Contents lists available http://www.kinnaird.edu.pk/

Journal of Natural and Applied Sciences Pakistan

Journal homepage: http://jnasp.kinnaird.edu.pk/

AUTONOMIC RECOVERY PLAN WITH SERVER VIRTUALIZATION

Seemab Hameed[1], Muhammad Saady[2], Muhammad Saad[3], Sidra Anwar[4*]
[1]Department of Computer Science, Government College Women University Sialkot Pakistan
[2]Department of Information, Technology, Virtual University Sialkot Pakistan
[3]Department of Software Engineering University of Gujrat Sialkot, Pakistan
[4]Department of Computer Engineering Memorial University of Newfoundland Canada

Article Info

*Corresponding Author
Email: sidraa@mun.ca

Abstract

For autonomic recovery with server virtualization, a cogent plan that includes recovery techniques and backups with virtualized servers can be developed instead of assigning an idle server to backup operations. In addition to hardware cost reduction and data center trail, the disaster recovery plan can ensure system uptime and to meet objectives of high availability, recovery time, recovery point, server provisioning and quality of services. This autonomic solution would also support disaster management, testing and development of the recovery site. In this research, a workflow plan is proposed for supporting disaster recovery with virtualization providing virtual monitoring, requirements engineering, solution decision making, quality testing and disaster management. This recovery model would make Disaster recovery lot easier, faster and less error prone.

Keywords
Autonomous Intelligence, Disaster Recovery, Cloud Computing, Server Virtualization

1. Introduction

Cloud computing is a reality in this era and virtualization, a key technology, is used to create computing clouds. Due to this emerging paradigm, the needs for standards to enable the interoperability have been increased. Today, users are willing to take advantage of the flexible services on reduced costs offered by cloud computing but are also concerned about getting locked into one vendor's cloud. Virtualization is increasing the use of existing physical resources for the customers and also reducing the number of systems deployed and managed (Jain *et al.,* 2013). From the perspective of IT and engineering, Cloud Computing with virtualization can improve scalability, high availability, and other non-functional services of the application (Rittinghouse *et al.,* 2016). This also includes the aspects of runtime performance, resource management, privacy and distributed data usage. Virtualization is also viewed as part of emerging IT trends in cloud computing that includes autonomous computing in which the IT environment would be capable to manage itself based on perceived activities (Motochi *et al.,* 2017). From all types of Virtualization, the Server Virtualization spares the user from understanding and managing complex details of server resources while increasing the resource sharing, distribution, utilization and maintaining the capacity to grow further (Binz *et al.,* 2014). Server virtualization is the hiding of server resources from server users. Server virtualization is used to free the client from understanding and accomplishing difficult details about resources of the server when usage and sharing of resource increased and having the capacity to further increase (Naeem *et al.,* 2016). It also minimizes the downtime, enables business continuity and disaster recovery, and simplifies data center management.

1.1 Design of Virtualized Technology

In distributed computing space/memory is essentially dispensed to the clients in the workers which requires a host (stage) on which hypervisor (programming which associates with the equipment) runs (Xing *et al.,* 2012) (Figure 1). The virtualization model is comprising of cloud clients, administration models, virtualized models and its host programming and just as their equipment. Virtualization programming makes it conceivable to run various working frameworks and numerous applications on a similar worker simultaneously," said Mike Adams, overseer of item promoting at VMware, a pioneer in virtualization and cloud programming and administrations (Malhotra *et al.,* 2016)

1.2 Why Virtualization?

With the assistance of virtualization, we can build the utilization of assets accessible to us in numerous to get more advantages. We ought to virtualize in light of the accompanying reasons:
a. Separation among clients: one client ought to be secluded from different clients with the goal that he/she may not get data about the others client's information and use and can't get to

other's information. b. Asset sharing: a major asset can be divided into numerous virtual assets so it tends to be utilized by different clients utilizing virtualization strategy. c. Dynamical assets: reallocation of assets, for example, stockpiling and computational assets is extremely troublesome however on the off chance that they are virtualized, at that point they can be effectively re-assigned. d. Conglomeration of assets: the little assets accessible can be expanded at a huge degree with the assistance of virtualization (Jain *et al.,* 2013).

