White Paper

Major Design Issues in Cloud Application Development

Atul Gupta
Architect, MphasiS

MPHASIS
an HP company

Contents

1. Abstract

The cloud has created tremendous value in the IT industry, ranging from the on-demand availability of enterprise class infrastructure via Infrastructure as a Service (IaaS) and the simplified deploy and manage semantics of Platform as a Service (PaaS), to business applications delivered via the Software as a Service (SaaS) model.

If approached incorrectly, however, the cloud is a double-edged sword. More specifically, writing applications for the cloud requires much more sophistication than writing traditional application architectures. Certain decisions that made sense in a pre-cloud era no longer make sense in the modern cloud world. In addition, the cloud has introduced some unique programming problems of its own that, if not understood, could lead to long-term negative consequences.

The cloud is not something that is public only in nature; enterprises are adopting true, private cloud semantics where compute power, storage and applications can be consumed with ease by end users internal to the enterprise. This means that potential mistakes exist both inside and outside the walls of enterprise IT.
This paper defines cloud application architecture and discusses some of the major cloud design choices, which an have a very negative impact if not properly addressed. Learning how to deal with these choices will provide a strong foundation to anyone tasked with building or managing cloud applications.

2. The Cloud Characteristics

The essential characteristics of cloud computing include both the delivery model and the commercial model. In terms of the delivery model, it should:

- Be available as a service over the Internet
- Accessible either from a Web browser or as a Web service
- Rapidly scale up and down in terms of available resources

Commercially:

- Users pay for service usage
- Both the overall maintenance effort and user costs are low.

Cloud computing presents both a model and new opportunities for technology buyers in an enterprise. It offers the opportunity to focus on the core capabilities of an enterprise by outsourcing certain aspects of IT and reducing IT costs. It also accelerates the provisioning and deployment with the support hassles that are transferred to the cloud-service provider. The cloud is altering the way in which organizations build their infrastructure and applications.

Technologically, the cloud is a culmination of standards and technologies that have come together over the past several years. They include server virtualization, web security, and web services. The implication for the enterprise is that dynamically scalable infrastructure is available transparently, without any IT involvement in building and managing the infrastructure.

The popular cloud-computing models include Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), as illustrated in the following diagram.

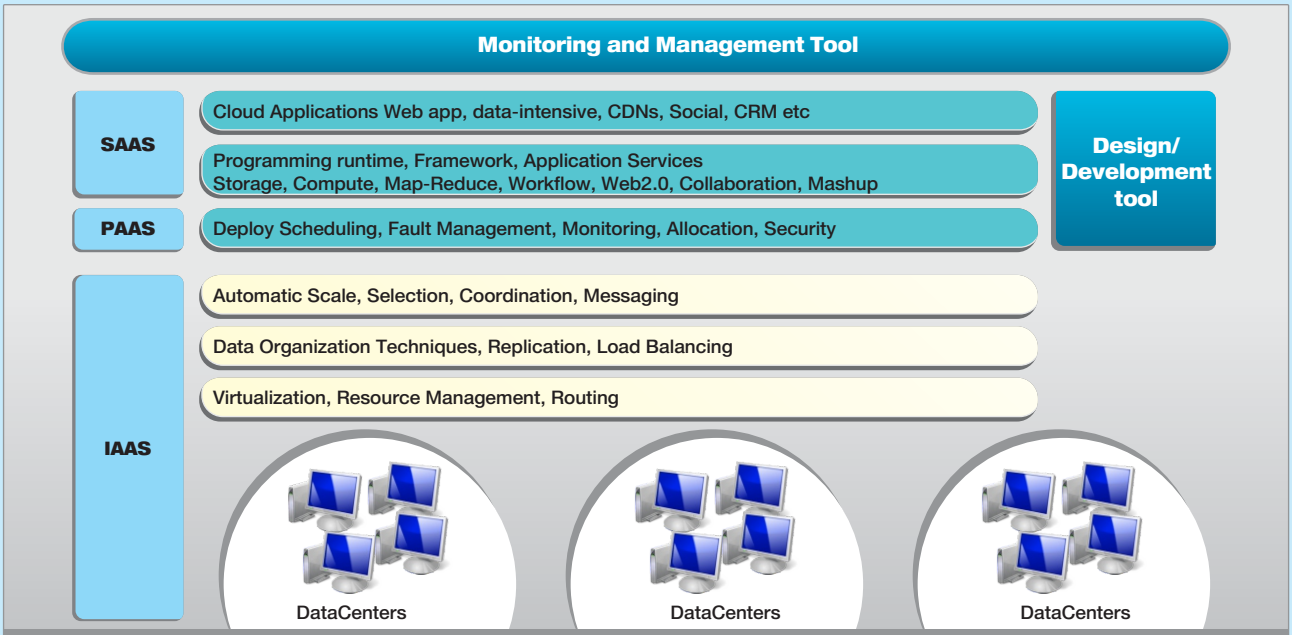

Figure 1: Cloud-Computing Model

The following are the characteristics of applications that are potential candidates for the cloud:

