

Software Reliability Assessment Paradigm of Multi Cloud Computing Systems

B.J.D Kalyani¹ and Dr. Kolasani Ramchand H Rao²

¹Research Scholar of Acharya Nagarjuna University,
Department of Computer Science and Engineering, India.

kjd_kalyani@yahoo.co.in

²Dr. Kolasani Ramchand H Rao,
Principal, ASN Degree College, Tenali, India,
ramkolasani@gmail.com

January 9, 2018

Abstract

Organizations shifting towards Multi Cloud Computing Systems (MCCS) for providing optimized services to users. MCCS consists of more than one cloud to mitigate the risk of service availability failure, data inconsistency, and vendor lock-in. In general, there is no service provider owns all the services that result in the total value being delivered to a customer. Hence Service delivery today requires two or more service providers working together efficiently to meet customer requirements most economically. This gives birth to multi cloud computing systems by combining best services with core competencies provided by several different providers into new value added service chain. This paper proposes a novel software reliability assessment paradigm consists of different levels of abstractions in MCCS by exploring the function of each component and provides plans to implement the paradigm.

Key Words:Multi Cloud Computing System, Software Reliability, Virtual Machine.

1 Introduction

It is very difficult to locate 100% resources required for a service in a single domain. This collection of services generally developed and available in multiple different service domains [1] and a service owner may not be able to directly control prerequisite services. Service delivery today needs multiple clouds and multiple service domains to work together in harmony throughout the entire lifecycle of that service [2]. Enterprises looking for MCCS to gain the advantages like

- To provide functional interfaces [3] that satisfy core value realized through service composition [4].
- To Support end-to-end provisioning [5] process.
- Guarantee service assurance with QoS [6] and SLA conformance [7].
- Provides event information for settlement and billing purposes.

Well known multi cloud service providers include RightScale [8], enStratus [9], Kaavo [10], mOSAIC [11] and Optimus [12]. These providers enhance the potential of multiple clouds by deploying services on multiple clouds, in terms of workload scheduling, data migration, and pricing, etc., to maximize utility in dynamic environments. The general performance metrics for evaluating MCCS involves operational cost, network delay, service response time and availability etc. Hence the QoS mechanism for MCCS includes High-level applications, Programming paradigms, and market mechanisms. Layered Software Reliability Assessment Paradigm (LSRAP) supports these mechanisms.

2 Related work

Now a days software reliability assessment in MCCS is an innovative theme for many researchers. Amir Vahid Dastjerdi [13] et al focuses on improvement of reliability to support SLA negotiations regarding resource utilization in cloud computing environment by

providing Service Negotiation Framework (SNF). Victor Munte-Mulero and Peter Matthews [14] et al specifies the mitigation of risk of various quality parameters like interoperability, integration, availability etc in order to improve reliability in MCCS. Hang Wei [15] et al assess the reliability of cloud application based on Semi quantitative Information and Evidential Reasoning by describing 4 Level System Architecture (4LSA).

3 Proposed paradigm

The recent trends in the field of network technologies are likely to facilitate the shifting from the single cloud to Multi Cloud. Network virtualization [16] techniques for distributed resources in various administrative domains can support computational and network resources from multiple clouds and network substrates to be integrated into a single virtual resource. The main motto of proposed paradigm is to avoid vendor lock-in [17]. The proposed paradigm provides solution to

- Business by defining strategies, guidelines, mode of use that reacts to changes of the offers by the providers.
- Semantic by constructing taxonomy for calls, responses and provides standard interfaces for the functionality of multi cloud.
- Application and services by facilitating automation or configuration of life cycle activities.
- Monitoring with rules, protocols, and standards in deployment or relocation.
- Maintaining the data with the help of the specificities of each Cloud by network allocation and admission procedures.

In order to tackle all the above solutions, MCCS architecture is developed by integrating task related proxies to a network in order to facilitate services to cloud customers. A proxy [18] can be a node that deploys a software instance for a user or a service provider can be permit operations to carry out on its behalf. The controller

component of proxies can work as intermediaries for merging different types of clouds. Hence in MCCS large pool of cloud service providers are integrated to access the proxy which itself is a cloud suggests proxy as service framework. The common technique for accessing this environment is a third party proxy service provider tackles the collaborated CSP providers as modeled in Fig 1.

