National Conference on Advances in Computer Science and Applications with International Journal of Computer Applications (NCACSA 2012)
Proceedings published in International Journal of Computer Applications® (IJCA)

Bottleneck Occurrence in Cloud Computing

Balvinder Singh
Priya Nain
M.Tech (CSE)
Lovely Professional University
Phagwara, Punjab

INTRODUCTION

It is conceivable that August 24, 2006 will go down as the birthday of Cloud Computing, as it was on this day that Amazon made the test version of its Elastic Computing Cloud (EC2) public [Business Week 2006]. This offer, providing flexible IT resources (computing capacity), marks a definitive milestone in dynamic business relations between IT users and providers. The target of Amazon's offer were developers, who had no wish to hold their own IT infrastructure, and instead, hired the existing infrastructure from Amazon via Internet. Nobody at this time spoke of Cloud Computing yet. The term first became popular in 2007, to which the first entry in the English Wikipedia from March 3, 2007 attests, which, again significantly, contained a reference to utility computing. Around this time, Dell attempted to trademark the word mark. This was successful in July, but the permission was revoked only a few days later.

In 2008, there was a glut of active parties in the increasingly popular field of Cloud Computing. Today, Cloud Computing generates over 10.3 million matches on Google. The scope of Cloud Computing grew from simple infrastructure services such as storage and calculation resources to include applications. However, this meant that forerunners such as application service providing and Software as a Service would also hence forth be included under the designation of Cloud Computing.

At the bottom of these developments was the eventual shifting of IT services away from local computers to the Internet or, generally speaking, in networks. Eventually, Cloud Computing realized an idea that had already been hit upon by Sun Microsystems long before the Cloud Computing hype.

What is Cloud Computing?

Cloud Computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the datacenters that provide those services. The services themselves have long been referred to as Software as a Service (SaaS), so we use that term. The datacenter hardware and software is what we will call a Cloud. Cloud computing is a computing paradigm, where a large pool of systems are connected in private or public networks, to provide dynamically scalable infrastructure for application, data and file storage. With the advent of this technology, the cost of computation, application hosting, content storage and delivery is reduced significantly. When a Cloud is made available in a pay-as-you-go manner to the public, we call it a Public Cloud; the service being sold is Utility Computing. Current examples of public utility Computing include Amazon Web Services, Google AppEngine, and Microsoft Azure.

Cloud Computing, the long-held dream of computing as a utility, has the potential to transform a large part of the IT industry, making software even more attractive as a service and shaping the way IT hardware is designed and purchased. Developers with innovative ideas for new Internet services no longer require the large capital outlays in hardware to deploy their service or the human expense to operate it. They need not be concerned about over provisioning for a service whose popularity does not meet their predictions, thus wasting costly resources, or under provisioning for one that becomes wildly popular, thus missing potential customers and revenue. Moreover, companies with large batch-oriented tasks can get results as quickly as their programs can scale, since using 1000 servers for one hour costs no more than using one server for 1000 hours. This elasticity of resources, without paying a premium for large scale, is unprecedented in the history of IT.

Cloud Computing Models

Cloud Providers offer services that can be grouped into three categories.

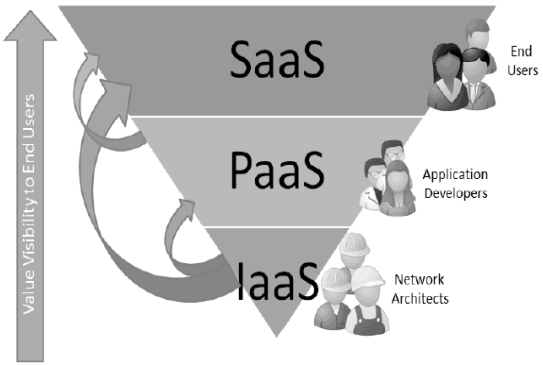
Software as a Service (SaaS): In this model, a complete application is offered to the customer, as a service on demand. A single instance of the service runs on the cloud & multiple end users are serviced. On the customers‟ side, there is no need for upfront investment in servers or software licenses, while for the provider, the costs are lowered, since only a single application needs to be hosted & maintained. Applications reside on the top of the cloud stack. Services provided by this layer can be accessed by end users through Web portals. Therefore, consumers are increasingly shifting from locally installed computer programs to on- line software services that offer the same functionally. Traditional desktop applications such as word processing and spreadsheet can now be accessed as a service in the Web. This model of delivering applications, known as Software as a Service (SaaS), alleviates the burden of software maintenance for customers and simplifies development and testing for providers

