ABHINAV
International Monthly Refereed Journal of Research In Management & Technology
ISSN – 2320-0073 Volume II, July'13

PORTABILITY/INTEROPERABILITY, RELIABILITY AND CLOUD PERFORMANCE IN CLOUD COMPUTING ENVIRONMENT

Hatim Kanpurwala

Consultant, Technology & Management, Mumbai, India
Email: hatimk@rediffmail.com

INTRODUCTION TO CLOUD COMPUTING ENVIRONMENT & ASSUMPTIONS

A Cloud consists of four basic entities, namely Application, Service, Platform or device/machine and Server e.g. Application could be a purchase & payment through say a HDFC credit card, Service could be a portal through which you as a user is making the purchase/payment powered by a payment gateway of a bank say ICICI Bank or Citibank, Platform could be your PC and a VeriFone swiping machine for the credit card and Server could be an IBM Server running Linux/Unix or Windows NT supporting the Application, Service and Platform. Now a Cloud could consist of several such Applications, Services, Platforms and Servers (maybe at a Datacentre) e.g. Cloud 1 could consist of Applications A1, A2…Am, Services S1, S2…Sn, Platforms P1, P2…Pl and Servers Sv1, Sv2…Svo. And the Cloud 1 could consists of several configurations e.g. Cfg1 may consist of A1, S2, P2,P3 and Sv3, Sv4 and so on and so forth for other Configurations. The observations made with respect to Cloud Computing have not been verified nor a literature survey done – this is just the authors understanding of Cloud Computing environment and a belief the way Cloud Computing Environment should function or works. N.B. Each configuration is serving some user needs such as u1, u2…un and the sum of all these user needs consists of a particular customer segment(s) the Cloud is meant for.

OBSERVATIONS

Suppose a configuration is not working for some reason either the Application or Platform or Service or Server is down due to failure. In such a case if the Platform, Service or Server is fault tolerant the backup device/software will take over i.e. operate. In absence of Fault tolerance there are two possibilities –

a) Either the cloud reconfigures itself for the user's needs/requirements or the user need/requirement is met by a different Configuration within the Cloud such that integrity of data is maintained and integrity of user's needs/requirements is met e.g. if a user wants to make a purchase using Citibank credit card on eBay portal (Let us call this need u1). Let us assume that this user need/requirement u1 is met using Cfg1 (namely A1, S2, P2, P3 and Sv3, Sv4). Now if u1 can be met by Cfgx in the Cloud, the Cloud need not reconfigure for the user need u1, in case Cfg1 becomes dysfunctional. But if u1 cannot be met by any Configuration in the Cloud, then the Cloud reconfigures itself such that u1,u2…un is met and the sanctity of data is maintained for u1, u2,…un.

b) There is either unilateral or bilateral Portability/Interoperability between the Cloud operators for its different configurations for different user needs/requirements. This entails knowing the entities in the Cloud and the ability to connect/network with/between them. Again just as in a) when a Configuration becomes dysfunctional for a user need, this may be met externally by using other Clouds resources (with or without reconfiguration).

N.B. The above Portability/Interoperability of user needs/requirements within and between Cloud environments necessitates Identity management and configuration management in Cloud Computing environment.

CONCLUSION

The higher is the Portability/Interoperability of user needs/requirements within and between Cloud environments and the higher is the Fault tolerance of a Cloud(s), the higher is the reliability/performance and uptime of the user needs/requirements for different Cloud(s).

We assume that the Identity Management and Configuration Manager in a Cloud(s) is typically hidden from users and the engine for Identity Management and Configuration management is typically governed by business alliances between user needs/requirements of a Cloud or between Clouds e.g. if there are

User needs u11, u12…..u1n for Cloud 1

User needs u21, u22,….u2n for Cloud 2

…….Etc.

Then bilateral business alliances between some combinations of say uxy and uij (User needs) may determine or govern the sharing of resources/configurations of uxy and uij e.g. let u11 be a Citibank credit card purchaser on eBay and u12 be an ICICI credit card purchaser on eBay. Let us say there is a business alliance between Citibank and ICICI, then when Citibank's payment gateway is down, the Citibank's credit card user might use ICICI bank's payment gateway transparently internally by reconfiguring Citibank's configuration to use ICICI bank's payment gateway when the former becomes dysfunctional till such a time that Citibank's payment gateway becomes operational. This ensures higher reliability and performance for a Citibank customer/user due to the Business alliance between Citibank and ICICI. N.B. u11 and u12 maybe either internal to each other in the same Cloud or across Cloud(s).

IMPLICATIONS AND POTENTIAL APPLICATIONS

To ensure higher reliability and performance the higher or more is the number of possible permutations/combinations of the Configurations for the User needs u1, u2…un etc. the higher is the reliability and performance of the Cloud(s).

SUMMARY AND SCOPE FOR FURTHER RESEARCH

The permutation/combinations for the entities in Clouds are large which can give the Configuration Manager endless possibilities for the engine to compute Configuration(s) for user need(s). This can lead to endless scope for further research.