International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 1 Issue 9, November- 2012

Service Composition Based on Multi Agent in Cloud Environment

Khushbu Virani[1], Dhara Virani[2]
[1]Post Graduate Student, Marwadi Education Foundation Group Of institutions, Rajkot, Gujarat.
[2]Post Graduate Student, Marwadi Education Foundation Group Of institutions, Rajkot, Gujarat.

Abstract

Web service composition is one of the challenging tasks which combine two or more service together from different service provider. There is no single web service which can satisfy the user requirement so we have to combine existing service together for that we require service composition. In this paper we present several methods on service composition based on agent paradigm. Agent concerns with the Cloud service discovery, service negotiation and service composition. Here we are presenting approach, framework for service composition and some techniques which are currently available for service composition.

1. Introduction

Cloud computing provide elastic services, high performance and scalable data storage to a large and everyday increasing number of users. Here we are using agent based technique to provide effective service. An agent is a computational entity that acts on behalf of another entity (or entities) to perform a task or achieve a given goal [3]. Generally a cloud service lifecycle consists of service requirements, service discovery, service negotiation, service composition, and service consumption [1]. There is no single web service which can satisfy the user requirement so we have to combine existing service together for that we require service composition so our main focus is on service composition [8].

Multi-Agent system represent computing paradigm based on multiple interacting agents that are capable to intelligent behaviour. Software agent used some AI approach based on some co-operation among several agents result in a solving large complex problem which keeps execution time low. Main focus is of cloud computing is efficient use of infrastructure in reduced cost. MAS can run on a cloud infrastructure or most compute-intensive part of it can be hosted in cloud whereas the light part can run on a local server or simply on client pc. So finally agent become more efficient and at the same time lighter & smarter [3].

First section of this paper presents a different approach for web service composition. It also includes language used for web service composition. Second section includes some related techniques which are currently used for web service composition. Third section includes challenges related to web service composition. Finally at last we present a comparative analysis of existing available techniques.

2. Approach

First of all we have to define web service model. Generally web service model consists of three individuals, the service provider, the service registry and the service consumer [13].

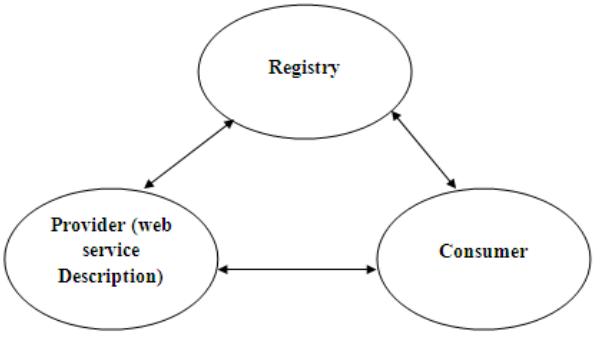

Fig. 1 Web Service Model

Basically, there is a 5 step model for service composition process. Each step requires different language as well as different platforms. Fig. 1 shows the web service model. Fig. 2 shows the steps of service composition. Steps include service presentation, translation, process generation, evaluation & execution [5].

There are 4 different languages which are generally used for service composition. 1) UDDI (Universal Description Discovery Integration), 2) WSDL (Web Service Description Language), 3) SOAP (Simple Object Access Protocol) & 4) BPEL4WS (Business Process Execution for Web Service) [8].

3. Related techniques

Tong [2] proposed that there are three categories of web service composition i.e. 1) Manual composition 2) Semiautomatic composition 3) Automatic composition. In manual composition BPEL4WS language is used and Domain is generally known to the user. But there are some disadvantages of manual composition like it is a labor intensive and error prone task. Again it is not used for large scale web service composition. Semiautomatic composition not used for large scale web service composition. For that they proposed AND/OR graph technique. Now a day mostly used automatic service which is based on AI planning. For that initially we have to give input as well as output parameters. Finally Tong [2] proposed agent based service composition approach which is not fully automatic solves some problems of automatic composition but it is still composition but describes coordinate & performance of composite web services.

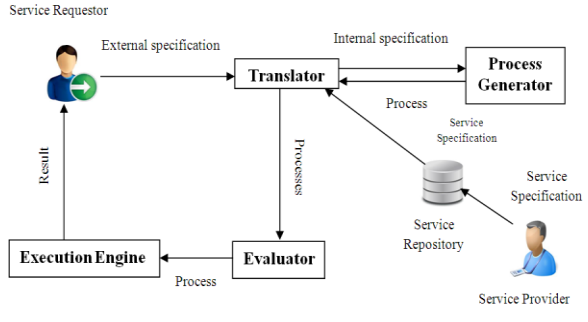

Figure 2. Framework of the service composition system

A) DPAWSC (Distributed Planning Algorithm for Web Service Composition)

DPAWSC (Distributed Planning Algorithm for Web Service Composition) is generally used for web service composition using service agent. Key idea of this algorithm is with smaller length has higher priority to be searched than one with larger length. Service agent model used 4 elements belief, goal, action, and plan. First have to collect requirement then apply distributed service agent planning then quality optimization & at last execution [2].

