Role of Cloud Computing in Cost Minimization: A Review

Pramod Kumar Sagar
Assistant Professor, Dept of IT,
SRM University, NCR Campus
Modinagar
pramodsagar.srm@gmail.com

Kanika Garg
Assistant Professor, Dept of IT,
SRM University, NCR Campus
Modinagar
kanikagarg.kg@gmail.com

Ruby Singh.
Assistant Professor, Dept of IT,
SRM University, NCR Campus
Modinagar
rubysinghit@gmail.com

ABSTRACT
Cloud computing has rapidly emerged as a new computation paradigm, providing agile and scalable resource access in a utility-like fashion. Processing of massive amounts of data has been a primary usage of the clouds in practice. While many efforts have been devoted to designing the computation models, one important issue has been largely neglected in this respect: how do we efficiently move the data, practically generated from different geographical locations over time, into a cloud for effective processing? The usual approach of shipping data using hard disks lacks flexibility and security. As the first dedicated effort, this paper tackles this massive, dynamic data migration issue. Targeting a cloud encompassing disparate data centers of different resource charges, we model the cost-minimizing data migration problem, and propose efficient offline and online algorithms, which optimize the routes of data into the cloud and the choice of the data center to aggregate the data for processing, at any given time.
This study is focusing on various issues in cloud computing and some suggestions and conclusion of the study.

Keywords— Cloud Computing, Utility Computing, Internet Datacenters, Distributed System Economics.

1. CLOUD COMPUTING
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.
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 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 will call a Cloud. When a Cloud is made available in a pay-as-you-go manner to the general public, we call it a Public Cloud; the service being sold is Utility Computing. We use the term Private Cloud to refer to internal datacenters of a business or other organization, not made available to the general public.Before the cloud era, each application has its own storage server, computing server, application logic as well as individual user interface. Nowadays, an emerging computing paradigm, called cloud computing, provides a shared resource pool, including shared storage resources, shared computing resources, even shared application logic. All resources are provided on demand and can be charged by pay-as-you-go policy. Before exploring more details about cloud computing, we should clarify the exact definition of cloud computing.

1.1 Challenges in the Cloud Era:
The recent cloud computing paradigm enables rapid on-demand provisioning ofserver resources (CPU, storage, bandwidth, etc.) to users with minimal management efforts.
Many cloud platforms have emerged, e.g., Amazon EC2 and S3 [1], Microsoft Azure [2], Google App Engine, Rackspace [3], GoGrid [4], which organize their sharedpool of servers from multiple data centers, and serve their users using different virtualization technologies. The elastic and on-demand nature of resource provisioning has made a cloud platform very suitable for running various applications, especially those computation-intensive ones [5]. More and more data-intensive Internet applications, e.g., Facebook, Twitter, are relying on

the clouds for processing and analyzing their peta byte-scale data sets, using a computing framework such as Map Reduce and Hadoop [6]. Many academic and industrial efforts have been devoted to designing better computing frameworks, e.g., Orchestra [7] is proposed to optimize the performance of large amounts of data transfer in the computation stages of Map Reduce. One important issue has largely been left out in this respect: How does one move the massive amounts of data into a cloud, in the very first place? The current practice is to copy the data into large hard drives and ship them to the data center [8], or to move machines entirely [9]. Such a shipping method inevitably introduces undesirable delay and possible service downtime, while outputs of the data analysis often needed to be presented to users in the quickest fashion [10]. It is also less secure, given that the hard drives can be infectious with a malicious program, or lost on the waydue to road accidents. A more flexible and intelligent data migration strategy is in need, to minimize any potential service downtime.

2. PROBLEMS FACED IN CLOUD COMPUTING

Problem 1: Availability of a Service

Organizations worry about whether Utility Computing services will have adequate availability and this makes somewary 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, we 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 single company 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.

Problem 2: 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.

Problem 3: Data Confidentiality and Auditability

"My sensitive corporate data will never be in the cloud." Anecdotally we have heard this repeated multiple times. Current cloud offerings are essentially public (rather than private) networks, exposing the system to more attacks. There are also requirements for auditability, in the sense of Sarbanes-Oxley and Health and Human Services Health Insurance Portability and Accountability Act (HIPAA) regulations that must be provided for corporate data to be moved to the cloud.

Problem 4: 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. 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 Cloud front service. 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 [11]. 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 (and even this could be mitigated by shipping extra disks with redundant data in a RAID-like manner).

Problem 5: Performance Unpredictability

Our 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. Figure 3(a) shows the average memory bandwidth for 75 EC2instances running the STREAM memory benchmark [12]. The mean bandwidth is 1355 MBytes per second, with astandard deviation of just 52 MBytes/sec, less than 4% of the mean. Figure 3(b) shows the average disk bandwidth for 75 EC2 instances each writing 1 GB files to local disk. The mean disk write bandwidth is nearly 55 MBytes per second with a standard

deviation of a little over 9 MBytes/sec, more than 16% of the mean. This demonstrates the problem of I/O interference between virtual machines.

