Proceedings of the 10th INDIACom; INDIACom-2016; IEEE Conference ID: 37465
2016 3rd International Conference on "Computing for Sustainable Global Development", 16th - 18th March, 2016
Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi (INDIA)

An Efficient Data Replication and Load Balancing Technique for Fog Computing Environment

SagarVerma, Arun Kumar Yadav, Deepak Motwani
ITM University, Gwalior, M.P. India
verma.sagar009@gmail.com, arun26977@gmail.com,
deepakmotwani@itmuniversity.ac.in

R.S. Raw, Harsh Kumar Singh
Associate Professor, Department of CSE
AIACTR, Delhi
rsrao08@yahoo.in, hksingh099@gmail.com

***Abstract*–The technological environment is a competitive ground used by IT industries for provisioning Cloud services. It is essential for the organizations to facilitate higher availability of quality services, computing resources and faster delivery. Fog computing can work with much the same Cloud computing in promoting faster IT services. The incorporation looks after affording, maintaining and provisioning of resources like storage space, hardware virtualization, high-capacity networks, service-oriented architectures, autonomic and utility computing etc, proximal to the access point. The maximal uniform distribution of the load across closer and more number of simpler nodes can help managing and providing the big data and large workloads more easily. The paper proposes an efficient load balancing algorithm for a Fog-Cloud based architecture. The algorithm uses data replication technique for maintaining data in Fog networks which reduces overall dependency on big data centers. A comparison of present load balancing techniques in 'Cloud-based' infrastructure to the presented 'Cloud-Fog' duo is also shown. The ultimate goal is to balance load through Fog networks and make internet less Cloud dependent by having data available closer to the user end.**

Keywords-Cloud Computing, Cloudlets, Data Replication, Edge Networks, Fog Computing, Foglets, Internet of Everything, Load Balancing

I. INTRODUCTION

The physical characteristics of IT resources are pulled together to facilitate multiple customers through internet from a single set of resources without failure, referred as Resource Pooling. The hierarchies of several data-centers are responsible for maintaining these resources and services. The Big Data and the information explosion are resulting in more loads all across the networks. To lower down the response time and better data availability is a major goal for using Edge networks. There should be minimal and evenly distributed workload on the data centers to function at their utmost potential and to control the degree of load distribution, deadlock avoidance and eradicate the problem of server overflow. In time, Edge computing is taking service deployment to a more advanced level due to its proximity to the user in the network. As a result, there is less delay in service provisioning due to expansion of the local networks and easy load distribution over more number of available servers. The Fog network nodes can effectively be used for data forwarding. The Fog Servers containing the replicas and records of the required data can help in reducing transaction failures and sustain the Quality of Services to the users through periodic Cloud updates, for the available data. The high performance virtualized environment considers memory or CPU processing efficiency as the service load. Load is a set of all the tasks which are to be serviced by virtual and real-time machines in the Cloud environment. Multiple aspects of load balancing, to be taken care of while managing the workload is as follows:

- Scalability.
- Overall Response Time
- Efficiency
- Related Overhead
- Resource Utilization

A better utilization ratio and QoS (Quality of Service) is provisioned by consistently monitoring the load and performing shifting of load when necessary.

The paper organization is as follows, section 2 examines the other techniques proposed by researchers for data replication and load balancing. In section 3, we have proposed the fog architecture and in section 4, an efficient algorithm for data replication and load balancing has been proposed. In section 5, proposed architecture and algorithm has been simulated on simulation tool and the result compared with existing techniques. Section 6 concluded the findings and section 7 gives the further directions for future research.

II. RELATED WORK

The section discusses frameworks to assist in maintaining load, latency and replication process in computing environment

and other collaborative works using Fog networks to further advance the use of Edge networks for example in parallel processing, dynamic environments, data grids, latency sensitive applications etc.

The load in hot spots along with the latency can be depreciated in distributed information processing through frameworks such as PAST [1] and CFS [2] for replication of data proximal to the request site and the sites carrying the data. The data in unstructured file system exhibit average latency, representing majority of the overall data. And because of data replication of unnecessary content, the network suffers from load imbalance. The node with the images of the required data recording the paths between the owner and the subscriber for update notifications might leave the network. Techniques like Scope [3] and Freenet [4] must be utilized for maintaining consistency and availability in the networks. The message casting for Edge network platforms are still unexplored and needed as above mentioned techniques are centralized in nature.