Non-core business services - For an enterprise, there are certain business functions that are core to the value that the enterprise delivers and drive more business and customers to the organization. For example, the core functions for a retailer are merchandizing and customer service. However, functions such as Web conferencing, enterprise-content management, and portal are non-core or non-differentiators; they are only enablers to support the business. Non-core applications are good candidates for the cloud and allow organizations to focus their business and IT staff on building the core.

Since it is difficult to predict the workload requirements for any social media based application like web conferencing or webinars, it is good to deploy them on the cloud to ensure the optimum hardware availability all the time. On the other hand, usually application QoS attributes are well defined and known in advance and can be addressed through proper capacity planning. They need not to be deployed on cloud.

High computing workloads over short time span - Applications that do not have uniform workload requirements. There will be spikes in usage for a small period; otherwise, however, resource utilization will be low most of time. Instead of acquiring infrastructure to support the peak workload, an enterprise can acquire space in the cloud and pay for usage.

For example Amazon Cloud Service charge you only for what you use. There is no minimum fee.

New business services and pilots - Applicable in scenarios in which enterprises need the flexibility to launch new business applications, without having to invest in full IT infrastructure upfront. The cloud offers a good platform for pilots, as businesses can terminate the contract, if needed—thus, helping them manage their risks better.

Web 2.0 collaborative applications - Applications (such as video sharing, discussion forums, and blogs) that generate data that is exposed to the public. As explained above, due to unpredictability in its workload, they are best suited to be modeled as cloud application.

Centralized applications - Applications that have cross- enterprise and cross-departmental reach. Instead of duplicating the effort in creating multiple applications or copies of the same software, these can be offered as cloud applications. Besides allowing enterprises to offload the infrastructure management to the cloud provider, they also offer economics of scale through their reuse across different departments.

3. Major Design Issues in Cloud Architecture/ Programming

3.1 Using Traditional Application Architectures

Traditional application architectures are not well suited for the cloud, public or private. Surely one can use the cloud, public or private, to host traditional applications and inherit deployment and management value. But if one is writing an application for a public or private cloud, a very different approach is needed. Cloud computing offers amazing elastic capacity. Having capacity available and the writing of an application that knows what to do with it, however, are very different things. Pre-cloud application architectures are typically built for fixed-server arrangements with rigid configuration: my database is over here, my application server over there, and some web services over here. This sort of approach is hardly capable of leveraging a dynamic infrastructure environment for scale-out, let alone intelligently handling high availability or providing for failure isolation. Furthermore, cloud computing platforms are shared in nature (as almost everything is virtualized). Sometimes, they can misbehave, thereby reducing reliability. Traditional software development practices with more rigid expectations tend to fail under these circumstances rather than planning for failure and working around it for continued operation.

3.2 Ignoring How Cost is Impacted by Architecture Decisions

When developing non-cloud applications, it was usually the case that writing code that was inefficient or didn't perform well, or choosing architectural patterns that were not cost-conscious, meant that the customers would have to spend more time and money operating that software. When building SaaS cloud offerings, the tables are turned. One is not only responsible for developing the applications, but also responsible for operating that application for his customers. Any inefficiency will increase its operating costs, drastically impacting its success profile. Interestingly, the cloud stack evolved from the top down. A number of companies built SaaS businesses worth hundreds of millions - if not billions - of dollars, and did so well before the advent of most cloud stack technologies. They evidenced that by making wise cost-savings motivated architectural decisions such as single instance, multi-tenancy, and shared nothing architectures. Using failure tolerant patterns, a company could push more revenue dollars to the bottom line or boost quality while keeping enterprise IT application delivery costs to a minimum. The cloud, public or private, is not inherently cheaper than traditional datacenters, but its flexibility and per-unit pricing approach does make it easy to fall into the trap of throwing money at a problem rather than using innovation to drive costs down.

