Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IMPACT FACTOR: 5.258

IJCSMC, Vol. 5, Issue. 7, July 2016, pg.106 – 112

ALLOCATING RESOURCES IN COLLABORATIVE CLOUD AND GRID COMPUTING USING LIS

Mr. Siddanna M Yaranal [1], Ms. Uma [2]
[1]P.G Scholar Student Department of Computer Science and Engineering
New Horizon College of Engineering
Bangalore, India
[2]Assistant Professor, Department of Computer Science and Engineering
New Horizon College of Engineering
Bangalore, India
[1] sidduyaranal2@gmail.com ; [2] umamam@gmail.com

Abstract-- A large resource of sharing system like collaborative cloud computing and grid computing generates way for sharing tremendous amount of resources distributed through Internet. Searching and locating desired resources is a critical component of resource sharing system. In addition with the resource discovery, efficiency and fidelity is also an important quality. Earlier systems either achieve high fidelity at low efficiency, or high efficiency at low fidelity .Proposed system introduces a resource information service that provides high efficiency and fidelity. Proposed service outperforms other services in terms of efficiency, fidelity, and flexibility without decreasing resource expressiveness. It reduces overhead and gives significant enrichment in efficiency and fidelity.

Keywords: Resources, Fidelity, Flexibility, Efficiency, Collaborative, Grid.

I. INTRODUCTION

The ability to find services or resources that satisfy some criteria is an important aspect of distributed systems[2]. Large-amount of resource sharing system generates a virtual space by providing a platform for sharing tremendous amounts of resources over the Internet [4]. For example we have collaborative cloud

computing (allows connection for multiple clouds and come forward to share resources) and grid computing (collection of computer resources from multiple locations to reach the common goal).
There are three challenges to get an effective resource information service

- First challenge is to bring about high efficiency- achieved by large scale scattering of resources and dynamics.
- Second challenge is to guarantee the high fidelity of resource location- achieved by locating the resources where it is kept and in which cloud.
- Third challenge is to conclude flexibility- achieved by getting similar type of resource in spite of searching same resource.

II. LITERATURE SURVEY

1. Entropia: Architecture and Performance of an Enterprise Desktop Grid System

Authors Andrew A. Chien, Brad Calder, Stephen Elbert, and Karan Bhatia Explained, It has ability to provide grid computing, distributed computing. It has some feature high-throughput computing, and scientific computing. The problem with the paper is system architecture provides a solid foundation to meet the technical challenges as the use of distributed computing matures-supporting the broadening the problems supportable by increasing the breadth of computational structure, resource usage, and ease of application integration.

2. MAAN: A Multi-Attribute Addressable Network for Grid Information Services

Authors Min Cai, Martin Frank, Jinbo Chen, Pedro Szekely Explained, It has ability to provide Grid Computing, Peer-to-Peer, Information Services. It has feature of Multi-attribute Range Queries. The problem with the paper is Centralized approaches have the inherent drawback of a single point of failure.

3. Cycloid: A constant-degree and lookup-efficient P2P overlay network

Authors Haiying Shena, Cheng-Zhong Xua, Guihai Chenb Explained, It has ability to provide cycloid, Peer-to-peer, Viceroy, Koorde. It has some feature Distributed hash table, Constant-degree DHT. The problem with the paper is choices are frequently more expensive than those of the cost-aware approach.

4. Efficient Event-based Resource Discovery

Authors Wei Yan, Songlin Hu, Vinod Muthusamy, Hans-Arno Jacobsen, and Li Zha Explained, It has ability to provide Service discovery, resource discovery, content-based publish/subscribe. It has some feature publish/subscribe applications, covering, and subscription similarity. Problem with the paper is limitation of the Mercury algorithm is that a multi-attribute query is decomposed into a set of single attribute queries that must be processed sequentially or in parallel.

5. Goal-Based Request Cloud Resource Broker in Medical Application

Authors Mohamad Izuddin Nordin, Azween Abdullah, and Maham at Issa Hassan explained, it has ability to provide Broker, Cloud Computing, and Medical Informatics. It has some feature Resources

Discovery, and Resource Selection. Problem with the paper is drawback from this action is the data owner can restrict the usage of services provided to the certain number of data which has permission to see.

