

P2301 and P2302 on Standards for Intercloud Interoperability and Federation have focused on the way the clouds can interoperate and standards defined. The present research was undertaken to understand and implement such a use case scenario where the standard interfaces, or, the cloud brokers were defined between the private network and the public one. The efficacy of the implementation was observed and it was seen that through some amount of fine tuning, an autonomic broker is a reality.

3. Federation in the Cloud

Federation in the cloud is achieved when a set of cloud providers voluntarily interconnect their infrastructures in order to allow sharing of resources among each other. The federation partnership in clouds leads to the establishment process which can be schematized according to the following three main phases:

- **Discovery:** The cloud looks for other available clouds for federation.
- **Match-making:** The cloud selects between the discovered clouds the ones that fit as much as possible its requirements.
- **Authentication:** The cloud establishes a trust context with the selected clouds.

When the federation in the cloud is established, a new phase becomes very relevant, that is, the “management” of the federated resources. Federation brings to fore new use cases to the cloud computing paradigm. In fact besides, the traditional scenario where cloud providers offer cloud-based services to their clients, federation triggers a new paradigm where cloud providers can buy and/or sell computing/storage capabilities and services to other clouds. For example, a cloud might need to procure resources from other clouds in the following manner:

- **Storage Space** - The cloud runs out of its storage/computing resources. In order to continue providing cloud-based service to its clients, it decides to procure resources from other clouds.
- **Distributed Availability** - The cloud needs to deploy a distributed cloud-based service in different geographical locations; hence, it acquires resources placed in those target locations.
- **Migration and Stability** - The cloud need to migrate cloud-based service instances in other clouds in order to accomplish service relocation, power saving, backup, etc. At the same time, a cloud can decide to provide resources to other clouds when it realizes that its datacenter is under-utilized at given times. Typically, datacenters are under-utilized during the night and over-utilized during the morning. Therefore, as the datacenter cannot be turned off, the cloud provider may decide to turn the problem into a business opportunity. This case may be applied to many different organizations, such as, universities, public administrations and enterprises.

4. Use Cases for Cloud Brokers Using Federated Clouds

The authors have conducted several simulations and real world scenarios implementation on the Greencloud 2.0 and the AWS cloud in the past for testing and simulating multiple cloud based request provisioning scenarios [13, 20, 21]. This has been done using multiple concurrent user requests and a Broker attempting to enable the seamless availability of services in the cloud. This research is based on live experiments made by the author on a federated cloud environment with multiple data centers talking to each other, either centrally or through a third party utility/cloud broker.

The authors in this paper have focused the research on cloud brokerage as the *discovery and matchmaking phase* needed for the creation of a federation of clouds. We envisage four possible federation schemas in which this cloud brokerage may take place. These are:

- Composite Schema
- Hierarchical Schema
- Distributed Schema
- Hybrid or Mixed Schema

While there are numerous use cases which can be extrapolated using the four schemas, for the present research, the authors have studied the Composite and Distributed Schemas and conducted actual experiments based on the scenarios provided below. It is pertinent to mention, that earlier work by the authors in the same field were on the implementation of the broker in the AWS cloud [20, 21].

4.1 Use Case I – Composite Federated Schema

The Composite Federated Schema is for a use case where a single broker, common to all clouds, is in charge of establishing the federation. Such schemas are often visible in private clouds which are disparate and distributed across geographical locations. The schema was extensively tested across the private network of the authors and the figure 1 illustrates the physical disposition of the data centers in India housing the servers. The authors implemented a private network to create a distributed engine based on the Drupal 7 platform which was used to create an application capable of hosting multiple service scenarios. The centralized broker was housed at the local facility of Jaipur, India and multiple request scenarios were generated and pushed through the broker. The analysis engine for recording the results was New Relic [22] and a customized version of the Drupal version 7 framework was used to create the central loading and testing utility. The PHP library SimpleCloud was used to implement the Cloud paradigm on the Drupal Interface.

In the case of the Composite Federated Schema, data centers at Jaipur, Delhi, Lucknow and Indore were used to create the cloud infrastructure. The data center at Kolkata, India was used as a limb to test certain externalities. The base tests were around the adoption and creation of new VMs across the server stacks and the usage of additional memory beyond the baseline memory usage available for standard servers.

Different use case scenarios depicted in the tables below indicate the different conditions and load tests generated.

