

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/283846908>

BroCUR: Distributed cloud broker in a cloud federation: Brokerage peculiarities in a hybrid cloud

Article · July 2015

DOI: 10.1109/CCAA.2015.7148472

CITATIONS

5

READS

63

3 authors:



Prashant Khanna

Waikato Institute of Technology

11 PUBLICATIONS 35 CITATIONS

[SEE PROFILE](#)



Sonal Jain

JK Lakshmipat University

23 PUBLICATIONS 376 CITATIONS

[SEE PROFILE](#)



Galgotias University

Galgotias University

235 PUBLICATIONS 6,309 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Cloud Computing [View project](#)



Fuel Cells [View project](#)

BroCUR: Distributed Cloud Broker in a Cloud Federation

Brokerage Peculiarities in a Hybrid Cloud

Prashant Khanna,
Institute of Engineering and Technology,
JK Lakshmipat University
Jaipur, India
perukhan@gmail.com,

Sonal Jain
Institute of Engineering and Technology,
JK Lakshmipat University
Jaipur, India
sonaljain@jkl.u.edu.in

BV Babu
Galgotias University
Greater NOIDA, India
vc@galgotiasuniversity.edu.in

Abstract—The paper describes the creation of a novel cloud broker created to provide efficient cloud brokering services for a government organization in India – the BroCUR. The BroCUR works in a volunteer federated, distributed schema with control being executed using a combination of threshold based metrics and triggers which launch task migration. The novel aspect of the BroCUR is the inclusion of the human manager in the decision making for task migration and the relevant inputs taken into consideration while prioritizing the scheduling and transfer of tasks in the cloud ecosystem. This paper describes its components, the logical placement of the BroCUR in a distributed schema, the results of the comparative experiments of the BroCUR against the Rackspace broker. This paper also amplifies the aspects related to the unique SLA management between multiple data centers in regards to their volunteering services and resources to the overall cloud ecosystem. The BroCUR has been implemented on live instances of data center under heavy to light load conditions and provides an efficient alternative to commercial offerings available in the market. The combination of the concepts of volunteer cloud federation and cloud brokerage with real time inputs from cloud data center human managers is unique to this research.

Keywords—Cloud Broker, Federation, Cloud Computing, Distributed Brokerage, Cloud SLA

I. INTRODUCTION

Cloud Computing and related technologies have seen an exponential increment in interest in the last decade, both in the industry as well as ACADEMIA. This is primarily due to the maturing of convergence and virtualization technologies at an appropriate time. The paradigm shift from organization and institutions using isolated networked entities, hosting legacy systems, to the cloud way of working had been further propelled by the constraints of budgets and corresponding reduction in organizations spending on IT management assets. The cloud way of working provides a very lucrative use case

for such organization to reduce overheads and associated network management issues.

This economic and technological driver has not been without its share of concerns and pitfalls. Security, interoperability issues and lack of standardization are some of the concerns that affect the optimal utilization of the cloud resources as they stand today. However, this is common for any new technological initiative or paradigm shift. The cloud ecosystem is indeed in a very exciting phase of maturing today and renewed effort in the standardization fields are seen on a regular basis. The role of the Cloud Broker as a ‘bridge actor’ or ‘bridge entity’ in assisting organizations to transcend from working in the typical siloed-networked-entities concept to a more optimal and seamless cloud way of working is a logical step towards the maturing of the cloud ecosystem. This research is an effort in the same direction. This research aims at providing a live use case where a government organization, with its data centers spread across 11 locations in Western India, used a customized cloud broker framework to migrate from a legacy networked framework to a more optimized, volunteer federation cloud framework. The research has been conducted on live traffic in actual data centers having real time load scenario and a distributed cloud brokerage framework was conceptualized and implemented. The customized brokerage framework was called Broker for Common Usage of Resources in a cloud ecosystem or BroCUR. Results on the performance of the BroCUR were published by the authors in [1]. This research presents further optimizations on the construction and implementation of the BroCUR and its implementation in a distributed schema, while using a novel algorithm for selection of the data centers for transferring the live load and tasks in application queue. This research is also a part of the IEEE P2302 [2] work group efforts and the basic structure proposed by the mOSAIC framework[3] and customized to cater for the present experimental setup. The brokerage structure was changed to

suite the specific requirements of the organization and it was also due to management requirements of optimization, which is very different from the one proposed in literature by the mOSAIC project.

