IaaS cloud computing services for libraries: cloud storage and virtual machines

Yan Han
The University of Arizona Libraries, Tucson, Arizona, USA

Received 2 September 2012
Revised 29 October 2012
Accepted 20 November 2012

Abstract

Purpose – The purpose of this article is to provide an overview of current uses of cloud computing (CC) services in libraries, address a gap identified in integrating cloud storage in IaaS level, and show how to use EC2 tools for easy backup and resource monitoring.

Design/methodology/approach – The article begins a literature review of CC uses in libraries, organized at the SaaS, PaaS and IaaS levels. The author presents his experience of integrating cloud storage services S3 and GCS. In addition, he also shows how to use virtual machine EC2 tools for backup and monitoring resources.

Findings – The article describes a case study of integrating cloud storage using S3 and GCS. S3 can be integrated with any program whether the program runs on cloud or locally, while GCS is only good for applications running on GAE. The limitation of the current GCS approach makes it hard to use for a stand-alone cloud storage. The author also discusses virtual machines using EC2 and its related tools for backup, increase storage, and monitoring service. These services make system administration easier as compared to the traditional approach.

Research limitations/implications – The article presents current CC uses in libraries at the SaaS, PaaS, and IaaS levels. CC services are changing quickly. For example, Google has stated that its APIs are experimental. Readers should be aware of this.

Practical implications – The author shows his experience of integrating cloud storage services. Readers can understand the similarities and differences between S3 and GCS. In addition, readers can learn the advantages and concerns associated with implementing cloud computing. Readers are encouraged to consider questions such as content, skills, costs, and security.

Originality/value – There are many uses of CC services in libraries. However, gaps are identified: in IaaS cloud storage, a few libraries used Amazon S3 and Microsoft Azure, but none explored using Google Cloud Storage (GCS); none provided implementation details, difficulties, and comparisons of S3 and GCS; and a few articles have briefly discussed implementations on Amazon EC2, but have not provided specific details about upgrade and backup. This article addresses those gaps.

Keywords Cloud computing, Google cloud storage, Cloudwatch, Libraries, Online operations, Virtual worlds

Paper type Case study

1. Introduction

Cloud computing (CC) is gaining popularity not only in libraries, but also in other industries. Over the years, CC providers have continued to improve their infrastructure to enhance existing services, and at the same time introduce new services addressing every aspect of computing. For example, Elastic MapReduce for Apache Hadoop, released in 2009, supports large data sets and data-intensive distributed applications. Amazon GovCloud, introduced in August 2011, is specially designed to allow US government agencies using CC for sensitive data to meet compliance requirements. In

OCLC Systems & Services: International digital library perspectives
Vol. 29 No. 2, 2013
pp. 87-100

1065-075X
DOI 10.1108/10650751311319296

June 2012 Google offered Google Compute Engine, allowing users to run large-scale computation on Google's infrastructure. Most recently, Amazon announced Glacier in August 2012 (Amazon, 2012b) as a low-cost archive service in supplement to its cloud storage service S3.

It is necessary to understand what CC is, and its service models. The National Institute of Standards and Technology published the following final definition of cloud computing:

> Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models" (Mell and Gance, 2011).

The three service models are defined as:

(1) Software as a service (SaaS) allows users to use the provider's applications on a cloud through a web browser or an application programming interface (API). The provider manages almost everything in the cloud infrastructure (e.g. physical servers, network, OS, applications). End users can run applications, but do not control the cloud infrastructure (Mell and Grance, 2011). The SaaS primary users are the general public. Gmail, Google Drive, Google Calendar, Windows SkyDrive, and Dropbox are popular SaaS services.

(2) Platform as a service (PaaS) allows users to deploy their own applications on the provider's cloud infrastructure under the provider's environment, such as programming languages, libraries, and tools. The end users can control their own applications, but do not have control of the cloud infrastructure (Mell and Grance, 2011). PaaS is targeted directly to software developers, who develop, test, and run applications on a PaaS platform. Google App Engine (GAE) is a PaaS service.

(3) Infrastructure as a service (IaaS) allows users to control and manage computing resources (e.g. storage, networks, computing power) so that they can deploy and run arbitrary software (Mell and Grance, 2011). Providers only manage underlying physical cloud infrastructure (e.g. physical servers and network). The users have maximum control of the infrastructure, as if they own the underlying physical servers and network. IaaS is primarily targeted at use by enterprises and integration with data and applications. IaaS providers include Amazon, Google, Microsoft, Rackspace and IBM.

2. CC in libraries

CC offerings range from hundreds of SaaS applications and storage services to only a few virtual machine and storage IaaS services offered by big IT companies. Libraries are in the business of creating, managing and delivering information. Almost all libraries have their own websites, maintain integrated library systems (ILS), provide access to digital collections through repositories, and maintain storage and backup of their content and data. A subject search of "Cloud Computing" and "Libraries" in the Library Literature and Information Science Full Text database showed that 80 results were found, of which 40 were peer-reviewed articles published since 2008. These

articles addressed public to academic libraries and concerned topics ranging from websites to repository systems. Libraries have been using SaaS and IaaS services since 2009. An analysis of these articles reveals that libraries are interested in:

- SaaS services (e.g. office applications, Google Doc, calendar, scheduling applications, and cloud storage for synchronizing and backup files) for daily work; and
- IaaS services (e.g. virtual machines and cloud storage) for content delivery such as websites, repositories, and online backup.