1.3 Disaster Recovery (DR) in Cloud

Disaster recovery (DR): The preparation process for continuation of organizational systems essential for enterprise survival. DR includes a focus on IT systems and specific control measures in three critical areas: preventive, detective, and corrective (Banerjee *et al.,* 2017). Disaster recovery plan (DRP): Organizational leaders use this term to outline strategies to reconstitute raw technical data from secondary systems and react quickly to replicate IT sources using stand-by applications dedicated before a disruption (Cervone *et al.,* 2017).

Generally, in the cloud environment, jobs of various sizes and different infrastructure requirement pop up concurrently. The major upcoming problem in cloud computing is effective disaster management. Disaster is an unexpected event that occurs in the cloud environment during its operational life time. The harmony among accessibility and all that else is one of the foundations of IT (Almorsy *et al.,* 2016). We as a whole need our frameworks to consistently be there when we need them so do our directors and heads. The issue comes in when you need to adjust accessibility against the stuff to pick up that accessibility. The worries are not just simply cost, it's the intricacy, aptitudes and testing that are the keys to making everything cooperate (Bandela *et al.,* 2015). The possibility that a solitary equipment or programming item will give the capacity to turn out to be more accessible simply doesn't exist today. While the autonomic reinforcement and calamity recuperation items we use today have become all the more wide- extending and viable, the applications have become considerably more intricate (Alhazmi *et al.,* 2013). This consistent race between such autonomic applications and accessibility of them brings about huge scope blackouts when the debacle recuperation items we have neglects to stay aware of our application needs and plans.

The term autonomic originates in biology referring to unconscious systems controlling basic functions like your heart rate; it takes on new significance when applied to innovation. For sure, autonomic knowledge has empowered programming capacities to advance from fundamental information assortment, announcing and making aware of really deciding, upgrading human intellectual capacity (Al-Shishtawy *et al.,* 2013). It empowers people to oversee progressively complex situations at new degrees of scale. Server farms and mists

driving applications like Google maps speak to another open door for the Autonomous Intelligence (AI) insurgency. For quite a long time, server farms expected people to make all application organization, position and estimating choices (Mayer *et al.,* 2013). Sadly, human oversight can no longer convey the exhibition vital for a web-scale world where "moderate" presently signifies "down." Today, a kind of autonomic knowledge ensures application execution across private, crossover and multi-cloud server farms by empowering applications themselves to choose the best foundation (Singh *et al.,* 2016) on which to run successfully self-arranging: An autonomic stage controlling cloud administrations.

There are various oversights and missteps that IT experts can make while making a catastrophe recuperation plan for such autonomic virtualized situations. One normal error is to choose an apparatus or administration that is essentially a lot for the business (Chang *et al.,* 2015). This could be an instrument or administration that is excessively mind boggling, with highlights and usefulness that the business doesn't utilize or even need, or devices that are hard for the accessible IT staff to design or use.

Another normal slip-up incorporates choosing an instrument that is lacking for the association's DR objectives (Ginis *et al.,* 2014). For instance, a device or administration that solitary ensures recuperation guide goals of 15 toward 20 minutes isn't the most ideal decision for a strategic Virtual Machine (VM) outstanding task at hand that requests practically constant replication. Likewise, an organization that requires simultaneous replication presumably won't accomplish satisfactory outcomes with far off replication locales because of abundance dormancy or system bottlenecks (Brender *et al.,* 2013). These sorts of errors happen when IT experts neglect to appropriately survey DR prerequisites or enough assess a DR device before they make a dedication.