Author Haiying Shen provided integrated file replication for centralized networks to keep track of all the replicas and their updates. This may help in achieving replication but not the consistency. For robust fault tolerance more space should be available to share more data [5].

Author Zacharia Fadika et al. [6] proposes MARLA which is based on dynamic scheduling mechanism. It supports Cloud and grid computing, parallel file systems. The space required and task parameters are not required to be predefined. The location of the data may be the location of the node, some other node or the job scheduling location. The job tracker is responsible for copying the data from remote node to the job scheduling node; which dominates the optimality of execution of job and increases the latency. Thus, the network hierarchies must follow the concept of data replication to contain data on nearby and multiple nodes. The majority of the work in Cloud environment and data grids is currently emphasized on job scheduling. As the structure of data is uniform in the data grids, in order to maintain the consistency, data replication techniques can append to the updates and forward replicas [7].

At the data centers, the replication strategies have the primary goal of maximal depreciation in bandwidth usage. The workload descriptions required by the servers for execution, delivery of updates among database replicas, and receiving database objects are provided using bandwidth in the downlink; whereas in the uplink ,the bandwidth is used for propagation of database requests and for the updates by the applications for modifying data items. The update rates for every data item and data access statistics monitoring are maintained and shared by all the databases[8].

For geo-spatial distribution of latency-sensitive future internet applications, a high level programming model, K. Hong have proposed on mobile Fog in [9]. At the core of the network - the Cloud, latency-tolerant large scope aggregation is handled by powerful resources whereas logical structures handling low-latency processing occurs closer to the Edge of the network. Capable of controlling network traffic and reduce latency. Mobile Fog is a generic model a set of event handlers and functions an application can call in use. For provisioning resources in Cloud and Fog environment, B. Ottenwalder proposes a placement and migration approach for ensuring application defined end-to-end latency restrictions and brings down the network utilization by pre-planning the migration of load. The method also describes the use of knowledge of applications to lower the use of required bandwidth of virtual machines when migration takes place. The description of network control policies for determining optimal routes for various applications is still left undone. Their work highlights the reliability requirements for Smart Grid, Cloud Sensors and Actuators but the absence of any basic concepts and strategies for making network of smart devices reliable in nearby networks makes Fog based projects relatively challenging [10].

J. Zhu et al.[11] applied existing methods for web optimization in a novel manner. Within fog computing context, these methods can be combined with the unique knowledge that is only available at the Fog devices. More dynamic adaptation to the user's conditions can also be accomplished with network's Edge specific knowledge. As a result, a user's web page rendering performance is improved beyond that achieved by simply applying those methods at the web server. BETaaS community represented 'local Cloud' a platform for machine-to-machine applications as a Cloud alternative. The functionality of 'local Cloud' is handled by various devices supporting Internet of Everything (IoE) ,real time data and analytics, home routers , smart phones and road-side units. [12].

There is even distribution of workload over all the web servers but as the algorithm depends upon the prior knowledge of the system resources, the mobility of load at any particular instant is not achievable. Therefore static algorithms are not suitable in areas with high variation of load in the system. On the contrary, the instantaneous shifting of load is possible in dynamic algorithms. The nodes with minimal load are tracked down in the network [13].

No previous work have been done in handling the issue of load balancing using data replication and Edge networks for data availability.

III. PROPOSED ARCHITECTURE

We are proposing a basic architecture for balancing load in the network and the Cloud servers using Fog servers. The following section also describes the components and the working of the system following Edge Balancing algorithm. Fog computing is another Cloud like paradigm, exhibiting less delay in service provisioning .Fog layer interacts as a middleware for the client's front end and back end of the Cloud data centers.