6. Javelin, scalability issues in global computing

Authors Michael O. Neary, Sean P. Brydon, Paul Kmiec, Sami Rollins and Peter Cappello Explained, It has ability to provide distributed computing. It has some feature high performance computing, Java; internet, scalable. Problem with the paper is choices are frequently more expensive than those of the cost-aware approach.

III. EXISTING SYSTEM

Present resource discovery approaches in clouds and traditional approaches for grids are in a centralized manner. These centralized approaches are insufficiently scalable due to cannot share same space at a same time (bottlenecks) and if a part of system fails entire system get stopped (single point of failure). Resource providers report their available resource information to a resource information server (one or several brokers), and then resource requesters contact the server for their desired resources [5]. The information server can be overloaded in a large-scale system leading to service inefficiency in turn the entire resource discovery mechanism will fail.

Problems

- Current DHT based resource information service systems either provide high fidelity at low efficiency or achieve high efficiency at low fidelity.
- Current systems have limited flexibility by only allowing exact-matching services or describing a resource using a present attribute list.
- Current approaches have high fidelity, they experience a high overhead for resource pooling, searching, and merging.
- One-point mapping systems allows high efficiency at the cost of low 0fidelity.

IV. PROPOSED SYSTEM

Distributed Hash Tables (DHTs) provide a favour solution for resource information services with their permanent properties of high scalability, self-organization and fault-tolerance in large scale distributed systems [4]. Presenting LSH-based resource information Service (LIS) that combines efficiency, fidelity, and flexibility.LIS offers high efficiency and fidelity without stopping resource expressiveness, while providing a similar-matching service. It states to achieve high efficiency, fidelity and flexibility. Three algorithms are introduced to transform a resource description for set of integers. Further building LSH functions by combining the algorithms with min-wise independent permutations. LIS also includes a load balancing algorithm to balance the load for storing resource information and process the resource requests.

Advantages

- Proposed Scheme provides flexibility but also helps to reduce on high, since a requester only receives the information of similar type of resources.

- Get high efficiency (increase in the speed), fidelity (Finding exact location of resources) and flexibility (sharing of resources).

- It is successful in locating the resources in an environment with an large number of resource attributes.

System design

The system design process builds up general framework for generating design.

i) Design Consideration

The reason for the design is to arrange the problem determined by the necessities report. This stage is the start phase in moving from issue to the arrangement space.

ii) System Architecture

The architectural configuration procedure is concerned with generating up a fundamental basic system for a framework. It includes recognizing the existing parts of the framework and exchanges between these segments.

Data Flow Diagram

The DFD is straightforward graphical representation that can be used to speak to a framework as far as the information to the framework, different preparing did on this information and the yield information generated by the framework. A DFD model uses an exceptionally predetermined number of primitive images to speak to the capacities performed by a framework and the information stream among the capacities.

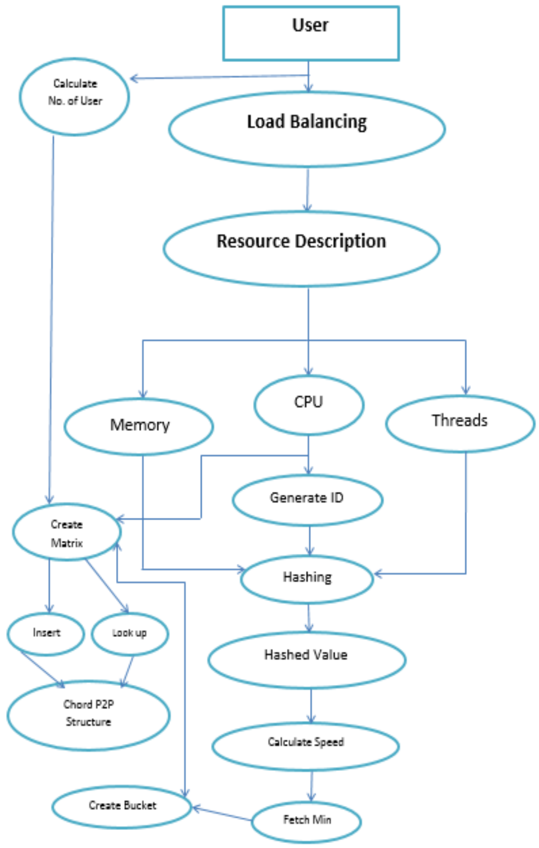

Fig 1. DFD of LIS

V. IMPLEMENTATION

Module 1: Resource Attribute Transformation

A resource description consists of a set of attributes (including values). To transform each attribute of a resource into an integer, we propose three attribute transformation algorithms with similarity preserving features for each attribute.