A third issue territory is lacking thought of the replication or DR site. For instance, it's conceivable to back up or repeat outstanding burdens to nearby capacity, yet this represents a solitary purpose of disappointment for the business. On the off chance that the neighborhood site is undermined, nearby DR substance may be useless or unrecoverable. Managers ordinarily influence distant destinations or cloud suppliers for versatility, however it's imperative to choose a site for its security and area (Latif *et al.,* 2014). You need a stage that is sufficiently close to limit idleness, yet far enough to dodge nearby risks, such as flooding or flames.

At last, an absence of testing regularly sabotages DR plans. IT staff cautiously create DR plans, yet neglect to test those plans normally and neglect to modify plans as business needs move after some time (Dutta *et al.,* 2013). The outcome is a recuperation plan that is obsolete and troublesome and, at times, even difficult to actualize when it's really important. When an administrator has made and actualized a DR

plan, he should test the arrangement as much of the time as could be expected under the circumstances and survey it normally to guarantee he meets its evolving needs. Present day virtualized DR devices and administrations for the most part uphold nondisruptive recuperation testing, which permits IT staff to confirm the capacity to actualize recuperation rapidly and adequately.

Calamity arrangement and recuperation arranging stays a basic cycle for each business. Virtualized frameworks, for example, VMware vSphere, help rearrange a few parts of the VMware DR plan, yet IT experts should in any case select the most reasonable devices and execute a proper arrangement, and all the while dodge basic impediments and traps that plague undertaking DR endeavors. There is not a viable alternative for satisfactory DR consideration and arranging.

2. Recovery Plan

This is the problem solved, by generating an autonomous platform plan for specific workload placement, sizing and provisioning decisions based on real-time demand at every layer of the stack. Furthermore, it allows a recovery site to automate these decisions to prevent performance degradation before it even started. Server administrators can choose from a variety of outstanding tools to handle failures/outages/ disaster recovery but tools are not enough, and even the best tools can be ineffective in the absence of a cogent disaster recovery plan to be followed by any autonomic platform. It can also be considered as autonomous planning and decision making for outages and disaster recovery solutions. Most well-considered aims of this platform can be broken down into following general steps i.e.

2.1 Server Provisioning

For improved virtual disaster recovery, it's important to keep an eye on server provisioning. Production machines that are not enabled for DR can cause big headaches later. As good DR doesn't just happen; it has to be planned, managed effectively and tested appropriately.

2.2 Assess Your DR Goals

Assessment is basically the process of gathering disaster recovery (DR) requirements with thorough determination of DR goals. Assessment involves:

- Rate current performance
- Review past incidents
- Inventory current assets and capabilities
- Map regulatory requirements
- Detail performance goals and expectations

2.3 Design a DR Plan

The design process crafts a DR plan. This often includes to process:

- Prioritize data, asset and application recovery process
- Identify technology and service needs
- Determine desired compute and storage targets

2.4 Deploy A DR Platform

When a DR plan is set up and a reasonable device is chosen, the business can implement the new DR stage for the virtual condition. Deployment may include:

- Installing and configuring additional infrastructure as needed
- Installing and configuring the new native or third-party DR platform
- Test and refine initial DR activity
- Actively manage the DR tool or service
- Reviewing DR messages and reports
- Verify that workloads are being protected as intended
- Making periodic changes to DR plan activity as business workloads and requirements
- Change over time.

2.5 Manage the DR Plan

Finally, a Virtual DR plan should never be a static entity. It should be reviewed and updated the DR plan periodically on the basis of sound decision making to ensure that it meets the needs of changing workloads and the business. For example, if a system's workload is more or less important to the business, it might be possible to adjust the resources allocated to protect it while new workloads should easily be added to the DR plan.

2.6 Testing

- Test effectiveness of plan through scenarios and live tests.
- Assess effectiveness of DR plans when activated.
- Measure performance vs. standards and expectations.

2.7 Control/ Monitor

- Implement management and oversight of DR assets
- Administer change management
- Conduct periodic audits
- Schedule system updates
- Schedule policy plan reviews
- Test systems at least annually

2.8 Conceptual Framework

The proposed workflow plan is modeled in "Fig.1" for supporting disaster recovery with virtualization providing virtual monitoring, requirements engineering, solution decision making, quality testing and disaster management. At each levels of the framework, following activities are supposed to be performed according to autonomic/solution recovery plan:

a. Level 0 (L0):

Virtualization of all stacks, dependencies and resources based on requirement engineering

b. Level 1 (L1):

Incident management in Table 1, regarding following variables.