Gaps are also identified:

- for IaaS cloud storage, a few libraries used Amazon S3 and Microsoft Azure, but none had explored using Google Cloud Storage (GCS);
- in addition, no articles provided implementation details, difficulties, and comparisons of S3 and GCS; and
- with regard to IaaS virtual machines, a few articles briefly discussed implementations on Amazon EC2, but did not provide specific details about upgrades and backup.

The author feels that these gaps are important to understand the full range of available IaaS services, and also to understand the differences between these services to make an informed decision.

2.1 Uses of SaaS services in libraries

Libraries have been using SaaS services in almost all aspects of library work, from instruction to scheduling to regular office applications. Some SaaS services are built on top of PaaS and/or IaaS platforms. For example, Dropbox uses S3 for its storage. DuraSpace uses cloud storage services such as S3 and Microsoft Azure. The SaaS services used by libraries include Gmail, Google Calendar, Google Docs/Drive, Dropbox, and the OCLC WorldShare Management Service. A literature review shows that libraries used the following SaaS services:

- the Eastern Kentucky University Library used Google Docs and Google Calendar for instruction (Kroski, 2009);
- the District of Columbia Public Library used Google Docs for staff (Tonjes, 2010);
- library discovery systems such as Summon, Primo, and WorldCat Local utilized CC (Breeding, 2011);
- Murray State University used Dropbox for instruction (Bagley, 2011);
- New York City College of Technology used Google Calendar for instruction scheduling and daily work (Leonard, 2011);
- the University of Wisconsin-Eau Claire used Google Forms for reference and instruction (Miller, 2011);
- Union Presbyterian Seminary librarians utilized Ning, a social network site, for summer courses (Deeds *et al.*, 2011);
- the University of Washington used the free web conferencing software Dimdim, but later it became hard to switch due to its acquisition by another company (Gleason, 2011);

- VoiceThread was used for collaboration and library instructions by many institutions (Ditkoff and Young, 2011);
- the Kindura project is using DuraCloud to interact with cloud storage (Waddington *et al.*, 2012);
- Davidson College (Milberg, 2012) and the Australian public library systems (CILIP Update, 2012) presented analysis, evaluation, selection, and/or cost to use the cloud-based OCLC WorldShare Management Service; and
- many libraries are using Google Analytics.

2.2 Uses of PaaS and IaaS services in libraries

The most popular IaaS and PaaS services include Amazon Web Services (AWS) such as EC2 and S3, Google App Engine (GAE), the GCS, Rackspace Cloud Files and Microsoft Azure. A literature review shows that libraries have been using IaaS and PaaS services since 2009, mostly in utilizing virtual machines such as EC2 and cloud storage such as S3 and Microsoft Azure.

EC2 is basically an IaaS scalable virtual machine service. It is the most popular IaaS service used by libraries to host websites, repositories, and integrated library systems (ILS), because it gives users complete control of virtual resources and is easily scalable. Working as a central hub, an EC2 instance can be integrated with other CC services to improve effectiveness and efficiency.

- The University of Arizona Libraries reported using EC2 instances, Linode instances, and GAE for a DSpace repository, an ILS, and the Afghanistan Digital Library's websites (Han, 20011).
- OhioLink implemented AWS for their repository (Kroski, 2009).
- The District of Columbia Public Library utilized EC2 instances to host its website and image archive (Tonjes, 2010).
- The Z. Smith Reynolds Library at Wake Forest University used EC2 for its ILS (Mitchell, 2010).
- The University of Arizona Libraries reported two cases of total costs in running a DSpace instance using EC2. The cost of an EC2 instance was more cost-effective than a traditional server, and using S3 cloud storage might not be cheaper than local storage (Han, 2011).
- Two librarians wrote how to put Koha in the Cloud (Nighswonger and Engard, 2011).
- OCLC offered CC web services for data sharing and reuse (Coombs, 2011).
- The Ohio University system and private colleges built DSpace repositories with Amazon (Davison, 2011).
- Rice University evaluated CC and its usefulness for the library IT department, and it considered which types of project are good candidates for CC and which are not. Glavin and Sun presented their case of using EC2 instances. They found that some projects were better suited to CC than others. Flexibility and cost savings were the best reason to use CC. However, there were also good reasons not to move some projects into CC (Galvin and Sun, 2012).

