

Cloud Computing-Platform as Service

Gurudatt Kulkarni, Prasad Khatawkar, Jayant Gambhir

Abstract: Cloud computing is an increasingly popular paradigm for accessing computing resources. In practice, cloud service providers tend to offer services that can be grouped into three categories: software as a service, platform as a service, and infrastructure as a service. This paper discuss the characteristics and benefits of cloud computing. The advantages in using the cloud for start-up and small businesses that lack infrastructure have been shown to far outweigh the disadvantages. Cloud platform services, also known as Platform as a Service (PaaS), provide a computing platform or solution stack on which software can be developed for later deployment in a cloud.

Index Terms: PaaS, SaaS, Cloud, Auto-scaling,

1.1 WHAT IS A CLOUD COMPUTING?[1]

Cloud computing is Internet("CLOUD") based development and use of computer technology ("COMPUTING").Cloud computing is a general term for anything that involves delivering hosted services over the Internet. It is used to describe both a platform and type of application. These cloud applications use large data centers and powerful servers that host Web applications and Web services. Anyone with a suitable Internet connection and a standard browser can access a cloud application.

1.2 BENEFITS OF CLOUD COMPUTING [4]

- ❖ Cloud technology is paid incrementally, saving organizations money.
- ❖ Organizations can store more data than on private computer systems.
- ❖ No longer do IT personnel need to worry about keeping software up to date.
- ❖ Cloud computing offers much more flexibility than past computing methods.
- ❖ Employees can access information wherever they are, rather than having to remain at their desks.

Manuscript published on 30 December 2011.

* Correspondence Author (s)

Gurudatt Anil Kulkarni*, Department of Electronics & Telecommunication, Marathwada Mitra Mandal's Polytechnic Pune, Pune, India, Phone/ Mobile No-9525571214., (e-mail: gurudatt.kulkarni@mmpolytechnic.com).

Prasad Ramakant Khatawkar, Department of Electronics & Telecommunication, Marathwada Mitra Mandal's Polytechnic Pune, Pune, India, Phone/ Mobile No-9525571214., (e-mail: khatawkarpr@mmpolytechnic.com).

Jayant Hari Gambhir, Department of Electronics & Telecommunication, Marathwada Mitra Mandal's Polytechnic Pune, Pune, India, Phone/ Mobile No-09175767011, (e-mail: gambhirjh@mmpolytechnic.com).

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](http://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>.



Cloud Computing

Having secure access to all your applications and data from any network device

FIG:-1.1 GENERAL CLOUD DIAGRAM

1.2.1 TECHNICAL BENEFITS OF CLOUD COMPUTING [7]

Automation – “Scriptable infrastructure”: You can create repeatable build and deployment systems by leveraging programmable (API-driven) infrastructure.

Auto-scaling: You can scale your applications up and down to match your unexpected demand without any human intervention. Auto-scaling encourages automation and drives more efficiency.

Proactive Scaling: Scale your application up and down to meet your anticipated demand with proper planning understanding of your traffic patterns so that you keep your costs low while scaling.

Improved Testability: Never run out of hardware for testing. Inject and automate testing at every stage during the development process. You can spawn up an “instant test lab” with pre-configured environments only for the duration of testing phase.

Disaster Recovery and Business Continuity: The cloud provides a lower cost option for maintaining a fleet of DR servers and data storage. With the cloud, you can take advantage of geo-distribution and replicate the environment in other location within minutes.

1.3 WHAT IS DRIVING CLOUD COMPUTING?

The CLOUD COMPUTING is driving in two types of categories as follows:

- Customer perspective
- Vendor perspective

1.3.1 CUSTOMER PERSPECTIVE:

- In one word: economics
- Faster, simpler, cheaper to use cloud computation.



- No upfront capital required for servers and storage.
- No ongoing for operational expenses for running datacenter.
- Application can be run from anywhere.

1.3.2 VENDOR PERSPECTIVE:

- Easier for application vendors to reach new customers.
- Lowest cost way of delivering and supporting applications.
- Ability to use commodity server and storage hardware.