Algorithm 1: Building of min-wise independent permutation had been done. Later attributes are transformed in to integers. A set of hash value is generated for resources, lastly using XOR operation ID is generated for resource.

Algorithm 2: Storing the resource predicates had been done by using generated ID's of resource. Operating system while storage cloud provides database and large capacity hard disk to store [1].

Algorithm 3: Querying the resource had been done by using pseudo-code.

Module 2: Locality-Sensitive Resource Translation

Using the introduced attribute transformation algorithms and min-wise independent permutations, develop new LSH functions that do not need a pre-defined attribute list.

Module 3: DHT-Based Resource Information Service

LIS is built on top of a DHT overlay formed by all nodes in the system to achieve multi-attribute resource searching. In the above sections, we introduced how to translate resources to IDs. Using these resource IDs, resource providers store the information of their available resources to the DHT, and resource requests will be forwarded to the DHT nodes storing the information of the required resources. Thus, the DHT functions as a matchmaker between resource providers and requesters. A DHT overlay is a decentralized system in the application level that partitions ownership. We use Chord as a representative of DHT overlays to explain the LIS resource information service [3].

Load Balancing

To avoid overloading at any node, aim to achieve every node via load balancing algorithm [6].

VI. EXPERIMENTAL RESULTS

Graph depicts the query processing latency of different services. It shows that MAAN leads to dramatically higher query processing latency than LIS. For a query consisting of attributes, MAAN generates l queries and searches all resources owning each attribute. Therefore, MAAN needs a very long time satisfying resources in the emerging phase. In LIS, the resource vector size is equal to the number of attributes in the resource description, so it produces the lowest query processing latency among all methods. This result shows the benefits of avoiding building a multi-dimensional space to produce resource IDs, and demonstrates that LIS outperforms all others in terms of query processing latency.

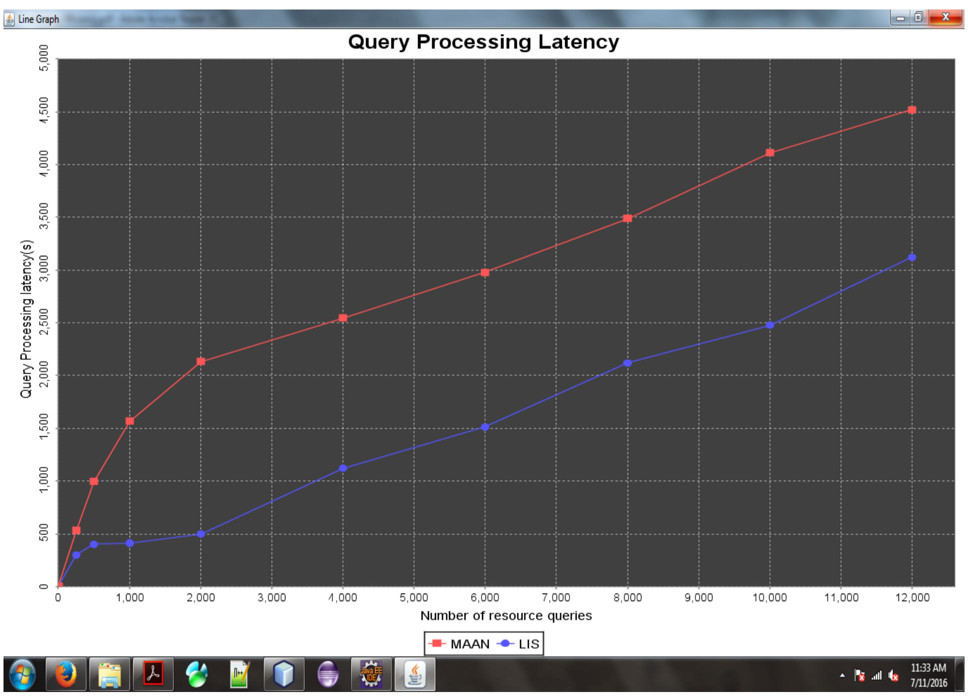

Fig 2.Experimental Results

VII. CONCLUSION AND FUTURE WORK

As earlier proposed resource information services for large scale resource sharing systems have limited flexibility. Presenting an efficient and high-fidelity LSH based resource Information service (LIS) effective in locating satisfying resources and Provides high flexibility. LIS built on DHT overlay which allows efficient resource data pooling and searching in large-scale resource sharing systems.. SHA-LIS only offers attribute exact-matching in resource discovery, similarity search that can find attributes with similar characters (e.g., MEM and memory). Future work is creating a cluster of users having similar type of resource.