Cloud storage services include Amazon S3, GCS, Rackspace Cloud Files, and Microsoft Azure. Libraries have been exploring using on-demand cloud storage since 2009. Several cases have been reported to use S3, while one case implemented Rackspace and Microsoft Azure. In comparison, there is no reported use of GCS in libraries:

- the Library of Congress and DuraCloud launched a pilot program to test the use of CC (Library of Congress, 2009);
- Central Connecticut State University Libraries used Amazon S3 for digital preservation (Iglesias and Meesangnil, 2010); and
- DuraCloud stated that it uses Amazon S3, Rackspace Cloud Files, Microsoft Azure, and San Diego Supercomputer Center (SDSC) cloud storage (Branan, 2011).

Besides using commercial providers, several institutions have launched their own private clouds, including:

- Hochschule Furtwangen University in Germany built a private cloud infrastructure, and ran e-learning applications on it. A paper by Doelitzscher *et al.* (2010) concludes that CC gives flexibility and better management of resources.
- The University of Southern California (USC) deployed Nirvanix private cloud storage of 8 petabytes of data for its digital repository (USC Shoah Foundation, 2011).
- SDSC launched the largest academic cloud storage (San Diego Supercomputer Center, 2011).

3. Integrating IaaS cloud storage

3.1 Cloud storage in SaaS and IaaS: one provider, two service offers

Cloud Storage providers are generally large IT companies such as Apple, Amazon, Google, Nirvanix, Rackspace, and Microsoft. Amazon, Google and Microsoft are some well-known cloud storage providers in both the SaaS and IaaS areas. As mentioned before, SaaS services are primarily used by the general public. Popular SaaS cloud storage services are Apple iCloud, Amazon Cloud Drive, Google Drive, DropBox, SugarSync and Microsoft SkyDrive. Some SaaS cloud storage providers are building their services on top of the IaaS providers. For example, both Dropbox and SugarSync use S3 as backup or backend. IaaS cloud storage providers are limited to these large IT companies, offering services such as S3, GCS and Microsoft Azure. In most cases, both IaaS and SaaS cloud storage services provide APIs for accessing data. For example, Google SaaS storage (Google Drive) and its IaaS storage (GCS) have different APIs.

The line between the SaaS and IaaS levels is sometimes fuzzy, but the two levels serve different purposes: IaaS for enterprise and SaaS for individuals. Cloud storage in IaaS is intended for developers to store and access data primarily through an API for enterprise-level applications, while SaaS cloud storage is for individual use and private content sharing. Uses of IaaS cloud storage include online archives, backups, and big data. In comparison, typical uses of SaaS cloud storage are personal file synchronization, share and backup. The above literature review shows that cloud storage in SaaS such as Google Drive and Dropbox are popular in libraries for individual uses in instruction and daily work. Regarding integrating IaaS cloud storage for online storage and archiving, there are only a few cases reported of using

S3, SDSC cloud storage, and Windows Azure, without much detail. Neither reported using GCS, nor provided a comparison of these IaaS services.

3.2 Integrating IaaS cloud storage: Amazon S3 and Google Cloud Storage

Amazon S3 is an IaaS cloud storage service through web services interfaces such as REST and SOAP. It was launched in 2006 and has reduced its prices over the years. Many services such as Dropbox, Ubuntu One, and Posterous use S3, while library uses of S3 include DuraCloud and Central Connecticut State University Libraries. Quint (2008) and Murray (2008) discussed issues and costs when comparing the OCLC Digital Archive with S3, and suggested that it was an apples-to-oranges comparison. Their article claimed that S3 missed critical components of preservation systems such as access control and a content backup/restore facility in functionality, while they agreed that S3 was much cheaper in cost (Quint, 2008; Murray, 2008). Murray also suggested that S3 lacked functions for digital preservation such as fixity checks, format verification and related digital preservation metadata (Murray, 2008).

Amazon claims that data in S3 is secure and that access is highly reliable at 99.99 percent availability. Amazon's Identify and Access Management (IAM) is integrated with these AWS (Amazon, 2008, 2012a). The Central Connecticut State University Library developed a system using S3 for digital preservation storage. They compared the cost of using OCLC Digital Archives with that of S3 for 5 TB data storage, and came to a conclusion that S3's cost is only one third of that of the OCLC Digital Archives (Iglesias and Meesangnil, 2010).

GCS, announced in May 2010, is a late competitor in IaaS cloud storage. Access to GCS can be through the RESTful interface or through its API for GAE (Google, 2012a). In comparison, Nirvanix started its Storage Delivery Network in 2007. In 2011, Nirvanix and USC deployed over 8 petabytes of private cloud storage for the USC digital repository and other units for archiving high-resolution videos and audios (USC Shoah Foundation, 2011). To the author's knowledge, this is the largest private cloud storage deployed for libraries. Due to its higher cost, the author did not consider Nirvanix.

The author and a student developed a digital preservation program called "Archivebox" trying to utilize both S3 and GCS. The software uses S3 for on-demand storage for files, directories and related metadata. The official documentation seems inadequate and was dated 2006. Although sometimes coding with the API is trial-and-error, the data model and its API are very simple, and the integration of functions (e.g. create, upload, and download) is simply a few lines of Java code. The program eventually comes with fewer than one hundred lines of code in the integration of S3. In comparison, the author encountered difficulties in the process of integrating the GCS, and found it is difficult to use GCS without running on GAE. After some research, the author believes that integrating the GCS will take too much effort. Its current API and integration is not as simple as that of S3.

