ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 2, Issue 1, January 2013

Challenges in the Cloud Application Development

B.J.D Kalyani, Asst.Professor, Department of Information Technology,
Swarna Bharathi Institute of Science & Technology, Khammam

***Abstract*— Anyone who develops and tests software is well-known with the Software Development Life Cycle (SDLC). By its very nature, cloud-based development offers an organization a high degree of agility; correspondingly, the developmental processes themselves should be agile in nature. This paper provides an overview of the challenges faced in developing cloud applications, and discusses an agile framework that offers a repeatable process for building cloud-based applications.**

***Index Terms*— Agility, Software Development Life Cycle, Security, Reliability.**

I. INTRODUCTION

The simple truth is that applications are increasingly being assembled from many small components, which are aggregated into an integrated collection that operates as a single deliverable. The development and deployment approach must support constant change in individual components and continuous integration of all components into the final deliverable.

Instead, there needs to be an approach to development and operations across all phases of an application to achieve application agility [1], which enables business agility. That need is intuitively obvious once one accepts that rapid application evolution is never-ending, not an occasional exception or irritating interruption.

The solution to the cloud application lifecycle challenge is clear: An approach to application assembly, delivery and deployment that can support continuous development and integration and that is used by all parties in the IT delivery chain. Examining the implications of the requirements of agile application lifecycle and deployment in a malleable hybrid cloud environment makes it clear that a new approach to application management is necessary where deployment in a cloud environment—especially a cloud environment in which the final image must be capable of running in dissimilar cloud environments. Linking resource, configuration, and cloud management will be key for Enterprises going forward. Cloud service development requires a different approach than the traditional software development lifecycle as the cloud provider becomes a critical success factor of the overall project. In a traditional software development setting, more emphasis is put on the functional aspects because it is deployed on an on-premise infrastructure with implicit security [2], compliance, control, operational transparency and perceived service level requirements. Another important factor is the cost of operations.

II. WHY IS CLOUD DEVELOPMENT DIFFERENT?

The Cloud Computing environment is very different compared to a traditional data centre in terms of how applications are deployed, configured, run, and managed. This requires certain fundamental attributes to be present for an application built for a Cloud environment [3]. The following table highlights some of the key differences between traditional applications and Cloud based applications.

Traditional Apps	Cloud Computing Apps
Application components co-located in same environment (deployed as one bundle)	Components are mostly scattered around one or many Clouds (deployed as multiple modules)
Run-time infrastructure is structured and controlled during development	Run-time infrastructure is un-structured and managed by Cloud fabric
Security is enforced by application architecture	Security is built into the service contracts
Support for Multi-tenancy is typically not required	Multi-tenancy support is assumed
Typical user base is known at Design-Time and is consistent	User base may not be known and could be dynamic
Deployment requires traditional packaging and deployment tools	Along with traditional tools, deployment requires knowledge and utilization of vendor specific Cloud API and tools
Application is tested in controlled environment	Application is tested on the Cloud to ensure seamless orchestration between services on one or many Clouds focus on across the network communication.

Table 1: Traditional Versus Cloud Applications

All Rights Reserved © 2013 IJARCET

III CHALLENGES FOR THE DEVELOPMENT TEAM

Just as Cloud based applications are "required" to be different, they present unique challenges to application architects and developers [4]. Here are some key challenges that one may encounter while building apps for the Cloud.

3.1 Challenges at design time to map to Cloud layers

- Infrastructure (server and installed software)
- Storage (RDBMS, Google's Bigtable, Amazon's SimpleDB, etc)
- Platform (solution stacks such as Ruby on Rails, LAMP, Python Django, etc)
- Application (custom app, Google Docs, SalesForce, Facebook, Flickr, etc)
- Services (custom web services, PayPal, Google Maps, etc)
- Client (thin/thick desktop clients, mobile clients like Symbian, Android, iPhone, etc)

3.2 Challenges in providing business solutions into:

- Services – Independent business tasks that can be started/stopped separately across the Cloud
- Service Applications - Applications that get deployed "into" a service (Java web app into a Tomcat service)

3.3 Challenges while Design with SOA principles in mind:

- Standardized Service Contracts, Loose Coupling, Abstraction, Reusability, Service Autonomy, Statelessness, Composability.

3.4 Challenges while building Cloud APIs to build complex business applications:

- Security - Use WS-security, SAML (or other methods) tokens that can be passed seamlessly between application and services
- Use standard service-access techniques e.g.: SOAP, REST, .NET and Spring Remoting
- Use common storage device (e.g.: SAN) accessible to all workers in the Cloud for storing resources (configuration files, databases, etc)
- Remember interoperability before using Cloud Fabric specific APIs
- Regard resources as services (storage, database, JMS queues, MOM, etc) and build service-enabled access to them
- Design framework components (logging, auditing, security, etc) as services

3.5 Challenges at design time regarding migration:

- **Reliability** – Choose a service provider with mirrored sites
- **Data security** – Check vendor policies before hosting sensitive data
- **Vendor lock-in** – Build methods to migrate data out of a provider, Avoid design using vendor specific APIs

IV DRIVERS FOR THE ADAPTION OF CLOUD COMPUTING

Eight attributes of cloud computing can be seen as drivers [5] for the adoption of cloud computing. The attributes are availability, collaboration, elasticity, lower infrastructure costs, mobility, risk reduction, scalability, and virtualization. Table 2 describes how these attributes can serve as drivers for cloud computing adoption.

