# Cloud-enabled Digital Information Service Framework

**Ahmet F. Mustacoglu[1,*], Ahmet E. Topcu[2], Ferhat O. Catak[1], Geoffrey C. Fox[3,4]**
[1]TUBITAK BILGEM – Informatics and Information Security Research Center, TURKEY
[2]Yildirim Beyazit University, TURKEY
[3]School of Informatics and Computing, Indiana University, Bloomington, Indiana, USA
[4]Community Grids Lab, Indiana University, Bloomington, Indiana, USA
[E-mails: afatih.mustacoglu@tubitak.gov.tr, aetopcu@ybu.edu.tr, ozgur.catak@tubitak.gov.tr, gcf@indiana.edu]

*Corresponding author: Ahmet F. Mustacoglu

**ABSTRACT:**

There are several important trend driving computing. We have the Data Deluge from Commercial (e.g. Amazon, e-commerce), Community (e.g. Facebook, Search), and Scientific applications (e.g. Analysis of LHC data, Genomics). We have light weight clients from smartphones, tablets to sensors. Clouds with cheaper, greener, easier to use IT for (some) applications are growing in importance. They enable the lightweight clients by acting as a backend resource and answer the difficult question "what do we do all with all those cores on a chip". As that's not so easy to answer on a conventional client, this is one driver to lighter weight client but on a server, each core can host a separate cloud service. These developments drive both research and education and will weave together as we look at data analysis in the clouds. We explore the use of cloud computing in the Digital Information Service framework and investigate the performance metrics by migrating its services from a cluster-based environment to the Amazon Public Cloud. The Digital Information Service consists of tools and web services for supporting Cyberinfrastructure based scientific research. This system supports a number of existing online Web 2.0 research tools (social bookmarking, academic search, journal and conference content management systems) and aims to develop added-value community building tools. We introduce a real life practical application of our proposed framework running on the Amazon Cloud and present its evaluation. As the results indicate, the cloud implementation can perform well enough and achieves federation and unification of digital entities.

## Keywords:

*Cloud Computing, Amazon Cloud, Information Retrieval and Management, Federation and Unifications.*

## 1. Introduction

Cloud computing has become one of the most powerful and popular technology that provides access to a shared pool of computing resources that can be rapidly provisioned and released

with minimal management effort [1]. Clouds offer improved functionality and better cost-performance than traditional approaches in many areas of scientific research, computational science and engineering [2]. Many of these opportunities have not been explored in depth as there is currently no viable business model as clouds charged as operating funds (bearing overhead) must compete with no-cost resources available through universities and federal initiatives. On general principles one can expect clouds to be the most economical computing resource as they offer economies of scale (one has around 100,000 servers in a large cloud data center) and their internet access model can allow cloud centers to be placed in optimal locations where operating costs are low and environmental impact is minimal. Of course current national supercomputer resources operate near 100% utilization (whereas clouds typically operate below full utilization allowing an attractive interactive model) and often are directly or indirectly subsidized by the host organization and this obscures the comparison of cloud and traditional scientific computing approaches.

Clouds offer interesting opportunities as both infrastructure (IaaS) and software (PaaS) levels. Their software model has been developed for the largest scale data intensive applications in the e-commerce, social media and search arenas. These have been reinforced by the commercial cloud focus as general next generation enterprise data center technology. Comparing clouds, grids (distributed systems) and supercomputers, clouds have synchronization and communication costs that lie between those of distributed systems and supercomputers. Further clouds tend to be optimized for external access and not for inter node communication performance [3]. Thus highly parallel large scale simulations are not likely to move to clouds in the near future and should remain staple of traditional supercomputers. However there are two important classes of applications where clouds could perform well and offer attractive cost-performance, interactive elastic (on demand) use and powerful new software platforms. These classes are;

- Pleasingly parallel applications and with some overlap
- Data-intensive applications

Clouds offer an interesting high throughput computing model for the pleasingly parallel case where there are two important cases – namely parallelism over users and usages. The former is illustrated by the many users of a Web 2.0 site in commercial applications and by support of the "long tail of science" (the many small users with individual jobs) in scientific case. The success of the European Venus-C project on the Azure cloud is a good example here. Parallelism of usages could be illustrated by particle physics data analysis (each event set can be analyzed independently) or the support of Sensor nets or more generally the "Internet of Things" where over 20 billion devices are predicted on the Internet by 2020; each sensor is naturally connected elastically (as individual sensors such as smart phones do not have 100% duty cycle) to a core in the cloud.