3.3 Similarities

S3 is intentionally built with simplicity in mind. Its data model consists of buckets and objects:

> A bucket is a container for objects stored in Amazon S3. Every object stored in Amazon S3 is contained in a bucket. [...] Within a bucket, you can use any names for your objects, but bucket names must be unique across all of Amazon S3 (Amazon, 2006).

The simplicity of this model makes it flexible in implementing a variety of services such as a traditional hierarchical file/directory structure. On the other hand, it also means that additional APIs or codes have to be implemented either by Amazon, a third-party, or users themselves. In comparison, Google has almost the same data model: buckets "are the basic container [. . .] everything [. . .] must be contained in a bucket". Objects have two components: "object data" and "metadata" (Google, 2012b). Both S3 and GCS share a lot of technical details in common, such as the data model, file naming, and access to APIs.

3.3.1 File and directory structure. S3 and GCS are not considered the same as a conventional hierarchical file structure. In S3 and GCS, every object is contained in a bucket, pretty much the same as the data structure concept "bag". Theoretically this data model is simple, and by nature this is a flat structure. The objects can be logically viewed in linear, hierarchical (traditional storage structure) or even more complex ways. In order to mimic traditional files and directories, one can create file names like "photos/2006/123.jpg" or "photos/2006/124.jpg". These two files are logically stored in the directory "photos/2006". Our program, S3 and GCS browser tools can display files and directories in this traditional view.

3.3.2 Bucket's name. A bucket's name must be unique. The author designed a consistent way to name a bucket using the system's time stamp.

3.3.3 API. Both S3 and GCS have a simple API for integration. The program's code for integration with S3 is fewer than 100 lines. Coding and testing time is less than ten hours.

3.3.4 Web interface. S3's web interface is straightforward and suitable for manual operations. One can create buckets and upload files, but with limitations on uploading directories. In other words, the current "Upload" command (see Figure 1) cannot upload any directory. An "Enhanced Uploader (BETA)" tool resolves this issue, but requires an "enabled Java applet" to run. The GCS manager is the web interface, which is very similar to that of S3.

3.4 Differences: limitations

The author believes that there is a significant difference between S3 and GCS in their approaches to integration with programs. That is, S3 can be integrated with any

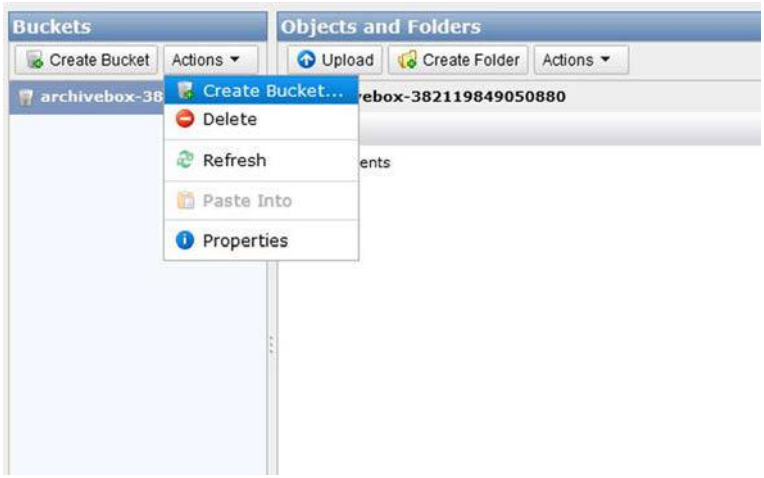

Figure 1.
Amazon S3 web interface

program whether the program runs on cloud or on local, while GCS is difficult to use for applications that are not running on GAE. In the above case, the author's application is required to read and analyze local files so that it can generate file identification information. Therefore, running it on a local machine is the most effective and efficient way. The application integrated with S3 with Amazon AWS SDK wrapped inside. In contrast, Google claims that the GCS API is "experimental, innovative and rapidly improving" (Google, 2012a). Given the same technical environment, the author finds that it is very difficult to integrate with GCS. At present, the author feels that integration with GCS requires additional work and that future code updates can be expected. In the case of implementing the author's program, integration with S3 is straightforward and it takes a day or two to develop and test the code. This reveals a huge limitation in using GCS as an on-demand cloud storage service.

The current GCS implementation is more suitable for programs running on the GAE platform. GAE is a PaaS platform, which is intended to make it easy to write applications without worry about managing the underlying OS, networking, and programming languages. However, this is a huge pitfall for programs that cannot be run on it. It is identified that GAE has at least the following limitations:

- It currently supports Java, Python and Go, although Google plans to support more languages in the future.
- the database is GQL, which does not have SQL's join statement (Han, 2011), though Google Cloud SQL can be used. This design has its advantages, but the drawbacks are also obvious. As a result, only a range of applications can be run on GAE, or they have to be modified.