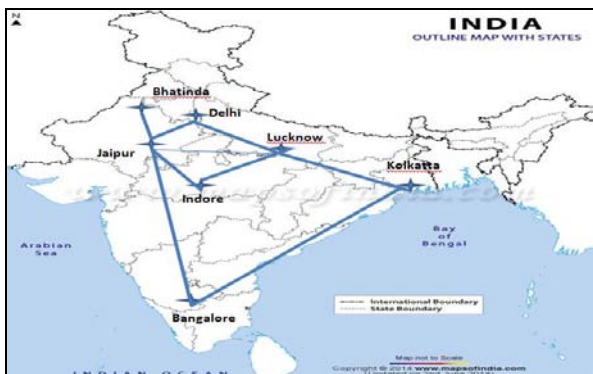


Figure 1: Location of the Data Center Servers within the Author's Country and Private Cloud

Extensive test cases were created and the load conditions were generated to test the cloud provisioning. The test case scenario for the Composite Schema load tests under live traffic conditions was as under in Table 1:-

Table 1: Load testing on the private cloud of the authors using the Composite Federated Schema

Scenario	Load Test Requests	Servers in Use	Time Duration	Type of Requests
Scenario 1	100 Searches	5 Concurrent Servers at Three Locations	60 - 400 secs	Singular Jobs
Scenario 2	60 Searches on the Server + 40 across servers	5 Concurrent Servers at Two locations	60 - 400 secs	Batch Jobs
Scenario 3	50 + 50 Search and Swaps on the Server	4 Concurrent Servers at Three Locations	60- 400 secs	Periodical Jobs (50% singular mixed)

The test results for the load generated for the Scenario depicted above using the data centers that are geographically distributed, as shown in figure 1 and having a similar hardware configuration. The tests cases illustrated in Table 1 were conducted across a seven day period and to normalize the results, 10 separate sets of similar experiments were conducted for each use case to eliminate network latency issues and to ensure that latency errors were reduced to a minimum. Three types of jobs, singular, batch, and periodical jobs were sent to the cloud broker through the network and the response of the broker was analyzed against in terms of successful job execution. The results of the test are depicted in Table 2.

Table 2: Load test results for Composite Federated Schema using multiple request scenarios

Scenario	Servers	Function	Test Duration	Request per Sec	Request Time	CPU Load
Scenario 1	5	Search	377 Sec	0.27	3.7	10%
Scenario 2	5	Search	141 Sec	0.71	1.4	13%
Scenario 3	5	Search + Swap	125 Sec	0.79	1.2	18%

The CPU and memory load using the customized cloud Broker in the two main scenarios were as depicted in the graphs below. It is pertinent to note that the load conditions on the servers were severe but the resultant distribution and accomplishment of the requests was Very High when the composite scenario was used with the proposed broker located at Jaipur. In the scenarios depicted, the broker used initially was a linear model cloud broker which acted in a centralized fashion and used a simple FIFO queue to push the cloud request to the cloud stack. However, an improved version of the same broker using a semi-recommender system to remember the cloud stack usage in previous requests and utilizing the knowledge to provide better services for future request in the second scenario to a much better result.

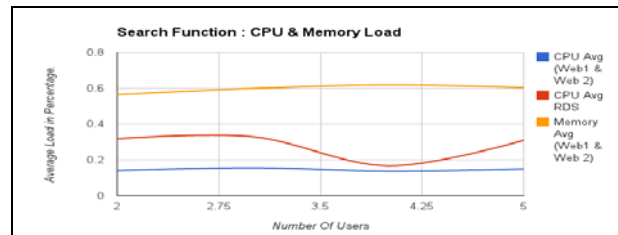


Figure 2: The CPU load condition in composite federated schema

The load condition for a combination of searches and swaps on a distributed engine that allows clients to search for properties and home exchanges (a multiple, mixed load) shows improved results even when the loading factor is high. The load (higher number of users per second) is a direct reflection on the load generator capability and simulates the efficacy of the cloud broker. This observation is further asserted in the next figure which shows an even better performance of the Broker under severe load conditions, across geographically disparate locations and variable conditions of request arrival. A predictive load analyzing engine was adopted in the broker configuration to improve the efficiency of the system and increase the number of accepted requests.

4.2 Use Case II – Distributed Federated Schema

In the Distributed Federated Schema tested by the authors, the broker is acting as medium to integrate the services which are available across distributed servers/data centers which do not necessarily talk to each other. The broker would be required to implement suitable mechanisms to make the service providers talk to each other and often acts as a third party to implement the resource request. There is a use case here for a broker to act as an intermediary and an aggregate or of multiple existing or new resources. The schema was implemented across Jaipur, Indore, Kolkata, Bangalore (CAIR) and Bhatinda in the present research. The data centers across the locations were given a base line install and the traffic requests were routed through the broker available at Jaipur and Kolkata. The brokers available for the test were using a serial logic to implement the request. The use of a recommender system developed by the authors (in research) was used to optimize the request provisioning. The General Meta-Broker Service, as described by [23], was utilized for describing the requested service and as far as concerned, the

provisioning of the service was transparent to the client utilizing the service.

