

## A Taxonomy and Survey of Distributed Computing Systems

Mohammadfazel Anjomshoa, Mazleena Salleh

Department of Computer Science & Information Systems, Faculty of Computing, Universiti Teknologi Malaysia, Skudai 81300, Johor, Malaysia  
[fazel\\_loyalboy@yahoo.com](mailto:fazel_loyalboy@yahoo.com)

**Abstract:** Technology is the combination of knowledge and working hard. When users want to accomplish something using special technology, they do not want to know how it works. It means that users only want to employ technology without any expert skills. So technologies are coming to solve and ease our complex problems. Computing paradigm is one of the most concerns in a complex problem. Computing paradigm is one of the most concerns in a complex problem. Distributed computing is one of technology that uses to solve large and complex computational problems. It employs Distributed Systems to address computational problems. In this paper we are going to highlight the most well-known computing technology and explain the technology building blocks of them. We address a full explanation of cloud computing and volunteer computing, as a grid branch, along with their advantages and also their open issues. [Anjomshoa MF, Salleh M. **A Taxonomy and Survey of Distributed Computing Systems**. *N Y Sci J* 2024;17(12):7-14]. ISSN 1554-0200 (print); ISSN 2375-723X (online). <http://www.sciencepub.net/newyork>. 02. doi:10.7537/marsnys171224.02.

**Keywords:** Volunteer Computing; Desktop Grids; Grid Computing; Cloud Computing; Computing; Clouds@home

### 1. Introduction

Grid computing is provided promising paradigm to enable high performance computing (HPC) by employing powerful computers as well as high speed network (Berman, Fox et al. 2003, Foster and Kesselman 2003). In the 1990s, the term grid was established to enable users to employ computing power on demand (Foster, Zhao et al. 2008). Since Know researcher developed the Grid idea in many different approaches. One of these approaches is desktop grid (DG) that uses from idle desktop computers of individuals owners (Fedak, Germain et al. 2001).

Volunteer computing is a type of DGs that employ idle CPU cycles of those public members who donate their computer's resources to solve a complex scientific problem (Anderson, Korpela et al. 2005). With the advent of Cloud computing (Mell and Grance 2011) which is the combination of many different technologies; virtualization, service-oriented architecture (Huhns and Singh 2005), scalability, quality of service, utility computing, green computing, fail over systems, cluster computing, the dream of computing become to reality (Rimal, Choi et al. 2009, Aversa, Avvenuti et al. 2011). With the combination of cloud computing and volunteer computing we can coin new computing paradigm that named Clouds@home (Distefano, Cunsolo et al. 2010).

This article intends to provide the comprehensive explanation of these computing paradigms. The rest of paper is organized as follow. Section two provides the taxonomy of computing paradigms including cloud computing (see section 2.1), volunteer computing (see section 2.2) and

Clouds@home (see section 2.3). Finally, section three concludes the paper.

### 2. Taxonomy

This section explains different forms of computing that are in common used today. Cloud computing, Volunteer computing and Clouds@home are addressed.

#### 2.1. Cloud Computing

Cloud computing is computing paradigm that has considered as a revolution in computing. Finding a unique definition for Cloud computing is not possible. Mell (Mell and Grance 2011) defines cloud computing as "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". Another definition for cloud computing is "Cloud computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the data centers that provide those services" (Fox, Griffith et al. 2009, Armbrust, Fox et al. 2010).

According to the (Mell and Grance 2011) cloud computing has lots of features that can be addressed the recent problem of computing. Vaquero (Vaquero, Rodero-Merino et al. 2008) defines cloud as form of grid computing, that virtual resources are dynamically allocated on a pay-per-used model. Cloud computing has two important portions. Cloud computing deployment models and cloud computing

services. As Rimal (Rimal, Choi et al. 2009) mentioned cloud computing has three types of models:

- i. Private cloud: Private clouds refer to the clouds that data and process are managed from inside of organization.
- ii. Public cloud: Public clouds refer to those types of clouds that cloud infrastructure is available to the public and can be accessed via web.
- iii. Hybrid cloud: Hybrid clouds are the combination of multiple clouds (private, public) with the goal of portability between different clouds that requires standardization technology.