4. Uses of IaaS virtual machine with Amazon EC2 and CloudWatch

EC2 is the central piece of Amazon AWS, which can be operated independently or integrated with other AWS services. Multiple articles have mentioned that it allowed a library to rebuild, upgrade, and backup web applications quickly. In comparison, rebuilding and/or upgrading traditional servers and/or applications can be cumbersome and time-consuming.

An EC2 instance comes with standardized resources. For example, a small instance has one EC2 compute unit, 1.7 GB memory and 160 GB storage. For a typical library repository, such an instance has enough computing power to handle incoming requests, but may not have enough storage for growing digital collections. The Afghanistan digital collections repository using DSpace (see www.afghandata.org) uses an EC2 small instance, hosting 1,800 titles of 200,000 images in 2010. Since then, an additional 1,200 titles of 150,000 digitized images have been added. As a result, the basic storage of a small EC2 instance is not enough and additional storage must be added. Typical system administration of a server involves regular backup and increasing storage from time to time. The following shows how to perform these operations, which are easy-to-use and can be performed much more quickly and easily than managing a traditional server.

4.1 Increase storage

Because CC is by its nature an on-demand service, users can buy just enough resources (e.g. CPU, memory, and storage) for their planned use. Additional resources, such as storage, can be added at a later time when needed. In AWS, one can use "Create

Volume" to create a new Elastic Block Store (EBS), which can be any reasonable size. A new EBS volume is unformatted block storage. When attached to an EC2 instance, it shows up as a device. Then a file system in a volume can be created and mounted to the instance as storage. This volume can be backed up independently, increased at a later time, detached, and re-attached to another instance if needed, without the creation of any downtime. These operations are straightforward, and very easy to do. This feature is very useful for website and repository systems to avoid downtime.

4.2 Back up instance and creation of a machine image

Backups are necessary and critical for recovery and data review. Their primary purpose is to recover data in the case of system failure and data loss. Sometimes users might want to examine and review data from an earlier time for business purposes. Why is backup important? A survey of 2,205 US adults revealed that 54 percent of adults personally have and/or know someone who has lost files. Nineteen percent of men and 30 percent of women do not back up (Seagate, 2012). Backup methods can be full or incremental. A full backup aims to record the state of a server, and is generally completed with system imaging backup software, while an incremental backup is records changes since the last incremental backup. An incremental backup is usually desirable for saving storage space without backing up duplicated files. On a locally managed server, one must use open source or proprietary software to create a full backup. For incremental backup, database data can be dumped and files can be synchronized. Both backups require staff time and computing resources, i.e. the installation, maintenance and operation of backup software and storage spaces, preferably located in a remote facility. As a result, backup can be time-consuming and expensive to run over time.

The EC2 web interface provides an easier way. For a full backup, one can use an existing feature to create an image of an existing instance. In AWS this is called Amazon Machine Image (AMI), which fully captures an existing EC2 instance. An AMI can be only accessed by certain users, or can be shared with desired users. For an incremental backup, one can synchronize files with a different EBS. Creating backups is just a few mouse clicks with the AWS web interface, or a program can be developed to create an automatic backup schedule. In order to restore and recover the system, one can re-launch an instance from an existing image in a few seconds (see Figure 2).

4.3 Monitoring a virtual machine using CloudWatch

Consistently high CPU usage, low memory, and excessive IO usage are important indicators for server/instance overload, meaning that the current instance struggles to handle IO requests and tasks. If the situation persists, it is a generally good practice to upgrade the server/instance. In the past, the author configured and maintained system monitoring tools and services such as Nagios. They are critical tools for system administration, but this traditional approach requires staff time and resources for maintenance. Amazon CloudWatch is a cloud monitoring service for AWS resources. The basic monitoring service is free of charge and watches EC2's resources such as CPU utilization and disk read/write. A similar product is Rackspace's monitoring service. Compared to their traditional counterparts, these monitor services save administrators' time and resources. To enable the basic monitoring service for an EC2 instance, one can set up alarms for the instance's health, such as CPU usage, disk

