Cloud Computing

[1]D.Arvind Varma
[1]Computer science and engineering
[1]Saveetha School of engineering, Saveetha University

Abstract--- Cloud computing is a computing concept which is currently becoming popular over the Internet .It is invariably a storage paradigm which provides dynamically scalable and virtualized resources for hosting data and information on the net. The term cloud is actually a metaphor which depicts the underlying storage concept which shrinks and expands like a cloud containing rain water. Currently everything is hosted on the internet for a fixed price for storage. Website hosting servers have to be paid a fixed sum of money for hosting information and files. In turn they provide a fixed storage which can only be used . If additional storage is needed then price relative to the amount of storage needed has to be paid in addition to waiting time till the server allocates the resources. This price has to be paid initially and even if it is not fully utilized yet the money has to be paid. Cloud computing brings in the concept of 'pay as you use'. The present aim of Cloud computing is to provide virtually infinite workspace and storage for any client that has registered with a cloud computing server. This concept is presently adopted by the Hosting servers which provide this facility as a service. This article aims at illustrating the concept of cloud computing and also explains about the pros and cons of cloud computing technology.

I. INTRODUCTION

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). The datacenter hardware and software is what we call a Cloud.

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 App Engine, and Microsoft Azure.

We use the term Private Cloud to refer to internal datacenters of a business or other organization that are not made available to the public. Thus, Cloud Computing is the sum of SaaS and Utility Computing, but does not normally include Private Clouds. Cloud Computing allows deploying SaaS—and scaling on demand—without building or provisioning a datacenter.

II. WHEN IS UTILITY COMPUTING PREFERABLE TO RUNNING A PRIVATE CLOUD?

A first case is when demand for a service varies with time. Provisioning a data center for the peak load it must sustain a few days per month leads to underutilization at other times, for example. Instead, Cloud Computing lets an organization pay by the hour for computing resources, potentially leading to cost savings even if the hourly rate to rent a machine from a cloud provider is higher than the rate to own one. A second case is when demand is unknown in advance.

For example, a web startup will need to support a spike in demand when it becomes popular, followed potentially by a reduction once some of the visitors turn away. Finally, organizations that perform batch analytics can use the "cost associativity" of cloud computing to finish computations faster using 1000 EC2 machines for 1 hour costs the same as using 1 machine for 1000 hours.

III. FUTURE OF CLOUD COMPUTING

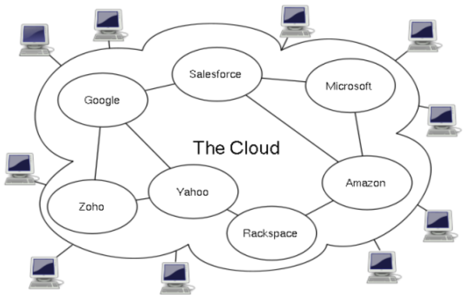

Fig. 1:

We can predict Cloud Computing will grow, so developers should take it into account. All levels should aim at horizontal scalability of virtual machines over the efficiency on a single VM. In addition.

- Applications Software needs to both scale down rapidly as well as scale up, which is a new requirement. Such software also needs a pay-for-use licensing model to match needs of Cloud Computing.
- Infrastructure Software needs to be aware that it is no longer running on bare metal but on VMs. Moreover, it needs to have billing built in from the beginning.
- Hardware Systems should be designed at the scale of a container (at least a dozen racks), which will be is the minimum purchase size. Cost of operation will match performance and cost of purchase in importance rewarding energy proportionality such as by putting idle portions of the memory, disk, and network into low power mode. Processors should work well with VMs, ash memory should be added to the memory hierarchy, and LAN switches and WAN routers must improve in bandwidth and cost.

IV. WHAT'S NEW IN CLOUD COMPUTING?

- The illusion of infinite computing resources available on demand, thereby eliminating the need for Cloud Computing users to plan far ahead for provisioning;
- The elimination of an up-front commitment by Cloud users, thereby allowing companies to start small and increase hardware resources only when there is an increase in their needs
- The ability to pay for use of computing resources on a short-term basis as needed (e.g., processors by the hour and storage by the day) and release them as needed, thereby rewarding conservation by letting machines and storage go when they are no longer useful.

V. BENEFITS OF BECOMING A PART OF THE CLOUD:

- Make a lot of money
- Leverage existing investment.

