Utilizing Content Delivery Network in Cloud Computing

Yale Li[#1], Yushi Shen[#2], Yudong Liu[*3]

[#]Microsoft
Redmond, WA, USA
[1]yuhang.li@microsoft.com
[2]yushis@microsoft.com
[*]Western Washington University
Bellingham, WA, USA
[3]yudong.liu@wwu.edu

Abstract— **Cloud Computing has the potential to trigger a major computing model transformation for the IT industry. This paper first briefly describes the business and technical benefits Cloud Computing will bring to us and explains technical challenges to adopt Cloud Computing including the network bottlenecks. One of the solutions to address network problems in Cloud Computing is Content Delivery Network (CDN). This paper digests the basics of Akamai CDN technology. Then, the authors conduct a CDN experiment in Microsoft public cloud, Windows Azure, to demonstrate the benefits of CDN integration with the cloud. The results show significant gain in large data download by utilizing CDN in Cloud Computing. Finally, a couple of academic research ideas are summarized for future cloud CDN improvements.**

Keywords*— *cloud computing, content delivery network* (*CDN*), *Windows Azure, cloud CDN

I. Cloud Computing Overview

Cloud computing has become a significant technology trend in recent years. Simply put, the goal of cloud computing in its early stage is to achieve utility computing, i.e. delivery of IT as a service. Cloud computing represents the 3rd major business computing model in IT history, with mainframe computing as the 1st major model and client/server computing as the 2nd major model. Cloud computing typically involves the provision of dynamically scalable and often virtualized resources as a service delivered over the Internet.

From business or customer's perspective, cloud computing is a subscription model to use services over the Internet. If one customer can use the browser somewhere and pay a fee online to get a service, he/she uses cloud computing. This broad definition is well described in [1].

From technical and provider's perspective, cloud computing has narrowed definition. To be qualified as "cloud", certain characteristics, delivery models, and service models have to be met. For example,as defined by National Institute of Standards and Technology [2], the five essential characteristics associated with cloud computing are:

- On-demand self-service,
- Broad network access,
- Resource pooling,
- Rapid elasticity,
- Measured service.

More introduction of cloud computing can be found in [1].

II. Opportunities and Challenges of Cloud Computing

Migration from client/server computing to cloud computing is a major computing model transformation. There are great opportunities in both business and technical innovation. However, tremendous challenges also present.

A. Business Opportunities and Challenges

On the business side, cloud computing will make business agile and utilize resources more effectively to achieve lower cost of total ownership. From cloud provider's perspective, [1] uses following formula to describe the profit for cloud service providers:

$$UserHours_{\text{cloud}} \times (revenue - Cost_{\text{cloud}}) \geq UserHours_{\text{datacenter}} \times (revenue - Cost_{\text{datacenter}} / Utilization) \quad (1)$$

The left-hand side represents the expected profit from using cloud computing. The right-hand side represents the expected profit from using traditional datacenter. Both sides perform the same calculation by multiplying profit per user-hour by the total user-hours.

Because the resource utilization in cloud computing is 100% (fully utilized) and the resource utilization in traditional is less than (partially utilized) or equal to 100% (fully utilized), the true cost of traditional datacenter is greater than or equal to the cost of cloud computing. The formula reveals that cloud providers have a better control over the cost per user-hour and therefore have a better opportunity to gain more profit.

From cloud customer's perspective, applications or services with following workload patterns will enjoy great benefit by cloud adoption:

- **Unpredictable Bursting**: An event happens and triggers heavy usage of resources,so normally the customer would have had to scale design considerations to try and "predict" what this resource usage requirement "could be",
- **Predictable Bursting**: Using Dominos Pizza as an example, the store is very busy on Friday nights. In

most days of the week, demand is much less. Even though the additional load is expected, it is expensive to maintain this capacity because it is under-utilized when demand is lower,

- **On and Off**: Similar to Predictable Bursting, On and Off can have seasonal or time-bounded workloads where there are all or almost nothing processing requirements. Important enterprise workloads including those that are run monthly, quarterly and annually exhibit this type of behaviour,
- **Growing Fast**: This is interesting in the case of smaller start-up companies or groups in larger companies. It can also be associated with new development. How to plan for rapid capacity increase is no longer an issue in the cloud, both during development and operation. Elasticity can be a huge opportunity/savings area.