This paper contains the practical utilization of the Digital Information Service framework in a real life scenario improving the previous work deployed on a cluster based environment [4]. The purpose of this empirical evaluation is to put our theoretical research into practice in order to evaluate and validate its utility in cloud environment. The literature fails to report on empirical case studies of a cloud based Digital Information Service framework designed

for unifying and federating different implementations of research tools for scholarly publications, so the main novelty of this paper is that it introduces a real life cloud-based practical application of our proposed framework. Along with the description of the architecture, this paper also contains the empirical evaluation of the framework's core services, analyzing its usefulness in a real situation. This study should inspire the design of other cloud-enabled information systems along with similar metadata management requirements.

The organization of the rest of the paper is as follows. Section 2 gives an overview about the cloud computing concept. Section 3 explains the architecture of the proposed system. Section 4 presents the evaluation test results for the prototype system running on Amazon Public Cloud. Last, we conclude with some final remarks and future work in Section 5.

## 2. Background

As explain in [1] cloud consists of five major characteristics, three service models and four deployment models. Cloud computing is an emerging technology one of whose main focuses is the on-demand service, measured service, scalable elastic service, broad any-time any-where network access, pooling of resources leading to economies of scale in performance and electrical power (Green IT) [5, 6]. These correspond to Infrastructure as a Service but there are also powerful new software models corresponding to Platform as a Service and Software as a Service that are also important [7]. Cloud technology provides powerful architectures to perform complex large-scale computing tasks and makes available various IT functions from storage and computation to database and application services. Many organizations have attracted by the cloud computing due to the economic and usage advantages to store, process and analyze large amounts of datasets [8]. A big number of scientific applications and the data have been migrated to the cloud environment due to the lack of resources in local computing environments, higher costs and increasing volume of data [9]. Also, cloud service providers have started to include parallel data processing frameworks to allow their users to have the related services during the deployment of their programs [10].

"Cloud computing is a model for ubiquitous, convenient, on-demand access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction" [1]. Cloud computing has various attractive features that allows organizations to focus on their core business rather than worrying about the issues such as infrastructure, various costs, flexibility, availability of resources and maintenance [11]. Moreover, the elastic environment, resources and the services provided by the cloud are excellent opportunities for scientists to perform their experiments [12, 13]. Cloud service models can be categorized into three groups:

- *Platform as a Service (PaaS):* In this models, cloud providers offer a computing platform such as operating system, database, code execution environments, application server and web server etc. Application developers can easily develop, compile and run their software solutions on a cloud platform without worrying about the cost and complexity of the underlying hardware and software layers. Google's

AppsEngine, Salesforce.com, Force platform, and Microsoft Azure are the popular examples to PaaS for end users.

- *Software as a Service (SaaS):* In this model, cloud providers offer application software that is installed and operated by the cloud provider in the cloud environment so that users can access the software from cloud client. Cloud users do not need to worry about managing the cloud infrastructure or platform where the software application is run. SaaS model is sometimes called as "on-demand software" due to the nature of pay-per-use or subscription fee based pricing policy. GoogleDocs, Gmail, Salesforce.com, and Online Payroll can be accessed through the Internet and can be given as examples to SaaS category [14].
- *Infrastructure as a Service (IaaS):* In this model, cloud providers offer physical machines, virtual machines or other resources so that users are abstracted from the detail of infrastructure such as physical computing resources, location, data partitioning, load balancing, security, backup and network etc. Flexi scale and Amazon's EC2 can be consumed by end users upon demand and can be given as examples to IaaS category.

We have developed Internet Documentation and Integration of Metadata (IDIOM) that is prototype of the proposed Digital Information Services Framework and deployed it on the Amazon Public Cloud (EC2). During the deployment of our prototype system on the cloud environment, we have used major services provided by Amazon Public Cloud such as database, web server, load balancers, execution runtime etc. Our prototype system running on Amazon Cloud [15] is offered to the end user as a SaaS model via cloud clients through the internet.