The remainder paper is structured as under: In section II we present a state of art on the aspects of cloud brokerage and cloud federation. The details of the BroCUR construction are presented in Section III with emphasis on the details of implementation of the broker in a distributed schema. Section IV highlights the experimental setup and metrics for performance of the BroCUR against the default broker available in Rackspace [4]. Section V presents the results of the experiment and concludes with notes on the future possibility of research and work in progress.

II. STATE OF ART

The present state of cloud computing research in general and, cloud federation and cloud brokerage is particular are seeing renewed research initiatives. This is partly driven by the interest in the academia to find an agnostic solution to a composite cloud interface solution. Several recent approached in theory have been attempted in recent years on arriving at solutions which are optimized in provisioning of cloud based services. In addition, there have been some efforts in recent times on co-joining the complimentary concepts of cloud brokerage and cloud federation. These concepts are complimentary and unique use cases are emerging that offer solutions which make the two concepts increasingly attractive to be co-joined.

Most authors have converged on the definition of a cloud broker, using the NIST [5] definition as a baseline, which generically defines a cloud broker as an entity that manages the use, performance and delivery of cloud services and negotiates relationships between cloud providers and cloud consumers. The traditional definition of the cloud broker requires a revision in the face of increasing adoption of federation of clouds [6]. This adoption has been necessitated due to a strong use case for converged service/data centers that seamlessly interact and share services. The interest in academia to generate an open standard for interoperability and service transfer between clouds across the application stacks is also driven by this use case. Customized cloud brokerage implementation and relevant research is at a cusp in development.

The emergence of the cloud broker as the agent to do cloud resource matching and provisioning amongst cloud providers would be a natural evolution of the cloud broker from its stated task. Recent work by Buyya [7], [8], [9] also cites the utility of such a use case. An excellent chapter by NM Calcavecchia compares the available frameworks that have proposed a brokerage strategy in a federated infrastructure [10]. Among different provider-centric approaches Hybrid Cloud, Cloud Federation and Inter-Cloud are the most prominent scenarios and are highly studied subjects [9]. Several recent works by the CLOUDS Labs, Melbourne and the work by the IEEE Workgroup P2301 [11] and P2302 [2] on Standards for Intercloud Interoperability and Federation have focused on the way the clouds can interoperate. However, a universal standard is still some distance away. In

such a scenario, federated clouds, which volunteer for establishing a relation, coupled by an efficient cloud service brokers are a plausible way ahead.

Gartner in 2010 anticipated that large enterprises will soon have a dynamic sourcing team responsible for ongoing cloud sourcing decisions and management. Today it is no longer a question of whether or not users will use cloud computing; organizations, both small and large, must ensure that they are cloud-ready and should evaluate opportunities by applying a structured framework to identify potential candidates for cloud-based services. Gartner, in a report in 2014 [12] indicated that cloud brokering services in the cloud service marketplace is emerging as a promising low-risk business model for offering new and value-added services through cross provider service delivery and partnership. It also indicates that the market for Cloud Service Brokerage will reach \$160 Billion [12]. This assertion has made a major impact on the industry as well as the academia.

The cloud brokering framework functions in the middle of the cloud provider and the cloud consumer to enable this dynamic provisioning. The position of a cloud broker in the model for provisioning of services and with the special emphasis of provision of automatic and dynamic services in the environment has made the concept of service brokering in the cloud the subject of many researches presently underway in the academia [13], [14]. This entails creation of toolkits and frameworks that can emulate and cloud environment, while saving costs on actually hiring and using services from providers. The same has once again been asserted by Gartner in 2015 [15].

