Venkatesh D *et al*, International Journal of Computer Science and Mobile Applications,
Vol.4 Issue. 5, May- 2016, pg. 34-40 **ISSN: 2321-8363**
Impact Factor: 4.123

InterCloud: The Cloud of Clouds

Venkatesh D[1], Vijay Kumar Damera[2], Suresh Pabboju[3]
[1]Department of I.T, SWEC, Hyderabad, INDIA, venkateshdamera@gmail.com
[2]Department of I.T, MGIT, Hyderabad, dameravijaykumar@gmail.com
[3]Department of I.T, CBIT, Hyderabad, INDIA, plpsuresh@gmail.com

Abstract: Cloud computing is a relatively new model for large data centers and highly distributed. This paradigm, with its pay- per-use model, provides high scalability and availability has completely changed the landscape of services and systems. Recently, the cloud computing model was expanded to include computing capabilities (for example virtual machines) and operating systems. However, till date, cloud computing proliferation has not lived up to expectations in the corporate segment. Furthermore, the clouds still cannot interoperate with each other. In this paper, we show Intercloud as a layer that encompasses all the clouds, with the aim of building a trusted cloud computing. We also discuss two operating propositions of Intercloud a proposal by Bernstein and Vij and another proposed by IBM.

Keywords: Cloud Computing, Intercloud, Data Centers.

1. Introduction

Cloud computing is becoming one of the key words of the IT industry. According Taurion [16] the term cloud computing came in 2006 in a lecture by Eric Schmidt , CEO of Google, about how his company managed their data centers. Today, cloud computing is presented as the core of a movement of profound changes in the world of technology.

The cloud is a representation for the Internet or communication infrastructure between the architectural components , based on an abstraction hides the complexity of the infrastructure. Each part of this infrastructure is provided as a service, and these services are normally allocated in data centers using shared hardware for computing and storage [11].

These services are known as Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). Below briefly define these services, according to[15] :

IaaS - Infrastructure as a Service: is the ability of the provider has to provide a processing and storage infrastructure transparently. Examples of IaaS include Amazon EC2, GoGrid and the Eucalyptus (open source).

PaaS - Platform as a Service : capacity offered by the provider to the user to develop applications that will be implemented and made available in the cloud. Google AppEngine and Microsoft Azure are examples of PaaS.

SaaS - Software as a Service: hosted applications in the cloud. Google Apps and SalesForce are examples of SaaS.

Along with these services mentioned above, cloud computing offers high scalability and availability , making it highly attractive for outsourcing data and calculation , both for consumers who want to share their photos with friends and for companies looking to reduce their IT budgets .

However, of course , there are security concerns related to the outsourcing of data and calculation in cloud computing . Even if the cloud provider is reliable, the issue of multi-tenancy (resource sharing) can be

©2016, IJCSMA All Rights Reserved, www.ijcsma.com

considered a matter of vulnerability. More specifically, these problems include the dimensions of the acronym CIRC :

- Confidentiality
- Integrity
- Reliability
- Consistency

For cloud systems are applied to larger areas such as the core of large companies, will be required QoS guarantee (Quality of Service) end-to- end, not only covering cloud computing resources , but also the networks involved , and meet the reliability requirements .

When services are provided by a single cloud (single cloud), an unexpected level of overhead (Internet traffic), or a natural disaster may require more computational resources . However , this scenario of a single cloud allocated resources are usually limited and the capacity of a single cloud to continue the services is also limited. In order to improve the cloud computing system to continue to meet QoS requirements, the organization of these clouds so that a cloud can interact with other cloud and thus obtain the necessary resources is required.

The Intercloud , also known as cloud of clouds, is a promising solution in improving the CIRC and QoS dimensions (compared to the model of computation in a single cloud) , and especially interoperation between clouds. It is clear that from a cloud, instructions can be issued through the Internet cloud to another. For example, the code running on Google AppEngine can also use the code that is stored on AWS (Amazon Web Services). However, there are implicit forms that cloud resources and services to interoperate.

This paper aims to explain the functioning of Intercloud and the survey of the main negative aspects of the current cloud computing model , single domain model . In the following section , we will explain the operation of the single domain computer and its limitations are lifted .

In section III the main objectives and plans of Intercloud are shown and discussed. In two subsections of this section show two operating propositions of Intercloud a proposal by Bernstein and Vij , and another proposed by IBM.

2. Cloud Computing as a Single Layer

Cloud computing consists of a single layer of distributed protocols designed to run on a single administrative domain, usually under the control of the service provider (eg, Amazon AWS and Google Apps) [1]. Figure 1 illustrates the cloud computing model of single domain. The protocols used in this context are meant for wide area systems , with scalability for a large number of customers and as the main goal, high availability.