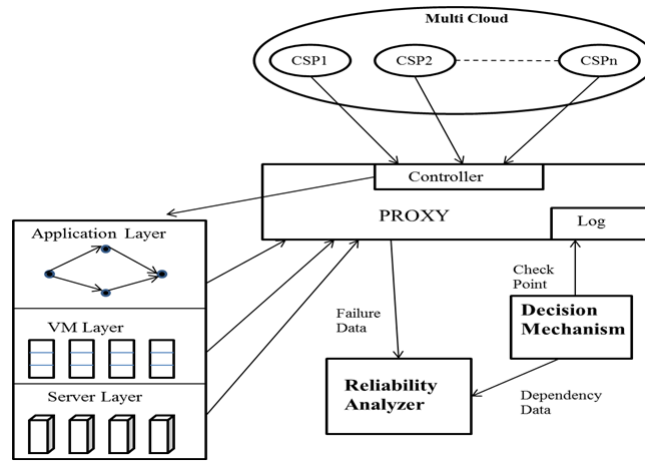


Fig 1: Layered Software Reliability Assessment Paradigm (LSRAP)

LSRAP uses bottom up software reliability assessment method which best suits for critical needs and identifies important parameters for evaluation. Bottom up approach supports accurate long term performance and life models. This approach is ease in locating problem areas at the low levels of an environment and also provides ease in observing the results. Hence early assessment of reliability is possible.

The heterogeneity of services [19] is a challenge for MCCS. A methodology is needed to satisfy diversified service requests from various user profiles. The optimal methodology that supports multiple criteria includes cost, policies, and performance. Diversity occurs to both low and high levels, between virtualization technologies and programming environments. Generally, the MCCS is hiding this heterogeneity which necessitates different levels of abstractions in MCCS architecture results in layered software rela-

bility assessment in MCCS.

3.1 Levels of Abstraction

Each cloud in MCCS has three layers like

- Application layer
- Virtualization layer
- Server Layer

3.1.1 Application Layer

Application layer provides user interaction with MCCS. Only the required data for the user can be viewed without knowing finer details regarding how his/her query is manifested by MCCS.

3.1.2 Virtualization Layer

Virtualization layer defines service load of user requests and also describes the load balancing mechanism [20] for MCCS. This layer assess the application reliability by considering three case studies includes

1. If all VMs are deployed on different physical servers.
2. If all VMs are deployed on same physical servers.
3. If only a group of VMs deployed on different group of physical servers.

3.1.3 Server Layer

Server layer allocates VM (Virtual Machine) required to process user request and allocates required resources like CPU, memory, Bandwidth etc to service the user request and defines resource scheduling mechanisms [21]. Virtual Machines can be allocated dynamically and also supports elasticity.

3.2 Decision Mechanism (DM)

Based on availability of resources like CPU, bandwidth, hard disk and memory etc, past history and service dependency data reliable virtual machine is identified. Recourses are estimated by comparing resources available with the resources requested. When requested resources are less than the available resources, those VMs are considered as reliable. Past history is taken from the log of proxy included in M CCS. After completion of the computing cycle, DM adds checkpoint in the log there by maintaining a global state [22]. This past history is useful for system recovery when all VMs fail. Service dependent data can be obtained by Cloudwatch API of Amazon Web Services. It provides whole system visibility of running VMs by supplying different metrics through Cloudwatch API like CPU utilization, bandwidth etc. Thus continuous supervision of user specified system resources and services through time bound checks on them by DM can take place.

3.3 Reliability Analyzer (RA)

Reliability Analyzer assesses the reliability of M CCS by combining reliability of each level of abstraction.

4 Implementation and Evaluation

Amazon EC2 can be used to create VM instances and to set up proxy in M CCS. With the help of AWS Management Console launch a new instance by clicking on Launch Instances as in Fig 2.

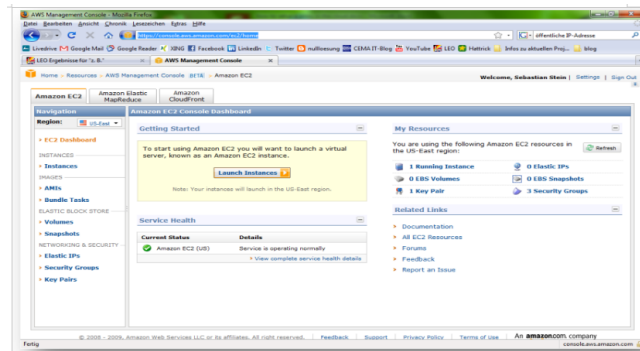


Fig 2: Launch Instances

Select number of instances using Launch Instance Wizard. Then create new VM by click on More Actions Get Windows Password as in Fig3.