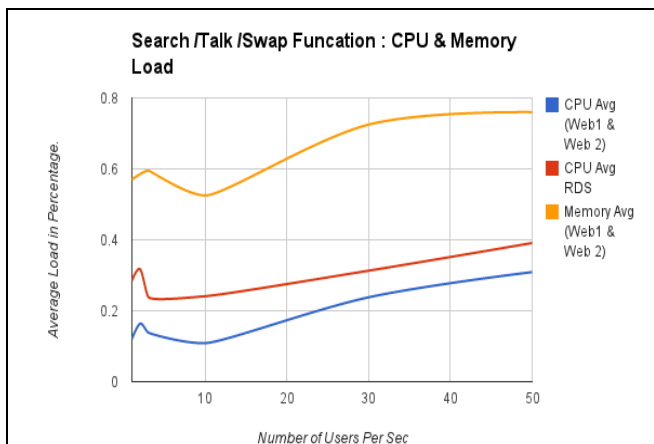


Figure 3: The load conditions in Scenario 2 (Search plus Swap) for Composite federated Broker

The test case scenarios for the Distributed Federated Schema load tests, given in Table 1, under live traffic conditions were similar to the Composite one. The metrics for obtaining the results were however dependent upon the response time a request took to be completed. The number of request which were not delivered as a result of over loading was also a metric which was checked again default conditions and against the special case of implementation of the broker.

Table 3: Results of Distributed Federated Schema results

No of clients per Sec	CPU Avg (Web1 & Web 2)	CPU Avg RDS	Memory Avg (Web1 & Web 2)
2	14.05%	31.80%	57%
3	15.60%	33.00%	60%
4	13.75%	16.70%	62%
5	14.95%	31.00%	61%

Similarly, the results of the distributed federated schema when multiple on the CPU load when multiple clients were accessing the services of the data centers are in Table 4. RDS indicates the Relational Database Server (MySQL) server instance used for provisioning of the web service.

Table 4: Federated Schema using multiple clients at Jaipur

No of clients per Sec	CPU Avg (Web1 & Web 2)	CPU Avg RDS	Memory Avg (Web1 & Web 2)
1	12.05%	28.50%	57%
2	16.35%	31.70%	59%
3	13.85%	23.90%	60%
10	10.87%	24.10%	53%
30	23.80%	31.30%	73%
50	30.90%	39.10%	76%

The time for search of a resource (the response time) and eventual provisioning of the resource when requested for by multiple clients in a Distributed federated Schema is presented in figure 4 below. The results indicate that the proposed broker is scalable and its impact does not deteriorate with the increase in the number of simultaneous clients requesting access to the same resource across multiple data centers.

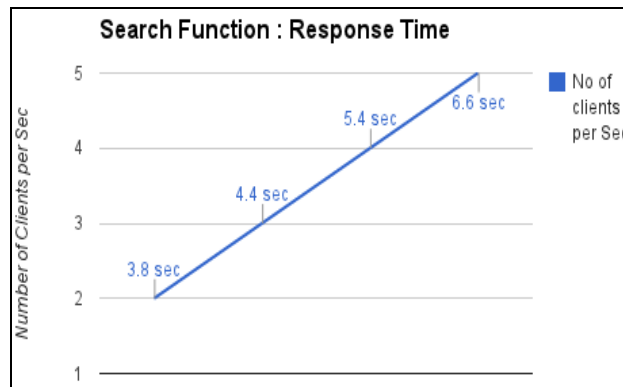


Figure 4: Response time of the broker for search and provisioning of web services in the Distributed Federated Scenario

5. Conclusion

The research is a step by the authors towards the actual implementation of a cloud broker in a multitude of scenarios using a Federated Cloud Schema. The tests conducted during the course of this research were aimed at reviewing the customized cloud broker schema created by the authors against the one available in the open domain. The authors have studied two salient use cases in the Federated Schema for Cloud provisioning, i.e. Composite and Distributed Schema, and would undertake another study to analyze the performance of the cloud broker in the other two in subsequent research.