Platform as a Service (Paas): Here, a layer of software or development environment is encapsulated & offered as a service, upon which other higher levels of service can be built. The customer has the freedom to build his own applications, which run on the provider's infrastructure. To meet manageability and scalability requirements of the applications, PaaS providers offer a predefined combination of OS and application servers.

In addition to infrastructure-oriented clouds that provide raw computing and storage services, another approach is to offer a higher level of abstraction to make a cloud easily programmable, known as Platform as a Service (PaaS). A cloud platform offers an environment on which developers create and deploy applications and do not necessarily need to know how many processors or how much memory that applications will be using. In addition, multiple programming models and specialized services (e.g., data access, authentication, and payments) are offered as building blocks to new applications.

Infrastructure as a Service (Iaas): Iaas provides basic storage and computing capabilities as standardized services over the network. Servers, storage systems, networking equipment, data centre space etc. are pooled and made available to handle workloads. The customer would typically deploy his own software on the infrastructure. Public Infrastructure as a Service providers commonly offer virtual servers containing one or more CPUs, running several choices of operating systems and a customized software stack. In addition, storage space and communication facilities are often provided. In spite of being based on a common set of features, Iaas offerings can be distinguished by the availability of specialized features that influence the cost benefit ratio to be experienced by user applications when moved to the cloud. The most relevant features are:

(i) Geographic distribution of data centres;
(ii) Variety of user interfaces and APIs to access the system;
(iii) Specialized components and services that aid particular applications (e.g., load balancers, firewalls);
(iv) Choice of virtualization platform and operating systems; and
(v) Different billing methods and period (e.g., prepaid vs. post-paid, hourly vs. monthly).

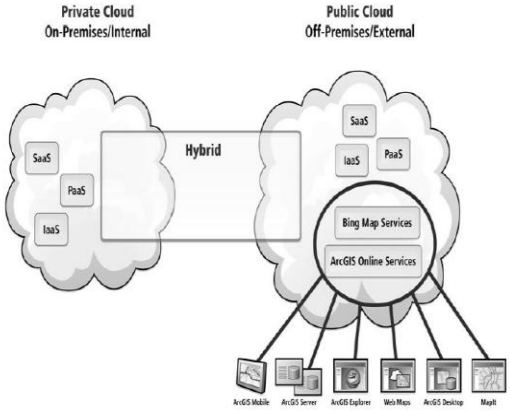


Understanding Public and Private Clouds

Enterprises can choose to deploy applications on Public, Private or Hybrid clouds. Cloud Integrators can play a vital part in determining the right cloud path for each organization.

Public Cloud: Public clouds are owned and operated by third parties; they deliver superior economies of scale to customers, as the infrastructure costs are spread among a mix of users, giving each individual client an attractive low-cost, "Pay-as-you-go" model. All customers share the same infrastructure pool with limited configuration, security protections, and availability variances. These are managed and supported by the cloud provider. One of the advantages of a Public cloud is that they may be larger than an enterprises cloud, thus providing the ability to scale seamlessly, on demand.

