International Journal of Engineering Research and Development
e-ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com
Volume 13, Issue 12 (December 2017), PP.51-60

A Comparison of Grid and Cloud Computing Technology

*Anmar Salih[1], Anthony O. Smith[2]
Florida Institute of Technology, Electrical and Computer Engineering,
150 W. University Blvd., Melbourne, FL 32940
Corresponding Author: Anmar Salih1

ABSTRACT: Cloud computing has evolved in this decade and become an intriguing field of study for researchers. Platforms and architectures are continuing to be developed, and every couple of years, new cloud computing techniques are discovered, with some of them deemed as revolutionary breakthroughs. One of the most amazing aspects of cloud computing is that researchers can develop a service and deliver it to users most likely through the Internet, without the need for an upgrade to the user side hardware. Cloud computing is the most famous but not the only paradigm to offer tremendous compute capability. Another form of comparable computing is grid computing which also offers the capability to process data in less time. This survey reviews both grid and cloud computing in order to clarify the understanding of applications, architecture, and topologies. We analyse three basic delivery models for cloud computing. Finally, we explore about link multiple clouds network, also known as inter-cloud or cloud of clouds.

Keywords: Grid computing, Cloud computing, Inter-cloud computing, Cloud architecture, Grid architecture.

Date of Submission: 19-12-2017 Date of acceptance: 11-01-2018

I. INTRODUCTION

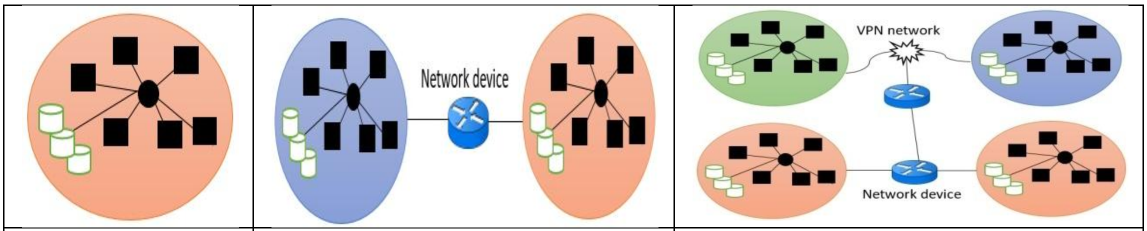
The development of the communications infrastructure allows cloud computing to provide daily services that depends on internet access. Cloud computing is the only technology to exist that combines multiple platforms, many locations, and provides users with a vast amount of services. Nowadays, smart devices utilize cloud computing [1] which make reaching cloud resources more accessible. Storage service is a simple example of how a cloud computing resource is utilized. Major corporations like Google, Microsoft, and Dropbox offer free limited cloud storage, with only a few required steps to register. Millions of users utilize cloud storage services to upload and share files such as images or documents. This sounds simple, but there are several hidden details. Other distributed systems like grid computing, utility computing, and distributed computing [2] share several aspects with cloud computing. The present work compares the details of grid computing and cloud computing technologies. We will explain grid computing in matter of classifications, architecture, applications, middleware, and topology. Then cloud computing in the matter of methodology, architecture, delivery models, technologies such as inter-cloud and intra-cloud, and platforms.

II. LITERATURE REVIEW

The focus of this survey is to understand cloud computing in relation to grid computing, and identify similarities between grid and cloud paradigms. A vast amount of research has been performed in this domain, and this is not an exhaustive study of cloud and grid related systems. We identify key components of ambiguity to explore, and direct the reader to [3]–[11] for more detailed comparisons. Cloud computing includes many services, that providers offer via the Internet. Dropbox, is a widely used example, and recognized as a cloud storage system [12]. Users can upload securely and save files on Dropbox servers eliminating the need for a user to retain data on a local computer. Buyya and Venugopal made a gentle introduction about grid computing, they explained components of grid computing, and stated operational flow from user's perspective [13]. Rekaby stated that grid computing is divided into two types, computing grid and data grid. Where computing grid is to have powerful processing grid while data grid is to have massive data storage environment [14]. We will focus more on works that comparing grid and cloud computing, because it is intersect more with our survey.The work in [3] directly compared and analysed 35 parameters shared between grid and cloud computing configurations in order to identify their relation to both grid and cloud. It was concluded that the main job of grid computing is collaborative sharing of resources while the main job of cloud is offering services over the Internet. Alisetty and Balachandrudu [15], considered six models (business, architecture, resource management, program, application and security). Their work illustrates how cloud and grid computing share many common aspects, but at the same time they have many differences such as security, compute model and applications. The research efforts in [16]–[21] have explored the architecture and infrastructure elements of both grid and cloud computing. All echoing a similar sentiment at the conclusion that both paradigms were complementary to each other. The research community in [14] , [22]–[25], has fostered more than a single definition for grid computing. Many of these studies share the similar view points, such as definitions, benefits of cloud and grid, and the various types of jobs of each is optimized to execute.The work by [26] summarizes the definitions, and clarifies grid computing in a few words, "grid computing is a system that coordinates distributed resources using standard, open, general-purpose protocols and interfaces to deliver nontrivial qualities of service.". Vaquero et al. [27], studied more than 20 definitions for cloud computing trying to extract a minimum one that contains all its characteristics to make a definition that clarifies the border between both grid and cloud computing. For years there was no consistency in the definition but fortunately National Institute of Standards and Technology (NIST) standardized the definition for cloud computing [28].oud network models have an important role in how these systems provide services. The two popular network models are intra-cloud and inter-cloud computing. Inter-cloud occurs when two services belong to single cloud provider work smooth and appear as a single service, and inter-cloud involves two or more providers that allow services and resources to appear as a single cloud [29]. To explain Inter-cloud by a few words we can define it as federate and interoperate of multiple clouds providers. [30] stated the problem with building Inter-cloud is not only technical protocols, but also to find a technically sound foundation and topologies. Designing Inter-cloud systems is a difficult task because of the lack of Inter-cloud standards. The authors in [31] identify the main points to consider when designing an Inter-cloud architecture. These include critical aspects such as: Service Level Argument (SLA), defined quality requirements, and resource management. A large portion of the Inter-cloud literature emphasize the importance of an efficient Inter-cloud computing configuration [32]–[34].Grozev and Buyya [34], agree there is no well-defined architecture for Inter-cloud, so they proposed a taxonomy based on the cloud type. Such as, volunteer federation cloud, where the provider collaborates and exchange their resources with each other, and independent cloud where multiple clouds are used in aggregation by a broker or application. They described the benefits of the taxonomy, like diverse geographical locations, better application resilience, and avoidance of vendor lock-in. [35] introduces a solution to help join multiple cloud storage providers, especially for small providers who have limited resources. The proposed solution was to utilize a dynamic Inter-cloud that could extend on request, in order to receive more resources from partner providers.The way two cloud networks collaborate can determine the correct name for their services. If users can jump from cloud A to cloud B serially, then we have a multi-cloud environment and the services are not cooperating. In such case, the providers have an agreement to join the same network, [36] so this is not an Inter-cloud system. NIST [28] also stated that when users jump from cloud to cloud we have a multi-cloud environment or serial clouds, while users using multiple services from different cloud providers have simultaneous clouds [28]. Both types (simultaneous and multi-cloud) are not Inter-cloud. A good explanation and application examples of multi-cloud has been stated in [36] , [37]. Inter-cloud happens when two services from different cloud providers collaborate to do a task as we mentioned before. According to the above, connecting multiple cloud providers becomes attractive and the lack of Inter-cloud standards and protocols encourages scientists to build an Inter-cloud model. The aforementioned works focus on comparing the infrastructure, models, architecture and services. However, there is an abundant amount of research that details the security aspect of cloud computing [38]–[45]. Our work will mention a few major concerns in security, but the security will not be the main purpose of this survey.