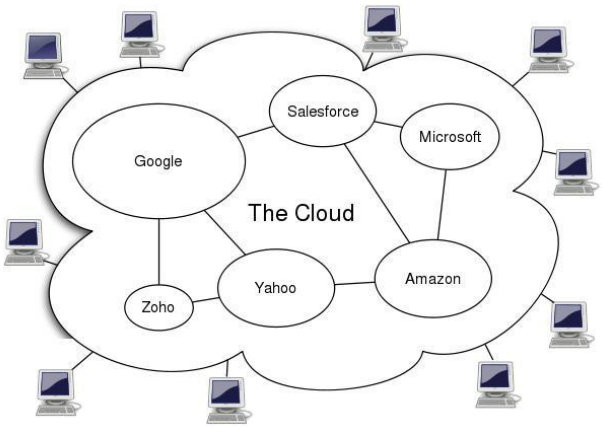

Figure 1: Model of Cloud Computing as Single Layer

©2016, IJCSMA All Rights Reserved, www.ijcsma.com

The reliability and safety of a single domain cloud, especially its confidentiality, integrity and isolation of data and calculations are receiving increased attention [1]. However, the design of reliable service , based on the offer of a provider of cloud X has its very visible limitations , since all the confidence in the system is reduced to trust X.

Other limitations on trust in a service provider in the cloud are related to reliability and consistency. While the clouds are designed to be highly available, interruptions can and do happen in any single domain provider. In addition, the network of a cloud provider is the single point of failure, especially in the case of cloud computing providers that not geographically distribute their services, ie their data centers are centralized in one location. Figure 2 illustrates this scenario and scenario data centers distributed.

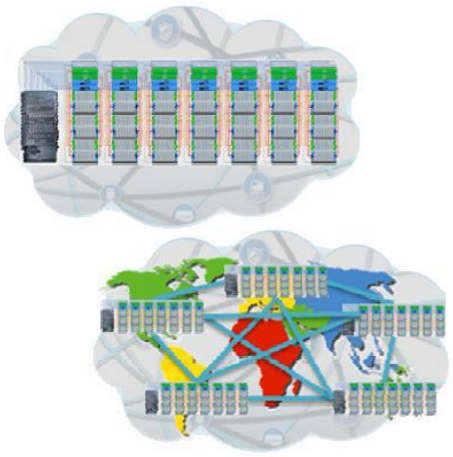

Figure 2: Data Centres can be centralized or distributed in different continents

Thus, the network connections are particularly vulnerable when the client is outside of North America and Europe, where broadband connections are not very stable. Thus, the eventual consistency offered by many cloud providers, may not be sufficient for some applications, such as operating systems. Cloud solutions of a single layer to cache data locally on the client , to avoid problems of consistency .

3. InterCloud

To understand better what is the Intercloud , let's make an analogy of this to the Internet: In a world www , data is ubiquitous and interoperable in a network known as the Internet. In a world of cloud computing, storage and content processing are ubiquitous and interoperable in a cloud network known as Intercloud, which is illustrated in Figure 3 .

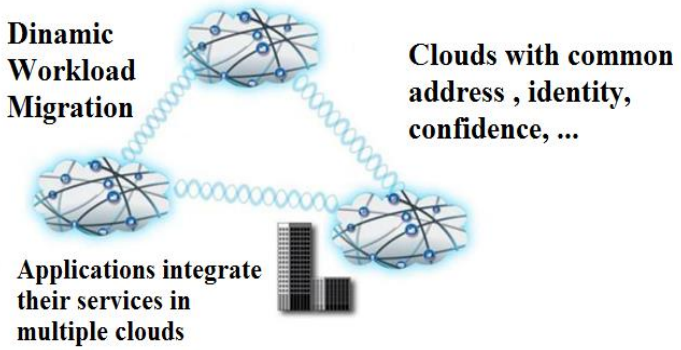

Figure 3: The vision of Intercloud

©2016, IJCSMA All Rights Reserved, www.ijcsma.com

The Intercloud offers a unique environment for the creation of reliable services. Consider, for example, fault tolerance as one of the key aspects of reliability. A fundamental assumption in virtually all systems tolerant to failure count is the fault assumption of independence. This event occurs in different forms, ranging from classical chance of failure limit for models without failure threshold. In the first case, failure independence is reflected on the assumption that only a maximum number of failed processes, ie by P but not P processes + 1 or more may fail in the second case, the failure of independence is assumed different sets of likely failures. However, the coverage of this assumption, in practice, it is sometimes small.