Attribute	Why It Can Draw an Organization Toward Cloud Computing
Availability	Users have the ability to access their resources at any time through a standard internet connection.
Collaboration	Users begin to see the cloud as a way to work simultaneously on common data and information.
Elasticity	The provider transparently manages a user's resource utilization based on dynamically changing needs.
Lower Infrastructure Costs	The pay-per-usage model allows an organization to only pay for the resources they need with basically no investment in the physical resources available in the cloud. There are no infra-structure maintenance or upgrade costs.
Mobility	Users have the ability to access data and applications from around the globe.
Risk Reduction	Organizations can use the cloud to test ideas and concepts before making major investments in technology.
Scalability	Users have access to a large amount of resources that scale based on their demand.
Virtualization	Each user has a single view of the available resources, independently of how they are arranged in terms of physical devices. Therefore, there is potential from a provider perspective to serve a greater number of users with fewer physical resources.

Table 2: Cloud Computing Drivers

V BARRIERS TO THE ADOPTION OF CLOUD COMPUTING

Some key organizational concerns can act as barriers to the adoption of cloud computing. These concerns are interoperability, latency, platform or language constraints, regulations, reliability, resource control, and security as shown in Table 3.

All Rights Reserved © 2013 IJARCET

ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 2, Issue 1, January 2013

Concern	Why It Can Act as a Barrier to Cloud Computing Adoption
Interoperability	A universal set of standards and/or interfaces have not yet been defined, resulting in a significant risk of vendor lock-in.
Latency	All access to the cloud is done via the internet, introducing latency into every communication between the user and the provider.
Platform or Language Constraints	Some cloud providers support specific platforms and languages only.
Regulations	There are concerns in the cloud computing community over jurisdiction, data protection, fair information practices, and international data transfer—mainly for organizations that manage sensitive data.
Reliability	Many existing cloud infrastructures leverage commodity hard-ware that is known to fail unexpectedly.
Resource Control	The amount of control that the user has over the cloud provider and its resources varies greatly between providers.
Security	The main concern is data privacy: users do not have control or knowledge of where their data is being stored.

Table 3: Cloud Computing Barriers

VI CLOUD DEVELOPMENT PROCESS SELECTION

Since Cloud development is different and poses its own challenges to the development team, it is important to consider a Cloud development process which leverages the strengths and benefits of Cloud development while helping the team manage the challenges.

A software application development process [6] provides information on *who* (roles) does *what* (workflows) *with* what (Work products) and *how* (guidance). Projects typically follow a specific process during their lifecycle. Cloud computing is a technology which introduces a original way of accessing IT resources. It allows access to hardware, virtualised computational power, data storage, development and execution platforms or applications, in a service-oriented manner. These are characterized by being particularly scalable and accessed through the Internet.

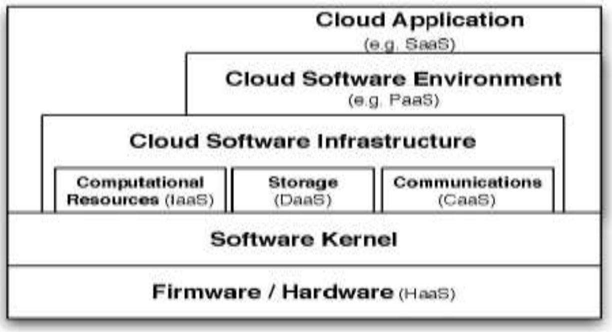

Figure 1: Cloud structure

The Five layers in Cloud Computing are :

•Hardware Layer liable for providing physical Infrastructure for higher layers, such as hardware and Networking systems.

• Software Kernel Layer which deals with basic control over physical infrastructure including OS kernels, hypervisors, virtual machine monitors and clustering middleware.

• Cloud Software Infrastructure Layer is accountable for delivering computational power, data storage and communications infrastructure in the form of virtualised abstraction. The service-oriented manner of delivering these artefacts are usually called Infrastructure as a Service (IaaS), Data as a Service (DaaS) and Communication as a Service (CaaS).

• Cloud Software Environment Layer handles run time environment for the applications that run on the top of Cloud's stack. It includes APIs, which allow applications to interact with the underlying Cloud infrastructure. The services provided by this layer are usually referred to as Platform as a Service (PaaS).