1.4. TYPES OF SERVICES:

These services are broadly divided into three categories:

- Infrastructure-as-a-Service (IaaS)
- Platform-as-a-Service (PaaS)
- Software-as-a-Service (SaaS).

1.4.1 INFRASTRUCTURE-AS-A-SERVICE (IAAS):[7]

Infrastructure-as-a-Service (IaaS) provides virtual servers with unique IP addresses and blocks of storage on demand. Customers can pay for exactly the amount of service they use, like for electricity or water, this service is also called utility computing.

1.4.2 PLATFORM-AS-A-SERVICE (PAAS):[7]

Platform-as-a-Service (PaaS) is a set of software and development tools hosted on the provider's servers. Google Apps is one of the most famous Platform-as-a-Service providers. This is the idea that someone can provide the hardware (as in IaaS) plus a certain amount of application software - such as integration into a common set of programming functions or databases as a foundation upon which you can build your application. Platform as a Service (PaaS) is an application development and deployment platform delivered as a service to developers over the Web.

1.4.3 SOFTWARE-AS-A-SERVICE (SAAS):

Software-as-a-Service (SaaS) is the broadest market. In this case the provider allows the customer only to use its applications. The software interacts with the user through a user interface. These applications can be anything from web based email, to applications like Twitter or Last.fm.

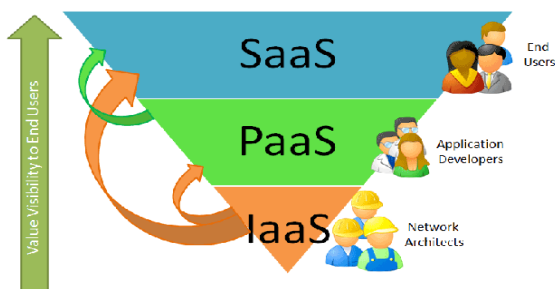


FIG 1.4.3 CLOUD COMPUTING STACK

You don't have to purchase anything. You just pay for what you use. A SaaS provider typically hosts and manages a given application in their own data center and makes it available to multiple tenants and users over the Web. Some SaaS providers run on another cloud provider's PaaS or IaaS service offerings. Oracle CRM on Demand, Salesforce.com, and Net suite are some of the well known SaaS examples.

2. HOW DOES CLOUD COMPUTING WORK?

Cloud computing aims to apply the power of supercomputer to problems like analyzing risk in financial portfolios, powering immersive computer games, in a way that users can tap through the Web. It does that by networking large groups of servers that often use low-cost consumer PC technology, with specialized connections to spread data-processing chores across them. Soon instead of installing a suite of software for each computer, you'd only have to load one application. That application would allow workers to log into a Web-based service which hosts all the programs. Remote machines owned by another company would run everything from e-mail to word processing to complex data analysis programs. It's called cloud computing. The only thing the user's computer needs to be able to run is the cloud computing systems interface software, which can be as simple as a Web browser, and the cloud's network takes care of the rest.

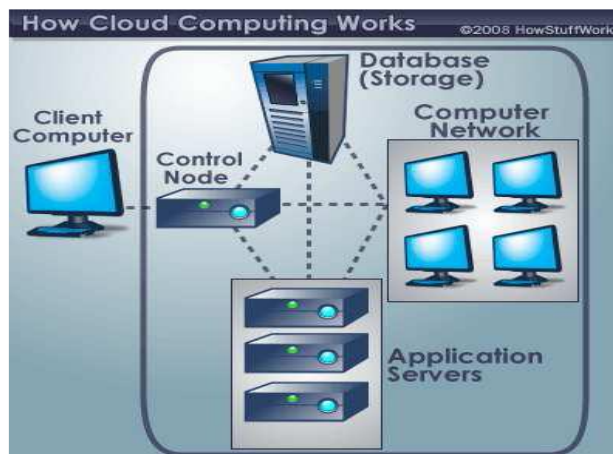


FIG 1.5 HOW CLOUD WORKS

Cloud computing providers deliver applications via the internet, which are accessed from a Web browser, while the business software and data are stored on servers at a remote location. In some cases, legacy applications (line of business applications that until now have been prevalent in thin client Windows computing) are delivered via a screen-sharing technology such as Citrix Xen App, while the computing resources are consolidated at a remote data center location; in other cases, entire business applications have been coded using web-based technologies such as AJAX.