This remains a weak point of many systems , and the results on the assumption often being criticized . In this respect, Intercloud is unprecedented: This comes with diverse geographical locations, power supplies, different administrative domains and middleware and different implementations of proprietary application. Best of all, this diversity comes to the customer, essentially for free, because the maintenance of diversified services is not the responsibility of a client, but is in the hands of cloud providers separately.

Since the resources provided by cloud computing are a commodity, its cost is absorbed by many customers for different tasks. By contrast, maintain enough diversity for a task, which include hardware and a variety of operating systems, software maintenance and management know-how, is prohibitively expensive. Also, trusting the diverse set of clouds through the distribution of trust between different cloud domains is an attractive alternative to trust (possibly critical) data for one cloud provider.

Clearly, Intercloud layer does not replace the single layer of cloud, but expands its reach. Most likely, the reliable protocols Intercloud will be, first, customer-centric , where client-side proxies orchestrate multiple clouds. Later , this will be followed by more sophisticated services involving communication between different cloud services (this is not easily possible today due to lack of standardization) .

Also, most likely the protocols in Intercloud They will add considerable value on the cloud, maximizing various metrics of QoS (Quality of Service) , including confidence in the broad sense , and CIRC in particular [1]. Here , briefly , we will see the model operation of Intercloud proposed by David Bernstein and Deepak Vij , which can be seen in more detail in [12] and [13] . Later , we will see another model proposed IBM (ICStore) [1].

3.1 Operating Proposition Intercloud by Bernstein and Vij

For better understanding, before explaining the operation of Intercloud proposed by Bernstein and Vij , let's look at Figure 4 :

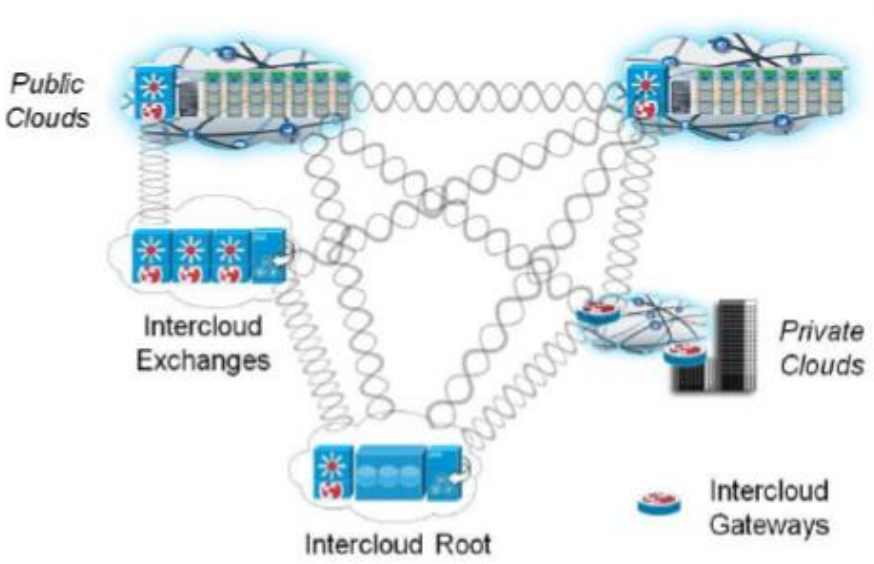

Figure 4: Intercloud topology proposed by Bernstein and Vij

©2016, IJCSMA All Rights Reserved, www.ijcsma.com

Venkatesh D *et al*, International Journal of Computer Science and Mobile Applications,
Vol.4 Issue. 5, May- 2016, pg. 34-40 **ISSN: 2321-8363**
Impact Factor: 4.123

Seeing the figure above, we see two characters unknown to date in Cloud scenario Computing. They are: Intercloud Root and Intercloud Exchanges.

The Intercloud Root will act as a mediator between exchanges in Intercloud, solving the problem N^2 (where all the clouds try communication with each other), allowing connectivity between different clouds. Intercloud providers Exchange together instances Intercloud Root will facilitate the negotiation of dialogue and negotiation between the different clouds.

Gateways Intercloud would provide mechanism to support all protocols and Intercloud standards. O Intercloud Root and Intercloud Exchange and facilitate mediate the initial negotiation process between the clouds. Once the initial negotiation process is completed, each of the clouds in question would directly collaborate with one another via one appropriate transport protocol, and for interoperability.