Fig. 1 demonstrates the above 4 application workload patterns.

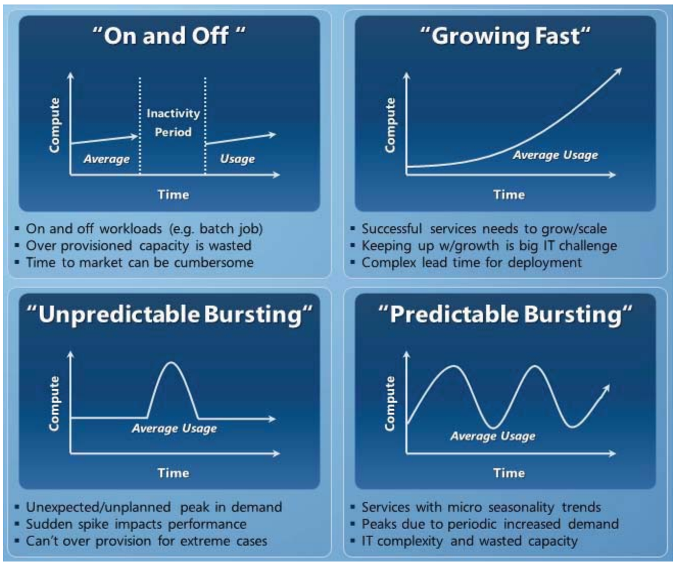

Fig. 1Cloud application workload patterns

The business challenge for cloud adoption is similar to IT outsourcing. It is critical to understand how the availability, security and supportability requirements will be met by cloud service providers.

It is also important to know the responsibility of each party, and have a formal service level agreement(SLA) and legal contract between the customer and provider.

B. Technical Opportunities

On the technical side, the study conducted in [1] has summarized the ten opportunities and obstacles for the growth of cloud computing. In Table I, the challenges are obstacles for cloud providers to deliver cloud services and blockers for customers to adopt cloud services. The opportunities are proposed technical solutions to overcome those obstacles and blockers.

Take the first item in the table as an example, the availability of cloud service might be lower than traditional service provided by enterprise datacenters due to Internetnetwork condition anddistributed denial of service (DDoS) attacks. The solutions are load distribution among multiple cloudproviders and absorbing DDoS attacks with the power of cloud elasticity. Other challenges and opportunities are also

well explained in [1], so there is no need to restate the rationales.

TABLE 1 CLOUD CHALLENGE AND OPPORTUNITY PROPOSED IN [1]

Challenges	Opportunities
Availability of Service	Use Multiple Cloud Providers; Use Elasticity to Prevent DDoS
Data Lock-In	Standardize APIs; Compatible SW to enable Surge Computing
Data Confidentiality and Auditability	Deploy Encryption, VLANs, Firewalls; Geographical Data Storage
Data Transfer Bottlenecks	FedExing Disks; Data Backup/Archival; Higher BW Switches
Performance Unpredictability	Improved VM Support; Flash Memory; Gang Schedule VMs
Scalable Storage	Invent Scalable Store
Bugs in Large Distributed Systems	Invent Debugger that relies on Distributed VMs
Scaling Quickly	Invent Auto-Scaler that relies on ML; Snapshots for Conservation
Reputation Fate Sharing	Offer reputation-guarding services like those for email
Software Licensing	Pay-for-use licenses; Bulk use sales

TABLEII OUR SUPPLEMENTARY CLOUD CHALLENGE AND OPPORTUNITY

Challenges	Opportunities
Network Congestion at Cloud Egress and Latency to Client	Integrate Cloud with Akamai Content Delivery Network Technology

However, [1] overlooked an important challenge and this issue should be mentioned or added in Table I. We add our supplement in Table II. Perhaps, the reason why [1] missed this point is that they focused on the cloud side and forgot that client is the other half of the entire cloud-client ecosystem.