Cloud computing is based on service-oriented-architecture that makes all resources on the cloud as a service (Tsai, Sun et al. 2010). As Figure.1 illustrates, general form of services in Cloud computing are,

infrastructure as a service-IaaS, platform as a service-PaaS, software as a service-SaaS (Mell and Grance 2011). These levels are supported by virtualization and management tools.

Cloud computing infrastructure as a service is composed of three important components; storage, servers and networks as is presented in Figure.2. The network is responsible for interconnecting entire resources; servers can be any type of servers and Storage that is attached to the servers. Amazon emerged as precursor of cloud computing since 2006 offering storage and basic processing via internet with its Amazon Elastic Compute (Shankar 2009) and Amazon Simple Storage Service (Palankar, Iamnitchi et al. 2008).

International Data Corporation (IDC) conducted a survey (Gens 2009) (refers to Figure.3) and asked from 263 IT organizations to rank the critical obstacles that prevent cloud computing from being adopted.



Figure 1: Cloud computing architecture

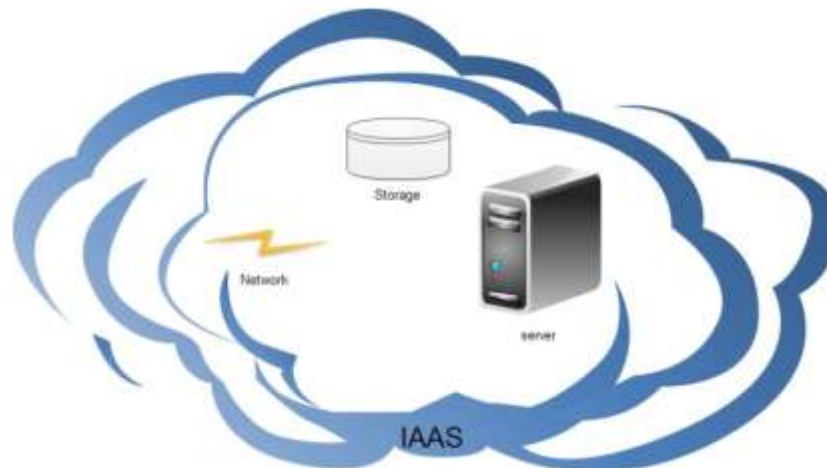


Figure 2: IAAS components

2.1.1. Challenging Issues for Cloud Computing

A glance at the Figure 3 provided, reveals that security, availability and performance are the top three concerns by organizations.

Without doubt, security has played an important role as an obstacle in cloud computing. Corporations and individuals are concerned about how security can be implemented in this new environment. Popovic (Popovic and Hocenski 2010) highlighted security concerns related to cloud computing infrastructure in three views; Security issues and challenges view, Security management standards view and Security management models view.



Source: IDC Enterprise Panel, 3Q09, n = 263

Figure 3: Results of IDC ranking security challenges

From the availability point of view, organizations worry about availability of cloud computing services. Cloud vendors offer service level agreements (SLAs) for the uptime and availability of services, while these availability SLAs should satisfy the needs of almost any cloud applications to convince enterprises and IT organizations to move into cloud. Table.1 provides comparison availability SLAs of well-known cloud providers.

Table 1: Comparing public cloud SLAs

Cloud vendor	Name of Services	Monthly Uptime Percentage
Google	Google Apps	< 99.9% - >= 99.0% (Google, (2013))
Amazon	Amazon EC2	< 99.5% - >= 99.0% (Amazon, (2013))
Amazon	Amazon S3	< 99.9% - >= 99.0% (Amazon, (2013))
Apple	iCloud	No-offered (Apple, (2013))
Microsoft	Cloud Services	<99.95%(Microsoft, (2013))
Dropbox	Dropbox	No-offered (Dropbox, (2012))
CloudFlare	CloudFlare	100% (Cloudflare, (2013))

However, infrastructure downtime is inevitable due to many unexpected issues. Table.2 illustrates the recorded cloud computing service outages in 2013 along with the error and duration. Google provided an application named "Apps Status Dashboard"(Amazon,(2013)) that provides for users to view for current status of all Google services.