Problem 6: Scalable Storage

Early in this paper, we 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.

Problem 7: 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.

Problem 8: Scaling Quickly

Pay-as-you-go certainly applies to storage and to network bandwidth, both of which count bytes used. Computations slightly different, depending on the virtualization level. Google App Engine automatically scales in response toload 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. Indeed, one RAD Lab focus is the pervasive and aggressive use of statistical machine learning as a diagnostic and predictive tool that would allow dynamic scaling, automatic reaction to performance and correctness problems, and generally automatic management of many aspects of these systems.

Problem 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. Or instance, blacklisting of EC2 IP addresses [13] 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 beheld liable, not Amazon).

Problem 10: Software Licensing

Current software licenses commonly restrict the computers on which the software can run. Users pay for the software and 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 [14]. 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.

3. CONCLUSION

The long dreamed vision of computing as a utility is finally emerging. The elasticity of a utility matches the need of businesses providing services directly to customers over the Internet, as workloads can grow (and shrink) far faster than 20 years ago. It used to take years to grow a business to several million customers – now it can happen in months. From the cloud provider's view, the construction of very large datacenters at low cost sites using commodity computing, storage, and networking uncovered the possibility of selling those resources on a pay-as-you-go model below the costs of many medium-sized datacenters, while making a profit by statistically multiplexing among a large group of customers. From the cloud user's view, it would be as startling for a new software startup to build its own datacenter as it would for a hardware startup to build its own fabrication line. In addition to startups, many other established organizations take advantage of the elasticity of Cloud Computing regularly, including newspapers like the Washington Post, movie companies like Pixar, and universities like ours. Our lab has benefited substantially from the ability to complete research by conference deadlines and adjust resources over the semester to accommodate course deadlines. As Cloud Computing users, we were relieved of dealing with the twin dangers of over-

provisioning and under-provisioning our internal datacenters. Some question whether companies accustomed to high-margin businesses, such as ad revenue from search engines and traditional packaged software, can compete in Cloud Computing. First, the question presumes that Cloud Computing is a small margin business based on its low cost. Given the typical utilization of medium-sized datacenters, the potential factors of 5 to 7 in economies of scale, and the further savings in selection of cloud datacenter locations, the apparently low costs offered to cloud users may still be highly profitable to cloud providers.

REFERENCES

[1]E. E. Schadt, M. D. Linderman, J. Sorenson, L. Lee, and G. P. Nolan, "Computational Solutions to Large-scale Data Management and Analysis," Nat Rev Genet,vol. 11, no. 9, pp. 647–657, 09 2010.

[2] Moving an Elephant: Large Scale Hadoop Data Migration at Facebook http://www.facebook.com/notes/paul-yang/ moving-an-elephant-large-scalehadoop-data-migration-at-facebook/10150246275318920.

[3] "A Conversation with Jim Gray," Queue, vol. 1, no. 4, pp. 8–17, Jun. 2003.

[4] R. J. Brunner, S. G. Djorgovski, T. A. Prince, and A. S. Szalay, "Handbook of MassiveData Sets," J. Abello, P. M. Pardalos, and M. G. C. Resende, Eds. Norwell,MA, USA: Kluwer Academic Publishers, 2002, ch. Massive Datasets in Astronomy,pp. 931–979.

[5] SenseWeb, http://research. microsoft. com/ en-us/ projects/ senseweb/.

[6] Amazon Elastic MapReduce, http://aws.amazon.com/elasticmapreduce/.

[7]"Cloud Computing ," National Institue of Standards and Technology, http://www.nist.gov/ itl/ cloud/ index.cfm.

[8] A. Borodin and R. El-Yaniv, Online Computation and Competitive Analysis.Cambridge University Press, 998, vol. 2.

[9] A. Borodin, N. Linial, and M. E. Saks, "An Optimal On-line Algorithm for MetricalTask System," J. ACM, vol. 39, no. 4, pp. 745–763, 1992.

[10] A. Karlin, M. Manasse, L. Rudolph, and D. Sleator, "Competitive SnoopyCaching," Algorithmica, vol. 3, pp. 79–119, 1988.

[11] A. R. Karlin, M. S. Manasse, L. A. McGeoch, and S. Owicki, "Competitive RandomizedAlgorithms for Non-uniform Problems," in Proceedings of ACM SODA.

[12] R. J. Brunner, S. G. Djorgovski, T. A. Prince, and A. S. Szalay, "Handbook of MassiveData Sets," J. Abello, P. M. Pardalos, and M. G. C. Resende, Eds. Norwell,MA, USA: Kluwer Academic Publishers, 2002, ch. Massive atasets in Astronomy,pp. 931–979.