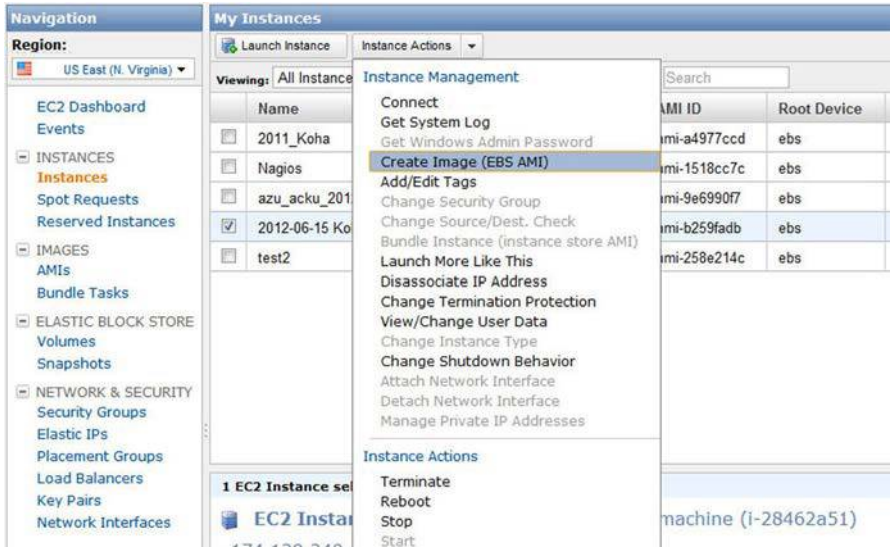

Figure 2.
Create Image in the Amazon EC2 web interface

read/write and network status (see Figure 3). The author has set up two alarms to monitor an EC2 instance running a DSpace repository. One alarm was set up to notify of high CPU usage (e.g. CPU usage is over 50 percent), while the other is used to warn of system failure (e.g. system check failed).

5. Discussion

The author has been using CC services at all levels over the past few years. His experience and the above literature review show that CC has advantages:

- On-demand – Libraries are no longer required to plan ahead for IT resources. Large-scale computing power and Petabyte storage can be acquired at any time. For example, DuraCloud and the Central Connecticut State University Libraries just purchase required storage as they go, and do not need to plan ahead.

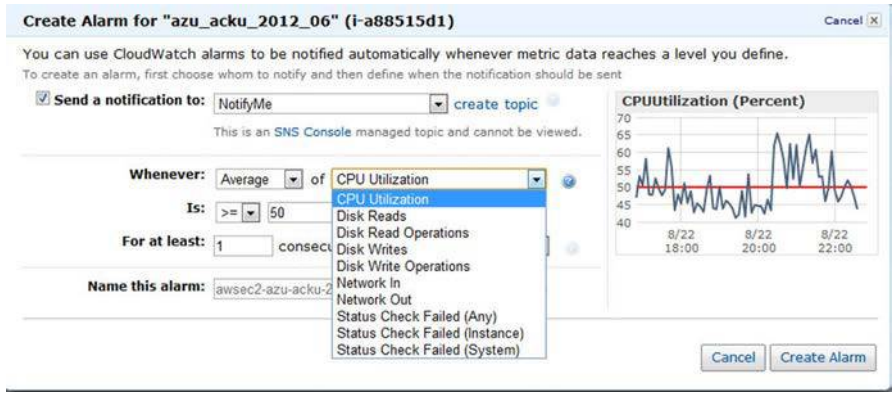

Figure 3.
Create An Alarm in Amazon CloudWatch

- Cost-effective – Some of SaaS services are free of charge, while IaaS services are billed by the resources used. Multiple articles show that CC in general is more cost-effective than locally hosted service. Users are encouraged to consider the total costs of ownership.
- Scalability – Without CC, Mike Nilson would have not been able to run 20 EC2 instances simultaneously for his short 40-hour project (Nielsen, 2012). Without CC, the Institute of Systems Biology would have not been able to analyze cancer data sets in a short time (Google, 2012c). CC opens a new way for scientists to work more effectively.
- On-demand cloud storage can be acquired at any time. With many services to choose from, libraries are no longer limited to the boundaries of their own IT resources.
- Convenient – As mentioned above, increasing storage and the creation of a new instance can be completed easily without the creation of downtime. Library users will be happy not to see an e-mail informing them of system downtime.
- CC providers provide an easy way to monitor costs, as the services are billed by the resources used. The author is able to monitor and control the costs.
- High availability – Large IT companies have tremendous technical and financial resources. There is occasional downtime, but the availability of their CC services is much higher than that of locally maintained servers. The author's experience shows that the availability of CC services is very high.

Readers should not forget that security is a major concern. There are a few known incidents of security concerns. For example, in 2011 Dropbox had a four-hour time period during which users could log in with any password (Singel, 2011). Eranki, head of Dropbox Server Engineering, stated that "I think a lot of services (even banks) have serious security problems [. . .] so figure it out if it really is important to you [. . .] before you go and lock down everything" (Eranki, 2012).

Not all CC services are created equal, even though they might provide similar services. For example, S3 is better than GCS if a program running locally requires on-demand cloud storage. If you are developing a program solely on GAE, then GCS is obviously a better choice. The EC2 virtual machine comes with useful tools such as the monitoring service and imaging creation tools. This gives it an advantage over other IaaS virtual machine services.

For users who are considering CC for individual uses and/or enterprise integration, the author suggests considering the following:

- *Content* – What is the data? How much? Is the data sensitive? How long will you keep the data? For huge amount of data for short-term usage, CC is probably the way to go. For sensitive data, one must follow policies and perform encryption to secure it in the cloud.
- *Users* – Who are the end users? If individual use for daily office routines, consider SaaS-level services.
- *Skills* – Integration of PaaS and IaaS requires programming and database administration skills, although system administration skills have been removed. Technical skills and ongoing support are still required for integration.