- Defend a franchise
- Attack an incumbent
- Leverage customer relationships
- Become a platform

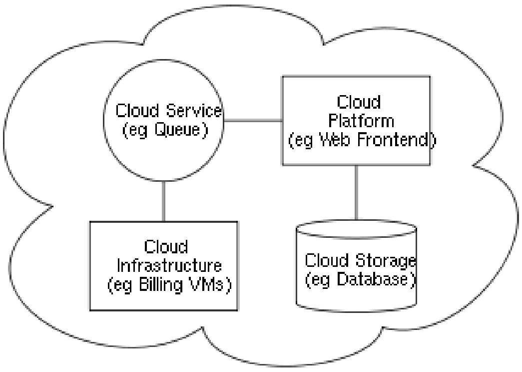

Fig. 2:Sample Cloud Architecture

VI. TOP 10 OBSTACLES AND THEIR SOLUTIONS

Not everything is perfect is in this world . Similarly though cloud computing hass lots of advantages yet it is hass some drawbacks about it. The following are the major obstacles faced by Cloud computing and my solutions to overcome them are proposed herewith:

- Availability of Service Use Multiple Cloud Providers to provide Business Continuity
- Data Lock-In Standardize APIs
- Data Confidentiality and Auditability
- Data Transfer and Data Backup/Archival
- Performance Unpredictability Improved Virtual Machine Support
- Scalable Storage
- Bugs in Large-Scale Distributed Systems
- Scaling Quickly
- Reputation Fate Sharing
- Software Licensing Pay-for-use licenses

A. Number 1 Obstacle: Availability Of A Service:

Organizations worry about whether Utility Computing services will have adequate availability, and this makes some wary of Cloud Computing. Ironically, existing SaaS products have set a high standard in this regard. Google Search is effectively the dial tone of the Internet: if people went to Google for search and it wasn't available, they would think the Internet was down. Users expect similar availability from new services, which is hard to do.

Just as large Internet service providers use multiple network providers so that failure by a single company will not take them off the air, I believe the only plausible solution to very high availability is multiple Cloud Computing providers. The high-availability computing community has long followed the mantra "no single source of failure," yet the management of a Cloud Computing service by a singlecompany is in fact a single point of failure. Even if the company has multiple datacenters in different geographic regions using different network providers, it may have common software infrastructure and accounting systems, or the company may even go out of business. Large customers will be reluctant to migrate to Cloud Computing without a business-continuity strategy for such situations.

The attack would therefore have to last 32 hours in order to cost the potential victim more than it would the blackmailer. A botnet attack this long may be difficult to sustain, since the longer an attack lasts the easier it is to uncover and defend against, and the attacking bots could not be immediately re-used for other attacks on the same provider. As with elasticity, Cloud Computing shifts the attack target from the SaaS provider to the Utility Computing provider, who can more readily absorb it and is also likely to have already DDoS protection as a core competency.

B. Number 2 Obstacle: Data Lock-In

Software stacks have improved interoperability among platforms, but the APIs for Cloud Computing itself are still essentially proprietary, or at least have not been the subject of active standardization. Thus, customers cannot easily extract their data and programs from one site to run on another. Concern about the difficult of extracting data from the cloud is preventing some organizations from adopting Cloud Computing. Customer lock-in may be attractive to Cloud Computing providers, but Cloud Computing users are vulnerable to price increases (as Stallman warned), to reliability problems, or even to providers going out of business.

For example, an online storage service called The Linkup shut down on August 8, 2008 after losing access as much as 45% of customer data [12]. The Linkup, in turn, had relied on the online storage service Nirvanix to store customer data, and now there is finger pointing between the two organizations as to why customer data was lost. Meanwhile, The Linkup's 20,000 users were told the service was no longer available and were urged to try out another storage site.

The obvious solution is to standardize the APIs so that a SaaS developer could deploy services and data across multiple Cloud Computing providers so that the failure of a single company would not take all copies of customer data with it.

C. Number 3 Obstacle: Data Confidentiality And Auditability:

"My sensitive corporate data will never be in the cloud." Anecdotally I have heard this repeated multiple times.Current cloud offerings are essentially public (rather than private) networks, exposing the system to more attacks.

I believe that there are no fundamental obstacles to making a cloud-computing environment as secure as the vast majority of in-house IT environments, and that many of the obstacles can be overcome immediately with well-understood technologies such as encrypted storage, Virtual Local Area Networks, and network middleboxes (e.g.firewalls, packet filters).

For example, encrypting data before placing it in a Cloud may be even more secure than unencrypted data in a local data center; this approach was successfully used by TC3, a healthcare company with access to sensitive patient records and healthcare claims, when moving their HIPAA-compliant application to AWS .

D. Number 4 Obstacle: Data Transfer Bottlenecks:

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 over WAN.