## 3. The Architecture of the Cloud-enabled Digital Information Services

The Digital Information Service forms an add-on architecture that interacts with the various social networking tools and unifies them in a higher-level system. In other words, it provides a unifying architecture, where one can assemble metadata instances of different web-based information services. The Digital Information Service framework achieves unification and federation of the major academic publication management tool implementations such as Delicious [16], Citeulike [17] etc. and support their communication protocols. Furthermore, the prototype implementation also supports ability to use major academic search tools (Microsoft Academic Search [18] and Google Scholar [19] etc.) to collect metadata and store them into a local system. The Digital Information Service achieves information federation by utilizing a global schema called Merged Schema. The merged schema consists of annotation tools' schemas, academic search tools' schemas, Dublin Core Metadata Initiative schemas and BibTex schemas. With these capabilities, the proposed Digital Information Service enables implementations of different digital metadata management and academic search tools to interact with each other and to share each other's metadata [4]. We have followed Web 2.0 design patterns [20] in designing the IDIOM prototype implementation of the Digital Information Service Framework running on Amazon Public Cloud [15]. Below, we list these patterns and discuss how they were applied in designing the IDIOM system:

*Delivering services, not packaged software:* The IDIOM is a collection of tools and services that can be accessed over the Web (either through a user interface or programmatically through Web services). It will evolve by introducing new features; still its users won't have to install new versions of the software.
*Producing hard-to-recreate data that gets richer as more people use the system:* By combining data from a variety of sources, the IDIOM will create added-value data and metadata generated with specific communities in mind.

*Harnessing collective intelligence:* Through its integration with the social bookmarking tools, the IDIOM can leverage data and metadata from a large number of researchers. Moreover, the system can handle both individual users and groups of users, and supports sharing and collaboration between group members.

*Leveraging the long tail through customer self-service:* The term "long tail" here refers to the concept formulated by Anderson [21] that non-hit products can collectively make up a market share that may exceed the relatively few current hits, bestsellers or blockbusters, provided the store or distribution channel is large enough (this business model is leveraged for example by Netflix or Amazon.com). The IDIOM aims to support research communities, such as the members of a research project, a group interested in a particular chemical compound and so on, by allowing them to create system accounts and to use the community-building tools for their specific usage scenarios.

*Software above the level of a single device:* Currently, the IDIOM user interface runs in a browser. However, because of its layered design and the use of J2EE technology, system front-ends for other devices, such as PDAs, can be developed at low cost.

Figure 1 shows the overall architecture of the IDIOM system running on Amazon Public Cloud [15]. This system consists of three main component: (a) the cloud clients; (b) the online resources; and (c) the IDIOM system running on Amazon Public Cloud. The cloud clients can be any clients such as smart phones, tablets, and laptop PCs etc. that interact with the system over the HTTP protocol. The online resources represent data sources located on the web such as repositories, social bookmarking and annotation tools, scientific databases etc. Finally, the IDIOM system is a collection of services for managing social data scattered on the internet. During the deployment phase of the IDIOM system on the Amazon Cloud, the properties of the Amazon Public Cloud have been utilized and its properties are described in detail:

*The hardware layer:* This layer is responsible for managing the physical resources of Amazon Public Cloud such as physical servers, routers, switches, power and cooling systems. In real life, data centers are the places where the hardware layer is typically implemented in. A data center generally contains around thousands of servers that are organized in racks and interconnected through switches, routers or other components. Hardware configuration, fault tolerance, traffic management, power and cooling resource management are the typical issues of the hardware level.