III. GRID COMPUTING

Grid computing was created in the early 1990s and best known through the Worldwide LHC (Large Hadron Collider) WLCG [46]. This form of computing has a distributed infrastructure usually based on heterogeneous systems that enable sharing of large scale resources. We can assume that a grid system is a virtual organization where all machines share resources, and is secure and flexible enough to add more components [47]. One of the early application of this paradigm is the film industry, to assist with animating the movies "*Toy Story*" and "*Shrek3*", which were rendered via grid computing on Sun and HP workstations [48] , [49]. In this scenario we can use any or many of these interconnected computers to process massive tasks. There are 3 key elements of grid computing definition; 1) Coordinate distributed resources, that may be more than one domain. Grid computing integrates all these resources no matter if they are in a single or multiple domain. 2) Using standard protocols and interfaces that can deal with authorization, authentication, and resource discovery. 3) Deliver reasonable qualities of services allows the use of its constituent resources in a coordinated manner to provide quality of service. For example, the work in [26] demonstrated availability, throughput, and response time.The primary machines in this paradigm are called "grid centers" which are represented by the number of interconnected servers. There are three elements that comprise a grid computing system. First, there must exist compute elements to manage the jobs and resources. Second, storage elements to offer the archiving capabilities, and third, worker elements to provide processing power to the system [46] , [50]. A grid platform can be utilized for various forms of computing, according to [51]–[53], there are five main types:



A | B | C

Fig.1: An illustration of the three basic network models for grid computing. **(A)** Intragrid for single grid organization. (**B)** Extragrid consist of multiple intragrid systems. **(C**) Intergrid configuration for many organizations multiple clusters.

3.1 Distributed supercomputing uses computational resources to reduce the completion time.

1. **High-throughput computing (HTC)** uses resources for a longer period of time. The goal here is use all idle processors for scheduled loosely coupled or independent tasks.
2. **On-demand computing** is typically used for short time jobs and the resources are made available to the user as needed.
3. **Data intensive computing** systems analyse high-volume dataflow, such as processing online streaming data or processing data coming from sensors in real time. The system works with big data and this type of computing usually must analyse so fast to decide in time.
4. **Collaborative computing** applications are designed to allow human to human interactions. As an example, two or more people accessing the same project and develop it at the same time.

An important property of a grid infrastructure is there is no single point of failure. This means, a machine failure will not affect processing the submitted job since there are other machines to complete the processing of the submitted jobs. Grid research community has summarized the benefits of grid computing into 3 categories [17], [54], [55]. First, **performance and scalability** considerations, deliver a high computational capacity. With a 10% processing time improvement, this would be deemed a huge time and cost savings. Second, **utilization of resources,** is a design to use multiple machines in a distributed environment more efficiently. By simply scheduling task to idle machines, we will get better utilization. The utilization design can be extended to design factors for processors exploits and more efficient storages and memories access. Finally, **system availability**, which is improved when grid computing in online and when a machine fails. Failures can be made transparent so long as there are plenty of other machines to continue processing a submitted job, then the failure will have a minimal impact.