• Cloud Application Layer is the visible part of the cloud to end users, exposing services in the form of software applications which are commonly known as Software as a Service (SaaS). Such applications are usually based on web technology and are accessible by means of the Internet.

6.1 Process Choices

Now to choose a specific process type for authoring the Cloud development process framework, we consider three broad types of processes which exist at present:

6.1.1 Waterfall Processes

In the Waterfall process, various project disciplines like Requirements, Analysis & Design etc. become sequential phases. Phase gates depend on signed document exchange. The process is good for repetitive work, but is otherwise perceived as rigid. The 'realized' risk typically rises with time, since risk mitigation opportunity diminishes with time.

All Rights Reserved © 2013 IJARCET

6.1.2 Iterative Processes

In an Iterative process [7], various project disciplines like Requirements, Analysis & Design etc. are interspersed within iterations which typically occur sequentially, and could be combined into phases. Iteration gates depend on specific objectives. The process is good for exploratory work. The 'realized' risk typically diminishes with time, since risk management happens early and within every iteration.

6.1.3 Agile Process

Agile processes are also typically iterative as above. However spiritually they draw from the Agile Manifesto [8] Agility within an information systems project. (What) Quick: Businesses want a quick "Time to-Market" – faster than the speed of competition – is the mantra. (How) Resourceful: To achieve such agility, resourcefulness is important. (How) Adaptable: The project must possess adaptability to be resourceful.

VII THE CLOUD DEVELOPMENT PROCESS

Based on the Goals for a Cloud Development Process [9] above, an agile process is chosen as the baseline upon which the Cloud Development process would be built. Today's Agile processes like Scrum and XP deliver agility on the terms mentioned above. If configures appropriately, they could facilitate quick discovery and assembly of resources and services available within the Cloud in order to build a software application.

Agile Development Benefits [10] are as follows:

- Predictability through "time boxing"
- Transparency
- Energy, fuelled by creativity and job satisfaction

Cloud computing is the ideal environment for agile development. It lets you get valuable functionality to your customers quickly, collect immediate feedback, and make rapid changes based on that feedback. These fast development cycles, an inherent advantage of cloud computing, are impossible to implement in the traditional development model because of the high cost of distribution.

VIII FUTURE VISION

As enterprises adopt more and more Cloud based applications, it is anticipated that they will need to interact with more than one Cloud provider for reasons of service capability/feature, reliability/availability, data security, compliance and regulatory requirements. Business processes supported by Cloud-based applications will need to "traverse" multiple Clouds to accomplish their business requirements.

Although there are some vendors offering secure and reliable connectivity between Clouds and a management console to manage and monitor environments and associated workloads, a mature set of development tools that can facilitate cross-Cloud development and testing is yet to emerge and this is definitely an area of growth in the future.

We also anticipate that the different nature of Cloud application development will catalyze changes in software development process frameworks; which will become more agile, practice-based and active.

CONCLUSION

It is crucial to understand the requirements and challenges of a Cloud application to be able to fully benefit from a Cloud environment. They are different from traditional development practices and norms because of the fundamental differences in Cloud architecture. Architects and designers must adapt to a new mindset to develop for the Cloud. Correspondingly, development processes and practices must also evolve to address the inherent expectations of agility and flexibility of a Cloud environment.

REFERENCES

[1] Craig Balding, "ITG2008 World Cloud Computing Summit", 2008 http://cloudsecurity.org/S

[2] Jonothan Erickson, "Best Practices for Protecting Data in the Cloud", 2008 http://www.ddj.com/security/210602698

[3] Dikaiakos, M.D; Katsaros, D.; Mehra, P.; Pallis, G.; Vakali, A.; (2010), "Cloud Computing Distributed Internet Computing for IT and Scientific Research".Vol.13 ,pp 10, Sept.-Oct. 2009.

[4] Mathur, P; Nishchal, N.; (2010), "Cloud Computing: New challenge to the entire computer industry", 2010 1st International Conference on Parallel, Distributed and Grid Computing (PDGC - 2010), pp 223.

[5] Cloud Testing- Issues, Challenges, Needs and Practice
Jerry Gao 1,2, Xiaoying Bai2, and Wei-Tek Tsai2,3, Software Engineering : An International Journal (SEIJ), Vol. 1, No. 1, SEPTEMBER 2011

[6] cloud computing, internet of Services and advanced Software Engineering ObjECTIvE IC T-2011.1.2 CAll 8 OF FP7-ICT

[7] Geva Perry, "How Cloud & Utility Computing Are Different", 2008 http://gigaom.com/2008/02/28/how-cloud-utility-computing-are-different/

[8] Michael Miller, "Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online", August 2008

[9] John Rittinghouse, "Cloud Computing: Implementation, Management, and Security", 2009

[10] Agile Development Meets Cloud Computing for Extraordinary Results at Salesforce.com

All Rights Reserved © 2013 IJARCET