- *Costs* – CC services remove startup costs for investing in physical IT equipment, but there are ongoing bills. Readers should be aware of the total costs. What are the total costs of ownership? Will running CC save money?
- *Security and privacy* – Libraries have a role to keep sensitive data safe as circulation records secure. Some technical and legal steps are suggested. Encryption of sensitive data and appropriate clauses in contracts with CC providers are good practices to keep data safe (Moulaison and Corrado, 2011).

6. Summary

The article began with NIST's definition to understand the three levels of CC. The literature review has shown that there are many articles discussing CC in libraries in terms of implications, potential uses, case studies, and costs. The author has analyzed these published articles and has shown how libraries utilize CC services at the three levels (i.e. SaaS, PaaS and IaaS). The author has also found that few studies mentioned integration with S3 and Microsoft Azure cloud storage. No studies provided detailed implementations or mentioned Google's GCS. This article describes a case study of integrating cloud storage using S3 and GCS. S3 can be integrated with any program whether the program runs on cloud or locally, while GCS is only good for applications running on GAE. The limitations of the current GCS approach make it hard to use for stand-alone cloud storage.

The author also discussed virtual machines using EC2 and its related tools for backup, increase storage, and monitoring service. These services make system administration easier as compared to the traditional approach. The advantages of CC are discussed by showing case studies from libraries. The security of data in the cloud is a great concern, and suggestions on this matter are proposed. Finally, some questions are given for readers to consider before using a CC service.

References

Amazon (2006), "Working with Amazon S3 buckets", available at: http://docs.amazonwebservices.com/AmazonS3/latest/dev/UsingBucket.html (accessed August 28, 2012).

Amazon (2008), "Amazon Web Services: overview of security processes", available at: http://aws.amazon.com/articles/1697?_encoding=UTF8&jiveRedirect=1 (accessed August 20, 2012).

Amazon (2012a), "Amazon Simple Storage Service (Amazon S3)", available at: http://aws.amazon.com/s3/ (accessed August 20, 2012).

Amazon (2012b), "Amazon Glacier overview", available at: http://aws.amazon.com/glacier/ (accessed August 31, 2012).

Bagley, C.A. (2011), "Parting the clouds: use of Dropbox by embedded librarians", in Corrado, E.M. and Moulaison, H.L. (Eds), *Getting Started with Cloud Computing: A LITA Guide*, Neal-Schuman, New York, NY.

Branan, B. (2011), "DuraCloud: a technical overview", available at: www.slideshare.net/DuraSpace/dura-cloud-technicaloverview201111finis1 (accessed October 22, 2012).

Breeding, M. (2011), "Library discovery services: from the ground to the cloud", in Corrado, E.M. and Moulaison, H.L. (Eds), *Getting Started with Cloud Computing: A LITA Guide*, Neal-Schuman, New York, NY.

CILIP Update (2012), "Cloud management system put to test", *CILIP Update*, Vol. 11 No. 8, p. 18.

Coombs, K.A. (2011), "Leveraging OCLC cooperative library data in the cloud via web services", in Corrado, E.M. and Moulaison, H.L. (Eds), *Getting Started with Cloud Computing: A LITA Guide*, Neal-Schuman, New York, NY.

Davison, J. (2011), "Building push-button respositories in the cloud with DSpace and Amazon web services", in Corrado, E.M. and Moulaison, H.L. (Eds), *Getting Started with Cloud Computing: A LITA Guide*, Neal-Schuman, New York, NY.

Deeds, L.R., Kisselo-lto, C. and Knox, A.T. (2011), "Ning: fostering conversations in the cloud", in Corrado, E.M. and Moulaison, H.L. (Eds), *Getting Started with Cloud Computing: A LITA Guide*, Neal-Schuman, New York, NY.

Ditkoff, J. and Young, K. (2011), "Speak up! Using VoiceThread to encourage participation and collaboration in library instruction", in Corrado, E.M. and Moulaison, H.L. (Eds), *Getting Started with Cloud Computing: A LITA Guide*, Neal-Schuman, New York, NY.

Doelitzscher, F., Sulistio, A., Reich, C., Kuijs, H. and Wolf, D. (2010), "Private cloud for collaboration and e-learning services: from IaaS to SaaS", *Computing*, Vol. 91 No. 1, pp. 23-42.

Eranki, R. (2012), "Scaling lessons learned at Dropbox, part 1: the security-convenience tradeoff", available at: http://eranki.tumblr.com/ (accessed October 10, 2012).

Galvin, D. and Sun, M. (2012), "Avoiding the death zone: choosing and running a library project in the cloud", *Library Hi Tech*, Vol. 30 No. 3, pp. 418-427.

Gleason, A.W. (2011), "Not every cloud has a silver lining: using a cloud application may not always be the best solution", in Corrado, E.M. and Moulaison, H.L. (Eds), *Getting Started with Cloud Computing: A LITA Guide*, Neal-Schuman, New York. NY.