Here main thing is the quality optimization stage which combines service based on actual templates according to the quality requirement. The output of this stage give effective web service composition and at last service will be delivered to the user. Here they are considering plan which containing smaller no. of plans is always better than having a more no. of plans. So there is a less collaboration among service agent. So management is easy and ability to produce high quality solution at low cost of communication [2].

B) Context Oriented Approach

Maamar [11] proposed web service composition using context oriented approach. In this approach web service composition according to the relevant information is used. For context they are considering how context is internally represented? Where it is stored? Whether it is locally or on network? How fast we can get and after then conversation is established. Conversation can be succeeded or failed. Some services are mandatory and some services are optional. Mandatory service includes the compulsory participation of all the component service in the execution process whereas in optional skipped some service during execution. Again they are considering two approaches i.e. 1) Proactive composition 2) Reactive composition. Proactive information gathers information in advance while reactive composition gathers service on demand. Reactive has several advantages over proactive composition. For reactive composition manager requires & ensures the identification, relationship of composite services. There are three input participant which are used in context development i.e. 1) Service, 2) User and 3) Both. User centric mechanism provides configuration mechanism based on user's personal preference. Service centric mechanism deals with

service availability and support on demand service composition.

Agent based context oriented approach dividing the agent into three categories. 1) Composite service agent, 2) master service agent, and 3) service agent. Here in this they are considering web service context which is work based on W-context which includes 1) Number of service which is currently running versus maximum no. of service that can be simultaneously run 2) Status of each service 3) Request time versus available time of service instance. First of all request originating from composite service agent then for processing service will pass to the master service agent and MSA take decision whether a web service authorized to join a composite service or not. For approval I-context is created. Authorization will reject because of some reason like non availability of service, overloaded status or exception situation. I-context provides following detail to W-context. 1) Execution status of service, 2) why service instance supposed to resume, 3) when execution completes, and 4) what are the corrective actions if service execution fails. For security identifying malicious pattern for that some pattern database is there in which some rules are satisfied under which circumstances a threat is malicious. So from this approach we can say that maximum no. of service can be executed at the same time and availability of instance for a certain period of time [11].

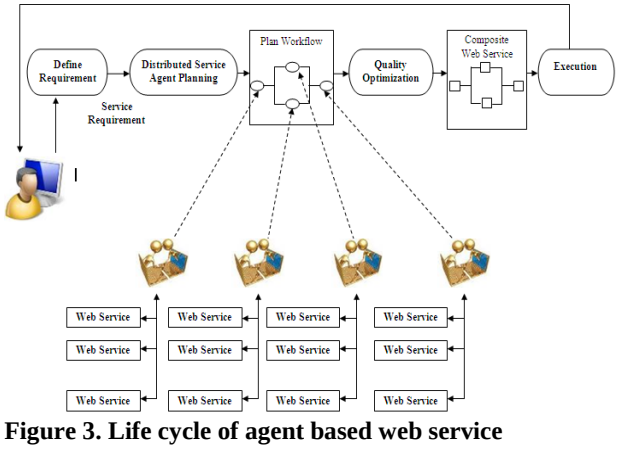

Figure 3. Life cycle of agent based web service composition

C) Web service composition based on FSCNP & SCT

Sim [1] suggest service composition based on FSCNP (Focus Selection Contract Net Protocol) and SCT (Service Capability Table). There is already available protocol CNP (Contract Net Protocol) which is work like 1) coordination and interaction among Cloud participants (consumers, brokers, and providers), 2) automation of service selection, 3) dynamic (re-)configuration of distributed and parallel services, and 4) dealing with incomplete information about the existence and location of Cloud providers and their services. Here SCT is used which contains the record of cloud service provided by other agent in cloud system. FSCNP is used for specifying the interaction of cloud agents. Analyzing the number of messages exchanged among agents in FSCNP in the worst case and showing that FSCNP enhances the efficiency of classical contract net protocol. In FSCNP agent focus on selecting relevant cloud service based on SCT. Thereby reducing the no. of message exchanged among cloud agents. For that agent have knowledge of acquaintance network and SCT. SCT includes service capability of agents in cloud system & states of cloud agents. State may be available, failed or busy. In CNP an agent attempt to select service from other agent by broadcasting its request to all other agent in the system. In FSCNP an agent consult with SCT according to SCT it should send its request message. So interaction among agent is more efficient and no. of message passing exchange is reduced. Thereby it provides relevant cloud services and provide higher success rate.