References

- [1] NIST, "The NIST Definition of Cloud Computing", National Institute of Standards and Technology, 2011.
- [2] Foster, I., Zhao, Y., Raicu I., Lu, S., 'Cloud Computing And Grid Computing 360-Degree Compared', IEEE Grid Computing Environments Workshop, , pp1-10, 2008
- [3] D. Bernstein, E. Ludvigson, K. Sankar., S. Diamond and M. Morrow, "Blueprint for the Intercloud - Protocols and Formats for Cloud Computing Interoperability," in Fourth International Conference on Internet and Web Applications and Services, Venice/Mestre, pp328-336, 2009.
- [4] C. A. Yfoulis and A. Gounaris, "Honoring SLAs on cloud computing services: a control perspective," in Proc. 2nd Workshop on Bio-inspired Algorithms for Distributed Systems, 2009.
- [5] M. Alhamad, T. Dillon and E. Chang, "A survey on SLA and performance measurement in cloud computing," in Confederated International Conference on the Move to Meaningful Internet Systems, pp469-477, 2011.
- [6] F. Jrad, T. Tao and A. Streit, "SLA Based Service Brokering in Intercloud Environments," in 2nd International Conference on Cloud Computing and Services Science, pp76-81, 2012.
- [7] M. Alhamad, T. Dillon and E. Chang, "SLA-Based Trust Model for Cloud Computing," in 13th International Conference on Network-Based Information Systems, pp321-324, 2010.

- [8] D. D. Lamanna, J. Skene and W. Emmerich, "Slang: A language for defining service level agreements," in Ninth IEEE workshop of Future Trends of Computing, pp100-106, 2003.
- [9] M. Macas, J. O. Fit and J. Guitart, "Rule-based SLA management for revenue maximisation in Cloud Computing Markets," in International Conference on Network and Service Management , pp354-357, 2010.
- [10] Toosi, A. N., Calheiros, R. N., & Buyya, R. (2014). Interconnected Cloud Computing Environments: Challenges, Taxonomy, and Survey. ACM Computing Surveys (CSUR), 47(1), 7.
- [11] R. Buyya, R. Ranjan and R. N. Calheiros, "InterCloud-Utility-Oriented Federation of Cloud Computing Environments for Scaling of Application Services," in 10th International Conference on Algorithms and Architectures for Parallel Processing, pp13-31, 2010.
- [12] D. M. Parrilli, "The determination of jurisdiction in grid and cloud service level agreements," in 6th International Workshop on Grid Economics and Business Models, pp128-139, 2009.
- [13] P Khanna, S Jain, BV Babu, "Cloud Broker: Working in Federated Structures A Hybrid Cloud Live Performance Analysis", in International Conference on Advances in Computing, Communications and Informatics (ICACCI-2014), IEEE, 2014
- [14] S. Venticinque, R. Aversa, B. Martino, M. Rak and D. Petcu, "A cloud agency for SLA negotiation and management," in Conference on Parallel Processing, pp587-594, 2010.
- [15] P. Balakrishnan and T. S. Somasundaram, "SLA enabled CARE resource broker," in Future Gener. Comput. Syst, pp265-279, 2011.
- [16] NIST, "500-293, US Government Cloud Computing Technology Roadmap, Release 1.0 (Draft), Volume II Useful Information for Cloud Adopters, November 2011," NIST Special Publication, Washington, 2011.
- [17] 10 Aug 2014. [Online]. Available: <http://aws.amazon.com/ec2>.
- [18] Gartner, "Gartner's Top IT Predictions for 2011-2015," 2010.
- [19] "Gartner Brokerage," 9 July 2009. [Online]. Available: <http://www.gartner.com/newsroom/id/1064712>. [Accessed 17 Apr 2014].
- [20] P. Khanna and B. V. Babu, "Cloud Computing Brokering Service: A Trust Framework," in the Third International Conference on Cloud Computing, GRIDs, and Virtualization, Nice, pp206-212, 2012.
- [21] P. Khanna and S. Jain, "Cloud Broker: Reducing Complexity through Autonomic Services," in Doctoral Conference Shodh 2013, JKLU, India, 2013.
- [22] "New Relic," 10 Jun 2014. [Online]. Available: <http://newrelic.com/>.
- [23] A. Maros, G. Kecskemeti, A. Kertesz and P. Kacsuk, "FCM: an Architecture for Integrating IaaS Cloud Systems.," in Proceedings of the Second International Conference on Cloud Computing, GRIDs, and Virtualization, Rome, 2011

Author Profile



Prashant Khana has over 20 years of industry and government experience in network and data center roll outs in India. He has authored several publications in national and international journals, including military publications. He is an ME in Internet Science and Engg from Indian Institute of Science, Bangalore and is presently pursuing his PhD from JKLU, Jaipur, India. He has interests in ICT rollouts, Cloud Computing and Brokering frameworks and Analytics in the Cloud. He is a member of IEEE P2302 Workgroup for Intercloud Standards.



Dr. Sonal Jain has over 14 years of academic experience. Dr. Jain has several publications to her credit and has presented research papers at National and International conferences organized by institutes like IITB, IITK and, NTU Singapore. Dr. Jain has co-authored several books including Personal Computer Software (Wiley India), Simplifying C (Dreamtech Publications) and five text books for Gujarat State Education Board. Dr. Jain is life member Computer Society of India and has served as Management Committee Member of Ahmedabad Chapter.