Private Cloud: Private clouds are built exclusively for a single enterprise. They aim to address concerns on data security and offer greater control, which is typically lacking in a public cloud. There are two variations to a private cloud:
On-premise Private Cloud: On-premise private clouds, also known as internal clouds are hosted within one's own datacenter. This model provides a more standardized process and protection, but is limited in aspects of size and scalability. IT departments would also need to incur the capital and operational costs for the physical resources. This is best suited for applications which require complete control and configurability of the infrastructure and security.
Externally hosted Private Cloud: This type of private cloud is hosted externally with a cloud provider, where the provider facilitates an exclusive cloud environment with full guarantee of privacy. This is best suited for enterprises that don't prefer a public cloud due to sharing of physical resources.
Hybrid Cloud: Hybrid Clouds combine both public and private cloud models. With a Hybrid Cloud, service providers can utilize 3rd party Cloud Providers in a full or partial manner thus increasing the flexibility of computing. The Hybrid cloud environment is capable of providing on-demand, externally provisioned scale. The ability to augment a private cloud with the resources of a public cloud can be used to manage any unexpected surges in workload.



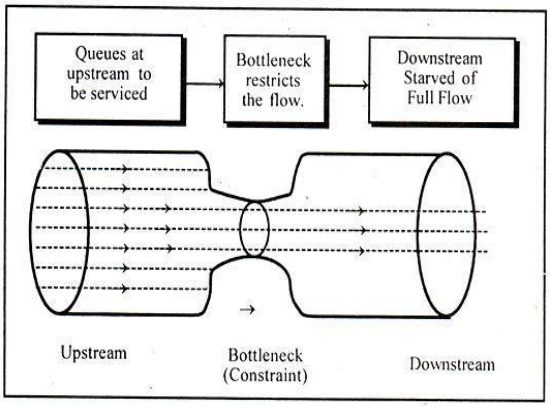
Speed and flexibility

Markets that change ever quicker are one of the main challenges that a company has to confront. Companies are successful when they see market opportunities early and react to them quickly. They will drive the issues and shape the market. Therefore in successful companies, organization and business processes are geared towards agility and flexibility.
In many companies it can be observed that the ICT (information and communication technology) cannot keep up with the pace and agility demanded by business. Instead of optimally supporting the business process with modern information and communications technology, in critical phases, the ICT turns out to be a bottleneck.
A company's processes and production systems are, in many cases, ready for the challenges of tomorrow – but in ICT methods often still prevail, which are at the level of industrialization via processor technology. Often, those responsible for ICT in a company are in a dilemma. The requirements of business of higher quality with decreasing costs appear to be two mutually opposing goals. With inflexible ICT structures, the mobility and dynamism are only possible with extreme limitations. Successful companies optimize added value with the help of innovative sourcing concepts, by outsourcing parts with which they do not differentiate or in which they are not implicitly competitive. Both cases involve working with a service provider with a command of specialist knowledge and who can achieve economies of scale through specialization. Hence, the challenge is in the creation of open cooperation models. This

is exhibited by successful companies, which only rarely work as closed, monolithic systems, but as dynamic, adaptive and networked systems.
Cloud Computing fulfils the demands, which companies place on ICT services, if the quick and flexible provision of resources and services on the one hand, a flexible adjustment of quantity on the other – as much from the top as from the bottom (scalability) – are SLA guaranteed. Services from within the cloud must therefore be designed in a way that they can be flexibly adapted to requirements.