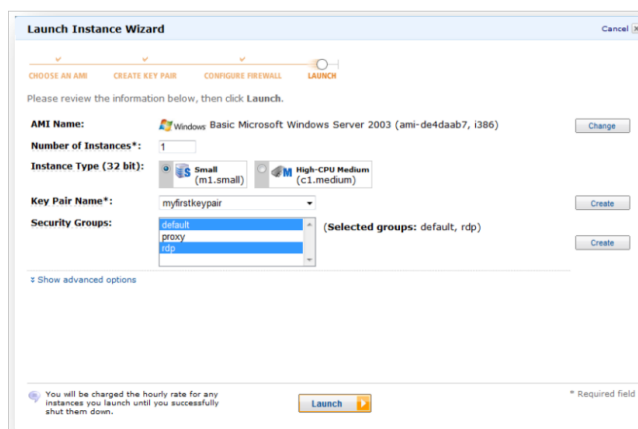


Fig 3: Get Window Password

Then connect the VM to MCCS by AWS Management Console Connect as in Fig 4.

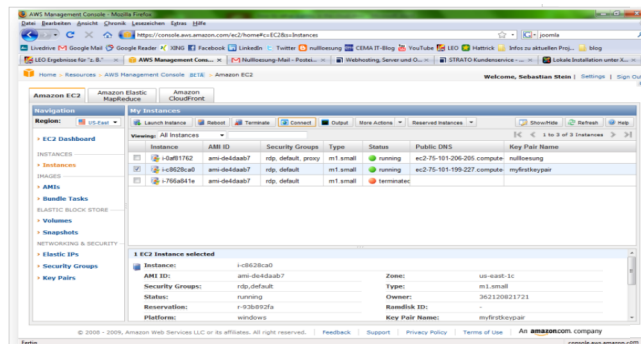


Fig 4: AWS Management Console

Install Free Proxy through wizard as in Fig5.



Fig 5: Free Proxy Wizard

Launch Free Proxy Control Centre and start the Service mode. Then select security options for MCCS and also change your browser proxy settings and Log settings. After invocation of proxy as a service, we implemented LSRAP with the help of Apache Tomcat application server at the application layer, Cloudify for distribution and placement of VMs at virtualization Layer and Ganglia to control and monitor at server layer. The performance of LSRAP is compared with SNF and 4 LSA by taking two parameters response time and invocation time into account is described by Fig 6.

Fig6...

5 Conclusion

This paper illustrates an abstract idea of Layered software reliability assessment paradigm with three layers and can be supported by proxy as a service in MCCS. Amazon EC2 can be used for creation of MCCS. But algorithms can be constructed for assessment of reliability of each layer to support dynamic and elasticity of MCCS environment. Thus we contributed a novel paradigm which facilitates assessment of application reliability in an easy and compartmentalized evaluation mechanism.

References

- [1] Khalid M., Yousaf M.M., Iftikhar Y., Fatima N. (2016) *Establishing the State of the Art Knowledge Domain of Cloud Computing*. In: Sulaiman H., Othman M., Othman M., Rahim Y., Pee N. (eds) Advanced Computer and Communication Engineering Technology. Lecture Notes in Electrical Engineering, vol 362. Springer, Cham, ISBN 978-3-319-24584-3, DOI https://doi.org/10.1007/978-3-319-24584-3_85, 2016.
- [2] Tran H.T., Feuerlicht G. (2015) *Service Repository for Cloud Service Consumer Life Cycle Management*. In: Dustdar S., Leymann F., Villari M. (eds) Service Oriented and Cloud Computing. Lecture Notes in Computer Science, vol 9306. Springer, Cham, ISBN 978-3-319-24072-5, DOI https://doi.org/10.1007/978-3-319-24072-5_12, 2015.
- [3] Pritee Gavhane, Pratibha Kurumkar , Madhuri Jagtap , Dharti Mahajan, *Cloud on Linux Operating System*, International Journal of Computer Science and Mobile Computing, ISSN 2320088X, Vol. 2, Issue. 5, May 2013, pg.6-13.
- [4] Mohan Manoharan and Dr. Swaraswathi, *Multi-Level Service Composition in Supervised Cloud Computing*, International Journal of Computer Science Issues, ISSN (Print): 1694-0814 — ISSN (Online): 1694-0784, Vol. 11, Issue 2, No 2, March 2014 pp 169-177.