The architecture of the cloud, whether public, private, community or hybrid, would make it non trivial to propose and implement binding frameworks in the absence of accurate measuring and monitoring mechanisms for service assurance. This is especially true for a use case when the broker is aggregating and arbitrating services from multiple heterogeneous cloud service providers and packaging them as a service bundle for the end client. Previous works on the subject include [16] and [17] that pertains to SLA formulation and using the SLA mechanisms using autonomic methods to enforce the SLA in a cloud. Most recent refereed work pertains to the relationship between the cloud provider entity and the cloud consumer with frameworks designed to emulate the relationship. However these frameworks do not illustrate the aspect of arbitrated and aggregated services for a service assurance model from the perspective of a cloud broker. This research is an attempt to address the research void.

Cloud federation presents a compelling use case as it can lead to Internet Scale Resource Optimization opportunities, the ability for cloud users and providers to upscale and down scale efficiently depending upon the utilization metrics/ratio of the data centers while arbitrating services [18]. Presently there is a flux when we come to using established standards governing the internal cloud management such as Cloud Data Management Interface (CDMI) [19], Open Cloud Computing Interface (OCCI) [20] and Open Virtual Machine Format (OVF) [21]. In this research the cloud broker which can act as an interface which achieves transactional integrity and

message parsing ability across these popular formats has been tested in a private cloud. Cloud interoperability however requires cloud providers to adopt and implement standard interfaces, protocols, formats and architectural components that facilitate collaboration.

III. IMPLEMENTATION DETAILS- *BROCUR*

The structure that has been created has a direct bearing on the actual use case scenario in a government organization which is migrating from a series of administrative networked entities to the cloud concept. The broker has been created such that it allows common usage of available resources in the cloud and stress has been laid on the best fit schema for the entire cloud ecosystem. Thus the Broker has been termed as **BroCUR** or *Broker for Common Usage of Resources*.

The broker has been created to accomplish the tasks associated with those defined for a cloud broker in the NIST framework and also handled the automatic and autonomic scheduling and placement of service requests received from local and remote data centers. The implementation of BroCUR has been done with the aim of achieving the following in a federated cloud ecosystem:

- Dynamical match between requested job and provider(s)
- Deployment in time and at runtime using simple interfaces
- Migration and elasticity
- Quality of Service using minimal management based SLA support
- Provider selection and integration at centralized level and system level in a distributed schema
- Enforced mechanisms through manual and automatic overrides
- Interoperability between existing administrative networked domains using simple and available hooks and interfaces
- Allow for federation to grow post adoption of a simple interface and request template

The components of the BroCUR are explained in the table 1 below. The interrelation between the components has been described in the succeeding figure and the control channel is using multicast protocols to keep the system stable. The human manager aspect has been illustrated in the table and the figures below.

TABLE I. TABLE 1. BroCUR COMPONENTS AND THEIR LOGICAL ASSOCIATION

BroCUR Component	Task Mandate of the Component
BroCUR Coordinator	The Coordinator is logically centralized but geographically distributed. It is responsible for the management, control and analytics function of the Brokering Framework. It has the overall control and privilege level access to obtain and share the cloud/data center's load provisioning statistics, take decisions on the scheduling and placement of the load in the cloud ecosystem and do the match making of the resource request and resource availability template. It passes the instructions to the BroCUR Placement Manager based on the inputs and threshold levels decided upon for individual Data Centers/Clouds.
BroCUR SLA Manager	The role of the SLA Manager is to obtain the WS