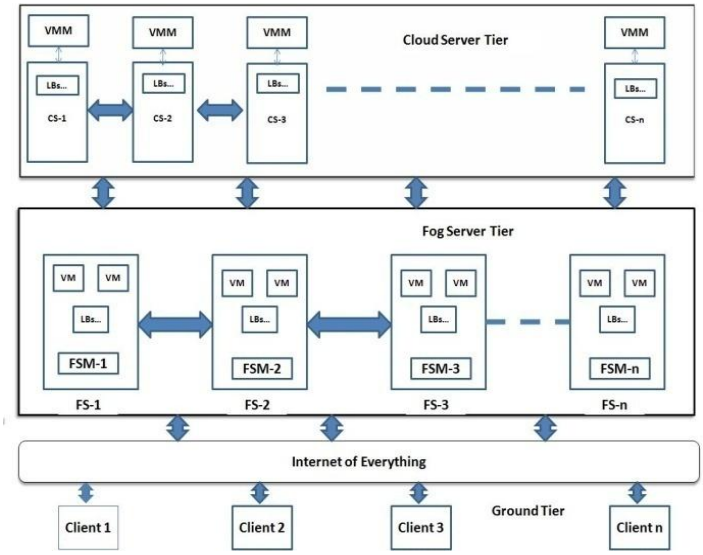

Fig. 1.Proposed Architecture using Fog Computing

We have presented a three tier architecture comprised of Ground Tier, Edge Tier and Core tier. Cloud as we know is a centralized virtual network of servers located at the Core tier, which are relatively at a farther distance from both the other tiers. The Core tier nodes are large data centers whereas Edge tier carry many user accessed devices like laptops, notebooks, mobile phones etc. The virtualized Edge tier nodes can interact with one another and the Cloud data centers for load management, handling user requests, monitor Cloud servers to avoid overloading in both the upper tiers, task assignment in Fog servers and request forwarding to the Cloud servers. The live migration between virtual machines is faster in Fog tier due to a significantly smaller distance between all Foglets than all the Cloud nodes in the Core tier. A Fog Server Manager (FSM) is associated with each server responsible for handling the data from mobile and static nodes present in various networks like Ad-hoc networks, VANETs, Wireless Sensor Networks (WSN) etc. All the clients in Ground Tier are connected to the IoE provisioning better connectivity and versatility in supporting protocols and technologies. The IoE houses the internet services circulating data through all the tiers. The Edge networks provide more resources, processing power and data migration capabilities.

A. System Components

Control and conduction of service environment requires more people, different backgrounds and expertise. Cloud being an expensive venture, the elements need to perform ideally in sustaining the paired virtual computing platform.

- Client-The routine starts and ends with the clients co-existing in the Ground Tier. The Client node houses the hardware and software components interacting with Cloud for applications, platforms, data resources, etc.
- Internet of Everything (IoE)-IoE developed by Cisco is an ideal platform for coping up with the fast technological changes and helping in reducing cost and complexity in server communication .It supports various kinds of client platforms and provides better connectivity and speed. The client zone interacts through Internet of Everything (IoE) circulating packets through all the three tiers. A secure link is established between the client and upper tier servers through IoE .Services and processes are launched through servers resolving the queries .Services are realized by the applications through heterogeneous platforms.
- Fog Server-The Fog servers functions as a middleware. They are the first service providers before Cloud servers. Fog servers are simpler and with less capabilities than Cloud servers and data centers. Foglets are basic fundamental devices present in an Edge network. All Fog servers in an Edge network can interact with other Fog nodes inside or outside the network. Each server reserves storage for network data.
- Fog Server Manager (FSM)-The roles for administration of server processes, IP packets, load and communication are optimized by FSMs. Each Fog server carries a single FSM. All the service storage and functions of Fog servers can be accessed only through their respective FSMs as it deals with the load balancing of the Fog servers. Any FSM is capable of interacting with other FSMs for request handling purpose on the basis of data availability and load handling capacity .It maps server availability against the number of requests. FSMs can make changes like migration in case of overloaded servers.
  The encrypted information travels in form of IP packets just like the queries.
- Cloud Server-The Cloud servers in the uppermost tier are mostly arrayed in data centers. The Cloud nodes form a hierarchy of data centers exchanging network data and facilitating clients with data and other resources..