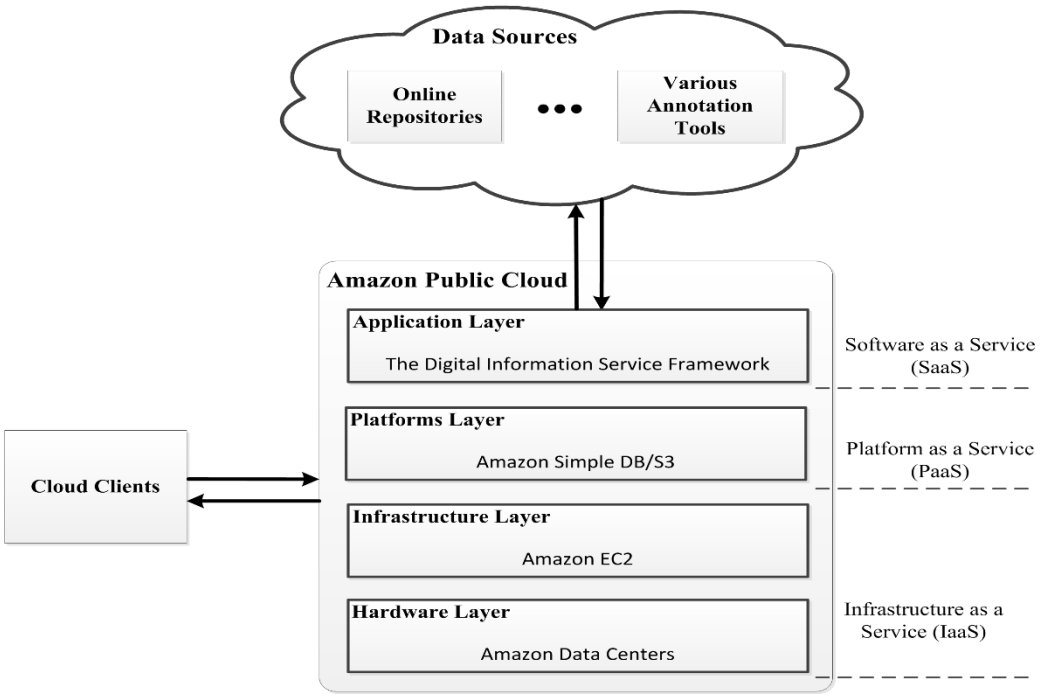


**Figure 1.** Architecture of the Cloud-enabled Digital Information Service Framework

*The infrastructure layer:* The infrastructure layer is also known as the virtualization layer, this layer generates a pool of storage and computing resources by partitioning the physical resources by using virtualization technologies such as Xen [22], KVM [23] and VMware [24]. This layer is a crucial component of cloud computing paradigm due to many key features are only made available through virtualization technologies such as dynamic resource assignment etc. Amazon EC2 service has been utilized for the deployment of the IDIOM services.

*The platform layer:* The platform layer is built on top of the infrastructure layer and it consists of operating systems and application frameworks. The main purpose of this layer is to minimize the burden of deploying applications directly into VM containers. For instance, Google App Engine operates on the platform layer to provide API support for implementing database, storage and business logic of typical web applications. Amazon Simple DB service has been used for satisfying the storage needs of the IDIOM services.

*The application layer:* The application layer is located at the highest level of the hierarchy and it consists of the actual cloud applications. Cloud applications can leverage the automatic-scaling feature to achieve better performance, availability and lower operating cost when compared to traditional applications. The IDIOM services has been deployed as a SaaS model on the Application Layer of the Amazon Cloud aiming to benefit from the automating-scaling feature.

## 4. The Evaluation of the Proposed System

We performed extensive series of measurements to evaluate the prototype implementation of the proposed architecture and investigate its practical usefulness in real life applications. We can run our client programs on several environments such as smart phones, tablets and PCs to reach IDIOM services and we have deployed our cloud-enabled IDIOM services on Amazon Public Cloud. We tested the IDIOM system running on Amazon Public Cloud deployed on Linux machines through EC2 services. Furthermore, we have also used Amazon Public Cloud Simple DB/S3 for our storage needs.

In our general experiments methodology, we have sent various requests from a client program to our proposed system implementation running on Amazon Public Cloud to test the performance metrics of our proposed system.