Table 1: LEVEL 1

Monitoring & Management	**Access**	**Inventory**
• Server • Dependencies • Environmental	• Remote • Virtual	• Physical • Virtual

c. Level 2 (L2): Disaster Management

- Capacity Forecasting
- Disaster Process Management
- Business Intelligence

d. Level 3 (L3): Recovery & Problem Management

e. Control

f. Automation

g. Level 4 (L4): Infrastructure Management

h. Testing

i. Real Time Optimization

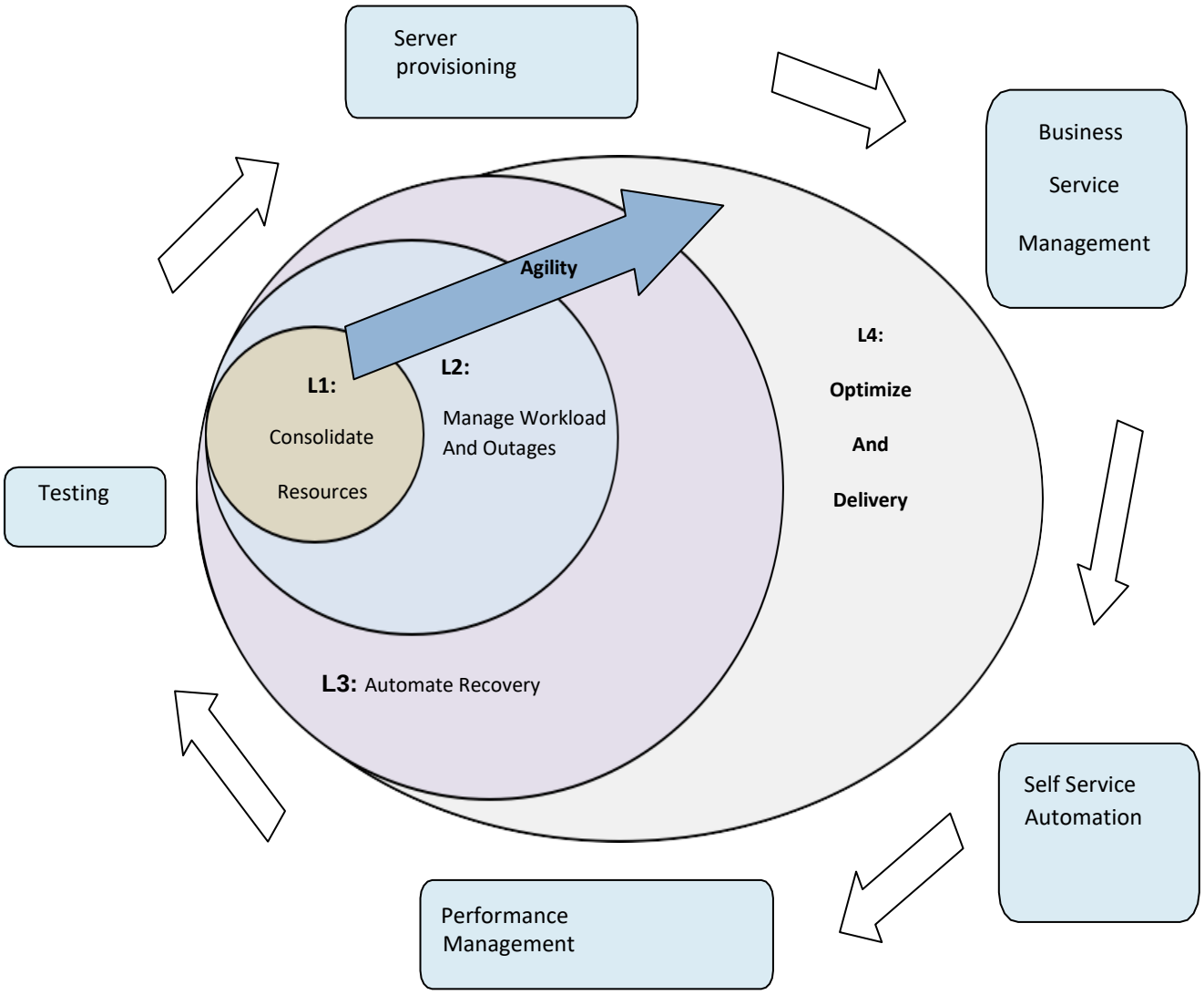

Figure 1: Recovery Framework

3. Discussion

As our environments scales, so does the challenge of understanding our resources, their interdependencies and the implications that an action might have on the entire system. Cloud Data centers supply multiple resources consumed by the demand of applications running in our environment, leaving our IT teams to determine the exact blend of resources each application needs to perform (Xiao *et al.,* 2013). Meeting that need requires implementable actions based on real- time understanding of application demand and resource supply (Avram *et al.,* 2014). With various elements of assets (CPU, MEM, Network, IOPS, and so on.), different planes of use request (reaction time, exchange throughput, and so forth.) and numerous moves to be made (arrangement, estimating and provisioning remaining burdens) (Chang *et al.,* 2015). Our IT team is left with a N- dimensional problem causing various disasters which is impossible to solve by humans alone or by a single recovery tool (Anwar *et al.,* 2015).

While people can absolutely make contents that complete activities on an on it and then base, the hardest part is choosing which activities are suitable in which situations, when to take them and what the expanding influences will be over the whole condition once they are taken. Imagine a scenario where notwithstanding robotizing the activities' execution, we could autonomically control the dynamic cycle behind them. We are past due for another path in taking a gander at debacle recuperation and blackouts. How about we plan our frameworks with the possibility that we will have a blackout rather than just attempting to forestall them. Grasping disappointment gives us genuine application versatility on the grounds that the disappointment insurance is not, at this point just surface thick. We would then be able to test and show that we can deal with disappointment.