Any client through internet network in Ground Tier creates an IP packet with the address of the nearby Fog server. The servers in the Fog tier functions like a middleware and sends the

user data to the nearest Cloud server for user authentication. Once the user has been verified in the Cloud server, a secure connection is established through which the encrypted packets are circulated between the client and the upper tiers. The servers present in Fog tier are required to be consistently monitored through their respective Fog Server Managers (FSMs). The FSMs systematically interact with FSMs of all the other servers in Edge networks. The server capable of handling the load at the time is chosen If not capable of providing the data and resources itself the request is forwarded to the next server for finally handling the request in the respective Edge network or a different Edge network.

The Fog tier server in case fails to provide the user with the data the request packet migrates upwards through the established connection along with the address of the active Fog server. The active Cloud server will administer the request and serve with the available data. In the last possible scenario, if the Cloud server also cannot deal with the request, it broadcasts query packet throughout the Cloud tier. The server with the required data to handle the request will acknowledge the initially active Cloud server. The newly active Cloud server instead of sending data to previous Cloud servers sends data directly to the active Fog server in the middle. Hence, the overhead of sending data to another Cloud server is skipped and data also gets replicated in Fog tier for similar future queries simultaneously. The Fog servers at last reaches the verified client with response packet and the secured connection among the three tiers gets terminated.

IV. PROPOSED ALGORITHM

R_i: The request from the Client i.
CL_i: Client i in the ground tier.
FS: Fog server handling the Client requests
FSM: Fog Server Manager for Fog server FS.
CS: Cloud Server in Cloud Tier

Step 1:CL_i sends request R_i to the Fog Server Tier.
Step 2: If (load FS_i< load threshold)
FS_i handles R_i;
Goto step 3
Else
Then check at next nearest FS_i
Go to step 2.
Step 3: If (FS_i carries the required data)
FS_i sends data to CL_i;
EoT //End of transaction
Else
FSM_i broadcasts message and address to the Edge networks in Fog Server Tier; // Multi-hopping
Step 4: If (FS_j carries the required data)
FSM_j responds to FSM_i and sends data to FS_i;
FS_i replicates data;
FS_i sends data to CL_i;
EoT;
Step 5:IfFSM_i receives no response for R_i by Ei
Fog Server TierFSM_i sends R_i and
FSaddress to the nearest CS_i
Step 6: If (CS_i load < load threshold)
CS_i handles the request;
Go to step 7
Else
Then check at next nearest CS_i
Repeat step 6;
Step 7: If (CS_i carries the data)
CS_i acknowledges FSM_i
and sends data toFS_i;
FS_i replicates data;
FS_i resolves R_i by sending data to CL_i;
EoT;
Else
CS_i broadcasts message and
address of FS_iinto the Cloud Server Tier;
Step 8: If (CS_j carries the required data)
CS_j sends data to FS_i and
FS_isends acknowledgement to CS_i
to end the communication;
FS_i replicates data;
FS_i resolves R_i by sending data to CL_i;
EoT;

V. SIMULATION SETUP

The simulation tool is used to demonstrate data and service mobility globally using Edge networks and achieved promising simulation results

A. Simulation tool (CloudSim 3.0)

The simulation tool provides six distinct regions on the world map to test various geographical server setups and service configurations. Various attributes of user bases and data centers across the network like response time and server cost on the basis of applied algorithms can be evaluated. The experiments can help analyzing the proposed models and results which can assist in determining algorithm's efficiency through comparison

with featured algorithms like Round Robin and Throttled. 'CloudSim 3.0' is a tool developed by CloudSim framework.

B. Simulation

The section shows the evaluation and comparison of the existing algorithms with the proposed algorithm under given constraints of the Cloud tool. The simulation of proposed model comprises of two different phases between three different layers. Our user set carries four user bases (UB1, UB2, UB3, UB4) across all the regions in the first layer i.e. the Ground Tier. The requests are handled by four data centers (DC1, DC2, DC3, DC4) in the second layer i.e. the Fog tier. The four servers are part of decentralized Edge networks near the user bases, these Fog servers reside in the middle layer. The four data centers in the first phase behave like Fog servers to each user. The middle layer servers are not as resourceful as Cloud data centers but still are capable of data storage and provisioning more rapidly. The user requests can reach any of the surrounding Edge servers lying in one or more Edge networks. The tool carries no concept of distinction in Fog-Cloud networks. It can only be presented by varying the distance and geography of the servers.