3.2 Grid Computing topology

There are three topologies in grid computing intragrid, extragrid, and intergrid. Intragrid [56], is used for single organization and single cluster. It is the simplest scheme to deploy, and considered the most secure because it operates in single security domain. A single organization, has characteristics of very high throughput and the fact that there is no additional need for security protocols. This is due to the fact that all computers are connected locally and all the hardware is located within the same organization. Extragrid [57], is a scheme most utilized by a many organizations, and operates across more than one security domain. Intuitively, extragrid is considered as an interconnected network of intragrids. Therefore, the system is more complex than intragrid, but at the same time provides more resources. Finally, intergrid [57] would be considered the most complex configuration. This system typically spans across many organizations and many multiple clusters. As the number of clusters and organizations increase, the complexity is also increased, the security becomes difficult to manage and control, but the system becomes powerful with the ability to process a larger job. Fig. 1 illustrates the three topologies.

3.3 nGrid computing middleware and architecture

Grid middleware is a collections of software, packages, and protocols used for the implementation of the grid [58]. The main job of grid middleware is to allow the sharing of resources, so we get better utilization of idle resources. There are other important jobs for the grid middleware, such as providing services to hosted applications. A few of these services include discovery, storage, execution, resource monitoring, and failure detection. The grid middleware performs another role to hide the heterogeneous nature of the system [59] , [60]. Grid architecture consists of five layers as shown in Fig. 2 these layers are:

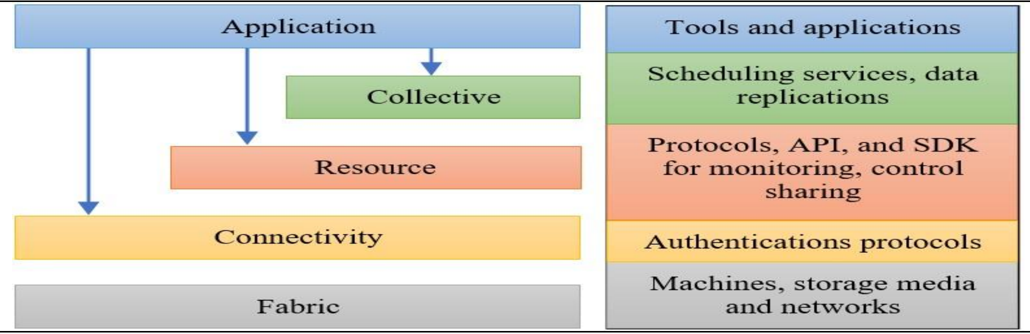


Fig.2: The five layers of grid computing middleware architecture, and the type of service that each layer can provide.

The fabric layer located at the bottom of the architecture. This layer provides access to resources such as storage, compute, and networks. By Fabric layer, the grid can understand what the limitations are of each computer in the grid. For example, the grid will have information about how fast the processor is and how much storage each machine has.Within the connectivity layer is the definition for core communications and authentications protocols. It provides all requirements to make data exchange possible between computers. By other words, it enables data exchange between Fabric layers. This layer provides authentication protocols to ensure identity of the users.The resources layer is built on the protocols and authentications of the connectivity layer. It defines protocols, API's, and SDK for monitoring and controlling sharing between computers. The collective layer deals with scheduling services, data replications services. While the resources layer is working with a single resource, the collective layer helps in coordinating multiple resources. For this reason, we call it the collective layer.The application layer these are user applications layer (languages and frameworks), comprises user applications and programs that operate within the grid environment. This layer can use all provided services from the other layers [52] , [61].

IV. GRID COMPUTING SUMMARY

The grid configuration can be heterogeneous and eliminate the need for all machines to have the same hardware and software. Two of the main advantages of a grid system are; powerful computational processing, because of the numerous networked machines, and abundant storage. For efficiency, the grid paradigm provides the ability to utilize idle resources at client machines. This level of flexibility is scalable, redundant with no single point of failure and makes it a viable system.Along with these benefits, grid computing also has a few drawbacks. The physical locations of the machines has become a concern of the past. This induces a need for fast network interconnections between machines (especially the servers). This is prevalent when applications need to be tweaked in order to take advantages of the distributed environment. Other challenges are, security, the heterogeneous nature of the system (hardware and software are not the same in all computers) and the simple case where machines working in different domains may not cooperate with each other seamlessly [62]. In summary there are many international grid projects, such as gLite [63], Unicore [64], Globus [65], Legion [52], all with unique capability and configurations.

V. CLOUD COMPUTING

Cloud computing shares a few common features with grid computing that include reduced cost of computing, increased reliability, and increased flexibility [4]. The main target of grid computing is to maximize application throughput, while the main target of cloud computing is delivering services over the Internet. One common scenario is that enterprises can rent hardware and software from the cloud and utilize the resource by network connection [42]. Providers such as Amazon [66]. Google [67], and Microsoft [68] provide this paradigm to allow customers to build virtual machines and define custom specifications, such as processor frequency, memory size, and hard disk drive size. The standard definition for cloud computing is introduced by the National Institute of Standard and Technology (NIST), stated as "Cloud computing is a model for enabling 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" [28].

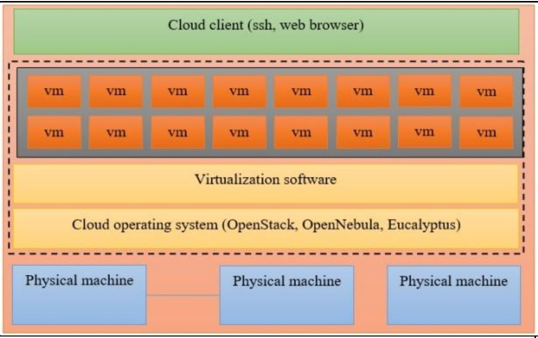


Fig.3: Cloud computing cluster, cloud operating system and virtualization software set on physical machines and host virtual components such as virtual machines (VM) or storage devices.