Table 2: Cloud computing service outages in 2013 (infoworld,(2013), Google,(2013), Cern,(2013))

Cloud vendor	Error	Duration	Date
Twitter	Timelines came up blank and tweets went undelivered	short time	January 3
Dropbox Around	Client-syncing and file uploading failure	16 hours	January10
Facebook	Friends' status updates failure	Two to three hours	January28
Amazon	Amazon home page goes down	49 minutes	January31
Microsoft	Office 365 editing suite and Outlook.com mail service both stuttered	Around two hours	February 1-2
Microsoft	Microsoft Azure Cloud suffered a worldwide service interruption	almost a full day	February22
CloudFlare	Company's own site and all of its services kicked the bucket	About an hour	March 3
Google	Slow load times or full-on timeouts on Google drive	About 17 hours total	March 18-19
Apple iCloud	Services mostly experienced failures in authentication	Several hours	April 3

Cloud computing infrastructure takes a large expense regarding datacenters deployment. Considering approximately \$53 million each year only for servers or about \$10 million for powering (Greenberg, Hamilton et al. 2008). Finding a proper solution to minimize the energy consumption of cloud infrastructure and consequently reducing maintenance Costs of cloud computing caused to the new cloud area named "Green Cloud Computing" (Buyya, Beloglazov et al. 2010). The other alternative solution is building cloud infrastructure from grid resources which named Clouds@home (Cunsolo, Distefano et al. 2010). Cloud computing also has a problem of data lock-in due to lack of standardization and so interoperability (Armbrust, Fox et al. 2010). Data and workloads cannot move from one cloud to another cloud in an easy way. This issue prevents some organization to adopt their needs and services into cloud.

## 2.2. Volunteer Computing

Volunteer computing is a type of distributed computing in which volunteers donate their own Internet-connected computer resources (processing power, storage and Internet connection) to do one or more distributed computing projects (Sarmenta 2011). Computers often employ only 10% to 20% of their total capability so there is a huge potential processing power available here in which by joining hundred or millions of them, it is possible to perform projects that essentially need huge processing power. The computing resources can be a desktop PC, laptops, mobile phones or tablets in the way that, by connecting

these resources together, a single and super powerful computer is established that can do large computational problems. Volunteer computing consists of two parts (Nov, Anderson et al. 2010):

- i. Computational aspects: Related to problem of allocating and managing large computational jobs.
- ii. Participation aspects: Related to encouragement and persuasion tasks to attract more numbers of individuals to donate their computing resources to a project.

There are lots of volunteer computing projects; SETI@home (Nov, Anderson et al. 2010) which is a flagship volunteer computing project that was started in 1999 and since now has had over 3 million volunteers, distributed.net, LHC@home and Rosetta@home. In volunteer computing projects, there is a big problem that needs to carry out by a huge computational power, so the problem is divided into many executable tasks and each of them is solved by one or more donated nodes in parallel.

### 2.2.1. Challenging Issues for VC

In this section, main challenging issues of existing VC systems are described (Choi, Kim et al. 2007, Rimal, Choi et al. 2009):

- i. Volatility

In volunteer computing it is common that computing machines are got out of the project by either resource's owners or some technical occurrences (e.g., system crashing or power problem). A scheduler should

support mechanisms to guarantee availability of services.

ii. Lack of trust

In Desktop Grid, processing tasks take place in owners' computer therefore volunteers should trust the jobs and workloads. It is possible to see corrupted results because of malicious interposition. A scheduler should take some actions and procedure to guarantee the correctness of results.

iii. Failure

Resource provider of volunteer computing systems are connected through the Internet and it is so often they are disconnected because of crashing or link failures. A scheduler should tolerate the failures and volatility.

iv. Heterogeneity

The resources provided by volunteers are in wide range of specifications and properties such as CPU, memory, network bandwidth and failure rate. These considerations cause the overall performance and make delay and so decision making looks difficult for the scheduler.

v. Voluntary participation

The resources provided by volunteers are free to join and leave even in execution time.

In the one hand, finding proper mechanisms to attract more users should be taken into consideration. In the other hand in order to have a long donating time by donors, a scheduler should take some actions on rewarding mechanism for resource providers.

### 2.2.2. VC Middleware

In order to compute scientific problems in volunteer computing, middleware should be assembled to handle computation tasks. The main goal of these platforms is to split the job and distributes the parts (tasks) all around the world and collects the results when ready. Middleware involves various elements (servers, networks, volunteers PCs, storage) to perform the jobs distributive. Actually volunteer computing middleware acts like an interface between OS and application software that need performance.

