Performance Analysis of Agent Based Framework

Sindhuja K a, Monisha A.V. b, Dr.S.Padmavathi c a 1, b, c

a Post Graduate Student Computer Science and Engineering, Thiagarajar College of Engineering, Madurai and 625015, India
b Under Graduate Student Computer Science and Engineering, Thiagarajar College of Engineering, Madurai and 625015, India
c Associate Professor Computer Science and Engineering, Thiagarajar College of Engineering, Madurai and 625015, India

Abstract

Cloud computing is the delivery of on-demand computing resources – everything from applications to datacenters over the internet on pay per use basis. Service composition in cloud is an important mechanism usually done through several scheduling policies or by dynamic resource allocation techniques. Services delivered to the users lack Quality of Service during this composition phase. Hence, an Agent-based cloud computing paradigm for resource allocation is proposed that consist of designing and developing software agents for cloud service discovery, finding appropriate service and service composition. Here, two types of agents are created – consumer agent that process the user requirements and service provider agent that clusters the output from consumer agent. Finally, each user is mapped to each producer. Experimental results shows efficient execution of user requests using agent with 100% success execution rate in a parallel manner than that of using Hadoop.

© 2015 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).
Peer-review under responsibility of organizing committee of the Graph Algorithms, High Performance Implementations and Applications (ICGHIA2014)

Keywords: Agent; Hadoop; Service discovery; Service composition; Consumer Agent; Service Provider Agent

1. Introduction

A cloud is a large group of interconnected computers that extend beyond a single company or enterprise. Cloud computing leverages virtualization technology to achieve the goal of providing computing resources as a

1 * Corresponding author. Tel.: +91-984-235-1031.
E-mail address: sindhuja.msec@gmail.com
utility. It shares certain aspects with grid computing and autonomic computing but differs from them in other aspects. Cloud computing has recently emerged as a new paradigm for hosting and delivering services such as IaaS (Infrastructure as a Service), PaaS (Platform as a Service), SaaS (Software as a Service) over the Internet. It is attractive to business owners as it eliminates the requirement for users to plan ahead for provisioning, and allows enterprises to start from the small and increase resources only when there is a rise in service demand. As cloud computing services rapidly expand their customer base, it has become important to share cloud resources, so as to provide them economically. In cloud computing services, multiple types of resources, such as processing ability, bandwidth and storage, need to be allocated simultaneously.

Web service composition is an important step towards development of web service applications. If no single web service can satisfy the functionality required by users, there should be possibility to combine existing services together in order to fulfil the request. Several initiatives such as usage of UDDI (Universal Description, Discovery and Integration), WSDL (Web Services Description Languages), SOAP (Simple Object Access Protocol) have been conducted that will allow easy integration of heterogeneous system. Despite all these efforts, web service composition still a complex task and it is already beyond the human capability to deal with whole process manually. Moreover, in current scenario, there might be very few service providers for cloud computing and there might be a very few services provided by them. But if the service providers and the services will increase in future, it would be difficult to find the optimized services on own. Hence several methods and frameworks have been proposed to carry out the process of service composition that includes Agents and Hadoop in cloud computing.

Efficiency of web service management can be improvised using hadoop. It can overcome the drawback which occurs in the traditional web service management infrastructure using Map -Reduce technique. These Map-Reduce tool usually offers automatic parallelization, load balancing, disk transfer optimization, handling of machine failure, and robustness. A framework for multi-agent simulation on a hadoop cloud where agents map themselves to data nodes has been proposed. Using this framework, applications involving thousands of nodes and peta bytes of data can be implemented. But problem arises in data replication phase during map-reduce and also when there is failure in the master node resulting in full system shutdown.

So, this paper stresses the need for some kind of automated procedure that will overcome drawbacks of hadoop and take care of the things no matter how many service providers are there and how many services they are providing. Agents can be used intelligently as a service which can be delivered to cloud consumers which may act as a solution to several issues in cloud computing services. The outcome of this work would deliver services to users on demand. Agents do the work of discovering services, negotiation the best price and finally giving the user a composite service.

The rest of this paper is organized as follows: Section 2 describes Agent Based System, desirable features of agents and its advantages. Section 3 includes the proposed work of agents and its architecture. Section 4 presents a comparative study and performance evaluation of the proposed strategy and Hadoop. Finally, the paper is concluded in Section 5.

2. Agent Based System

The association of Agent and the Cloud is beneficial for both the parties: Cloud User and Cloud Providers. A simple agent program can be defined mathematically as an agent function which maps every possible percept sequence to a possible action the agent can perform or to a coefficient, function or constant that affects eventual actions.