## 4.1. System Responsiveness Experiments

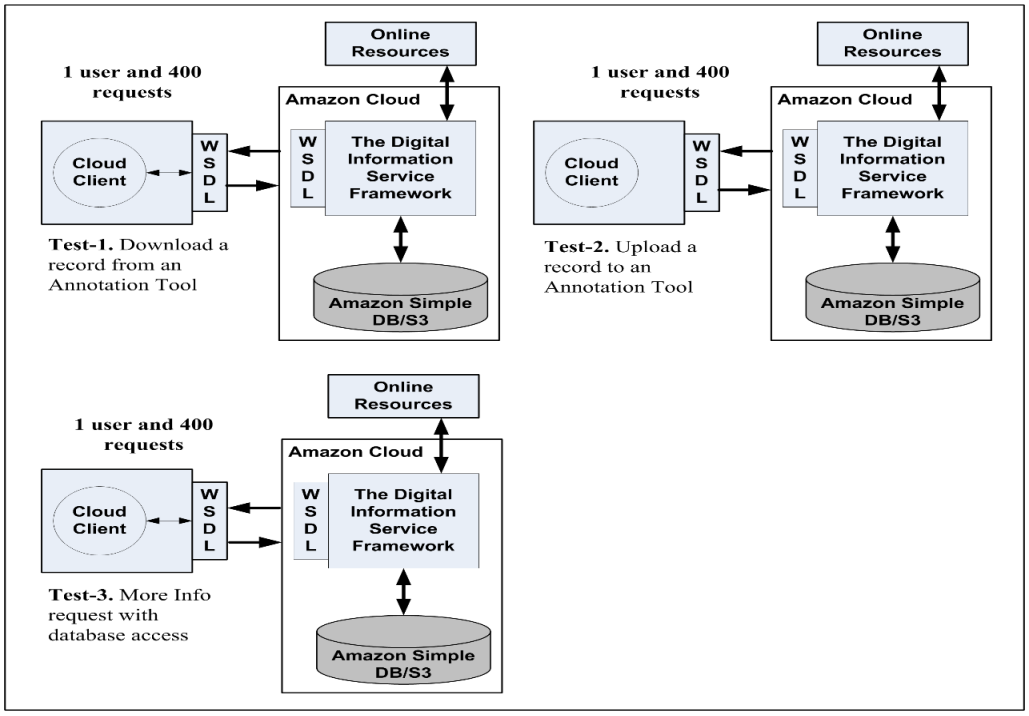


**Figure 2.** Testing Cases for System Responsiveness Experiment

Our main goal in doing this experiment is to measure the baseline performance of the IDIOM prototype implementation deployed on Amazon Public Cloud. We have tested the performance of the proposed system by measuring the times necessary to *download* a record from an annotation tool into Amazon Cloud repository, to *upload* a new record from Amazon Cloud repository to an annotation tool. Furthermore, we have also tested the *More Info* functionality to retrieve the metadata of a specified record forming a digital record from Amazon Cloud repository. The client programs were run on a personal laptop, while cloud-enabled IDIOM system was running on Amazon Cloud. In this experiment, we were exploring the performance metrics of our methodology for "*download*", "*upload*" and "*more*

*info*" services of the proposed system. We have conducted the following test cases: a) A cloud client sends a request to download records from an annotation tool required to access to Amazon Cloud database; b) A cloud client sends a request to upload a record(s) from Amazon Cloud database to an annotation tool; and c) A cloud client sends a request to get a more info on a digital entity from Amazon Cloud repository. In our each testing case, the clients send 400 sequential requests for *download*, *upload* and *more info* standard operations. We recorded the average round trip time and this experiment was repeated 5 times. Figure 2 shows the design of these experiments.

## 4.2. System Responsiveness Experiment Results

We conduct experiments where we investigate the base performance of the proposed system. *Figure 3*, *Figure 4*, *Figure 5*, and *Table 1*, *Table 2*, *Table 3* represent basic responsiveness results of our system. In this experiment we first recorded round trip times for: a) calling the *download* service to measure the response times of our implemented service; b) calling the *upload* service to measure the response times of our implemented service; c) calling *More Info* service to measure the response times of our implemented service. Downloading a new entry requires to store this entry as a major event in Amazon Cloud database and it is one of the major services provided by the prototype IDIOM system. Furthermore, the IDIOM propagates the updates via push mechanism by using upload service of the system in order to maintain consistency. This experiment shows the necessary time requirements for these major services to download or to upload a digital entity between Amazon Cloud database and annotation tools (replicas).