C. Result and Comparisons

The result of simulation for the three tiers orderly shows evaluation on the basis of Overall Response Time, Data Center Processing time and Total Cost. The simulation for comparison in performance between a Cloud network and Fog network is presented Phase 1 shows interaction of client nodes with Fog servers. The second stage occurs only if the Edge networks fail to serve the user with the data requested. Both the phases can occur in sequence depending on server load and data availability.

The edge network servers receiving data in the first step acts as a second line users (UB') in the Phase 2. The second phase presents interaction between a single Cloud server and the users (UB'). The interaction scenario in phase 2 takes place when in case the requested data and resources goes missing in the entire Fog tier. In case of failure, the request will be forwarded to a nearby Cloud server (in the Cloud tier) from the particular Fog server handling the data request in the Edge network. As second phase involves transmission to the distant Cloud server for getting the data, the maximum value of distant transmissions is considered. Figure shows a global setup with four distinct regions (R0,R1,R2and R3).

There are four Edge servers in respective regions (DC1,DC2,DC3and DC4).

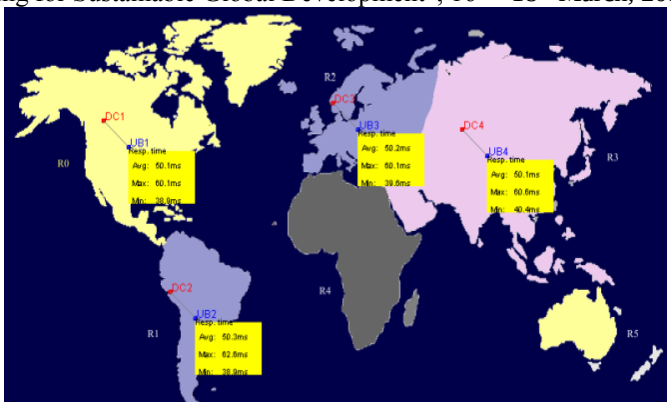

Fig. 2 First phase representing an interactive mesh of proximal Fog servers and users

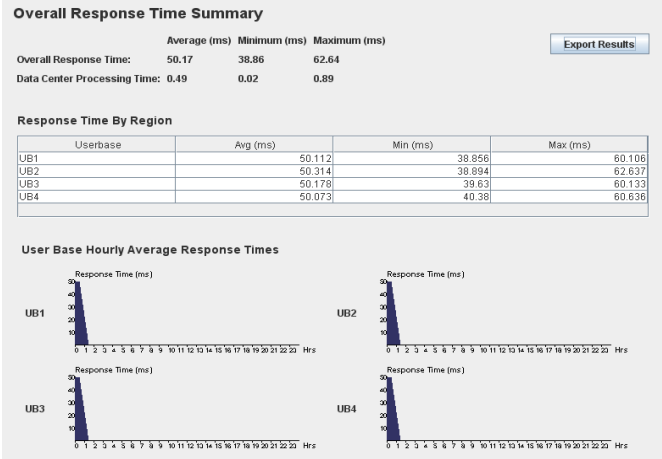

Fig. 2(a) Response time summary for Fog Tier interaction

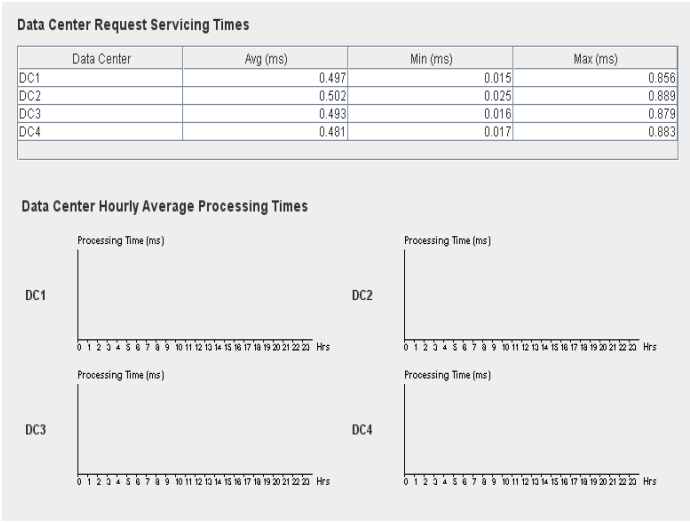

Fig. 2(b) Processing time for Fog servers.