Google (2012a), "Google Cloud Storage API overview", available at: https://developers.google.com/storage/docs/developer-guide (accessed October 10, 2012).

Google (2012b), "Google Cloud Storage Java API overview", available at: https://developers.google.com/appengine/docs/java/googlestorage/overview?hl=en (accessed October 20, 2012).

Google (2012c), "Cancer investigators use Google Compute engine to accelerate life-saving research", available at: https://cloud.google.com/files/ComputeISBCaseStudy.pdf (accessed October 20, 2012).

Han, Y. (2011), "Cloud computing: case studies and total cost of ownership", *Information Technology and Libraries*, Vol. 30 No. 4.

Iglesias, E. and Meesangnil, W. (2010), "Amazon S3 in digital preservation in a mid-sized academic library: a case study of CCSU ERIS digital archive system", *The Code4Lib Journal*, Issue 12, available at: http://journal.code4lib.org/articles/4468 (accessed January 5, 2011).

Kroski, E. (2009), "Library Cloud Atlas: a guide to cloud computing and storage", Library Journal, available at: www.libraryjournal.com/article/CA6695772.html (accessed August 1, 2012).

Leonard, A. (2011), "From the cloud, a clear solution: how one academic library uses Google Calendar", in Corrado, E.M. and Moulaison, H.L. (Eds), *Getting Started with Cloud Computing: A LITA Guide*, Neal-Schuman, New York, NY.

Library of Congress (2009), "Library of Congress and DuraCloud launch pilot program using cloud technologies to test perpetual access to digital content", available at: www.loc.gov/today/pr/2009/09-140.html.

Mell, P. and Gance, T. (2011), "The NIST definition of cloud computing", available at: http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

Milberg, C.I. (2012), "A tale of two systems: a case study on the implementation of two discovery systems at Davidson College", *College and Undergraduate Libraries*, Vol. 19 Nos 2-4, pp. 264-277.

Miller, R.E. (2011), "Integrating Google Forms into reference and instruction", in Corrado, E.M. and Moulaison, H.L. (Eds), *Getting Started with Cloud Computing: A LITA Guide*, Neal-Schuman, New York, NY.

Mitchell, E. (2010), "Using cloud services for library IT infrastructure", *The Code4lib Journal*, Issue 9, available at: http://journal.code4lib.org/articles/2510/

Moulaison, H.L. and Corrado, E.M. (2011), "Perspectives on cloud computing in libraries", *Getting Started with Cloud Computing: A LITA Guide*, Neal-Schuman, New York, NY.

Murray, P. (2008), "Long-term preservation storage: OCLC Digital Archive versus Amazon S3", Disruptive Library Technology Jester, available at: http://dltj.org/article/oclc-digital-archive-vs-amazon-s3/

Nighswonger, C.R. and Engard, N.C. (2011), "Koha in the cloud", in Corrado, E.M. and Moulaison, H.L. (Eds), *Getting Started with Cloud Computing: A LITA Guide*, Neal-Schuman, New York, NY.

Nilsen, M. (2012), "How to crawl a quarter billion webpages in 40 hours", available at: www.michaelnielsen.org/ddi/ (accessed October 20, 2012).

Quint, B. (2008), "OCLC introduces high-priced digital archive service", Information Today, available at: http://newsbreaks.infotoday.com/nbReader.asp?ArticleId=49018.

San Diego Supercomputer Center (2011), "SDSC announces scalable, high-performance data storage cloud", available at: www.sdsc.edu/News%20Items/PR092211_sdsccloud.html (accessed October 20, 2012).

Seagate (2012), "Seagate reinvents backup for your digital life", available at: www.seagate.com/about/newsroom/press-releases/backup_plus_launch_master_pr/ (accessed August 22, 2012).

Singel, R. (2011), "Dropbox left user accounts unlocked for 4 hours Sunday", The Wired, available at: www.wired.com/threatlevel/2011/06/dropbox/ (accessed October 20, 2012).

Tonjes, C. (2010), "Cloud computing at DCPL", LITA Cloud Computing Session, available at: www.slideshare.net/ctonjes/chris-tonjes-cloud-computing

USC Shoah Foundation (2011), "Institute's archive now in the Cloud", Institute News, available at: http://dornsife.usc.edu/vhi/news/3294 (accessed October 10, 2012).

Waddington, S., Zhang, J., Knight, G., Hedges, M., Jensen, J. and Downing, R. (2012), "Kindura: repository services for researchers based on hybrid clouds", *Journal of Digital Information*, Vol. 13 No. 1, available at: https://journals.tdl.org/jodi/article/viewFile/5877/5887 (accessed August 30, 2012)

Further reading

DuraCloud (2011), "DuraCloud Dictionary", available at: www.duracloud.org/duracloud_dictionary

Kincaid, J. (2011), "Dropbox security bug made passwords optional for four hours", available at: http://techcrunch.com/2011/06/20/dropbox-security-bug-made-passwords-optional-for-four-hours/ (accessed October 10, 2012).

Corresponding author

Yan Han can be contacted at: hany@u.library.arizona.edu