3.0 PLATFORMS AS SERVICE [3]

“Cloud computing” has dramatically changed how

business applications are built and run.

At its core, cloud computing eliminates the costs and complexity of evaluating, buying, configuring, and managing all the hardware and software needed for enterprise applications. Instead, these applications are delivered as a service over the Internet. Cloud computing has evolved to include platforms for building and running custom applications, a concept known as “platform as a service” (or PaaS). PaaS applications are also referred to as on-demand, Web-based, or software as a service (or SaaS) solutions. PaaS (Platform-as-a-Service) is a business model in the cloud computing era, which provide a server platform or development environment for developers. Compared with the SaaS (Software-as-Service) to end-user, PaaS is more flexibility. Google App engine is a typical PaaS platform, which allows developers to write and run their own applications on the platform. It also helps developers store data and manage the server.

3.1 THE TRADITIONAL MODEL

Building and running on-premise applications has always been complex, expensive, and risky. Each application required hardware, an operating system, a database, middleware, Web servers, and other software. Once the stack was assembled, a team of developers had to navigate complex programming models like J2EE and .NET. A team of network, database, and system management experts was needed to keep everything up and running. Inevitably, a business requirement would require a change to the application, which would then kick off a lengthy development, test, and redeployment cycle. Large companies often needed specialized facilities to house their data centers. Enormous amounts of electricity also were needed to power the servers as well as the systems to keep them cool. Finally, a failover site was needed to mirror the data center so information could be replicated in case of a disaster.

3.2 THE NEW MODEL

Just as Amazon.com, eBay, Google, iTunes, and YouTube made it possible to access new capabilities and new markets through a Web browser, PaaS offers a faster, more cost-effective model for application development and delivery. PaaS provides the entire infrastructure needed to run applications over the Internet. It is delivered in the same way as a utility like electricity or water. Users simply “tap in” and take what they need without worrying about the complexity behind the scenes. And like a utility, PaaS is based on a metering or subscription model so users only pay for what they use. With PaaS, ISVs and corporate IT departments can focus on innovation instead of complex infrastructure. By leveraging the PaaS, organizations can redirect a significant portion of their budgets from “keeping the lights on” to creating applications that provide real business value. This model is driving a new era of mass innovation. For the first time, developers around the world can access unlimited computing power. Now, anyone with an Internet connection can build powerful applications and easily deploy them to users wherever they’re located. The PaaS provider will deliver the platform on the web, and in most cases you can consume the platform using your

browser. There is no need to download any software. This combination of simplicity and cost efficiency empowers small and mid-size companies, or even individual developers, to launch their own Cloud SaaS. Without any initial investment, they can leverage the power of these platform providers. Figure 3.2.1 shows Platform of Service base model. PaaS Layers