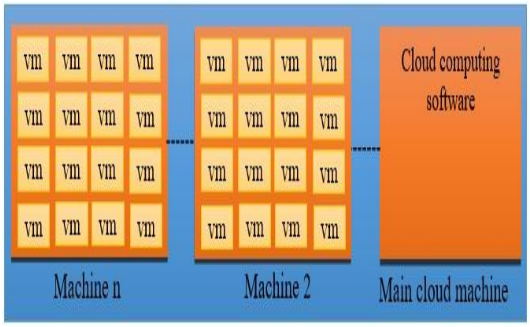


Fig.4: Simulation for Cloud computing cluster, cloud main software set on one machine and the use of other machines as containers to host virtual operating systems.

A cloud computing cluster may host hundreds of virtual operating systems. The cluster can comprise of many physical machines networked by high speed network, where each physical machine host several virtual operating systems. Fig. 3 provides an illustration of a cluster, with a cloud management platform (or cloud operating system) and the virtualization software set on the physical machines. The paradigm would be configured on platform using virtualization software such as Kernel based virtual machine (KVM) to enable hosting and control the virtual operating systems images. OpenStack (which is cloud computing operating system) that supports multiple virtualization software such as KVM, Quick Emulator (QEMU), and others [69]. Customers can use a web browser or secure shell to access a virtual operating system. Fig. 4 is another example of a cluster, where the main cloud software is set on one machine and the remaining machines are used to host virtual images for operating systems.

3.4 Cloud computing methodologies

Cloud computing methodologies are based on two main techniques, Service Oriented Architecture (SOA) and Virtualization [70]. SOA is a suite of flexible services that can be used within multiple businesses, these services are designed to be able to communicate with each other and coordinate to accomplish a job. We can assume the services designed by following SOA are like Lego parts, each service is like a separate part, which can be reshaped to build the needed model. SOA services are distributed across the network and may belong to different ownership domains [71].

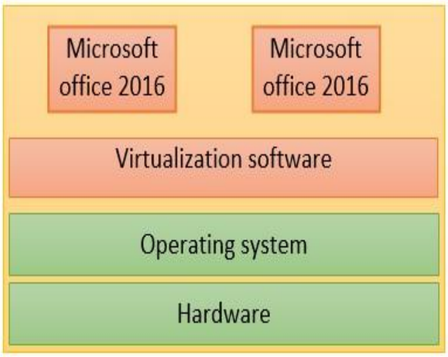


Fig.5: Application virtualization, multiple copy from the same program can sit on the same operating system. Each copy is isolated and there are no conflicts between both copies.

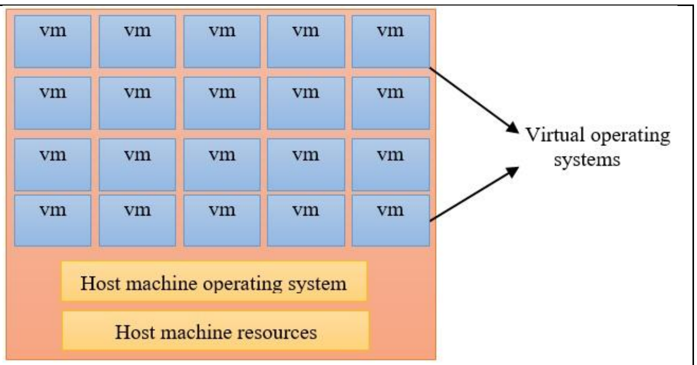


Fig.6: Sever virtualization, single machine hosting multiple virtual operating system images.

Virtualization is the technique of making a virtual image of something like an operating system. By using virtualization, a single machine can host multiple machines. The basic two types of virtualization are applications virtualization and server virtualization [72]. Application virtualization encapsulate computer program from the operating system and other installed programs. The virtualized applications will run normal and behaves like it have direct connection with the operating system while the truth it is isolated and sandboxed. Fig. 5 is an example of application virtualization that provides isolationThere are also several disadvantages, for example applications that rely on heavy integration with the operating system cannot be virtualized. Also, any application that needs direct contact with the graphics card

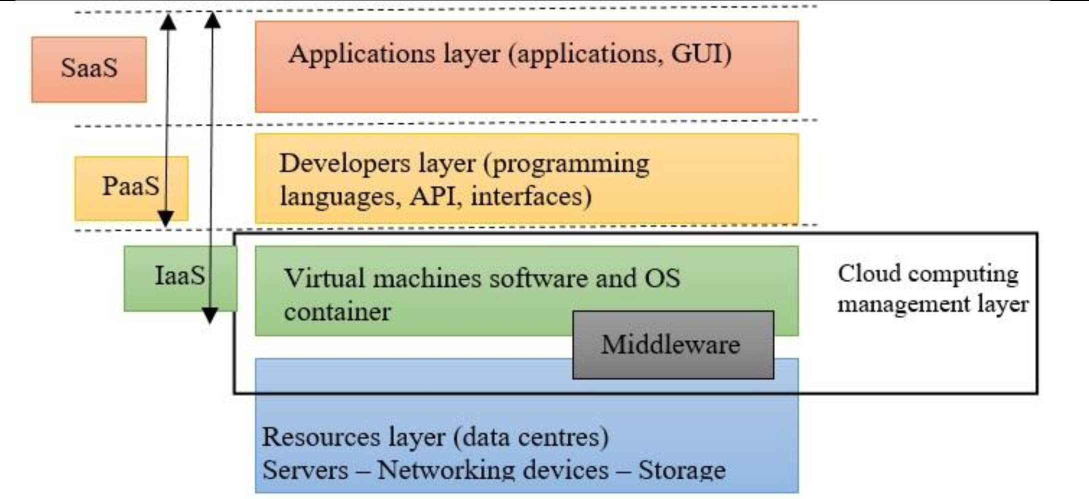


Fig.7: A cloud computing architecture showing the available services for each delivery model. Infrastructure as a service (IaaS) has low level access. Platform as a service (PaaS) provides development tools and interfaces. Software as a service (SaaS) is completed tools and capabilities.

cannot be virtualized. Server virtualization shown in Fig. 6, delivers virtual machines hosted in multiple physical machines with each physical machine potentially hosting many virtual operating systems. The specifications and resources of each physical server decides how many virtual machines can be made available. In a virtual environment, the host machine's resources are shared among all hosted virtual machines [73]. As shown in Fig. 6 each hosted image has an allotment of memory and processing resources. These hosted images are independent and do not realize they are sharing the host machine resources.