BroCUR Component	Task Mandate of the Component
	Agreement based templates from multiple agencies participating in the federation. It provides the threshold for load, QoS and time threshold for service request restrictions to the BroCUR Coordinator. It is the repository of the cloud statistics in terms of performance and load levels. It provides the repository of load and usage templates to clouds that are part of the federation. The local data center provides the information of the running load to the SLA Manager and in turn the SLA manager processes it and offers it to the Coordinator. The SLA Manager also collates the statistics obtained from public cloud analytics engine mapped to the BroCUR Coordinator and stores the same in a separate template. The SLA Manager is also responsible keeping an updated state of definitions and observed violations occurrences of service requests in the cloud ecosystem when in a federation.
BroCUR Resource Placement Engine	The BroCUR Resource placement engine is responsible for Management of the Virtual Machines (VMs) and the Virtual Networking functions. The placement engine is a pseudo-forwarding plane that is akin to a distributed router, using multicasting at the network layer. It is responsible for all VM related functions and also extends the network from the physical router and switches in data centers into virtual overlays which map a network hosted virtualized infrastructure. The base VM functions which the Placement engine performs are : VMCreate VMCopy VMSave VMRemove VMModify VMShare VMMigrate VMRollBack
Administrator and Management Engine	This is the non-autonomic segment of the BroCUR that provides the inputs for the SLA Manager and BroCUR Controller to take decisions on the Placement and SLA management. The Neo-AHP algorithm is used for the ranking structure of the available clouds and the inputs to the Neo-AHP algorithm are provided by the Administrators and Management Layer.

The illustration in figure 1 below illustrates the structure of the components of BroCUR and the interconnects between the components .

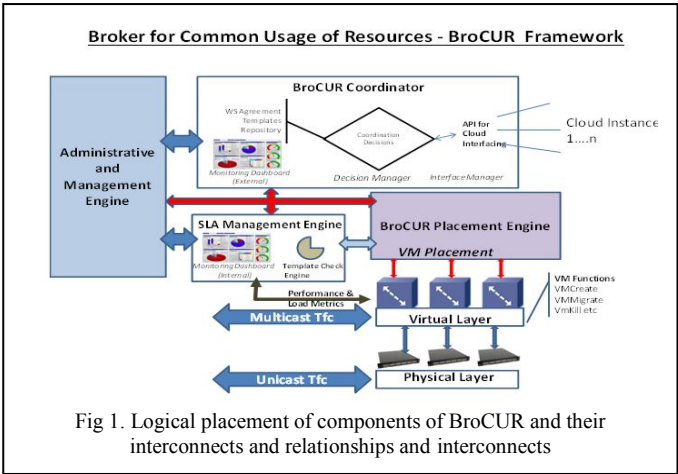


Fig 1. Logical placement of components of BroCUR and their interconnects and relationships and interconnects

It is pertinent to mention here that the BroCUR can be located in a geographically distributed manner, but acts in a centralized manner with the Coordinator working in tandem using Multicasting on the Network layer for passing control and update information. This aspect has made the performance of the broker more optimized and evolutionary in nature. The BroCUR also uses the concept of Virtual Networks (VNs). Virtual networks are logical constructs implemented on top of the physical networks. Virtual networks are used to replace VLAN-based isolation and provide multi-tenancy in private virtualized data centers. Each tenant or an application can have one or more virtual networks. Each virtual network is isolated from all the other virtual networks unless explicitly allowed by security policy. When the Zone based scheduling was implemented, the tags for specific zones were provided in the virtualized routers for more efficient routing.