The program agent, maps every possible percept to an action. Percept refers to the agent's perception inputs at any given instant. A multi-agent system consists of number of agents, which interact with one another. Adopting multi-agent system in applications will bring the following advantages:

1. The operation will be faster due to the parallel processing
2. The demand for communication bandwidth is lower because the information processing is carried out near the
(3) Errors of one Agent will not affect the whole system, so the system has higher reliability and
(4) System has higher response speed.

2.1 Desirable Features of Agents

Agents typically include a set of features. The main features of agents include the following:

- **Autonomy**: The capacity to act autonomously to some degree on behalf of users or other programs also by modifying the way in which they achieve their objectives.
- **Pro-activity**: The capacity to pursue their own individual set goals, including by making decisions as result of internal decisions.
- **Re-activity**: The capacity to react to external events and stimuli and consequently adapt their behaviour and make decisions to carry out their tasks.
- **Communication and Cooperation**: The capacity to interact and communicate with other agents (in multiple agent systems), to exchange information, receive instructions and give responses and cooperate to fulfil their own goals.
- **Negotiation**: The capability to carry out organized conversations to achieve a degree of cooperation with other agents.
- **Learning**: Agents improve performance and decision making over time when interacting with the external environment.

3. Proposed Work

The proposed work is to allocate cloud resources to the consumers dynamically by enabling collaboration between consumer agents and service provider agents in an automated manner. This work can be achieved using Contract net protocol which can successfully compose cloud services by autonomously selecting services. Agents in contract net protocol have two roles. 1) Initiator 2) Participants. Consumer adopting initiator role broadcast call-for-proposals to satisfy user’s request. The participants may reply with proposal to carry out the task or a refusal message.

The phases involved in an agent-based cloud computing starting from getting the user requirement from cloud consumers, processing it, mapping it to the producers and delivering the services are given below in Fig. 1.

![Service Composition process](image-url)
3.1 System Architecture

The agent-based architecture is composed of

**Consumer agent** provides a single virtualized service to cloud consumers. This is achieved by receiving consumer’s requirements, mapping the requirements to available cloud resources and selecting the best (cheapest) agent. The consumer agent act on behalf of cloud consumers by selecting best producers with the specified constraints.

**Service provider agent** will cluster the producers based on location. This agent allows user to select producers from specific location. It acts as an intermediate agent between producers (repository) and consumers as it helps in selecting producers from list provided by consumer agent.

![Diagram of Interaction between consumer and service provider agent](image)

Interaction between consumer and service provider agent is shown in Fig. 2. In cloud, there can be large number of consumers requesting for services which are mapped to agents. If there are ‘n’ numbers of consumer, it can be either mapped to ‘n’ agents or to a single agent that can do the allocation process.

4. Results and Discussions

An agent-based approach for cloud service composition was implemented using JADE (Java Agent Development) framework and these results were simulated using CLOUDSIM. Using JADE, consumer and service provider agents are created. Their working is formulated mathematically with variables like quantity, budget, price and location. There are two constraints (availability, budget) to be checked so as to optimally select the producer who provides best services. Finally, the output is passed to cloudsim which maps user request to virtual machines. All user requests are processed at the same time as every user runs in separate virtual machines. The experiment was carried out on a computer with following specifications: Intel Core 2 Duo CPU, 2.10 GHz, 2 GB RAM, with a Windows 8 (32 bits) operating system.
4.1 Comparative study

In this study, two platforms have been analysed and it is noted that JADE improves reliability of Hadoop using agents. Table 1 illustrates the differences between Hadoop and Agent (JADE).

| FACTORS               | HADOOP                                      | JADE                                      |
|-----------------------|---------------------------------------------|-------------------------------------------|
| Architecture          | Client/Server file system                   | Distributed multi-agent framework         |
| Language              | Java                                        | Java                                      |
| License               | Open-Source                                 | Open-Source                               |
| Startup time          | Long                                        | Less                                      |
| Performance           | Less                                        | Better                                    |
| Reliability           | Reliable                                    | More Reliable                             |
| Modules used          | Map-Reduce, HDFS (Hadoop Distributed File System) | AMS (Agent Management System), DF (Directory Facilitator) |
| Fault Tolerance       | Master Node failure – system shutdown       | Failure of one agent does not affect the system |
| Year founded          | 2005                                        | 2000                                      |
| Mobility              | N/A                                         | Support                                   |
| Allocation Tasks      | Support                                     | Support                                   |
| Scheduling Tasks      | N/A                                         | Support                                   |
| Methodology           | Object-Oriented                             | Object-Oriented                           |
| Application           | Marketing Analytics, Image Processing, Machine Learning | E-Commerce, Military, Mobile Telecommunication |

Thus, the comparison table gives a clear idea of both Hadoop and Agent Framework. Several tasks can be scheduled and executed effectively using agents. Agent mobility is the ability for an agent program to migrate or to make a copy (clone) itself across one or multiple network hosts. With JADE framework, it is possible to build agents powering these abilities. But such property is not applicable using hadoop. Many such comparisons are made that proves higher efficiency of agents.