### 2.2.3. Deployment Challenges

This section provides exhaustive account of challenges and technical issues that arise on the design of volunteer computing platforms. The main aim of this section is to introduce requirements and necessities of designing such platforms and also to highlights the difference between the successful platforms and the unsuccessful ones.

Due to the nature of volunteer computing which is a wide distributed public computing, in the one hand users who own the resources are in a wide range of technical knowledge. In other hand users might own different resource architecture and specifications for instance different OSs and different

software installed on that. From the user perspective, to attract and involve more users and consequently more computing resources the platform from user view should be ease of use which is including user interface design and abstracting the difficulties.

From diversity of resources provided by donors, platform independence should be taken into consideration, which means Macintosh or Linux owners can participate into the project as well as Windows users.

Security is the main challenge in volunteer computing as the whole computing jobs take place in volunteers' resources so they should trust the project that is migrated into their systems and performed. Basically, the scheduler should guarantee the security concerns from user views in order to encourage the volunteers from donating their resources.

The other important obstacle is application portability. Ideally, project owners who are willing to employ volunteer's resources must not be involved in the variety of hardware and platforms to perform their applications. As developing parallel application is not easy and maintaining and programming multiple versions of application for each platform might discourage the scientist to use volunteer computing platform.

Scalability and adaptability might become a hurdle since the project that is powered by volunteer computing platform is performed by a large number of volunteers distributed in all around the world and resources provided by these users are in the risk of volatility and failures as public and free computing resources.

### 2.2.4. Examples of Platforms

There are many volunteer computing frameworks that have been developed such as BOINC (Anderson, Cobb et al. 2002), Xtremweb (Fedak, Germain et al. 2001), and HTCondor. In this section the overview on some of the well-known volunteer computing middleware are provided.

### 2.2.5. BOINC

The Berkeley Open Infrastructure for Network Computing (BOINC) is the most well-known framework for Volunteer computing that is designed in a client-server approach.

The BOINC Client that is installed on volunteer's machine is responsible to perform the jobs and the BOINC server has a manager and coordinator role in the system. The client application periodically seeks for any job available on the server and if it finds, downloads the workload and returns the result back to the server after computation. Each project requires a separate assembled server but volunteers can



participate in multiple projects through a BOINC client.

BOINC is supported by approximately 2,700,000 users that hold about 7.8 petaflops (flops means floating point operations per second) (Boincstates, (2013)). By comparing the power of BOINC with the most powerful supercomputer in the world, it is possible to argue that BOINC can be placed in the sixth place of the most powerful supercomputers in the world (Top 500 supercomputer sites, (2013)).

#### 2.2.6. XstreamWeb

Xstreamweb is a multi-platform volunteer computing platform consists of three components, Client, Coordinator and Workers (Fedak, Germain et al. 2001). The workers send a message to server to get jobs and the server sends related files and also sends the application if it is not already existed in the workers machine. After finishing the execution process, worker contacts the coordinator to send the results back to the server. In comparison with BOINC the architecture of both are almost the same with the central server and many workers/clients which are responsible for performing the jobs by pulling from the server. However, the most conspicuous feature of XstreamWeb in contrast with BOINC is that some users possess the right to submit new jobs to be executed by the rest workers on the network grid.

The other stands out feature that in XstreamWeb functionality is that it is possible to not follow the centralized architecture in the way that workers might be able to send the results directly to client (Xstreamweb, (2013)).

#### 2.2.7. HTCondor

Htcondor is a powerful resource management for workstation environment that is based on remote unix (Litzkow 1987). HTCondor provides a job queueing mechanism, scheduling policy, priority scheme, resource monitoring, and resource management (HTcondor, (2013)). It is possible to use HTcondor as the cluster management of dedicated compute nodes. HTcondor provides transparently job check pointing and migrating from user's view.

In contrast with BOINC and XstreamWeb, HTcondor has a push model for its task distribution since the workers in condor grid network trust the application and so the jobs will be pushed from server to workers. HTcondor is composed of four different machines each servers more than one job in the same time. Central machine, which exists only one per each condor pool, is responsible for collecting information and also acts like an interface between resources and resource requests. The other machines running under the same pool will contact the central machine and send their update status over the network.