3.2 IBM Intercloud by the operating Proposition (ICStore)

No different in this model, for better understanding Intercloud of the working model proposed by IBM , we see Figure 5, which represents the high design ICStore level.

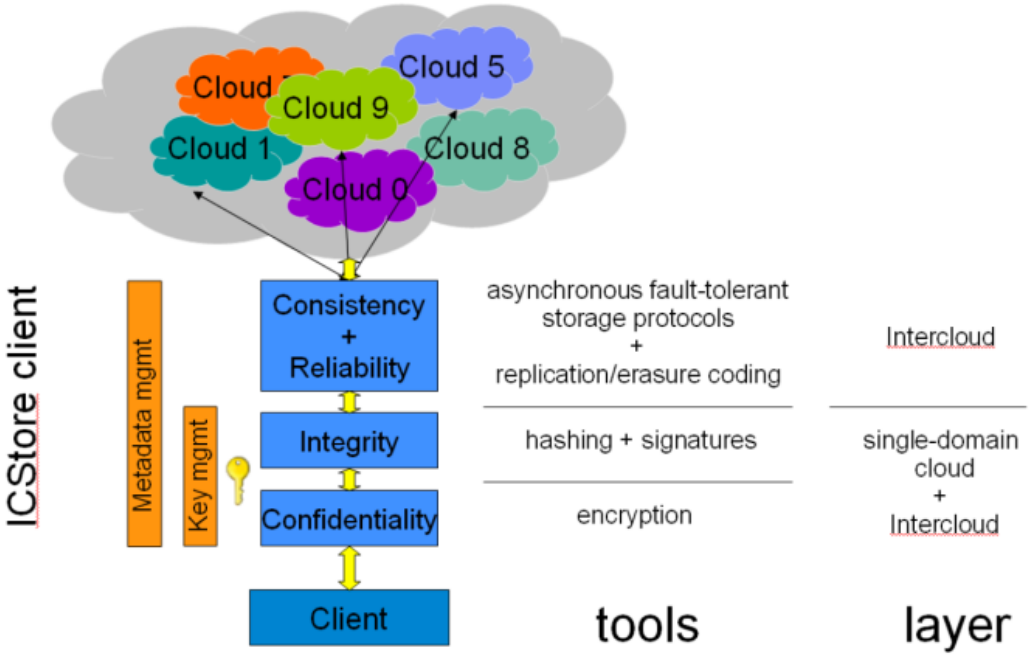

Figure 5: High-level Design ICStore

In the heart of its design, there is a customer that ICStore orchestra convenience of multiple services cloud storage in Intercloud (Examples: Amazon S3, Eucalyptus / Walrus, Nirvanix) [1] and provides a service transparent storage to the end customer, which does not You need to have knowledge about the details of ICStore, so as the number of different Clouds and their APIs. The client ICStore offers the end customer a key stored value with reading and writing simple operations, which is a service common offered by storage providers a cloud. The initial implementation of IBM sees interface ICStore / Client as a subset of Amazon interface S3 (including, for example, this operation: put (key, version, value); get (key); delete (key)). With this project, the customer ICStore appears to the end customer as a cloud virtual, which simplifies the porting of existing cloud storage applications (example, those built upon Amazon S3) to ICStore. Several end users can access the ICStore through your own customer ICStore.

On the other side, on the back end of Intercloud , the client ICStore connects to separate cloud providers computing , protecting the end user for their diversified APIs. New cloud computing providers can be modularly inserted through individual writing adapters individual provider API. API for more standardized ICStore of which is connected to the end customer.

©2016, IJCSMA All Rights Reserved, www.ijcsma.com

Finally, the client ICStore consists of three layers with different goals:

1. Confidentiality
2. Integrity
3. Reliability and Consistency

This layered approach allows individual layers to be "on" and "off " to provide different levels of trust so you can match the client's goals , also with performance and the possibility of financial constraints. Note that the ICStore client functionality need not be implemented on the client side, or the client may reside remotely ICStore acting as a separate service and as a gateway (gate) for Intercloud . However, in the IBM initial prototype, the ICStore client is implemented as a library the final customers.

The following are briefly to ensure that each of the three layers work in ICStore model:

Confidentiality: In this layer, the client performs a simple key symmetric encryption of customer data received. The key challenge in this layer is key management. To this end, the ICStore project supports the enterprise-level use of key managers, assuming that these key managers are properly replicated to fornececimento a desired level of fault tolerance.

Integrity: The integrity layer provides cryptographic protection against unauthorized modification of data. When only a single client access untrusted cloud storage, data integrity can be performed primarily with hash trees. To allow multiple readers and writers to access the stored data, the integrity layer depends on public key infrastructure clients.