C. Network Congestion at Cloud Egress and Latency to Client

In today's client/server computing model in enterprise and business IT, the resources (e.g., applications, emails, databases, files) are typically delivered to end user's client machines from servers over the LAN. The network congestion is not an issue because resources are highly distributed. The network hops over LAN are also few between client and server. With the advent of cloud computing model, the resources will be highly concentrated in the cloud datacenter. The network congestion at cloud egress becomes an issue because of the topology and usage pattern. Between a datacenter and clients, the network hops over the Internet can increase. Thus, the probability of latency due to data transmission delay or retransmission may also increase, especially when the datacenter and clients are located in different continents.

D. Integrating Content Delivery Network with Cloud

One of the solutions to resolve network congestion/latency problems is to integrate content delivery network (CDN) with cloud computing data centers. To use an analogy, the data center is the manufacturing plant and CDN servers are the warehouses of the distributors. Therefore, global load balancing will be achieved with CDN. The largest CDN in the world at present is the Akamai network which is described in Section 3. In Section 4, we will demonstrate the benefits of Cloud CDN as derived from an experiment in Microsoft Windows Azure cloud. Section 5 presents academically proposed solutions for further optimizingcloud CDN performance in the future.

III. Akamai CDN Technology

A. Content Delivery Network

A content delivery network (CDN) is a collection of web servers distributed across multiple locations to deliver content to users more efficiently. When optimizing for performance, the server selected for delivering content to a specific user is based on a measure of network proximity. For example, the CDN may choose the server with the fewest network hops or the server with the quickest response time.

A CDN service provider typically shares its CDN servers across all its clients. One drawback to rely on a CDN is that one customer's response time can be affected by traffic from other clients, possibly even a customer's competitors. Another drawback is the occasional inconvenience of not having direct control of the CDN servers. For example, modifying HTTP response headers must be done through the service provider rather than directly by customer's operation team. Finally, if CDN service provider's performance degrades, customer's performance degrades accordingly.

B. Akamai CDN Technology

Akamai CDN [3] is the mostly used CDN in the world. Akamai's original approach was to sell customers a distributed content-caching service. Its goal was simply to resolve bandwidth issues, and it solved that problem very well.

Fig. 2 below articulates Akamai CDN technology and architectural design, which comprises 7 steps:

- In Step 1, a client browser visits a web site, for example *http://www.cbc.ca*. If *cbc.ca* is integrated with Akamai CDN, the URL is mapped to Akamai network. In this case, the *.net* root name server is queried for the name resolution,
- In Step 2, the *.net* root name server sends domain delegation response to a lower level *.net* domain name server. In this process, the *.net* domain name server (DNS) returns an NS record for *akamai.net* sub-domain. This NS record is the Akamai CDN top-level name server,

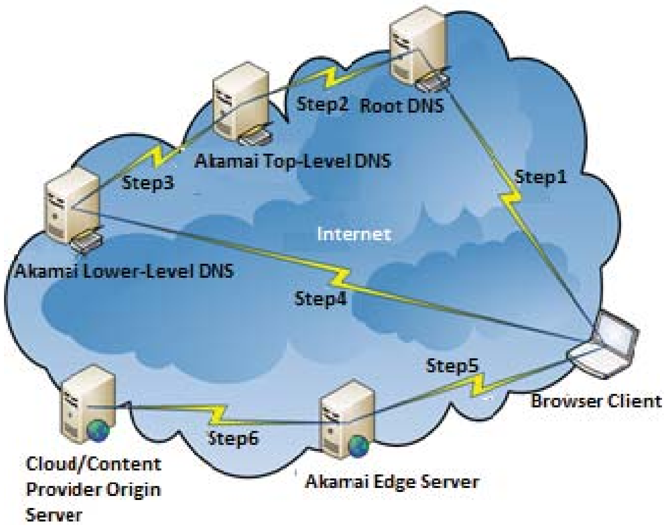

Fig. 2Akamai CDN technology [3]