Bottleneck in Cloud Computing

Cloud computing is gaining popularity as a way to virtualized the datacenter and increase flexibility in the use of computation resources. Virtualization technology has transformed the modern datacenter. Instead of installing applications directly onto physical machines, applications and operating systems are installed into virtual machine images, which in turn are executed by physical servers running a hyper visor. Virtualizing applications provides many benefits, including consolidation running multiple applications on a single physical machine and migration transparently moving applications across physical machines for load balancing and fault tolerance purposes. In this environment, the datacenter becomes a pool of interchangeable computation resources that can be leveraged to execute whatever virtual machine images (applications) are desired. Not every application, however, is suitable for deployment to public clouds operated by third party vendors. Medical records or credit card processing applications have security concerns that may be challenging to solve, and many other business applications may require higher levels of performance, quality-of-service, and reliability that are not guaranteed by a public cloud service. Thus, there is a motivation to maintain the administrative flexibility of cloud computing but keep all data behind the corporate firewall. This is referred to as private cloud computing. In recent years, several frameworks have been introduced to facilitate massively-parallel data processing on shared-nothing architectures like compute clouds. While these frameworks generally offer good support in terms of task deployment and fault tolerance, they only provide poor assistance in finding reasonable degrees of parallelization for the tasks to be executed. However, as billing models of clouds enable the utilization of many resources for a short period of time for the same cost as utilizing few resources for a long time, proper levels of parallelization are crucial to achieve short processing times while maintaining good resource utilization and therefore good cost efficiency.
Applications continue to become more data- intensive. If we assume applications may be "pulled apart" across the boundaries of clouds, this may complicate data placement and transport. At $100 to $150 per terabyte transferred, these costs can quickly add up, making data transfer costs an important issue. Cloud users and cloud providers have to think about the implications of placement and traffic at every level of the system if they want to minimize costs. This kind of reasoning can be seen in Amazon's development of their new Cloudfront service.



Example: One opportunity to overcome the high cost of Internet transfers is to ship disks.
Jim Gray found that the cheapest way to send a lot of data is to physically send disks or even whole computers via overnight delivery services.
Although there are no guarantees from the manufacturers of disks or computers that you can reliably ship data that way, he experienced only one failure in about 400 attempts.
To quantify the argument, assume that we want to ship 10 TB from U.C. Berkeley to
Amazon in Seattle, Washington. Garfinkel measured bandwidth from three sites and found an average write bandwidth of 5-18 Mbits/second. Suppose we get 20
Mbits/second over a WAN link. It would take

$10 * 10^{12}$ Bytes / ($20*10^6$ bits/second) = ($8*10^{13}$) = ($2*10^7$) seconds = 4,000,000 seconds,

This is more than 45 days. Amazon would also charge you $1000 in network transfer fees when it received the data. If we instead sent ten 1 TB disks via overnight shipping, it would take less than a day to transfer 10 TB and the cost would be roughly $400, an effective bandwidth of about 1500 Mbits/seconds.

Conclusion: The general idea of a bottleneck is that it is either a task or a communication channel in the cloud that slows down other parts of the processing chain and that the processing time of the entire cloud would improve if the bottleneck were to be alleviated in some way. We shall define two different types of bottlenecks:

CPU bottlenecks are tasks whose throughput is limited by the CPU resources they can utilize. CPU bottlenecks are distinguished by the fact that they have sufficient amounts of input data to process, however, subsequent tasks in the processing chain suffer from a lack thereof.

I/O bottlenecks are those communication channels which are requested to transport more records per time unit than the underlying transport infrastructure (e.g. network interconnects) can handle. Through this abstraction our approach becomes independent of the concrete physical compute and communication resource, which may be hard to observe in (shared) virtualized environments like Iaas clouds.

REFERENCES

[1] Torry Harris (2010) *"Cloud computing-An overview"*

[2] Simon Malkowski, Markus Hedwig, and Calton Pu *"Experimental EvaluationOf N-tier Systems: Observation and Analysis of Multi-Bottlenecks"*

[3] Jeffrey Shafer, Rice University (2009) *"I/O Virtualization Bottlenecks in Cloud Computing Today"* Houston, TX

[4] Technical Report No. UCB/EECS-2009-28 (2009) *"Above the Clouds: A Berkeley View of Cloud Computing"*

[5] Dominic Battr'e, Matthias Hovestadt, Bj'orn Lohrmann, Alexander Stanik andDaniel Warneke (2010) *"Detecting Bottlenecks in Parallel DAG-based DataFlow Programs"* IEEE papers