3.5 Cloud computing architecture and delivery models

A cloud computing architecture is shown in Fig. 7 and is best described by dividing it into four layers, which help clarify how cloud computing works [74]–[76]. The lower layer, which is the resources layer (data centre), contains all the hardware, such as processors, memories, storage devices, network devices, power supplies, and all needed resources to power up multiple nodes, stacked together to form a cluster of machines; Performance of the cluster is highly depending on this layer. Cloud computing management and VM software is the next layer, where cloud management (cloud operating system) administrate and provide services like monitoring and accounting. The cloud operating system provides oversight and control over the virtualization software so we can deploy, manage, monitor, and control all virtual operating system instances. The following layer, is the developer layer known as the cloud programming environment and tools layer. In this layer we can find developer tools such as programming languages and libraries installed and ready to use. The applications layer is the top layer in the stack and traditionally offers web based applications. This layer eliminates the need to install software in local computers.There are three basic service delivery models in the cloud computing. The first, is the Infrastructure as a Service (IaaS) model that allows providers to deliver virtualized machines, networks, and storage, through the Internet. The model has this name because it provides a complete infrastructure over the internet. IaaS users have full control over the operating system. This model is best for new organizations that have no space to host hardware nor do they want to invest in hardware. Another common scenario is companies that are growing rapidly with the need for more resources. Amazon Web Services Elastic Compute (EC2) and Amazon Secure Storage Service (S3) are examples for IaaS [28] , [77]. The second delivery model is associated with the developer layer, Platform as a Service (PaaS). This offers a complete set of tools and programming languages to help developers write codes and test it on cloud environment. Google App Engine is example of this type of delivery model [74] , [78]. The last layer is Software as a Service (SaaS) provides applications running on cloud infrastructure. Customers access these applications usually by using web client (web browser). Google docs and Google mail (Gmail) are examples for SaaS [28] , [79].

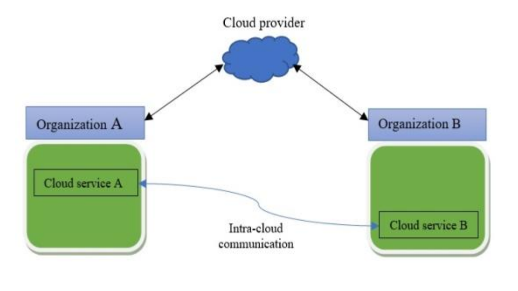


Fig.8: An intra-cloud network model with two services belong to two organizations collaborating together and both services are from single cloud provider.

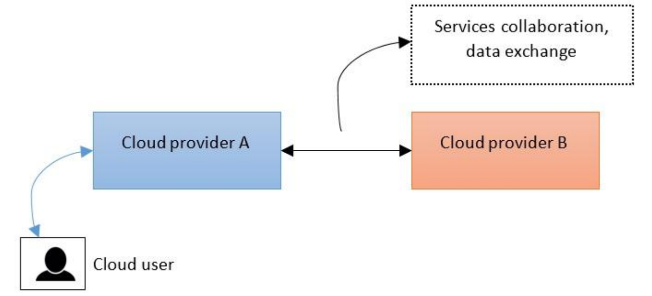


Fig.9: Inter-cloud communications for two services from different providers (Amazon and Microsoft for example) connected.

3.6 Inter-cloud and Intra-cloud

There are two network models related affiliated with cloud computing: Inter-cloud and intra-cloud. Within an intra-cloud model two services that belong to the same cloud provider collaborate to achieve a goal. For example, if we have two organizations using the same cloud provider, and the goal is to collaborate and exchange data by using cloud services. This form of collaboration between two or more services that belong to the same provider is detailed in [80] , [81]. Fig. 8 shows example for Intra-cloud communications model. The second model is inter-cloud computing and introduces a new layer. Inter-cloud is a connection between two cloud computing providers that connects them to make data transfers and applications collaboration possible. In inter-cloud, nodes from one cloud would use resources, data, or services from another cloud [33]. Fig. 9 shows simple inter-cloud model where cloud A and cloud B have communication channel to share data and applications to serve a cloud user.

Despite the recent cases of new inter-cloud companies starting [82] , [83], inter-cloud is still under development and there is no well-defined architecture for this model [34]. In fact, the work in [34] , [84]–[86] suggest various architectures, but none have been accepted as a standard. One of the challenges is that different providers may not understand each other services. [33].

3.7 Cloud computing summary

The main advantage of cloud computing is cost reduction by eliminating the need to purchase both hardware and software. Cloud computing removes the need for maintaining expensive computing hardware, because the maintenance work is shifted to the responsibility of the provider. There are several cloud computing platforms such as Abicloud, Eucalyptus, NIMBUS, OpenNebula, OpenStack, etc. [87]. OpenStack and OpenNebula are two IaaS open source cloud computing platforms that are most attractive for both the research and industrial community.