Virtualized disaster recovery plan has taken the complexity out of DR, but many administrators misunderstand the benefits of this paradigm shift in the virtual world. Critically,

Virtual DR plan must still remain true to the end result of DR. The key pillars of this DR Plan are:

• Recovery time objective: How some time before frameworks are accessible once again.

• Recovery point objective: The point in time from which the information ought to be accessible.

• Functionality: Ensures the DR example capacities true to form, and that servers come up as arranged and work as expected.

• Autonomicity Objective: The complexity of DR through the achievement of self-governance (autonomy) and self-management (autonomicity) is controlled.

• Recovery Test Objective: After the creation and implementation of a DR plan, testing the plan would ensure the system meets its changing needs.

• Server Provisioning Objective: Virtual Servers are well planned, managed effectively and tested appropriately.

4. Conclusion

By generating an autonomous plan for specific workload placement, sizing and provisioning decisions based on real-time demands at every layer of the stack, it allows automating such decisions to prevent performance degradation before it even starts. Server administrators can choose from a variety of outstanding tools to handle failures/outages/ disaster recovery but tools are not enough, and even the best tools can be ineffective in the absence of a cogent disaster recovery plan to be followed by any autonomic platform. It can also be considered as autonomous planning and decision making for outages and disaster recovery solutions. The preliminary requirement to use the offered system is the virtualization all the dependencies and whole stack too.

5. References

Al-Shishtawy, A., & Vlassov, V. Elastman, (2013), "Autonomic elasticity manager for cloud-based key-value stores", In Proceedings of the 22nd international symposium on High-performance parallel and distributed computing.115-116.

Alhazmi, O. H. & Malaiya, Y. K. (2013) "Evaluating disaster recovery plans using the cloud," in Reliability and Maintainability Symposium (RAMS), Proceedings-Annual, 1–6.