Execution machine is another part of condor architecture that is in charge of executing the job and provides the resources (CPU speed, memory and swap space) of condor pool. The next machine is the submit machine that it can be any one in the condor pool (including the central manager). Note that the resource requirements for this machine should be much more than the execution machine.

Submit machine is in charge of saving the checkpoints of the jobs and also all system calls performed as remote procedure calls back to the submit machine. Moreover, binaries of jobs that are submitted to the execute machines are saved on the submit machine.

#### 2.3. Cloud@homes: The Vision and Issues

With the advent of virtualization, it has been using in many computing infrastructure to enhance the infrastructure functionalities. Cloud computing is the service oriented computing that offers the services by using virtualization. The other distinctive idea regarding to the virtualization is volunteer cloud computing or Clouds@home (Distefano, Cunsolo et al. 2010). The idea is based on enabling virtualization technology in volunteer computing resources and make cloud-like infrastructure from volunteer resources. This type of cloud infrastructure addresses the problem of interoperability which is exists in cloud computing and also builds a customizable computing infrastructure in a lower scale and also in an affordable manner. To achieve the volunteer cloud system goals there are many obstacles need to be addressed.

The first problem that needs to be solved is adopting virtualization technology in volunteer computing environment. One of the biggest challenges of moving from costly and modern cloud's datacenters to volunteer's resources is the volatility and availability of resources. In volunteer computing it is common that computing machines are got out of the project by either resource's owners or some technical occurrences (for example system crashing or power problem). Consequently fault-tolerance has considered as an important effort to establish volunteer cloud. Another important problem that needs to be addressed is convincing volunteers to donate their resources in a different level of accessibilities than before as resources are not used only for scientific problems but also used by commercial providers.

### 3. Conclusion

In this paper the overview on computing system technology was presented. The different types of computing including cloud computing, volunteer computing and volunteer cloud computing were addressed. The advantages and disadvantages of cloud computing as well as cloud computing services and

models have been highlighted. Cloud issues that are ranked by IDC were shown. Volunteer computing as a powerful computing prototype has been explained. The issues of volunteer computing systems were introduced. Most examples of VC frameworks were explained. And at the end the idea of Clouds@home or volunteer cloud was described.

#### Corresponding Author:

Mohammadfazel Anjomshoa,  
Department of Computer Science & Information Systems,  
Faculty of Computing,  
Universiti Teknologi Malaysia, Skudai 81300, Johor, Malaysia  
[fazel\\_loyalboy@yahoo.com](mailto:fazel_loyalboy@yahoo.com)