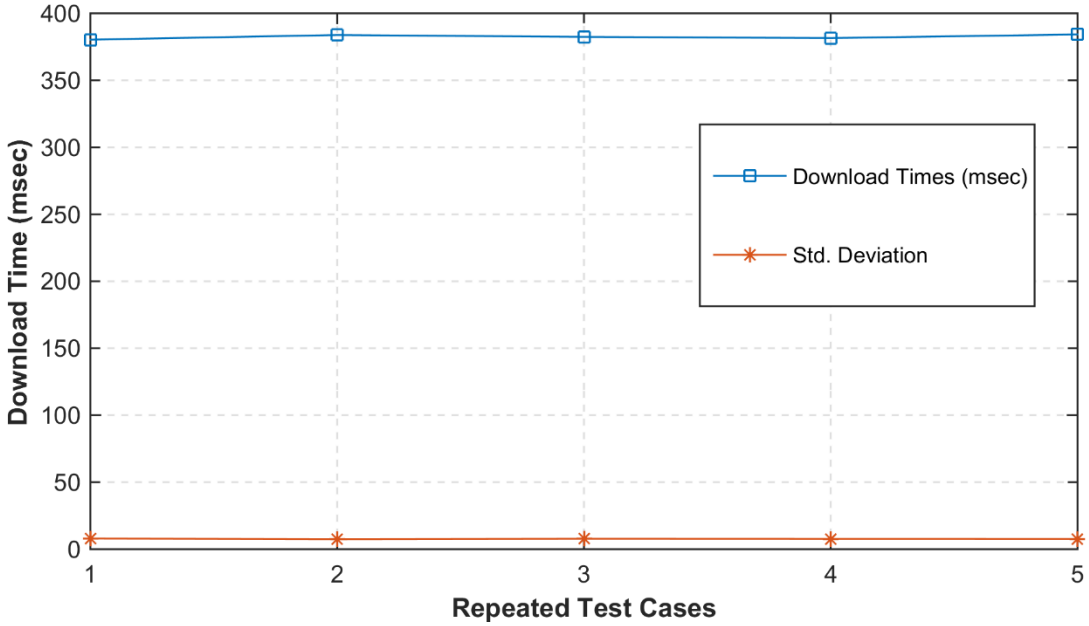


**Figure 3.** Depiction of Downloading a Record

**Table 1**. Statistics of the Experiment Depicted in *Figure 3*

| Repeated Test Cases | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| **Download Round-trip Time (msec)** | 380.3 | 383.3 | 382.4 | 381.5 | 384.2 |
| **Download STDev** | 7.97 | 7.39 | 7.76 | 7.65 | 7.59 |

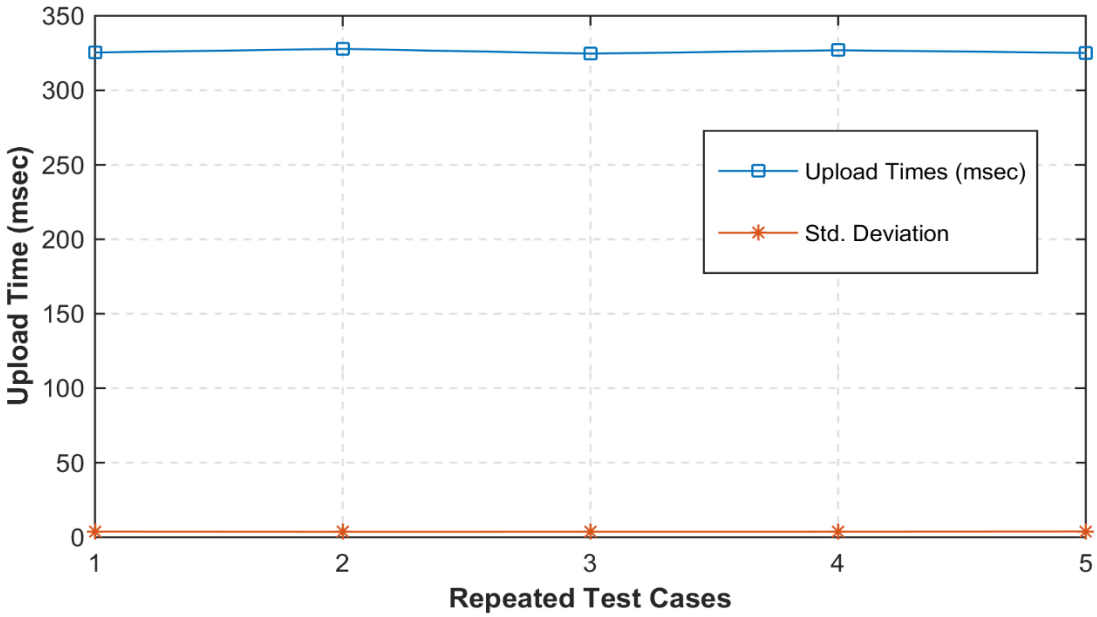


**Figure 4.** Depiction of Uploading a Record

**Table 2.** Statistics of the Experiment Depicted in *Figure 4*

| Repeated Test Cases | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| **Upload Round-trip Time (msec)** | 325.4 | 327.8 | 324.6 | 326.9 | 325.1 |
| **Upload STDev** | 3.8 | 3.64 | 3.68 | 3.72 | 3.88 |