- In Step 3, the Akamai Top-level DNS server returns a domain delegation or *.g.akamai.net* to a Low-Level Akamai name server with a TTL (Time To Live) of about one hour,
- In Step 4, the Akamai low-level DNS server returns the IP addresses of an available edge server, for example *a7.g.akamai.net*. This resolution has a short TTL (several seconds to one minute). The purpose of short TTL is to encourage frequent refreshes of the DNS resolution. Therefore, Akamai CDN is able to direct requests to edge servers in other locations based on network status measurements,
- In Step 5, the IP address for server *a7.g.akamai.net* domain is returned so the browser can access the edge server URL *http://a7.g.akamai.net*,
- In Step 6, if the requested web content is not cached on edge server (normally the first hit only), the content will be fetched from the origin server of content provider or cloud provider.

The criteria for network status measurements include service requested, server health, server load, network condition, client location, and content requested. This measurement determines an edge server for client access, and finds the path from the client to the edge server.

The integration of CDN with traditional web sites and cloud services is very similar. Content origin servers are located at customer's site or cloud provider's datacenters, and they are configured to communicate with Akamai infrastructure and edge servers.

IV. CDN Evaluation in Microsoft Cloud

A. Windows Azure Overview

The Windows Azure Platform [4] is Microsoft's PaaS (Platform as a Service) Public Cloud. Customers create applications on top of Azure Platform, by using programming languages (e.g. c#) and tools (e.g. Visual Studio) supported by Azure. As a PaaS cloud provider, Microsoft is responsible for OS (Operating System)and network builds and maintenance.

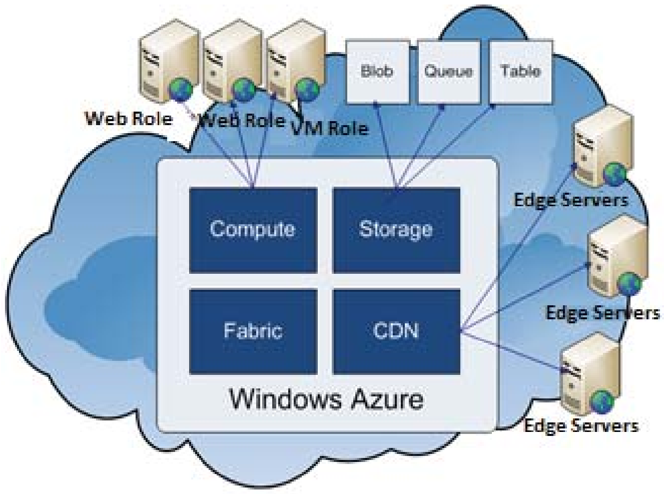

Fig. 3Windows Azure high-level architecture

The Windows Azure Platform is located in Microsoft datacenters. It provides a cloud operating system called Windows Azure that serves as a runtime for the applications and provides a set of services that allows development, management and hosting of applications off-premises. Following is detailed technical description of Azure Platform architecture and components [5], as shown in Fig. 3:

- **Compute**: This is customer's VM (virtual machines) running Windows Server 2008 Operating System in Web Role, Worker Role, and VM Role,
- **Storage**: This customer's scalable and persistent storage (in Blobs, Tables, and Queue forms) for large-scale needs. A blob storage container is similar to a network file drive in the cloud,
- **Fabric**: This makes up the building blocks of the Windows Azure platform. It is a collection of the network of interconnected nodes consisting of servers, high-speed connections, and switches,
- **CDN**: This is Azure Content Delivery Network [5]. It is based on Akamai CDN technology. It integrates with Azure Storage to provide better user experience for large file downloads. The integration of CDN with Azure or any cloud service is very similar to integration of CDN with traditional web sites.