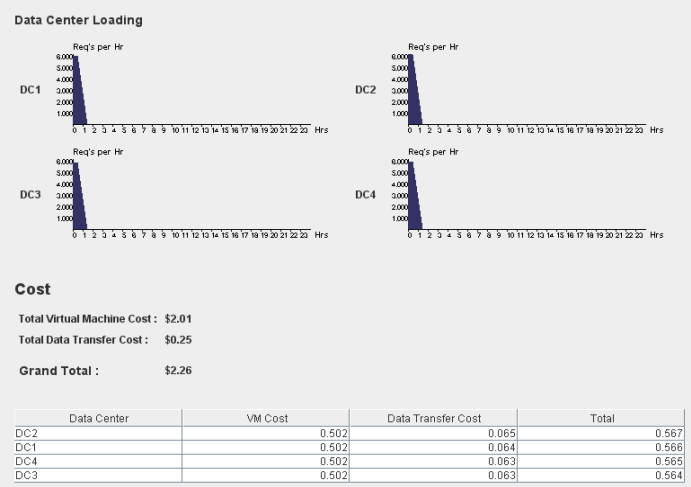

Fig.2(c) Server load and overall cost in Fog Tier

The second transaction occurs to provide data from Cloud to the Edge server currently handling the request. The Cloud server sequentially resolves the user request by providing the required data and replicating it into the active Edge server to avoid future Edge network failures. The overall time and resources consumed are considered as the sum of data transfers from Ground tier to Fog tier and maximum of attribute values for a single Cloud server transaction from Cloud to Fog tier i.e. inclusion of values consumed per transaction from Cloud data center in case of missing data in Edge networks. The Cloud tier thus provides fault tolerance in Edge networks through updates and replication to a certain degree, if the current Cloud data center fails to resolve the query; it broadcasts the message, the address of the user and the active Fog node into the Cloud tier. The Data center with the required data acknowledges the inquiring Cloud server and responds directly with the Fog tier and replicates data into the active Fog server.

In this second phase, the interaction occurs between one of the users (UB0',UB1',UB2',UB3') and the nearest Cloud data center. The second phase setup displays the basic interaction between the Cloud servers and Fog nodes.Note that the users in phase 2 are the Fog servers from phase 1. We have presented a scenario where in transmission, data travels to the Cloud server based on One-on-One interaction for resolving the query. We take single Cloud data centers (DC1' in R0,DC2' in R1) to hande all the requests from the Phase 2 users (UB2' in R1 and UB1' in R0).

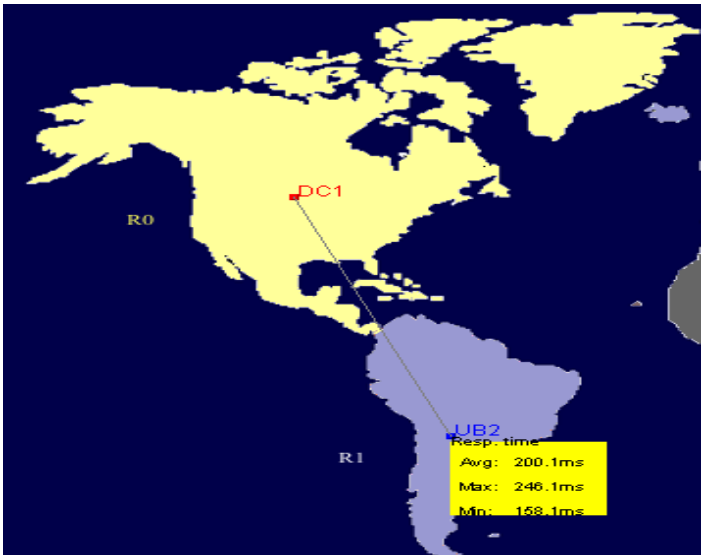

Fig. 3(a) Second phase: Cloud Tier-Interaction between Fog Server (UB2') and Cloud Server (DC1') (Fog Tier fails to resolve the user query)