- Cloud OS
- Cloud Middleware

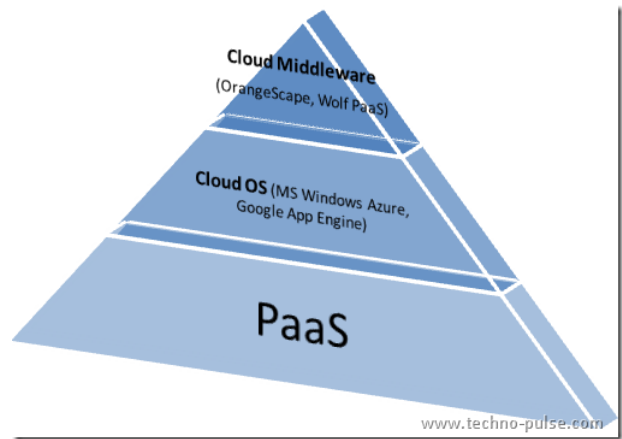


FIG 3.2.1 PaaS Layers

Google App Engine and Windows Azure are examples of Cloud Operating Systems (OS). Orangescape and Wolf PaaS are cloud middleware. Microsoft Windows Azure and Google App Engine are for hard-core developers. Existing .Net developers can easily move up to the Azure platform and create their own scalable Cloud SaaS. Java and Python developers can use Google Apps Engine to deliver cloud apps. OrangeScape and Wolf PaaS are on-demand browser based platforms for rapidly designing and delivering *multi-tenant* Cloud SaaS apps. They can be used as a 5 GL platform which automates many standard software development tasks to simplify application development. The USP of these platforms is that they can be used by business analysts (non developers) as well. You don’t need to write one piece of code! With a little patience, anyone with good analytical ability and domain knowledge can get started with his or her first Cloud SaaS. PaaS is suitable for both business and professional developers. Business developers who don’t write code can use platforms like Wave Maker to develop their applications with drag and drop components, and professional developers can use platforms like Google App Engine to build and deploy web applications. PaaS provides the facilities to support the end-to-end life cycle of building and delivering web

applications and services. Some of the major PaaS providers are Google App Engine for Java™ and Python, Microsoft® Azure for .Net and Java, and Force.com.

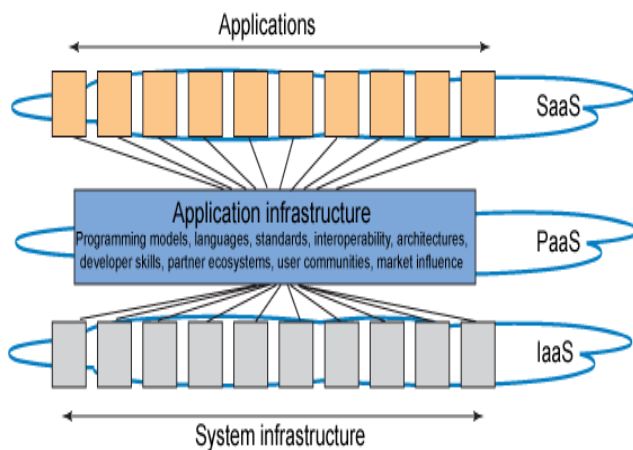


FIG 3.2.2 TRADITIONAL MODEL

PaaS platforms also have functional differences from traditional development platforms.

These include:

- Multi-tenant development tool: traditional development tools are single user - a cloud-based studio must support multiple users, each with multiple active projects.
- Multi-tenant deployment architecture: scalability is often not a concern of the initial development effort and is left instead for the sys admins to deal with when the project deploys. In PaaS, scalability of the application and data tiers must be built-in (e.g., load balancing, failover need to be basic elements of the dev platform itself).
- Integrated management: traditional development solution usually does not concern them with runtime monitoring, but in PaaS, the monitoring ability needs to be baked into the development platform.
- Integrated billing: PaaS offerings require mechanisms for billing based on usage that are unique to the SaaS world.

3.3 PaaS ELEMENTS:

1. Browser-based development studio - if you have to install something on your computer to develop applications, that's not PaaS!
2. Seamless deployment to hosted runtime environment - ideally, a developer should be able to deploy a PaaS application with one click. If you have to talk to a person to get your app deployed, that's not PaaS!
3. Management and monitoring tools - while cloud-based solutions are very cost effective, they can be tricky to manage and scale without good tools. If you have to bolt on DIY monitoring to scale your cloud app, that's not PaaS!

4. Pay as you go billing - avoiding upfront costs has made PaaS popular. If you can't pay with your credit card based on usage, that's not PaaS!

3.4 BENEFITS OF PaaS

The benefits of PaaS lie in greatly increasing the number of people who can develop, maintain and deploy web applications. In short, PaaS offers to democratize development of web applications much the same way that Microsoft Access democratized development of client/server application. Today, building web applications requires expert developer with three, highly specialized skill sets:

- Back end server development (e.g., Java/J2EE)
- Front end client development (e.g., JavaScript/Dojo)
- Web site administration.