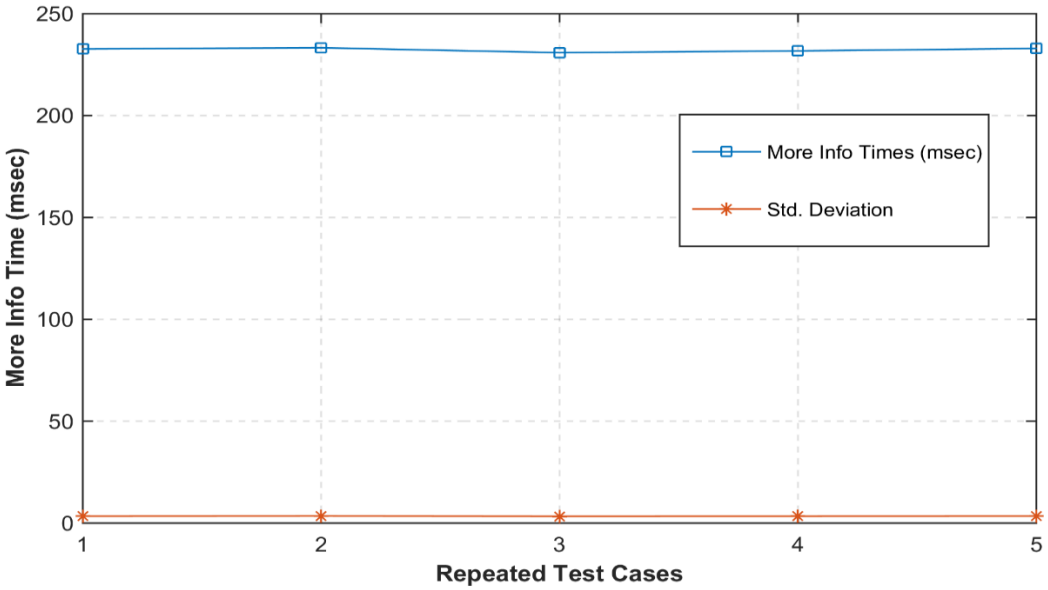


**Figure 5.** Depiction of More Info Service for a Record

**Table 3.** Statistics of the Experiment Depicted in *Figure 5*

| Repeated Test Cases | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| **MoreInfo Round-trip Time (msec)** | 232.7 | 233.3 | 230.86 | 231.78 | 232.96 |
| **MoreInfo STDev** | 3.40 | 3.49 | 3.34 | 3.38 | 3.44 |

## 5. Conclusion

Cloud computing is an emerging computing paradigm for managing and delivering services over the Internet. The increasing popularity of cloud computing is rapidly changing the landscape of information technology, and eventually turning the long-held promise of utility computing into a reality. Cloud computing (such as Amazon's EC2 and S3 online services, the Google App Engine, and Microsoft's SkyDrive) outsources basic computing infrastructure such as storage, computing, and hosting. The interiors of such systems are interesting and make use of virtual machine technologies, but such details are not exposed to the user. This technology movement is closely related to the advent of multicore, which are well matched to virtualization technologies such as VMWare, Xen, and OpenVZ. Programming models for clouds such as MapReduce-based Hadoop and Dryad are suitable for large scale clustering, dimensional reduction, and other data-mining techniques.

Although cloud computing offers significant benefits, the current technologies are still not matured enough to realize its full potential. Further researches are still being carried on for many key challenges in this domain, including automatic resource provisioning, power

management and security management. Therefore, we believe that there is still huge opportunity for researchers to make great contributions in this field resulting in significant impact to their development in the industry.

In this paper, we have discussed cloud computing paradigm, covering its essential concepts, architectural designs, prominent characteristics, key technologies as well as research directions. In particular, we discuss the proposed "Cloud-enabled Digital Information Service Framework" that is Web 2.0 and Cloud system and its potential for handling and managing metadata coming from different sources such as search tools, social annotation tools etc. We intend to further improve this approach to be able to scale up to a high number of distributed metadata sources such as video collaboration domain (YouTube etc.) and social networking domain (Facebook etc.). An additional area that we intend to research is an information security mechanism for the distributed Digital Information Service and machine learning techniques to identify typing errors within the documents. As the development of cloud computing technology is still at an early stage, we hope our work will provide a better understanding of the design challenges of cloud computing, and pave the way for further research in this area.

# References

[1] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," NIST, [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf. [Accessed 30 December 2015].