The secong phase values are analysed only for a single Cloud transaction. The second phase can send data from R0 to R1 and from R1 to R0. The expenses will show negligible deviation for either of the configurations. So the transmission cost and time from UB1' to DC2 is assumed to be similar. The maximmum of cost and time (worst case scenario)used in a single transaction from Edge server to the Cloud server is added to the overall cost and time between the User and Fog tier transactions to calculate the total consumption of resources in the entire process between the user,the Edge and the Cloud networks collectively.

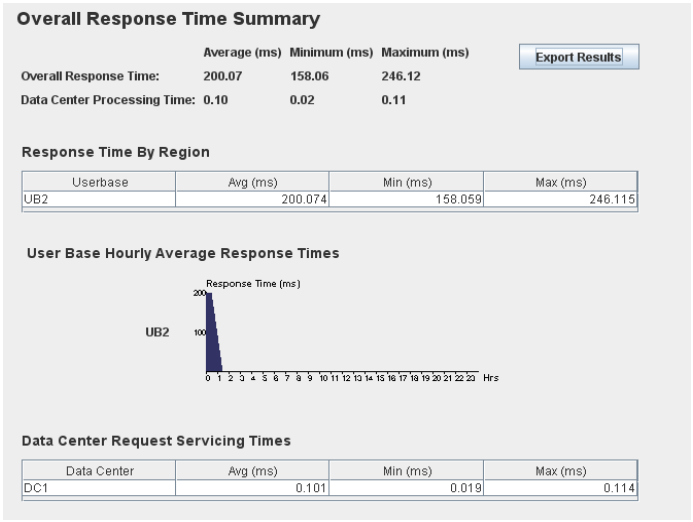

Fig. 3(b) User Response time summary for Fog-Cloud interaction

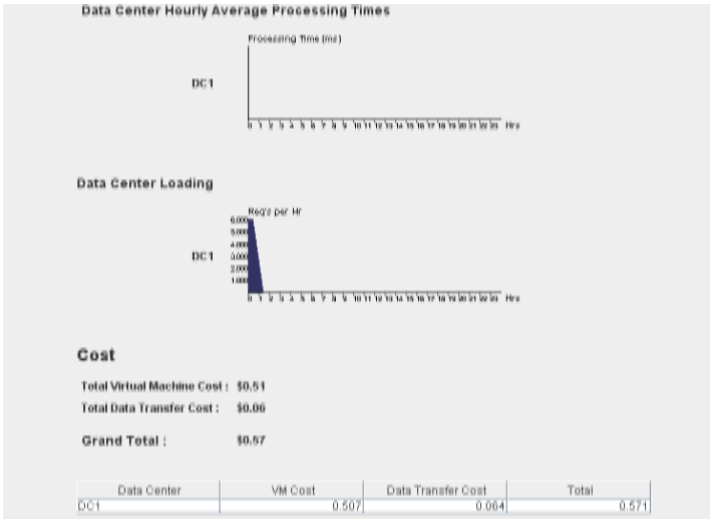

Fig. 3(c) Second phase: Processing Time,Server Load and Overall Cost in Fog-Cloud interaction

The simulation result is compared with results for Round Robin, Equally Spread Current Execution Load and Throttled policies used for transmitting data.

TABLE1. Comparison of algorithms on the basis of cost and duration in response among different tiers.