Reliability and Consistency: The layer of reliability and consistency is distributed tolerant data gaps protocols that scatter the data Intercloud after the data (optionally) pass through the layers of confidentiality and integrity. IBM intends, this layer to support a variety of scatter data protocols , which should be selected depending on the final application goals, observing the performance and monetary constraints.

4. Conclusion

Despite being a new term , cloud computing will probably be the new architecture for hardware and software , as well as being typically outsourced services, cloud computing brings to the customer largest economy with infrastructure. However, Cloud Computing brings concern safety issues. To improve this point, a solution called Intercloud, still under study, is being developed. Because it is a model that aims, as well as interoperability between clouds, improvement of data security-related terms, and despite being a still theoretical term, the Intercloud is certainly a model that will most likely be used in future Cloud Computing. Another factor that proves the relevance of Intercloud, is that IBM, large private company in the area, is investing heavily in the model.

References

[1] Kleinrock, L.: A Vision for the Internet. ST Journal of Research 2(1), 4–5 (2005)

[2] Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., Zaharia, M.: Above the Clouds: A Berkeley View of Cloud Computing. University of California at Berkley, USA. Technical Rep UCB/EECS-2009-28 (2009)

[3] Buyya, R., Yeo, C., Venugopal, S., Broberg, J., Brandic, I.: Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility. Future Generation Computer Systems 25(6), 599–616 (2009)

[4] London, S.: Inside Track: The high-tech rebels. Financial Times (September 6, 2002)

[5] The Reservoir Seed Team. Reservoir – An ICT Infrastructure for Reliable and Effective Delivery of Services as Utilities. IBM Research Report, H-0262 (Febuary 2008)

[6] Buyya, R., Abramson, D., Giddy, J., Stockinger, H.: Economic Models for Resource Management and Scheduling in Grid Computing. Concurrency and Computation: Practice and Experience 14(13-15), 1507–1542 (2002)

[7] Weiss, A.: Computing in the Clouds. NetWorker 11(4), 16–25 (2007)

[8] VMware: Migrate Virtual Machines with Zero Downtime, http://www.vmware.com/

[9] Barham, P., et al.: Xen and the Art of Virtualization. In: Proceedings of the 19th ACM Symposium on Operating Systems Principles. ACM Press, New York (2003)

©2016, IJCSMA All Rights Reserved, www.ijcsma.com

[10] Buyya, R., Abramson, D., Venugopal, S.: The Grid Economy. Special Issue on Grid Computing. In: Parashar, M., Lee, C. (eds.) Proceedings of the IEEE, vol. 93(3), pp. 698– 714. IEEE Press, Los Alamitos (2005)

[11] Yeo, C., Buyya, R.: Managing Risk of Inaccurate Runtime Estimates for Deadline Constrained Job Admission Control in Clusters. In: Proc. of the 35th Intl. Conference on Parallel Processing, Columbus, Ohio, USA (August 2006)

[12] Yeo, C., Buyya, R.: Integrated Risk Analysis for a Commercial Computing Service. In: Proc. of the 21st IEEE International Parallel and Distributed Processing Symposium, Long Beach, California, USA (March 2007)

[13] Sulistio, A., Kim, K., Buyya, R.: Managing Cancellations and No-shows of Reservations with Overbooking to Increase Resource Revenue. In: Proceedings of the 8th IEEE International Symposium on Cluster Computing and the Grid, Lyon, France (May 2008)

[14] Chu, X., Buyya, R.: Service Oriented Sensor Web. In: Mahalik, N.P. (ed.) Sensor Network and Configuration: Fundamentals, Standards, Platforms, and Applications, January 2007. Springer, Berlin (2007)

[15] Amazon Elastic Compute Cloud (EC2), http://www.amazon.com/ec2/ (March 17, 2010)

[16] Google App Engine, http://appengine.google.com (March 17, 2010)

[17] Windows Azure Platform, http://www.microsoft.com/azure/ (March 17, 2010)

[18] Spring.NET, http://www.springframework.net (March 17, 2010)

[19] Chappell, D.: Introducing the Azure Services Platform. White Paper, http://www.microsoft.com/azure (January 2009)

[20] Venugopal, S., Chu, X., Buyya, R.: A Negotiation Mechanism for Advance Resource Reservation using the Alternate Offers Protocol. In: Proceedings of the 16th International Workshop on Quality of Service (IWQoS 2008), Twente, The Netherlands (June 2008).

©2016, IJCSMA All Rights Reserved, www.ijcsma.com