The most effective solution to this is to reduce the WAN costs.

E. Number 5 Obstacle: Performance Unpredictability:

My experience is that multiple Virtual Machines can share CPUs and main memory surprisingly well in Cloud Computing, but that I/O sharing is more problematic.

The reason for this is because not all virtual machines use the same set of access speeds. One opportunity is to improve architectures and operating systems to efficiently virtualize interrupts and I/O channels. Technologies such as PCIexpress are difficult to virtualize, but they are critical to the cloud. One reason to be hopeful is that IBM mainframes and operating systems largely overcame these problems in the 1980s, so we have successful examples from which to learn.

Another possibility is that flash memory will decrease I/O interference. Flash is semiconductor memory that preserves information when powered off like mechanical hard disks, but since it has no moving parts, it is much faster to access (microseconds vs. milliseconds) and uses less energy. Flash memory can sustain many more I/Os per second per gigabyte of storage than disks, so multiple virtual machines with con icting random I/O workloads could coexist better on the same physical computer without the interference we see with mechanical disks The lack of interference) might extend to semiconductor storage as well, thereby increasing the number of applications that can run well on VMs and thus share a single computer. This advance could lower costs to Cloud Computing providers, and eventually to Cloud Computing consumers.

F. Number 6 Obstacle: Scalable Storage:

Early in this paper, I identified three properties whose combination gives Cloud Computing its appeal: short-term usage (which implies scaling down as well as up when resources are no longer needed), no up-front cost, and infinite capacity on-demand. While it's straightforward what this means when applied to computation, it's less obvious how to apply it to persistent storage. The opportunity, which is still an open arbitrarily up and down on-demand, as well as meeting programmer expectations in regard to resource management for scalability, data durability, and high availability research problem, is to create a storage system would not only meet these needs but combine them with the cloud advantages of scaling arbitrarily up and down on-demand, as well as meeting programmer expectations in regard to resource management for scalability, data durability, and high availability

G. Number 7 Obstacle: Bugs In Large-Scale Distributed Systems:

One of the difficult challenges in Cloud Computing is removing errors in these very large scale distributed systems. A common occurrence is that these bugs cannot be reproduced in smaller configurations, so the debugging must occur at scale in the production datacenters. One opportunity may be the reliance on virtual machines in Cloud Computing. Many traditional SaaS providers developed their infrastructure without using VMs, either because they preceded the recent popularity of VMs or because they felt they could not afford the performance hit of VMs. Since VMs are de rigueur in Utility Computing, that level of virtualization may make it possible to capture valuable information in ways that are implausible without VMs.

H. Number 8 Obstacle: Scaling Quickly:

Pay-as-you-go certainly applies to storage and to network bandwidth, both of which count bytes used. Computation is slightly different, depending on the virtualization level. Google AppEngine automatically scales in response to load increases and decreases, and users are charged by the cycles used. AWS charges by the hour for the number of instances you occupy, even if your machine is idle.

The opportunity is then to automatically scale quickly up and down in response to load in order to save money, but without violating service level agreements

I. Number 9 Obstacle: Reputation Fate Sharing:

Reputations do not virtualize well. One customer's bad behavior can affect the reputation of the cloud as a whole. For instance, blacklisting of EC2 IP addresses by spam-prevention services may limit which applications can be effectively hosted. An opportunity would be to create reputation-guarding services similar to the "trusted email" services currently offered (for a fee) to services hosted on smaller ISP's, which experience a microcosm of this problem.

Another legal issue is the question of transfer of legal liability—Cloud Computing providers would want legal liability to remain with the customer and not be transferred to them (i.e., the company sending the spam should be held liable, not Amazon).

J. Number 10 Obstacle: Software Licensing

Current software licenses commonly restrict the computers on which the software can run. Users pay for the softwareand then pay an annual maintenance fee. Indeed, SAP announced that it would increase its annual maintenance fee toat least 22% of the purchase price of the software, which is comparable to Oracle's pricing . Hence, many cloud computing providers originally relied on open source software in part because the licensing model for commercial software is not a good match to Utility Computing.

VII. CONCLUSION

Thus 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

REFERENCES

[1] San Murugesan (Editor), "Cloud Computing: IT's Day in the Sun" ,Cutter Consortium, 2009.

[2] Luis M. Vaquero et al., "A Break in the Clouds: Toward a Cloud Definition", ACM SIGCOMM Computer Communication Review, Volume 39, Issue1 (January 2009)

[3] Berkley Rad Labs, "Above the Clouds",(February 2009) 1000 hours.