#### References

- (2012). "Dropbox, Dropbox terms of service." from <https://www.dropbox.com/privacy>.
- (2013). "B. states, Boincstates." from <http://boincstats.com/en>.
- (2013). "C. Staff, The 10 biggest cloud outages of 2013 (so far)." from <http://www.crn.com/slideshows/cloud/240158298/the-10-biggest-cloud-outages-of-2013-so-far.htm?pgno=11>.
- (2013). "A. Choudhury, Google outage." from <http://webmasterfacts.com/google-outage/>.
- (2013). "A. EC2, Amazon ec2 service level agreement." from <http://aws.amazon.com/ec2-sla/>.
- (2013). "G. Team, Apps status dashboard." from <http://www.google.com/appsstatus#hl=en&v=status&ts=138365225>.
- (2013). "Google, Google apps service level agreement." from <http://www.google.com/apps/intl/en/terms/sla.html>.
- (2013). "HTcondor, What is htcondor." from <http://research.cs.wisc.edu/htcondor/description.html>.
- (2013). "I. CloudFlare, Business service level agreement." from <http://www.cloudflare.com/business-sla>.
- (2013). "J. D. H. S. Hans Meuer, Erich Strohmaier, Top 500 supercomputer sites." from <http://www.top500.org/>.
- (2013). "J. Raphael, The worst cloud outages of 2013 (so far)." from <http://www.infoworld.com/slideshow/107783/the-worst-cloud-outages-of-2013-so-far-221831#slide4>.
- (2013). "Microsoft, Service level agreements." from <http://www.windowsazure.com/en-us/support/legal/sla>.
- (2013). "A. S3, Amazon s3 service level agreement." from <http://aws.amazon.com/s3-sla/>.
- (2013). "X. team, Introduction to xtremweb." from <http://www.xtremweb.net/introduction.html>.
- (2013). "A. Inc, icloud terms and conditions." from <https://www.apple.com/legal/internet-services/icloud/en/terms.html>.
- Anderson, D. P., et al. (2002). "SETI@ home: an experiment in public-resource computing." *Communications of the ACM* **45**(11): 56-61.
- Anderson, D. P., et al. (2005). *High-performance task distribution for volunteer computing*. e-Science and Grid Computing, 2005. First International Conference on, IEEE.
- Armbrust, M., et al. (2010). "A view of cloud computing." *Communications of the ACM* **53**(4): 50-58.
- Aversa, R., et al. (2011). *The Cloud@ Home project: towards a new enhanced computing paradigm*. Euro-Par 2010 Parallel Processing Workshops, Springer.
- Berman, F., et al. (2003). *Grid computing: making the global infrastructure a reality*, John Wiley and sons.
- Buyya, R., et al. (2010). "Energy-efficient management of data center resources for cloud computing: A vision, architectural elements, and open challenges." *arXiv preprint arXiv:1006.0308*.
- Choi, S., et al. (2007). *Characterizing and classifying desktop grid*, Cluster Computing and the Grid, 2007. CCGRID 2007. Seventh IEEE International Symposium on, IEEE.
- Cunsolo, V. D., et al. (2010). Open and Interoperable Clouds: The Cloud@ Home Way. *Cloud Computing*, Springer: 93-111.
- Distefano, S., et al. (2010). A taxonomic specification of Cloud@ Home. *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*, Springer: 527-534.
- Fedak, G., et al. (2001). *Xtremweb: A generic global computing system*. Cluster Computing and the Grid, 2001. Proceedings. First IEEE/ACM International Symposium on, IEEE.
- Foster, I. and C. Kesselman (2003). *The Grid 2: Blueprint for a new computing infrastructure*, Elsevier.
- Foster, I., et al. (2008). *Cloud computing and grid computing 360-degree compared*. Grid Computing Environments Workshop, 2008. GCE'08, Ieee.
- Fox, A., et al. (2009). "Above the clouds: A Berkeley view of cloud computing." *Dept. Electrical Eng. and Comput. Sciences*,

- University of California, Berkeley, Rep. UCB/EECS 28: 13.
29. Gens, F. (2009). "New IDC IT cloud services survey: top benefits and challenges." IDC exchange.
  30. Greenberg, A., et al. (2008). "The cost of a cloud: research problems in data center networks." ACM SIGCOMM Computer Communication Review **39**(1): 68-73.
  31. Huhns, M. N. and M. P. Singh (2005). "Service-oriented computing: Key concepts and principles." Internet Computing, IEEE **9**(1): 75-81.
  32. Litzkow, M. J. (1987). Remote Unix: Turning idle workstations into cycle servers. Proceedings of the Summer USENIX Conference.
  33. Mell, P. and T. Grance (2011). "The NIST definition of cloud computing (draft)." NIST special publication 800(145): 7.
  34. Nov, O., et al. (2010). Volunteer computing: a model of the factors determining contribution to community-based scientific research. Proceedings of the 19th international conference on World wide web, ACM.
  35. Palankar, M. R., et al. (2008). Amazon S3 for science grids: a viable solution? Proceedings of the 2008 international workshop on Data-aware distributed computing, ACM.
  36. Popovic, K. and Z. Hocenski (2010). Cloud computing security issues and challenges. MIPRO, 2010 proceedings of the 33rd international convention, IEEE.
  37. Rimal, B. P., et al. (2009). A taxonomy and survey of cloud computing systems. INC, IMS and IDC, 2009. NCM'09. Fifth International Joint Conference on, Ieee.
  38. Sarmenta, L. F. (2011). Volunteer computing, Ph.D. thesis. Massachusetts Institute of Technology.
  39. Shankar, S. (2009). Amazon elastic compute cloud, CS.
  40. Tsai, W.-T., et al. (2010). Service-oriented cloud computing architecture. Information Technology: New Generations (ITNG), 2010 Seventh International Conference on, IEEE.
  41. Vaquero, L. M., et al. (2008). "A break in the clouds: towards a cloud definition." ACM SIGCOMM Computer Communication Review **39**(1): 50-55.

11/22/2024