VI. GRID AND CLOUD COMPARISON

Cloud computing is easy to use, customers do not need to have experience to take advantage of this environment. Most of the cloud computing providers such as Amazon, Google and, Microsoft offers web based control panel to allow customers to manage their account. Grid computing on the other hand typically require expert knowledge. This is required because they have to provide administration and configure middleware.Recall that cloud computing is based on virtualization, see Fig. 3. Grid computing has limited virtualization in middleware such as Nimbus [5], and the core of cloud computing is virtualization to provide better utilization or resources. For the high performance computing matter, grid computing uses physical machines which makes processing data more efficient than process it in virtual machines. Cloud computing provides a capability to dynamically resize the amount of resource for more machines and storage devices to handle the job demand Loose coupling is another comparison point, where a system is configured with a group of items, and each item is partially separated and has a minimal effect on other items within the system. Cloud computing is a loose coupling system because of its use of virtualization. Each VM is isolated from the others and has information about other VMs in the system even though they are co-located on the same physical machine. While grid computing is partially loose coupled since each machine has to process part of the job.Virtualization make the cloud paradigm strong fault tolerant environment. It is easy to deploy new machines to replace any broken resource. On the contrary, grid computing does not support strong fault tolerance because in the case that several machines stop working, the system will not be able to continue processing job smoothly.Finally, resource sharing is limited in a cloud computing paradigm. Each virtual machine has specific hardware and it will not share with other virtual machines. While grid computing has high resources sharing capabilities. Grid computing is based on idle resources sharing.

VII. CONCLUSION

In an effort to clarify some ambiguity and misunderstanding related to two very powerful and popular paradigms, we have explored several key components of cloud and grid computing. Both have advantages and disadvantages when applied to specific needs and problems. It is fair to say that when deployed in the right environment, configured properly, and taking all the necessary consideration both can be deemed as viable systems.

REFERENCES

[1]. V. Yu, F Richard and Leung, Advances in mobile cloud computing systems. CRC Press, 2013.
[2]. G. Mittal, N. Kesswani, and K. Goswami, “A Survey of Current Trends in Distributed, Grid and Cloud Computing,” IJASCSE, vol. 2, no. 3, 2013.
[3]. S. M. Hashemi and A. K. Bardsiri, “ARPN Journal of systems and software cloud computing vs. grid computing 1,” vol. 2, no. 5, 2012.
[4]. I. Foster, Y. Zhao, I. Raicu, and S. Lu, “Cloud computing and grid computing 360-degree compared,” in Grid Computing Environments Workshop, 2008. GCE’08, 2008, pp. 1–10.
[5]. R. Buyya, C. S. Yeo, and S. Venugopal, “Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities.”
[6]. S. Shuai Zhang, X. Xuebin Chen, S. Shufen Zhang, and X. Xiuzhen Huo, “The comparison between cloud computing and grid computing,” in 2010 International Conference on Computer Application and System Modeling (ICCASM 2010), 2010, pp. V11-72-V11-75.
[7]. L. Youseff, M. Butrico, and D. Da Silva, “Toward a unified ontology of cloud computing,” in Grid Computing Environments Workshop, 2008. GCE’08, 2008, pp. 1–10.
[8]. F. M. Aymerich, G. Fenu, and S. Surcis, “An Approach to a Cloud Computing Network.”
[9]. L. Wang et al., “Cloud computing: a perspective study,” New Gener. Comput., vol. 28, no. 2, pp. 137–146, Apr. 2010.
[10]. T. Dillon, C. Wu, and E. Chang, “Cloud Computing: Issues and Challenges,” in Proceedings of the 2010 24th IEEE International Conference on Advanced Information Networking and Applications, 2010, pp. 27–33.
[11]. S. Zhang, S. Zhang, X. Chen, and X. Huo, “Cloud computing research and development trend.”
[12]. I. Drago et al., “Inside dropbox,” in Proceedings of the 2012 ACM conference on Internet measurement conference - IMC ’12, 2012, p. 481.
[13]. R. Buyya and S. V. Database, “A gentle introduction to grid computing and technologies,” gridbus.cs.mu.oz.au.
[14]. A. Rekaby and M. A. Rizka, “A comparative study in dynamic job scheduling approaches in grid computing environment,” Int. J. Grid Comput. Appl., vol. 4, no. 3, p. 1, 2013.
[15]. K. Alisetty and K. Balachandrudu, “Cloud computing & grid computing environments,” Int. J. Innov. Eng. Technol., vol. 3, no. 2, 2013.
[16]. A. Doavi, “Comparing the architecture of Grid Computing and Cloud Computing systems,” Int. J. Comput. Sci. Netw. Solut. Sep, vol. 3, no. 9, 2015.
[17]. H. Alhakami, H. Aldabbas, and T. Alwada ’n, “COMPARISON BETWEEN CLOUD AND GRID COMPUTING: REVIEW PAPER,” Int. J. Cloud Comput. Serv. Archit., vol. 2, no. 4, 2012.
[18]. N. Sadashiv and S. M. D. Kumar, “Cluster, grid and cloud computing: A detailed comparison,” in 2011 6th International Conference on Computer Science & Education (ICCSE), 2011, pp. 477–482.
[19]. M. Obali and A. E. Topcu, “Comparison of cluster, Grid and Cloud Computing using three different approaches,” in 2015 23nd Signal Processing and Communications Applications Conference (SIU), 2015, pp. 192–195.
[20]. L. Yuxi and W. Jianhua, “Research on Comparison of Cloud Computing and Grid Computing,” Res. J. Appl. Sci. Eng. Technol., vol. 4, no. 2, pp. 120–122, 2012.
[21]. Q. Zhang, L. Cheng, and R. Boutaba, “Cloud computing: state-of-the-art and research challenges,” J Internet Serv Appl, vol. 1, pp. 7–18, 2010.
[22]. I. Foster and A. Iamnitchi, “On Death, Taxes, and the Convergence of Peer-to-Peer and Grid Computing 3 Comparing Grids and P2P,” LNCS, vol. 2735, pp. 118–128, 2003.
[23]. M. Bote-Lorenzo, Y. Dimitriadis, and E. Gómez-Sánchez, “Grid characteristics and uses: a grid definition,” in Grid Computing, 2004, pp. 291–298.
[24]. P. V. Coveney, “Scientific Grid Computing,” Philos. Trans. Math. Phys. Eng. Sci., vol. 363, pp. 1707–1713.
[25]. M. Armbrust et al., “A View of Cloud Computing Clearing the clouds away from the true potential and obstacles posed by this computing capability,” Commun. ACm, vol. 53, no. 4, 2010.
[26]. I. Foster and C. Kesselman, The grid : blueprint for a new computing infrastructure. Morgan Kaufmann,