Windows Azure became commercially available on 1stFeb 2010. Windows Azure CDN feature is offered as an optional service. This feature is implemented as an "Enable CDN" button in Azure Storage service. As of 2010, the Azure CDN enables worldwide low-latency delivery of static content from Azure Storage to end-users. Microsoft has 6 datacenters worldwide and some of the datacenters have as many as 300,000 – 400,000 servers.

The locations of Microsoft cloud data centers are:

- North America:
  - North-central US - Chicago, IL
  - South-central US - San Antonio, TX
- Asia
  - East Asia - Hong Kong
  - South East Asia – Singapore
- Europe
  - North Europe - Amsterdam, Netherlands
  - West Europe - Dublin, Ireland

B. Azure CDN Evaluation

We took advantage of the Microsoft Free Trial offered until June 30, 2011 for the CDN evaluation. Three Azure external storage services were created in three different data centers. Then, a public blob container *\yinghua* was created under each storage service.

The full URLs to access each public container are listed below:

- Located in US South Data Center: *http://b2bblob.blob.core.windows.net/yinghua*
- Located in Europe West Data Center: *http://europe.blob.core.windows.net/yinghua*
- Located in Asia East Data Center: *http://asiaeast.blob.core.windows.net/yinghua*

In this experiment, a free utility called "Azure Storage Explorer" was downloaded and installed on the PC after creating 3 blob storage containers. This utility was used to upload a large file (699 MB Ubuntu ISO image) to all 3 storage containers. Fig. 4 is a screenshot of uploaded file in Windows Azure Storage.

Without CDN enabled, the large file from the three cloud data centers was downloaded to a laptop computer located in Seattle, US. Table III shows the download time in three measurements made at different times (morning, noon, evening, respectively) on the same day. The accuracy is rounded to the nearest 10 second by IE software.

TABLE III EXPERIMENTAL RESULTS WITHOUT CDN ENABLED

Data Center Location	Asia	Europe	US
1st download time	30 min 20 sec	37 min 10 sec	11 min 0 sec
2nd download time	36 min 10 sec	30 min 30 sec	10 min 50 sec
3rd download time	37 min 30 sec	31 min 10 sec	12 min 30 sec
Average download time	34 min 40 sec	33 min 0 sec	11 min 30 sec
Standard deviation	3 min 50 sec	3 min 30 sec	1 min 20 sec

Afterwards, the CDN was enabled in the Windows Azure Portal tool. It took up one hour to propagate the content from blob storage container in Microsoft cloud data centers to the Azure CDN network globally. In Azure management tool, the authors observed that CDN used different URLs for content delivery of the same file. The msecnd.net CDN network is equivalent to akamai.net CDN network. The URLs are listed below:

- Europe: *http://az25036.vo.msecnd.net/yaleli/*
- US: *http://az25037.vo.msecnd.net/yaleli/*
- Asia: *http://az25038.vo.msecnd.net/yaleli/*

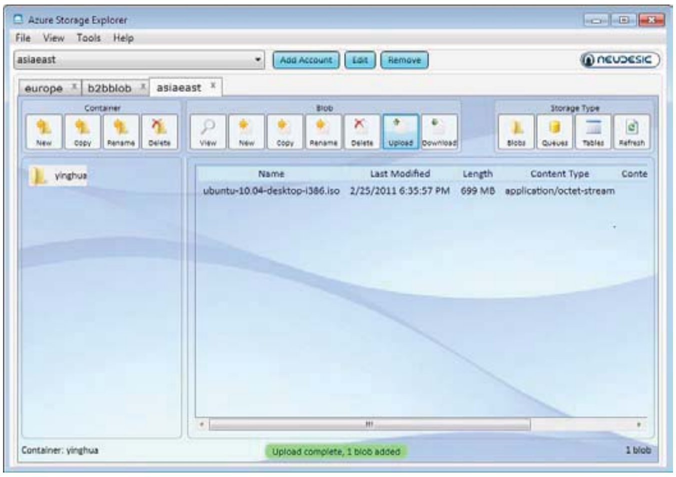

Fig. 4A screenshot of uploaded file in Windows Azure Storage

Then, the download time was measured on the same file again at different times (morning, noon, evening, respectively) on the same day. Table IV shows the results. The accuracy is rounded tothe nearest second by IE software.