The placement of the Broker in the researcher's cloud ecosystem was as presented in Fig 2. The multicasting approach to transfer of control information and the management layer information is illustrated in the VM interaction. The broker for this research has been created with Linux (Ubuntu 14.02 LTS) on the Domain 0 and versions of WINE/XEN on other domains

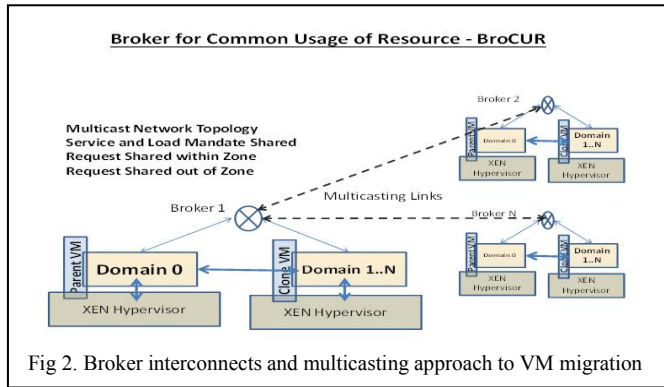


Fig 2. Broker interconnects and multicasting approach to VM migration

IV. SLA STRUCTURE AND BROCUR INTERACTION

The BroCUR API is a RESTful and resource oriented API that has been modeled on the DMTF Cloud Infrastructure Management Interface (CIMI) HTTP RESTful HTTP Model [21]. For querying and documenting the resource requirements of the MPEG requirements used in the experiments, the MPEG Query Format (MPQF) standard was used to ensure compatibility. The SLA structure was created and published as a template for the service providers and the BroCUR to access, on required basis. This was housed on the centralized broker in the composite schema for brokering and on the individual instances of the brokers in the distributed schema. It was appreciated that a distinction was necessary between the content of the SLA, which is a representation of the context shared by the parties involved and also defines a reference point, and the use of the and the use that each actor makes of this context in its own operations throughout the SLA lifecycle (publish offer/discover/select/agree/provision/use/terminate). The SLA was required to provide a complete description of the service for both Consumer and Provider to avoid uncertainty and disputes which benefit neither. An unambiguous, clearly

defined SLA was required as it had benefits to both Consumer and Provider when arbitrated by a Broker. Towards this aim, the BroCUR SLA framework was created with the following objectives:

1) *Clarity on Service Offer.* For the broker, this allows comparison between different providers, and for the provider, this allows the scope and extent of a service offer to be clearly defined and used to offer services, comparing with other data centers.

2) *Unambiguous definition of expectations and obligations for all players in the cloud ecosystem and the test bed.* For the consuming data center, this allows the impact on organizational processes to be determined, including any changes required to enable cloud services to be used effectively. This includes proper consideration of any management, organisational policy and threshold (load or time) considerations. Penalty would be defined by the Administrative and Management Engine so that the environment is aware of the metrics. For the providing data center, this allows services to be managed and prioritised effectively, particularly where many consumers are being served from a multi-tenanted infrastructure.

3) *Stability of the Federation.* This was necessary to implement as the administrative entities which were collaborating in the creation of the federation were under a unified management layer, but had individual local priorities for service completion. This constraint had to be managed such that the resource provisioning did not affect the local service requests, while still allowed the data center to be part of the federation and oblige to their stated commitments. The SLA negotiation algorithm was thus constructed with a bias on local service requests and on the availability of resources in the data center after servicing local requirements. This ensured the federation stability.

Keeping the constraints and requirements mentioned above in mind, the SLA architecture in the case of BroCUR ecosystem was deliberately kept as a mixture of a SLA based and co-joined with Trigger based approach. As this was a volunteer based federation, the BroCUR was created as an internal agent with inputs from the administrative and management engine (trigger approach), while maintaining communication with the BroCUR coordinator for obtaining the external repository WS Agreement based templates (SLA Approach) [14] and [22].

The logic was that a trigger would become active once a predefined condition triggers a series of actions considering the externally visible data center application performance indicators reach a threshold. An action is executed when a correspondent trigger becomes active. Actions usually include scale up/down or other provisioning activities. In the case of BroCUR ecosystem, a data center was in the availability zone for provisioning once the data center local resource utilization thresholds were below a pre-defined level. For example, a trigger would become active if the number of active TCP/IP sessions becomes more than 5000 or if the memory

consumption of an allocated VM exceeds 512MB. An associated action may be the allocation of a new VM.