Algorithms		**Overall Response time (Max.)(ms)**	**Data Center Processing Time (Max.)(ms)**	**Total Cost (VM+ Data Transfer ($))**
Client-Cloud Tier	RR	375.11	4.01	0.96
	ESCEL	375.11	4.01	0.96
	TLB	375.11	0.95	0.96
Client-Fog-Cloud Tier	Edge Balancing (Proposed Algorithm)	308.76	1.0	1.63

VI. CONCLUSION

In this paper, we presented a new setup of the networks and users in an attempt to demonstrate how we can evolve the functionalities of the worldwide online environments through the Edge networks. To the best of our knowledge, this is an initial paper binding the study of load balancing, replication, grid and mesh networks with Fog computing platforms. We hope our work inspires and instigates other works in this primordial field of Cloud and Fog computing oriented collective research.

VII. FUTURE WORK

The reliability in a network system can be established by ensuring security of mobile data and authentication of user nodes and access points. The Confidentiality and Authentication of the user is essential. The future work emphasizes on creating security for the networks links and transactions providing privacy to the clients.

References

[1] I. Rowstron and P. Druschel "Storage Management and Caching in PAST, a Large Scale, Persistent Peer to Peer Storage Utility", SOSP '01 Proceedings of the eighteenth ACM symposium on Operating systems principles.

[2] F. Dabek, M. F. Kaashoek, D. Karger, R. Morris and I. Stocia, "Wide Area Cooperative Storage with CFS",SOSP '01 Proceedings of the eighteenth ACM symposium on Operating systems principles..

[3] X. Chen, S. Ren, H. Wang and X. Zhang "SCOPE: Scalable Consistency Maintenance in Structured P2P System" Proc. Institute of Electrical and Electronics Engineers INFOCOM, 2005.

[4] I. Clarke, O. Sandberg, B. Wiley and T. W. Hong "Freenet: A Distributed Anonymous Information Storage and Retrieval System ", ProcInt‟l Workshop Design Issue in Anonymity and Unobservability, pp. 46-66, 2001

[5] Haiying (Helen) Shen "IRM: Integrated File Replication and Consistency Maintenance in P2P System" Institute of Electrical and Electronics Engineers, Transactions On Parallel And Distributed Systems, Vol. 21, No. 1, January 2010.

[6] ZachariaFadika, ElifDede, Jessica Hartog and MadhusudhanGovindaraju "MARLA: MapReduce for Heterogeneous Clusters" 12th Institute of Electrical and Electronics Engineers ACM International Symposium on Cluster, Cloud and Grid Computing, 2012

[7] PriyaDeshpande, AniketBhaise, PrasannaJoeg"A Comparative analysis of Data Replication Strategies and Consistency Maintenance in Distributed File Systems" International Journal of Recent Technology and Engineering, International Journal of Recent Technology and Engineering ISSN: 2277-3878, Volume-2, Issue-1, March 2013.

[8] DejeneBoru ,DzmitryKliazovich , FabrizioGranelli , Pascal Bouvry , Albert Y. Zomaya "Energy-Efficient Data Replication in Cloud Computing Datacenters" Globecom 2013 Workshop - Cloud Computing Systems, Networks, and Applications.

[9] K. Hong, D. Lillethun, U. Ramachandran, B. Ottenwälder, and B. Koldehofe, "Mobile fog: A programming model for large-scale applications on the internet of things," in Proceedings of the Second SIGCOMM Workshop on Mobile Cloud Computing, ser. MCC'13. Association of Computing Machinery (ACM) ,pp 15–20.

[10] I. Ottenwalder, B. Koldehofe, K. Rothermel, and U. Ramachandran, "Migcep: Operator migration for mobility driven distributed complex event processing," in Proceedings of the 7th International Conference on Distributed Event-based Systems, ser. DEBS'13. Association of Computing Machinery (ACM) ,pp183–194.

[11] J. Zhu, D. Chan, M. Prabhu, P. Natarajan, H. Hu, and F. Bonomi, "Improving web sites performance using Edge servers in Fog computing architecture," in Service Oriented System Engineering (SOSE), 2013 Institute of Electrical and Electronics Engineers7th International Symposium 2013, pp. 320–323.

[12] "Building the environment for the things as a service," BETaaS, Tech. Rep., Nov. 2012.

[13] Willis Lang Jignesh M. Patel Jeffrey F. Naughton "On Energy Management, Load Balancing and Replication" Computer Sciences Department University of Wisconsin-Madison, USA