2004.
[27]. L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A Break in the Clouds: Towards a Cloud Definition."
[28]. F. Liu et al., "NIST cloud computing reference architecture," NIST Spec. Publ., vol. 500, no. 2011, p. 292, 2011.
[29]. C. Esposito, M. Ficco, F. Palmieri, and A. Castiglione, "Interconnecting Federated Clouds by Using Publish-Subscribe Service," Cluster Comput., vol. 16, no. 4, pp. 887–903, Dec. 2013.
[30]. D. Bernstein and D. Vij, "Intercloud Security Considerations," in 2010 IEEE Second International Conference on Cloud Computing Technology and Science, 2010, pp. 537–544.
[31]. T. Aoyama and H. Sakai, "Inter-Cloud-Computing," Wirtschaftsinformatik, vol. 53, no. 3, pp. 171–175, 2011.
[32]. A. N. Toosi, R. N. Calheiros, and R. Buyya, "Interconnected Cloud Computing Environments," ACM Comput. Surv., vol. 47, no. 1, pp. 1–47, May 2014.
[33]. J. Lloret, M. Garcia, J. Tomas, and J. J. P. C. Rodrigues, "Architecture and protocol for intercloud communication," Inf. Sci. (Ny)., vol. 258, pp. 434–451, 2014.
[34]. N. Grozev and R. Buyya, "Inter-Cloud architectures and application brokering: taxonomy and survey," Softw. Pract. Exp., vol. 44, no. 3, pp. 369–390, Mar. 2014.
[35]. M. Fazio, A. Celesti, M. Villari, and A. Puliafito, "How to Enhance Cloud Architectures to Enable Cross-Federation: Towards Interoperable Storage Providers," in 2015 IEEE International Conference on Cloud Engineering, 2015, pp. 480–486.
[36]. D. Petcu, "Multi-Cloud: Expectations and Current Approaches."
[37]. F. Paraiso, N. Haderer, P. Merle, R. Rouvoy, and L. Seinturier, "A Federated Multi-cloud PaaS Infrastructure," in 2012 IEEE Fifth International Conference on Cloud Computing, 2012, pp. 392–399.
[38]. K. Vieira, A. Schulter, C. B. Westphall, and C. M. Westphall, "Cloud computing & grid computing environments," IT Prof., vol. 12, no. 4, pp. 38–43, Jul. 2010.
[39]. R. Rajagopal and M. Chitra, "Trust based interoperability security protocol for grid and Cloud computing," in 2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT'12), 2012, pp. 1–5.
[40]. R. Kaur and J. Kaur, "Cloud computing security issues and its solution A review," in Computing for Sustainable Global Development, 2015 2nd International Conference on, 2015, pp. 1198–1200.
[41]. D. Zissis and D. Lekkas, "Addressing cloud computing security issues," Futur. Gener. Comput. Syst., vol. 28, no. 3, pp. 583–592, 2012.
[42]. S. Carlin and K. Curran, "Cloud computing security," 2011.
[43]. "Gartner: Seven cloud-computing security risks," 2008.
[44]. A. and S. (Online service) Chakrabarti and S. (Online service), "Grid computing security," 2007.
[45]. A. Chakrabarti, A. Damodaran, and S. Sengupta, "Grid Computing Security: A Taxonomy," IEEE Secur. Priv. Mag., vol. 6, no. 1, pp. 44–51, 2008.
[46]. D. Garlasu et al., "A big data implementation based on Grid computing," in 2013 11th RoEduNet International Conference, 2013, pp. 1–4.
[47]. K. Czajkowski, S. Fitzgerald, I. Foster, and C. Kesselman, "Grid information services for distributed resource sharing," in Proceedings 10th IEEE International Symposium on High Performance Distributed Computing, pp. 181–194.
[48]. F. Berman, G. Fox, and T. Hey, "The Grid: past, present, future."
[49]. S. N. J. Dupire, "Entertainment Computing--ICEC 2009."
[50]. D. E. Kaplan et al., "DIRAC: a community grid solution," J. Phys. Conf. Ser, vol. 93, no. 119, 2008.
[51]. M. Baker, R. Buyya, and D. Laforenza, "Grids and Grid technologies for wide-area distributed computing," Softw. Pract. Exp., vol. 32, no. 15, pp. 1437–1466, Dec. 2002.
[52]. F. (Frédéric) Magoulès, Fundamentals of grid computing : theory, algorithms and technologies. CRC Press, 2010.
[53]. K. Krauter, R. Buyya, and M. Maheswaran, "A taxonomy and survey of grid resource management systems for distributed computing," Softw. Pract. Exp., vol. 32, no. 2, pp. 135–164, Feb. 2002.
[54]. S. N. Pardeshi, C. Patil, and S. Dhumale, "Grid Computing Architecture and Benefits," Int. J. Sci. Res. Publ., vol. 3, no. 8, pp. 2250–3153, 2013.
[55]. D. Minoli, A Networking Approach to Grid Computing. Wiley-Interscience, 2005.
[56]. "Distributed data mining in grid computing environments," Futur. Gener. Comput. Syst., vol. 23, no. 1, pp. 84–91, Jan. 2007.
Y. Zhu and L. M. Ni, "A Survey on Grid Scheduling Systems."
F. Magoules, J. Pan, K.-A. Tan, and A. Kumar, Introduction to grid computing. CRC press, 2009.
[57]. G. Von Laszewski, G. W. Pieper, and P. Wagstrom, "Gestalt of the Grid," Tools Environ. Parallel Distrib.