REFERENCES

[1] M. Nordin, A. Abdullah, and M. Hassan,"Goal-based request cloud resource broker in medical application,"WASET, vol. 50, no. 74,pp. 770–774, 2011.
[2] W. Yan, S. Hu, V. Muthusamy, H. Jacobsen, and L. Zha,"Efficient event-based resource discovery,"in Proc. Int. Conf. Distrib. Event-Based Syst. (DEBS), 2009, pp. 1–10.
[3] H. Shen, C. Xu, and G. Chen,"Cycloid: A scalable constant-degree P2P overlay network,"Perform.Eval., vol. 63, no. 3, pp. 195–216, 2006.
[4] M. Cai, M. Frank, and P. Szekely,"MAAN: A multi-attribute addressable network for grid information services,"Grid Comput.,vol. 2, no. 1, pp. 3–14, 2004.
[5] A. Chien, B. Calder, S. Elbert, and K. Bhatia,"Entropia: Architecture and performance of an enterprise desktop grid system,"J. Parallel Distrib. Comput., vol. 63, no. 5, pp.597–610, May 2003.
[6] M. O. Neary, S. P. Brydon, P. Kmiec, S. Rollins, and P. Capello,"Javelin++: Scalability issues in global computing,"Future Gener. Comput. Syst. J., vol. 15, no. 5–6, pp. 659–674, 1999.
[7] S. Surana, B. Godfrey, K. Lakshminarayanan, R. Karp, and I. Stoica,"Load balancing in dynamic structured P2P systems,"Perform. Eval.,vol. 63, no. 3, pp. 217–240, 2006.
[8] R. Ranjan, B. Benatallah, and M. Wang,"A cloud resource orchestration framework for simplifying the management of web applications,"in Proc.Int.Conf.Service-Oriented Comput. (ICSOC), 2012, pp. 248–249.
[9] J.Li,B.Li,Z.Du,andL.Meng,"Cloudvo: Building a secure virtual organization for multiple clouds collaboration, "In Proc. Int.Conf.Softw.Eng.Artif.Intell.Netw.Parallel/Distrib.Comput.(SNPD),2010, pp. 181–186.
[10] C.Liu,B.T.Loo,andY. Mao,"Declarative automatedcloud resource or chestration,"In Proc. Symp. Cloud Comput. (SOCC), 2011, p. 26.
[11] C. Liu, Y. Mao, J. E. Van der Merwe, and M. F. Fernandez,"Cloud resource orchestration: A datacentric approach,"In Proc. Conf. Innovat. Data Syst. Res. (CIDR), 2011, pp. 241–248.
[12] H. Bandara and A. Jayasumana,"On characteristics, and modeling of P2P resources with correlated static, and dynamic attributes,"In Proc. Global Commun. Conf. (GLOBECOM), 2011, pp. 1–6.
[13] K. Psounis, P. M. Fernandez, B. Prabhakar, and F. Papadopoulos,"Systems with multiple servers under heavy-tailed workloads,"Perform. Eval., vol. 62 (1–4), pp. 456–474, 2005.
[14] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson,"Self-similarity through high-variability: Statistical analysis of Ethernet LAN traffic at the source level,"IEEE/ACM Trans. Netw., vol. 5,no. 1, pp. 71–86, Feb. 1997.
[15] X. Zhang, Y. Qu, and L. Xiao,"Improving distributed workload performance by sharing both CPU, and memory resources,"In Proc.Int. Conf. Distrib. Comput. Syst. (ICDCS), 2000, pp. 233–241.
[16] H. Shen,"Combining efficiency,fidelity, and flexibility in grid information services,"In Proc. Int. Conf. Comput. Commun. Netw.(ICCCN), 2009, pp. 1–6.