PaaS offers the potential for general developers to build web applications without needing specialized expertise. This allows an entire generation of MS Access, Lotus Notes and PowerBuilder developers to start building web applications without the huge learning curve.

3.5 SALIENT FEATURES:-[9]

- Known also as PaaS (Platform-as-a Service)
- Empowers developers to deploy, deliver and manage their applications. They can build applications, upload (deploy) the same into the cloud platform and simply run and test them
- Developers can also leverage additional benefits like authentication and data access provided by the platform
- This cloud takes away the concept of servers, while providing an application centric environment
- While creating this kind of cloud computing platform, a vendor “builds a cloud platform first and then develops applications that run on it” (OR) “develops a host able application and then plugs it into the cloud”. But considering the advantages and disadvantages, the latter would be the better approach

3.6 PaaS ANATOMY:-

Now that you've come this far in learning about PaaS, let's explore what features you should consider when comparing PaaS providers:

- Application development framework. A robust application development framework built on technology that is widely used. Ideally, you should beware of the potential for vendor lock-in here. Open source platforms such as Java™ technology are usually a safe bet in this regard.

- Ease of use. A PaaS should come with easy-to-use WYSIWYG tools that have pre-built widgets, canned UI components, drag-and-drop tools, and support for some standard IDEs. It should facilitate rapid, iterative application development.
- Business process modeling (BPM) tools. You need a strong BPM framework that allows you model your business process and build the application around it.
- Availability. The platform of choice should be accessible and available anywhere, anytime.
- Scalability. The platform should be smart enough to leverage the elastic capacity of an underlying infrastructure to handle the loads the application will be put under.
- Security. To effectively combat threats, the platform should address things like cross-site scripting, SQL injection, Denial of Service, and encryption of traffic and make it ingrained into the application development. In addition, the platform must support single sign-on capabilities for you to be able to integrate it with your remaining on-premise applications or any other cloud applications.
- Inclusive. The platform should provide the ability to include, embed, and integrate other applications built on the same platform or others.
- Portability. The platform should be agnostic to the underlying infrastructure and allow companies to move the application from another IaaS to another.
- Porting tools. To facilitate an easy and quick migration of data from the legacy on-premise application to the application based on the new platform, bulk import transformation tools are a necessary part of the platform's toolkit.
- API. To perform tasks such as user authentication and storing and retrieving files (for example, Web application files and assets) and sometimes even making calls directly to a database, the platform should have a well-documented API. This will allow your business to have the flexibility of creating and customizing a software application to interface with the platform that meets the specific needs of the company.

3.7 PaaS FOR DEVELOPERS:- [2]

A common misunderstanding for developers is that cloud computing applies only to network administrators. But this misconception overlooks the many possibilities that cloud computing brings to development and quality assurance teams. Consider some of the things that are often problematic during the software development life cycle. In my experience, the process of setting up the server environment that will host the Web application the development team has been assigned to build can be a huge hassle. Even in the largest enterprises, there is typically a single network administrator resource assigned to several development teams. When PaaS is not being used, setting up a

development or test environment typically requires the following tasks:

- Acquire and deploy the server.
- Install the operating system, run time environments, source control repository, and any other required middleware.
- Configure the operating system, run time environments, repository, and additional middleware.
- Move or copy existing code.
- Test and run the code to make sure everything works.

3.8 ISSUES WITH EXISTING PaaS SOLUTIONS [8]

- Vendor lock-in (lack of cloud portability): No option was given to a developer or an organization to move from one cloud to another. For example, if a developer chose Google App Engine, then they must remain with Google App Engine unless the application is deleted using the Google App Engine console and the existing application is modified according to the new cloud provider. The application must then be deployed there.
- Deployment to public cloud only: Most of the existing PaaS solutions target only public cloud, which results in many enterprises not moving to PaaS. There are numerous enterprises that will never move to the public cloud, and they were given no choice to set up private cloud on their infrastructure or use a combination of private and public cloud.
- Stagnant mindset: Developers have to change their mindset for building applications on the public cloud. Previously, Google App Engine did not provide the option of relational databases. The application had to be coded against BigTable, which requires a change in mindset before building begins, and there is a learning curve associated with it. The developer can take an application WAR file and deploy to the cloud without any modifications.
- Support for diluted frameworks: Currently, Java programmers use frameworks, and less than 100 percent framework supports introduces difficulty when writing applications. Using Google App Engine to build a simple application using any common framework like Spring or Struts required a few hacks before using that framework.
- Tied to a particular environment: The existing PaaS offering is tied to particular environments such as Java and Python using BigTable as back end for Google App Engine,

Net and Java using SQL Azure database for Microsoft Azure, or Java for Amazon Beanstalk. Using any of these PaaS requires survival in that environment only.