Comput., p. 149, 2002.
[58]. G. von Laszewski and K. Amin, “Grid middleware,” Middlew. Commun., p. 109, 2004.
[59]. L. Ferreira et al., “Introduction to grid computing with globus,” IBM redbooks, vol. 9, 2003.
[60]. V. Berstis and others, Fundamentals of grid computing, vol. 28. IBM Redbooks, 2002.
gLite - Lightweight Middleware for Grid Computing.” .
[61]. “UNICORE | Distributed computing and data resources.” .
[62]. “Weblet Importer.” .
[63]. “Amazon Web Services (AWS) - Cloud Computing Services.” .
[64]. “Google cloud computing, hosting services & apis.” .
[65]. “Home | Microsoft Cloud.” .
[66]. “OpenStack Docs: Mitaka,” 2017. .
[67]. L. Wang, R. Ranjan, J. Chen, and B. Benatallah, Cloud computing: methodology, systems, and applications. CRC Press, 2011.
[68]. K. B. Laskey and K. Laskey, “Service oriented architecture,” Wiley Interdiscip. Rev. Comput. Stat., vol. 1, no. 1, pp. 101–105, Jul. 2009.
[69]. N. B. Ruparelia, Cloud Computing. Mit Press, 2016.
[70]. L. Malhotra, D. Agarwal, and A. Jaiswal, “VIRTUALIZATION IN CLOUD COMPUTING,” Int. J. Comput. Sci. Mob. Comput., vol. 38, no. 8, pp. 745–749, 2014.
[71]. R. Buyya, C. Vecchiola, and S. T. Selvi, Mastering cloud computing: foundations and applications programming. Newnes, 2013.
[72]. R. Hill, L. Hirsch, P. Lake, and S. Moshiri, Guide to cloud computing: principles and practice. Springer Science & Business Media, 2012.
[73]. H. T. Dinh, C. Lee, D. Niyato, and P. Wang, “A survey of mobile cloud computing: architecture, applications, and approaches,” Wirel. Commun. Mob. Comput., vol. 13, no. 18, pp. 1587–1611, Dec. 2013.
[74]. S. Bhardwaj, L. Jain, and S. Jain, “Cloud computing: A study of infrastructure as a service (IAAS),” Int. J. Eng. Inf. Technol., vol. 2, no. 1, pp. 60–63, 2010.
[75]. D. Beimborn, T. Miletzki, and S. Wenzel, “Platform as a Service (PaaS),” WIRTSCHAFTSINFORMATIK, vol. 53, no. 6, pp. 371–375, Dec. 2011.
[76]. W. M. Sinnett, Software as a service: experiences of SMBs. 2010.
[77]. H. S. Alqahtani and G. Kouadri-Mostefaoui, “Towards a Classification of Multiple-Cloud Computing Concepts and Terms,” in European Conference on Service-Oriented and Cloud Computing, 2014, pp. 271–277.
[78]. S. Chen, S. Nepal, and R. Liu, “Secure connectivity for intra-cloud and inter-cloud communication,” in 2011 40th International Conference on Parallel Processing Workshops, 2011, pp. 154–159.
[79]. “InterCloud - Home.” [Online]. Available: https://www.intercloud.com/. [Accessed: 27-Sep-2017].
[80]. “Intercloud Systems | Minds Together.” [Online]. Available: http://www.intercloudsys.com/. [Accessed: 27-Sep-2017].
[81]. Y. Demchenko, M. X. Makkes, R. Strijkers, C. Ngo, and C. de Laat, “Intercloud Architecture Framework for Heterogeneous Multi-Provider Cloud based Infrastructure Services Provisioning.,” Int. J. Next-Generation Comput., vol. 4, no. 2, 2013.
[82]. S. Sotiriadis, N. Bessis, and E. G. M. Petrakis, “An inter-cloud architecture for future internet infrastructures,” in International Workshop on Adaptive Resource Management and Scheduling for Cloud Computing, 2014, pp. 206–216.
[83]. Y. Demchenko, C. Ngo, R. Strijkers, and C. De Laat, “Defining inter-cloud architecture for interoperability and integration,” in In CLOUD COMPUTING 2012, The Third International Conference on Cloud Computing, GRIDs, and Virtualization, 2012.
[84]. J. Peng, X. Zhang, Z. Lei, B. Zhang, W. Zhang, and Q. Li, “Comparison of several cloud computing platforms,” in Information Science and Engineering (ISISE), 2009 Second International Symposium on, 2009, pp. 23–27.

*Anmar Salih.“A Comparison of Grid and Cloud Computing Technology.” International Journal Of Engineering Research And Development, vol. 13, no. 12, 2017, pp. 51–60.