The average (or mean) in both tables is calculated by:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i (2)$$

TABLE IVEXPERIMENTAL RESULTS WITH CDN ENABLED

Data Center Location	Asia	Europe	US
1st download time	7 min 19 sec	7 min 55 sec	6 min 34 sec
2nd download time	5 min 43 sec	5 min 34 sec	7 min 49 sec
3rd download time	6 min 43 sec	8 min 24 sec	5 min 59 sec
Average download time	6 min 35 sec	7 min 18 sec	6 min 48 sec
Standard deviation	48 sec	1 min 31 sec	56 sec

The standard deviation in both tables is calculated by:

$$\sigma = \frac{\sqrt{\Sigma(x-\bar{x})^2}}{n-1} (3)$$

Without CDN, the download speed isroughly 170% slower from US, 350% slower from Europe, and 420% slower from Asia. You can see that Azure CDN is very effective in improving the content delivery time for the cloud, particularly for cloud data centers far away from the end user's client machine.

C. Challenge of Cost Control in Cloud CDN

As shown in our experiments, CDN has greatly benefited cloud computing for large content delivery such as video and data file downloads. However, it also brings the challenge that the network consumption and cost control therefrom may become unpredictable for both cloud CDN providers and customers.

For example, in this experiment, our original plan was to repeat the testing for multiple days so that more experimental data could be collected. However, Azure CDN service consumes additional service charge and exceeds the free usage quota quickly. In fact, there is no way in Azure or Akamai CDN to set a fee limit for our global CDN usage.

In reality, many enterprise IT departments are cost centers. They have fixed budget for network usage even if they move to the cloud. Consumers also have the same desire to limit their monthly bill to use cloud services. To satisfy customer's requirements, the cloud providers need to provide a flat-fee billing model in addition to usage-based pricing. In order to provide a flat-fee service, a provider must be able to limit total network usage consumption, including CDN usage, in cloud computing environment.

To address the issues, more improvements should be added into CDN service in the future. In the next section, some potential solutions to this problem will be discussed, which include implementing a Distributed Rate Limiting (DRL) [6] and its variant in cloud CDN.

V. Future Improvements for Cloud CDN

Two possible improvements are to incorporate DRL [6] or CDN Redirection [7] in Azure or other cloud services. If DRL was provided by Azure CDN service, network consumption and monthly bill should not be worried after the free subscription is over. In the following, some potential solutions to this problem are discussed.

A. Fixed-cost Model for Cloud CDN

A fixed-cost model for cloud CDN is obviously a handy solution. This model works fine as long as the network consumption is predictable for both service providers and customers. However, such prediction turns out to be a significant challenge in a distributed environment such as cloud CDN. The service providers have to provide a fixed price for an aggregate, global usage, and allow services toconsume resources dynamically across various locations, subject tothe specified aggregate limit. Here we are more interested in the other two solutions.

B. DRL for Cloud Control

In [7], a good solution named DRL (Distributed Rate Limiting) was proposed to address the issue. DRL is a technology to control the total usage of network bandwidth in cloud computing, whether or not CDN is enabled.

One of their contributions is the DRL algorithms. They designed and implemented 2 DRL algorithms. The first algorithm is called GRD (Global Random Drop). The purpose of GRD is to approximate the number of packet drops. The 2nd algorithm is called FPS (Flow Proportional Share). The purpose of FPS is to improve scalability.