To the best of our knowledge, there is presently no working cloud broker that uses the combined approach of a trigger based cum threshold based approach to cloud tasks migration and VM migration. The novel structure proposed and created in this live experiment has been necessitated due to a live requirement of presenting an optimized approach to migration of the organisation's legacy networked entities into an efficient cloud based ecosystem. This research is also a major step in utilizing existing frameworks, in a customized manner, to achieve the planned transition over legacy network inventory without major changes in the capital overlay of purchasing new equipment.

V. BROCUR – COMPARATIVE PERFORMANCE AND RESULTS

The performance measurement of the BroCUR was conducted in three phases. In phase 1, the broker was created and tested within a local data center, with load being transferred across the virtualized servers and scheduling efficiency of the broker measured. In this phase, the broker was utilized akin to an Application Delivery Controller with clear mandates of load threshold and the algorithm of service transfer was introduced for migration of virtual machines across servers within the data center. The concept of a data center load metric dashboard was incorporated and min thresholds defined and implemented in this phase. The BroCUR SLA Manager took the dashboard metrics from the local data center/clouds and the threshold based decisions, or trigger based ones were taken for resource utilization.

In Phase 2, the broker was installed in a geographically distributed manner within the cloud ecosystem, created from the networked administrative entities, and the centralized schema as well as the distributed schema of brokerage implemented. The performances were analyzed on cost metrics for utilization of resources. The base was against cost incurred when the Amazon Web Services (AWS) were used in a similar load conditions. A bridge was created between the research lab and AWS server infrastructure using the OpenNebula API [23] (OpenNebula, 2015) and through the use of open source libraries. The scheduling and migration algorithms were tested and optimized for VM migration within the cloud ecosystem. The metrics for analysis were generated using the New Relic engine on the load on the AWS and the Openstack Metering Management and Dashboard [24] for the research cloud. This was made available in the BroCUR Coordinator Engine. The BroCUR was tested within the research lab in a centralized as well as distributed manner.

In the third phase of testing, the cloud federation concept were introduced and the breaking of a cloud federation was emulated through severing of the communication links between the cloud instances and delayed message passing and increased queue length extrapolated on the system to simulate a cloud not honoring the federation mandate. The Administrative & Management component of the BroCUR was also involved in this phase of simulations where the negotiation metrics were induced by Human decision makers or the Data Center Managers to reflect real world constraints

on data center priorities and management decisions. This phase of study was conducted locally on the research test bed and the performance metrics were based on BroCUR's ability to maintain the cloud stability in the wake of increased load in a cloud federation. This was done in the presence of the broker and then in its absence.

In phase 1 of the experiments on the construction of the BroCUR, the broker instances were placed on the edge of the Data centers in the centralized as well as a distributed schema. The instances were replicated on the Rackspace servers and the comparison of results between the performance of the BroCUR and the default Rackspace broker was done. The Rackspace instances were hired for the duration of the test the performance threshold was kept at a maximum of four virtual servers per set of experiments as the service requests increased. The maximum bandwidth allocated on the Rackspace server was 40 Mbps and each test was conducted over 60 seconds at the end of which the service request were increased to over 5000 concurrent requests per second and the application was tested for scalability and the response time. The load was generated using an online service called Loader.io [25] whose local instance was installed at the Local Data center at Jaipur .

The metrics of performance for the BroCUR in phase 1 of the experiment were:

- Responsiveness of the server - The response time indicates the efficacy of the broker to regulate and allots job requests to the servers.
- Error Rates from the Server - The broker would be able to concurrently service several client requests for the service available on the server and unless the same is not provisioned in a timely manner, the error rate rises.

An efficient broker would manage to keep the response time constant and the error rate low.