[2] X. Yang, D. Wallom, S. Waddington, J. Wang, A. Shaon, B. Matthews, M. Wilson, Y. Guo, L. Guo, J. D. Blower, A. V. Vasilakos, K. Liu and P. Kershaw, "Cloud computing in e-Science: research challenges and opportunities," *The Journal of Supercomputing,* vol. 70, no. 1, pp. 408-464, 2014.

[3] S. K. Garg and R. Buyya, "An environment for modeling and simulation of message-passing parallel applications for cloud computing," *Software: Practice and Experience,* vol. 43, no. 11, p. 1359–1375, 2013.

[4] A. F. Mustacoglu and G. C. Fox, "A novel digital information service for federating distributed digital entities," *Information Systems,* vol. 55, no. January 2016, p. 20–36, 2016.

[5] H. M. Lee, Y.-S. Jeong and H. J. Jang, "Performance analysis based resource allocation for green cloud computing," *The Journal of Supercomputing,* vol. 69, no. 3, pp. 1013-1026, 2014.

[6] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. D. Rose and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and Experience,* vol. 41, no. 1, pp. 23-50, 2011.

[7] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica and M. Zaharia, "A view of cloud computing," *Communications of the ACM,* vol. 53, no. 4, pp. 50-58, 2010.

[8] H. Liu, "Big Data Drives Cloud Adoption in Enterprise," *IEEE Internet Computing,* vol. 17, no. 4, pp. 68-71, 2013.

[9] S. Pandey and S. Nepal, "Cloud Computing and Scientific Applications — Big Data, Scalable Analytics, and Beyond," *Future Generation Computer Systems,* vol. 29, no. 7, pp. 1774-1776, 2013.

[10] D. Warneke and O. Kao, "Nephele: efficient parallel data processing in the cloud," in *In Proceedings of the 2nd Workshop on Many-Task Computing on Grids and Supercomputers (MTAGS '09). ACM*, New York, NY, USA, 2009.

[11] G. Aceto, A. Botta, W. d. Donato and A. Pescapè, "Cloud monitoring: A survey," *Computer Networks,* vol. 57, no. 9, p. 2093–2115, 2013.

[12] T. Gunarathne, B. Zhang, T.-L. Wu and J. Qiu, "Scalable parallel computing on clouds using Twister4Azure iterative MapReduce," *Future Generation Computer Systems,* vol. 29, no. 4, p. 1035–1048, 2013.

[13] F. Durao, J. F. S. Carvalho, A. Fonseka and V. C. Garcia, "A systematic review on cloud computing," *The Journal of Supercomputing,* vol. 68, no. 3, pp. 1321-1346, 2014.

[14] A. O'Driscoll, J. Daugelaite and R. D. Sleator, "'Big data', Hadoop and cloud computing in genomics," *Journal of Biomedical Informatics,* vol. 46, no. 5, p. 774–781, 2013.

[15] A. F. Mustacoglu, A. E. Topcu, F. O. Catak and G. C. Fox, "IDIOM Prototype Running on Amazon Cloud," [Online]. Available: http://ec2-52-27-148-245.us-west-2.compute.amazonaws.com:8080/IDIOM/login.jsp. [Accessed 30 December 2015].

[16] "Delicious," [Online]. Available: https://delicious.com/. [Accessed 20 January 2016].

[17] "Citeulike," [Online]. Available: http://www.citeulike.org/. [Accessed 20 January 2016].

[18] "Microsoft Academic Search," [Online]. Available: http://academic.research.microsoft.com/. [Accessed 20 January 2016].

[19] "Google Scholar," [Online]. Available: https://scholar.google.com. [Accessed 20 January 2016].

[20] T. O'Reilly, "What Is Web 2.0:Design Patterns and Business Models for the Next Generation of Software," 30 September 2005. [Online]. Available: http://www.oreilly.com/pub/a/web2/archive/what-is-web-20.html. [Accessed 30 December 2015].

[21] C. Anderson, The Long Tail: Why the Future of Business is Selling Less of More, New York: Hachette Books, 2008.

[22] C. S. Inc., "XenServer," [Online]. Available: https://www.citrix.com/products/xenserver/overview.html. [Accessed 30 December 2015].

[23] K. V. Machine, "Kernel Virtual Machine," [Online]. Available: http://www.linux-kvm.org/page/Main_Page. [Accessed 30 December 2015].

[24] VMware, "VMware ESXi," VMware, [Online]. Available: https://www.vmware.com/products/esxi-and-esx/overview. [Accessed 30 December 2015].