Then they designed a peer-to-peer limiter architecture for DRL. In their design, each limiter is not only functionally identical but also operates independently. The functionalities of limiter are divided into 3 subtasks. Task one is estimation for computing the average arrival rate. A standard EWMA (exponentially-weighted moving average) filter is also used to cancel out fluctuations in this task. Task two is communication for estimating number of dynamically chosen limiters. Task three is allocation for combining global estimates. In this task, local measurements are used to determine a limit for local rate enforcement.

After the design, 2 limiters are implemented and then evaluated. 3 metrics are developed in [7] for evaluation: utilization, flow fairness, and responsiveness. It is worth to point out that the flow fairness is described by Jain's fairness index [8]:

$$f = \frac{\left(\sum_{i=1}^{k} x_i\right)^2}{k\left(\sum_{i=1}^{k} {x_i}^2\right)} \quad (4)$$

Their results show that there is good potential for DRL to work with TCP based web services, even though more work still needs to be done for protocol agnostic DRL due to a UPD protocol issue in the cloud. As explained in previous section, if DRL was used in Azure CDN, the over charge after enabled CDN in our current Azure storage service would no longer be a concern. Currently, only CoDeeN CDN has unwillingly deployed an ad-hoc DRL solution. We highly recommend all public cloud providers to incorporate a DRL solution in their services.

C. Subscriber's Behaviour Based DRL Solution

The DRL solution makes distributed rate limiters collaborateto enforce a global rate limit across traffic aggregates in multiple sites, which can effectively support the flexibility requested by resource provisioning and accounting in the cloud computing environment.

Based on the DRL solution, we propose a further optimization for cloud DRL – a subscriber's behaviour based DRL solution. If the cloud CDN providerscan leverage the knowledge of subscriber's usage habits and network resource consumption patterns, a better estimation on the traffic demand of the aggregate at each limiter can be achieved, and capacity in proportion to that demand can be more accurately apportioned. Since fairnessbetween flows inside a traffic aggregate depends critically on accurate limiter assignment, this solution can potentially benefit the flow fairness. It also allows individual flows at different limiters to compete with each other for bandwidth in a more intelligent way so that it can further optimize the network bandwidth utilization and the response time. Given the consumption patterns are also changing, this solution should be made adaptive to such change.

VI Conclusions

Based on the study conducted inthis paper, we truly believe that cloud computing will be a new era in Information and Communications Technology industry. Cloud computing brings many benefits to customers and will dramatically increase business agility and reduce operational costs. However, there are many obstacles on the journey to the full adoption of cloud computing. One of the problems is network congestion at cloud egress and latency to client. It is suggested that all cloud providers should evaluate and consider utilizing CDN in cloud computing to resolve this issue.

CDN has been proven a good working solution for integration with cloud computing. Microsoft Windows Azure has adopted CDN technology and demonstrated the benefits. Our Azure CDN experimental results show significant download speed improvement during large content data downloading. Experiments have shown that CDN is a good complimentary service when it is bundled with cloud services. In the future, cloud CDN service could be continuously improved with new technologies, such as DRL and subscriber's behaviour based DRL solution.

However, CDN may have its limitation to benefit all types of cloud applications, for example highly transactional applications. The long term solution would be to increase the network bandwidth with new technologies, and utilize CDN as a transition step. The high speed connectivity from all clients worldwide to Internet could be the most challenging problem in achieving the goal. Recently, a practical solution prototype has been demonstrated to resolve "the last mile", with over 100 time faster speed than current technologies, by a Chinese company.

REFERENCES

[1] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy H.Katz, Andrew Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, Ion Stoica and MateiZaharia, "Above the Clouds: A Berkeley View of Cloud Computing", EECS department, University of California at Berkeley, Technical Report No. UCB/EECS-2009-28, Feb. 2009.

[2] Wayne Jansen, Timothy Grance, "Guidelines on Security and Privacy in Public Cloud Computing", Draft NIST Special Publication, 2011.