The performance metric of the BroCUR is shown in succeeding figures and it was found that the BroCUR performed better on both the response time as well as the error rates compared to the default Rackspace server broker under similar task load conditions. This was handled in a more efficient manner in the BroCUR once the federation concept was introduced and local decisions made on the data center capacity were made on performance metrics rather than cost based one. The results of this phase 3 of the testing were discussed by the authors in [16].

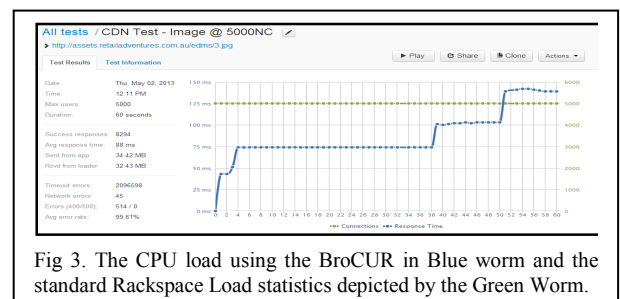


Fig 3. The CPU load using the BroCUR in Blue worm and the standard Rackspace Load statistics depicted by the Green Worm.

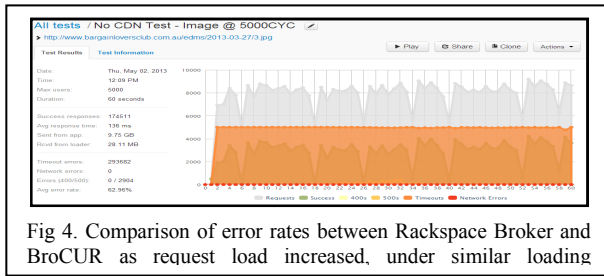


Fig 4. Comparison of error rates between Rackspace Broker and BroCUR as request load increased, under similar loading

As is evident from the results, Rackspace model for provisioning is guided by commercial interests and thus the responsiveness of the system is stable across load conditions, but there is excessive over-provisioning visible. The CPU load generation statistics visible are evidently catering for static loads and thus over provision – and over charge the users. In the case of error rate comparisons, it was observed that the rate at which the errors were recorded sharply fluctuated in the case of the Rackspace instance. This is illustrated by the shaded grey towers on Fig 4 above. The error rate was near constant in the case of the BroCUR as the delegation of tasks was done more efficiently due to a system view of resource availability available at the individual instance of the BroCUR riding on the multicast control channel.

In our previous publication, we had introduced the aspect of delay in the control information being passed to respective instances of BroCUR in a geographically distributed data centers. This is a work in progress in the present government organization and optimization in the BroCUR performance is now oriented towards handling delays in passage of resource requests.