Almorsy, M., Grundy, J., & Müller,I. (2016) "An analysis of the cloud computing security problem," *ArXiv Prepr*. ArXiv160901107.

Anwar,S &Nisa,K. (2015) "Web Based Goal Structuring Notation Tool". *International Journal of Research* (IJR), 7(2), 517–522.

Avram, M.-G. ,(2014) "Advantages and challenges of adopting cloud computing from an enterprise perspective," *Procedia Technology*., 12, 529–534.

Banerjee, P. K., & Biswas, S. US Patent No. 9,582,379. (2017). Washington, DC: US Patent and Trademark Office.

Binz,T., Breitenbücher, U., Kopp, O., & Leymann, F (2014) "TOSCA: Portable Automated Deployment and Management of Cloud Applications," in Advanced Web Services, Springer, New York, NY, , 527– 549.

Brender, N. & Markov, I. (2013) "Risk perception and risk management in cloud computing: Results from a case study of Swiss companies," *The International Journal of Information Management*., 33(5), 726–733.

Chang, V (2015) "Towards a Big Data system disaster recovery in a Private Cloud," Ad Hoc Networks,. 35, 1570-8705.

Cervone, H. F. (2017). "Disaster recovery planning and business continuity for informaticians", *Digital Library Perspectives*, 33, 78-81. doi:10.1108/DLP-02-2017-0007

Dutta, A., Peng, G. C. A. & Choudhary, A (2013) "Risks in enterprise cloud computing: the perspective of IT experts," *Journal of Computing and Information Science in Engineering*. 53(4), 39–48.

Ginis,G., Kerpez, K. J., Cioffi, J.M., Goldburg,M., Galli,S &. Silverman,P (2014) "Software-defined access networks," *IEEE Commun. Mag*., 52(9), 152–159.

Jain,R & Paul,S (2013) "Network virtualization and software defined networking for cloud computing: a survey," in *IEEE Communications Magazine*, 51(11), 24-31. doi: 10.1109/MCOM.2013.6658648.

Latif, R., Abbas, H., Assar,S.,& Ali,Q. (2014) "Cloud computing risk assessment: a systematic literature review," *Future Information Technology, Springer*. 285–295.

Malhotra,L., Agarwal, D & Jaiswal A (2014). "Virtualization in Cloud Computing" *Journal of Information Technology & Software Engineering*, 4:2 DOI: 10.4172/2165-7866.1000136

Mayer, P., Klarl, A.,Hennicker,R., Puviani,M., Tiezzi,F., Pugliese,R., Keznikl,J., Bureš,T. (2013):"The autonomic cloud: a vision of voluntary, peer-2-peer cloud computing," IEEE 7th International Conference on Self- Adaptation and Self-Organizing Systems Workshops (SASOW) , 89–94 , doi: 10.1109/SASOW.2013.16.

Memon , H M F.,.Naeem, M., Siddique, M., Rauf , A. (2016), "An overview of virtualization & cloud computing" *Science .International* .(Lahore),28(4),3799-3803.

Motochi, V., Mbuguah, S. M., & Mbandu, S. A. (2017) "Factors that influence the choice of virtualized environments in small medium enterprises". *International Journal of Advanced Research in Computer Engineering & Technology*.

Ramesh, G. (2015). "Survey On Cloud Computing Technologies And Security Threats", *International Journal of Research and Applications*. 2. 296-308. 10.17812/IJRA/2(6).

Rittinghouse, J. W., & Ransome, J. F (2016), "Cloud computing: implementation, management, and security", *CRC press,. Networking*., 35, 65– 82.

Singh,S & Chana,I (2016) "QoS-aware autonomic resource management in cloud computing: a systematic review," *ACM Computing. Survey. CSUR*, 48(3), 42.

Xiao, Z., Song, W., & Chen, Q. (2012). "Dynamic resource allocation using virtual machines for cloud computing environment", IEEE transactions on parallel and distributed systems, 24(6), 1107-1117.