4.2 Variation of User Request using Agent

Agent Response time by varying user request is analysed in Fig. 3. Homogeneous task with feasible constraints is compared with the heterogeneous task with less feasible constraints. Homogeneous task refers to user requests having similar requirements whereas heterogeneous task refers to user requests having no logical relations between several user requirements.
This feasibility constraint is based on the Producer Table as shown in Table 2. This analysis is carried out by taking list of all producers from the producer table. The producer table contains Producer name, Location, Variety, Quantity, Selling price. In the table created, quantity of the items is limited to be feasible (i.e. 25nos.). Thus, the execution time with feasible constraints (i.e., quantity \( \leq 20 \)) searches through the producer table and displays almost all possible data satisfying user’s requirement. Hence this may take long time. Whereas, execution time with less feasible constraints (i.e., quantity > 20) displays less or no producers from the table as only few user constraints are satisfied. Thus it may take less time to map the users to producers than mapping users to all feasible producers.

Table 2. Producer Table

| ID | PRODUCER_NAME | LOCATION | VARIETY | QUANTITY | SELLING_PRICE |
|----|---------------|----------|---------|----------|---------------|
| 1  | P1            | Chennai  | Large   | 8        | 7             |
| 2  | P9            | Madurai  | Small   | 10       | 3             |
| 3  | P10           | Salem    | Medium  | 18       | 5             |
| 4  | P2            | Erode    | Large   | 20       | 6             |
| 5  | P8            | Madurai  | Medium  | 15       | 6             |

4.3 Performance analysis

The execution time for service composition using JADE framework is compared and analysed with hadoop framework as shown in Fig.4.
In hadoop based web service management, Map-Reduce and a indexing mechanism is designed to manage web services with different properties and to retrieve interfaces which satisfy user’s requirement from HBase Table. User requests processed using hadoop with index formulation takes more time when compared to processing user requests using agent. If number of user request is 2, then the execution time for processing it with the help of hadoop is 1200ms while the execution time using agents is 127ms. Thus, success rate of service composition is higher in agent based cloud computing than in a hadoop framework.

5. Conclusion and Future Work

In this paper, we have analysed and compared service composition approach using agent and hadoop framework. Comparative study shows that execution time for processing user request using agents gives better results than that of using Hadoop. Maximum resource utilization and success rate in agent based cloud computing are achieved as there is a reduction in the negotiating time and also due to the parallel execution of user requests by creating number of VMs. Our future work will focus in designing a framework in real-time to evaluate the scalability of the agent-based cloud service composition approach.

References

1. Octavio Gutierrez-Garcia, Kwang Mong Sim, GA-based Cloud Resource estimation for Agent-based execution of Bag-of-Task, Springer, Applications and Information System Frontier; 2011. Vol.14, Issue 4.
2. Kwang Mong Sim, Agent-Based Cloud Computing, IEEE Transactions on Service Computing; 2012. Vol. 5.
3. Shangguang Wang, Wei Su, Xilu Zhu, Hongke Zhang, A Hadoop Based Approach for Efficient Web Service Management, International Journal of Web and Grid Services; 2012.
4. J.Octavio Gutierrez-Garcia, Kwang Mong Sim, A Family of Heuristics for Agent-based Elastic Cloud Bag-of-Tasks Concurrent Scheduling, Elsevier Future Generation Computer Systems; 2013. Vol. 29, Issue 7.
5. Rajesh, Web Service Composition using Multi-Agent System, International Journal of Research in Engineering & Advanced Technology; 2013 Vol. 1, Issue: 1.
6. A.K. Srivastava, Ms Vandana Srivastava, Ms. Richa Bhargava, Towards developing an Intelligent Agent for Cloud Computing, International Conference on Cloud, Big Data and Trust; 2013. pp. 13-15.
7. Prashant Sethia, Kamalakar Karlapalem, A Multi-agent simulation framework on small Hadoop Cluster, Springer, Engineering Applications of Artificial Intelligence; 2011.
8. Jinghai Rao, Xiaomeng Su, A survey of Automated Web Service Composition Methods, *International Workshop on Semantic Web Services and Web Process Composition*; 2005. vol. 3387.

9. Youssef, Gamal, Ayman, Mobile Agent based New Framework for Improvising Big Data Analysis , *IEEE Conference on Cloud computing and Big Data*; 2013.

10. Stuart Russell, Peter Norvig, *Artificial Intelligence: A Modern Approach*; 2nd Edition; 2003. chap. 2.

11. J.Octavio Gutierrez- Garcia, Kwang Mong Sim, Agent based Cloud Service Composition, *Elsevier Applied Intelligence*; 2013. Vol. 22, No. 2.