VI. REFERENCES

- [1] P. Khanna, B. Babu and S. Jain, "Cloud Broker: Working in federated structures - A Live Analysis," in *International Conference on Advances in Computing, Communications and Informatics (ICACCI, 2014)*, New Delhi, 2014.
- [2] "P2302," IEEE, 1 January 2015. [Online]. Available: <https://standards.ieee.org/develop/project/2302.html>. [Accessed 2 Feb 2015].
- [3] D. Petcu, "Consuming resources and services from multiple cloud," *Journal of Grid Computing*, p. 1–25, 2014.
- [4] "Rackspace," Rackspace, 1 January 2015. [Online]. Available: <http://www.rackspace.com/>. [Accessed 1 Feb 2015].
- [5] P. Mell and T. Grance, "The NIST definition of cloud computing," NIST Special Publication, Washington, 2011.
- [6] N. Gorez and R. Buyya, "Inter-Cloud architectures and application brokering: taxonomy and survey," in *SOFTWARE – PRACTICE AND EXPERIENCE*, Wiley, 2012, pp. 369-190.
- [7] A. N. Toosi, N. C. Rodrigo and R. Buyya, "Interconnected Cloud Computing Environments: Challenges, Taxonomy, and Survey," in *ACM Computing Surveys*, 2014.
- [8] R. Buyya, S. Pandey and C. Vecchiola, "Market-Oriented Cloud Computing and The Cloudbus Toolkit," in *Market-Oriented Cloud Computing and The Cloudbus Toolkit*, John Wiley Inc, 2013, pp. 319-158.
- [9] 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*, 2010.
- [10] N. M. Calcevecchia, A. Celesti and E. Di Nitto, *Understanding Decentralized and Dynamic Brokerage in Federated Cloud Environments*, igi-global.com, 2010.
- [11] "P2301," IEEE, 1 January 2015. [Online]. Available: [Standards.ieee.org, 'IEEE SA - P2301 - Guide for Cloud Portability and Interoperability Profiles \(CPIP\)', 2015. \[Online\]https://standards.ieee.org/develop/project/2301.html. \[Accessed 2 Feb 2015\].](https://standards.ieee.org/develop/project/2301.html)
- [12] T. C. Eng and K.-Y. Foong, "Market Trends: Cloud Market Opportunities for CSPs — Lessons Learned After Five Years," 17 July 2014. [Online]. Available: <http://my.gartner.com/portal/server.pt?open=512&objID=260&mode=2&PageID=3460702&resId=2802217&ref=QuickSearch&stkw=cloud+broker>. [Accessed 2 February 2015].
- [13] P. Khanna and S. Jain, "Cloud Broker: Application Brokering in a Federated Clouds," in *Shodh 2014*, Jaipur, 2014.
- [14] P. Khanna, S. Jain and B. Babu, "Distributed cloud brokerage: solution to real world service provisioning problems," *ARNP Journal of Engineering and Applied Sciences*, vol. 10, no. 5, pp. 2011-2016, 2015.
- [15] D. Smith and D. Plummer, "Agenda Overview for Cloud Computing 2015," 5 January 2015. [Online]. Available: <http://my.gartner.com/portal/server.pt?open=512&objID=260&mode=2&PageID=3460702&resId=2954528&ref=QuickSearch&stkw=cloud+broker>. [Accessed 2 February 2015].
- [16] P. Khanna, S. Jain and B. Babu, "Cloud Broker: Working in Federated Structures - A Hybrid Cloud Live Performance Analysis," in *3rd International Conference on Advances in Computing, Communication and Informatics*, NOIDA, 2014.
- [17] P. Khanna and B. V. Babu, "Cloud Computing Brokering Service: A Trust Framework," in *he Third International Conference on Cloud Computing, GRIDs, and Virtualization*, Nice, 2012.
- [18] 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, 2009.
- [19] "CDMI," SNIA, 1 Jan 2015. [Online]. Available: <http://www.snia.org/cdmi>. [Accessed 3 August 2014].
- [20] "OCCI," Jan 2015. [Online]. Available: <http://www.ogf.org/documents/GFD.183.pdf>. [Accessed 3 Aug 2014].
- [21] "OVF," DMTF, 2 Dec 2014. [Online]. Available: <http://www.dmtf.org/standards/ovf>. [Accessed 3 Aug 2014].
- [22] P. Khanna and S. Jain, "Distributed Cloud Federation Brokerage: A Live Analysis," in *IEEE/ACM 7th International Conference on Utility and Cloud Computing (UCC)*, 2014, London, 2014.
- [23] "OpenNebula," OpenNebula, 1 January 2015. [Online]. Available: <http://opennebula.org/>. [Accessed 1 March 2015].
- [24] "Open Stack," 15 Mar 2015. [Online]. Available: <https://www.openstack.org/>.
- [25] "Loader," [Online]. Available: www.loader.io. [Accessed 10 Jul 2014].
- [26] P. Khanna and S. Jain, "Cloud Broker: Definition Differential between Gartner and NIST," in *National Conference on Science and Engineering (NCSE)*, Jaipur, 2014.
- [27] P. Khanna and S. Jain, "Live Performance of Cloud Brokers in a Federation," *International Journal of Science and Research*, vol. 3, no. 12, December 2014.