- Not open-source: No existing PaaS solutions are open-source, making it difficult for developers to fork the code and add support for new features to share with the community. A compelling element of Cloud Foundry is its open-source nature, allowing developers outside the parent organization a major role in its success.

4.0 CONCLUSION:-

Most of the existing PaaS solutions target only public cloud, which results in many enterprises not moving to PaaS. PaaS applications are referred to as on-demand, Web-based, or software as a service (or SaaS) solutions. PaaS (Platform-as-a-Service) is a business model in the cloud computing era, which provide a server platform or development environment for developers. To fully benefit from all PaaS capabilities, developers need to adapt their applications to be able to react to changing conditions, which may require them to invoke the appropriate IaaS and PaaS services via API calls from within their applications. PaaS solutions have particular appeal for startups, projects with ambitious deadlines, and organizations with a limited budget for capital expenditures.

ACKNOWLEDGMENT

Gurudatt Kulkarni one of the authors is indebted to Principal Prof. Prasad Khatawkar for giving permission for sending the paper to the journal. Jayant Gambhir is also thankful to the Secretary Principal B.G. Jadhav, Marathwada Mitra Mandal for giving permission to send the paper for publication. We would also like to thanks our colleagues such as Lecturer Ramesh Sutar and Ravindra Desai for supporting us.

REFERENCE

1. <http://www.ibm.com/developerworks/opensource/library/os-springroo4/index.html>
2. "Detailed reference to security and geoprocessing services", byron ludwig and serena coetzee, Department of computer science, University of Pretoria, Pretoria, 0002
3. "Cloud Computing Platform as Service", InformationWeek 16 Oct. 2, 2009.
4. "THE IMPROVEMENT OF PAAS PLATFORM", ZENG SHU-QING; XU JIE-BIN; DEPT. OF COMPUTER. SCI. & TECHNOLOGY, SOUTH CHINA UNIVERSITY OF TECHNOLOGY. GUANGXHOU, CHINA, NETWORKING AND DISTRIBUTED COMPUTING.
5. "CLOUD COMPUTING: AN ANALYSIS", THAKUR RAMJIRAM SINGH, INTERNATIONAL JOURNAL OF ENTERPRISE COMPUTING & BUSINESS SYSTEMS, VOL. 1 ISSUE 2 JULY 2011.
6. "Controlling the Cloud: Requirements for Cloud Computing", F5 Networks' perspective on cloud computing: definition, architecture, and development, Lori MacVittie
7. L. Wang, G. Laszewski, M. Kunze and J. Tao, "Cloud computing: a perspective study", J New Generation Computing, 2010, pp 1-11.
8. <http://www.esscloud.com/paasBenefits.html>
9. <http://www.theresearchpedia.com/research-articles/top-10-benefits-of-paas>

AUTHOR PROFILE

Gurudatt Kulkarni completed BE Electronics & Telecommunication from Shivaji University Kolhapur. He is currently working in Marathwada Mitra Mandal's Polytechnic, Pune as Lecturer. He is Microsoft Certified System Administrator. He has presented two papers in National level Conference.

Prof. Prasad Khatawkar has completed Master in Engineering from COEP Pune. He is Life time member of ISTE Chapter. Currently he is working Principal in Marathwada Mitra Mandal's polytechnic Pune.

Jayant Gambhir completed BE Electronics from Pune University Pune. He is currently working in Marathwada Mitra Mandal's Polytechnic, Pune as Lecturer. He has presented two papers in National level Conference.