- [5] QiangDuan, *Modeling and performance analysis for composite networkcompute service provisioning in software-defined cloud environments*, Digital Communication and networks <https://doi.org/10.1016/j.dcan.2015.05.003>, Vol 1, issue 3, Aug 2015 pp 181-190.
- [6] Prerita Gupta, Dr. Harmunish Taneja, Dr. Gagandeep Singh Brar, *Quality of Services in Cloud Computing: Issues, Challenges and Analysis*, International Journal of New Innovations in Engineering and Technology, ISSN : 2319-6319, Volume 3 Issue 3 July 2015 pp 76-82.
- [7] Nie G., E. X., Chen D. (2012) *Research on Service Level Agreement in Cloud Computing*. In: Hu W. (eds) *Advances in Electric and Electronics. Lecture Notes in Electrical Engineering*, vol 155. Springer, Berlin, Heidelberg, ISBN 978-3-642-28744-2, DOI https://doi.org/10.1007/978-3-642-28744-2_5, 2012.
- [8] www.rightscale.com/
- [9] <https://techcrunch.com/tag/enstratus>
- [10] Kavvo — MarketBite <https://marketbite.wordpress.com/tag/kavvo/>
- [11] mOSAIC Cloud mosaic-cloud.eu/
- [12] www.optimus2020.com/
- [13] Vahid Dastjerdi and Rajkumar Buyya, *A dependency-aware ontology-based approach for deploying service level agreement monitoring services in Cloud SOFTWARE PRACTICE AND EXPERIENCE* Softw. Pract. Exper. 2012; 42 :501518 Published online 27 July 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/spe.1104.
- [14] Smrati Gupta, Victor Muntz-Mulero, Peter Matthews, Jacek Dominiak, Aida Omerovic, Jordi Aranda, Stepan Seycek: *Risk-Driven Framework for Decision Support in Cloud Service Selection*. CCGRID 2015: 545-554.
- [15] Hang Wei and Pei-Li Qiao *Reliability Assessment of Cloud Computing Platform Based on semi quantitative Information*

- and Evidential Reasoning*, Journal of Control Science and Engineering Volume 2016 (2016), Article ID 2670210, 10 pages <http://dx.doi.org/10.1155/2016/2670210>.
- [16] Wenbo Mayo, *The role and effectiveness of cryptography in network virtualization: a position paper*, proceedings of ACM SIGSAC symposium on Information, computer and communication security, ISBN 978-1-4503-1767-2, Hangzhou, China May 08 - 10, 2013, DOI 10.1145/2484313.2484337, 2013.
 - [17] B. Sandhya Rani, Patil. ShivaChandrika, *Switching from Vendor Lock-in to the Meta cloud*, International Journal of advanced research in computer and communication engineering ISSN (Online) 2278-1021 ISSN (Print) 2319-5940 Vol. 4, Issue 4, April 2015 pp 642-646.
 - [18] Dr. C.S Pillai, *A Framework for multi cloud environmental services*, International journal of advanced networking and applications, ISSN : 0975-0290 Volume: 6 Issue: 3, 2014 pp 2338-2341.
 - [19] Sarang Mandloi, Priya Sexena, *Implementation of cloud storage system using multiple cloud services*, International journal of emerging technology and advanced engineering ISSN 2250-2459 Volume 5, Issue 9, September 2015 pp 119-124.
 - [20] Mr. Avinash R. Dhok, Ms. Ashwini P. Kolhe and Ms. Neha V. Lutade, *A Survey on Scalable Data Security and Load Balancing in Multi Cloud Environment ISSN (online): 2349-6010 volume 1, issue 8, Jan 2015 pp 232-236*.
 - [21] Er. Parul Indoria and Prof. Abhishek Didel, *An Approach to Secure Resource Sharing Algorithm (SRSA) for Multi Cloud Environment*, COMPUSOFT, An international journal of advanced computer technology ISSN:2320-0790 Volume-II, Issue-XII, December 2013 pp 405-409.
 - [22] Eszter Kail , Peter Kacsuk and Miklos Kozlovsky , *a novel adaptive checkpointing method based on information obtained from workflow structure Computer Science DOI:http://dx.doi.org/10.7494/csci.2016.17.3.387 Vol 17 No 3, 2016 pp 387-406*.