[3] John Dilley, Bruce Maggs, Jay Parikh, HaraldProkop, Ramesh Sitaraman and Bill Weihl, "Globally Distributed Content Delivery", *Internet Computing, IEEE*, Volume 6, Issue 5,pp. 81 – 86, Sep./Oct. 2002.

[4] "What is Windows Azure?", Retrieved 21 February 2011 from Microsoft website: http://www.microsoft.com/windowsazure/

[5] "Windows Azure Content Delivery Network", Retrieved 1 March 2011 from Microsoft website: http://www.microsoft.com/windowsazure/cdn/default.aspx

[6] BarathRaghavan, KashiVishwanath, SriramRamabhadran, KennethYocum, and Alex C. Snoeren, "Cloud Control with Distributed Rate Limiting", *ACM SIGCOMME Computer Communication Review*, Volume 37 Issue 4, October 2007.

[7] Ao-Jan Su, David R.Choffnes, AleksandarKuzmanovic, and Fabian E. Bustamante, "Drafting Behind Akamai: Inferring Network Conditions Based on CDN Redirections", *IEEE/ACM Transactions on Networking*, Volume 17 Issue 6, December 2009.

[8] R. Jain, D. Chiu, and W. Hawe, "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems", DEC Research Report TR-301, 1984.

Yale Li is the earliest cloud security expert (CCSK) in the world. As Principle Security Architect, he is currently responsible for data security, applications development security, and emerging market security in SAFE-T (Security Accelerator for Emerging Technologies) team at Microsoft. Yale is also Board Member & Research Director of Cloud Security Alliance - Greater Seattle Area Chapter, Graduate Education Partner in the Center of Information Assurance and Cyber Security at University of Washington, CEO's Advisor of multiple Microsoft partner companies.

As a security professional, Yale has worked on corporate security strategy, security aspects of emerging technologies such as cloud computing, mobile computing, and big data, and emerging markets such as China. In his 14 years at Microsoft, Yale served as Windows source code manager, Tools development lead, and Web Platform chief engineer and other various positions in many Microsoft divisions. He has completed secure cloud adoption, Windows engineering excellence, shared source delivery, and large number of enterprise software development, deployment, and management. Prior to joining Microsoft, Yale was Chief Architect at an IBM company, co-founder/CTO of multiple technology startups, and research assistant to CERN Nobel Prize and Rutherford Medal winners. He is co-author of 《Microsoft 360》 and upcoming 《Cloud 360》.

Yushi Shen received his B.S. degree in electrical engineering from the Tsinghua University, Beijing, P. R. China, in 2001, M.S. and Ph.D. degrees in electrical and computer engineering from the University of California at San Diego (UCSD), La Jolla, in 2003 and 2006, respectively, and MBA degree from University of Washington (UW), Bothell, in 2012. He has worked in Microsoft Inc., Redmond, since 2006, and is currently media platform architect, cloud strategist, and Chair of Chinese Microsoft Employee Network (CHIME). Dr. Shen has published five books and over twenty archived papers in the areas of cloud computing, wireless communications and video processing. He has served as reviewers, associate editors or technical program committee members for numerous international conferences and core journals. He is the organizers or keynote speakers at over 10 international conferences. And he is the recipient of various awards including the Chinese Government Award for Outstanding Students Abroad (2006).

Yudong Liu obtained her PhD in Computer Science from Simon Fraser University (SFU), Canada. Her research interest includes Statistical Natural Language Processing (NLP), Formal Languages, Artificial Intelligence (AI), Machine Learning and Security. She was the only nominee from SFU for Canadian AI Association Doctoral Dissertation Award 2010. She has a series of publications on multiple well-known NLP conferences and has given talks on these conferences, and once won "Best Oral Presentation Award". She is also a co-author of "Fuzzy Semirings with Applications to Automata Theory" (Springer, 2012).

Prior to joining WWU, Yudong worked as a Translation Tool Engineer for SAP, Canada, where she was mainly in charge of the software translation automation.