3.3 Not Leveraging Purpose-Built Middleware and Tooling

Building cloud applications the right way is a difficult, costly, and time-consuming adventure. A majority of the challenges are orthogonal to your primary competency and development focus, and tackling those challenges head on is not worth it. SaaS and cloud pioneers had to build things from scratch. One of the largest time-sinks and risk points in software development is re-inventing the wheel. The software industry has reached an inflection point where everything from virtualization abstractions to true single-instance, multi-tenancy and codified, inheritable operating workflows are available in application server. Not leveraging middleware and tooling that already solves a majority of the cloud architecture challenges is one of the top mistakes. In fact, by doing research and deciding to use middleware packages that already solve most cloud architecture challenges, one will already be ahead of the curve with respect to the other mistakes identified above.

3.4 Ensuring Availability of the Cloud

This proves to be crucial as clouds host critical aspect of core business applications, for whom downtime would mean monetary losses. Effective monitoring and load balancing solutions are to be built. Most clouds employ virtualization technology to get the most out of any resource. In such cases, tools should be written to figure out a resource hog early and move the application to a more powerful grid or a machine, so that the other users get their share of the cloud without delays.

3.5 Ignoring Operational Lock-in Risk

A number of cloud platforms, particular some PaaS offerings; tout an all-in-one holistic stack. When the vendor's API and specific infrastructure patterns are exploited, it is possible to quickly realize a highly valuable solution. However, attention has to be paid to the risk of vendor lock-in. In many cases, the resulting code will only run on that particular platform and nowhere else. This might seem like a fair trade, but it's important to understand the implications: making an R&D decision of what runtime and stack to use is a very different one than committing to a day-in-and-day-out operating and hosting environment, whose reliability, security, and performance can changeover time. Try to choose public or private cloud technologies that are broadly adopted (even if proprietary), or better yet, cloud platforms that draw the line at providing infrastructure value and allow you to make operating independent architecture and programming decisions. In the latter case, identifying purpose-built cloud middleware that can be layered on top of cloud infrastructure can give you the best of both worlds and some without the associated risk.

3.6 Ignoring Traditional Approach of Application Code Review

Deploying an application on the cloud demands the correctness of the program in execution. Erroneous conditions, like infinite loops, can not only put the machine at the risk of being overloaded and unavailable, but also cost the user a significant amount of money. Tools like static analysis should be used to analyze code uploaded on the cloud and it should be checked for infinite loops, possible race conditions, null references, unreachable code etc. The code uploaded should also be optimized or suggestions should be provided to the users about how they could optimize code to best utilize the available resources.

4. MphasiS Cloud Capabilities & Offerings

MphasiS professional services for the cloud address various aspects of cloud adoption. It provides companies with business agility while reducing total cost of ownership.

MphasiS services include:

- Cloud strategy and adoption for developing your cloud roadmap
- Cloud builder for a secure private and service-provider cloud
- Cloud services aggregation and migration for best-in-class business services
- Cloud application builder for native cloud services
- Cloud orchestrator to bring together Public and Private clouds and enterprise assets
- Cloud sustenance for QoS and ecosystem governance

5. Conclusion

It is extremely important to understand the differences between traditional and cloud application architectures. By understanding these differences, you will be able to avoid common mistakes and fast-track your journey to cloud success.

6. Acknowledgments

The author wishes to thank **Bert Hooyman** and **Amber Marfatia** for their support, feedback and comments

About the Author

Atul Gupta

Atul Gupta works as an Architect at the Enterprise Web Technology Practice Group at MphasiS, USA. He has over 12 years of IT experience in enterprise application architecture and development. His current focus is on cloud computing related consulting services.

ABOUTMPHASIS.

MphasiS is a $1 billion global service provider, delivering technology based solutions to clients across the world. With over 41,000 people, MphasiS services clients in Banking and Capital Markets, Insurance, Manufacturing, Communications, Media & Entertainment, Healthcare & Life Sciences, Transportation & Logistics, Retail & Consumer Packaged Goods, Energy & Utilities, and Governments around the world. Our competency lies in our ability to offer integrated service offerings in Applications, Infrastructure Services, and Business Process Outsourcing. To know more about MphasiS, log on to www.mphasis.com

For more information, contact: sales@mphasis.com.

USA: 460 Park Avenue South, Suite #1101, New York, NY 10016, USA
Tel.: +1 212 686 6655, Fax: +1 212 686 2422

UK: 88 Wood Street, London EC2V 7RS, UK
Tel.: +44 20 85281000, Fax: +44 20 85281001

AUSTRALIA: 9 Norberry Terrace, 177-199 Pacific Hwy, North Sydney, 2060, Australia
Tel.: +61 2 99542222, Fax: +61 2 99558112

INDIA: Bagmane Technology Park, Byrasandra Village, C.V. Raman Nagar, Bangalore 560 093, India
Tel.: +91 80 4004 0404, Fax: +91 80 4004 9999

1011

MphasiS
an HP company