D) Web service composition based on Acquaintance N/W & CNP

Octavio [4] proposed self-organizing agent technique which is based on acquaintance network & CNP. For that web service specification is created and some agents are used like 1) Resource Agent, 2) Service Provider Agent, 3) Broker Agent, and 4) Consumer Agent. Acquaintance is a network or list of available agents. For that they are considering 3 different acquaintance networks. 1) Consumer agents acquaintance N/W, 2) Broker agents acquaintance N/W, and 3) Service provider agents acquaintance N/W. CNP is used for dynamically selecting cloud service. Agent in CNP have two roles i.e. initiator & participator. First of all initiator send a call-for-proposal to n participants. So participate will reply with price for fulfilling the requirements. Then initiator evaluates the received proposal & selects the best proposal. Again initiator will send reject proposal to

select agent. The complexity for the broker agent's behaviours is bounded by: O (m (m + 1)/2) + O (n (n + 1)/2). Finally it provides the complete information & dynamically selecting cloud services.

E) Web service composition in decentralized orchestration

Chafle [9] proposed composition of web services in decentralized organization. Generally a composite service executed by single co-ordinator node. So it receives client request, make required data transformation & call component service as per specification. So performance bottleneck problem is there. So it leads to traffic on network again web service generates a lots of data that is irrelevant to composite services. So disadvantage of centralized orchestration is poor scalability & performance degradation at high load. So decentralize orchestration technique is proposed. In this there are multiple engines, each executing a composite web service specification of distributed locations. Advantage is that there is no centralize coordinator which can be potential bottleneck. It reduces network traffic & improves concurrency. But in this decentralize orchestration some build time & run time issues are there. Build time issues include how to efficiently partition the centralized specification & error handling across partition. For that code partitioning techniques are used so code execute at distribute locations and can be called remotely. Again because of asynchronous messaging between the different composite web service partitions make error propagation in a decentralized setup more complex. Thereby they use additional fault handler so error propagate correctly in the decentralized setup. Finally we can say that centralize orchestration perform better then decentralize orchestration at low load whereas decentralize orchestration scale better at high loads.

F) Tree based web service composition

Ching-Seh Wu [7] proposed tree based heuristic search algorithm for web service composition. Tree will be created based on task sequence & service component. Service includes service component, input, output, Max QoS, Min QoS. Every path in a tree from the root node to leaf node represents complete composition solution. Then filter the tree or prune the tree to remove branches that violate user's threshold as well as remove illegal compositions. Filtering will increase efficiency & decrease response time. Then apply best-first-search algorithm so it will explore nodes & evaluate them according to distance.

G) Constraint oriented approach

Aggarwal [10] proposed approach for constraint representation, optimization & cost estimation for constraint analysis & best possible selection of web services. They proposed one framework METEOR-S (Managing End-To-End Operations). METEOR-S back end allows the producer to bind service based on given theoretical process, requirement & constraints. METEOR-S is for semantic web services. It explores three different kinds of semantics i.e. 1) Data semantics, 2) functional semantics, 3) QoS semantics. Data semantics is used for communication with each other so they can understand the semantics of each other data. Functional semantics are combination of data semantics & operational functionality. Quality of web service includes specification of cost, time, reliability, availability. Ontology can be used to represent and explain the semantics of this parameter.

H) SLA based web service composition

Yan & Lin [12] proposed SLA (Service Level Agreement) based service composition technique. The provision of a service composition implies consumption of a set of service that is dynamically purchased from other service providers.This buy & offer relationship is commonly governed by an agreement is called SLA. For agreement they are considering functional & non-functional requirements. Functional aspects of a SLA are about what interaction need to be carried out to offer the service. Non-functional aspect describes a set of quality of service constraint based on how well the service should be offered. In this technique they are defining two approaches. One approach is to have a single SLA agreed by all the parties involved, specifying all the aspects of the service composition including the QoS requirement. But such an agreement that is too complex to be managed. Another approach is having a multiple SLA. In this approach the non-functional aspects of

each SLA define the particular QoS constraint on the provision of single service. And collectively fulfill the end-to-end Qos requirement of the whole service composition. Agent expertise enables independent negotiation as a means of establishes service contracts. In this generally XML based WSDL language is used for web service specification. UDDI is used for locate & compose web services.

I) Template based web service composition

Wei-Tek Tsai [6] proposed template based service composition. BPEL & OWL-S are commonly used language for web service composition. Domain ontology is used to organize service interface & service implementation. Relationship between service interface & service implementation is one-to-many. Service interface can be specified by 2 ways i.e. by programming, using XML file. He also proposed testing at each phase using unit testing, integration testing. Finally it manages the cloud service at the time of service composition.

4. Challenges

1) Exception handling mechanism for the distributed execution of the composite web service as well as information exchange mechanisms among service agents [2] [11].
2) Need to deal with changing consumers' requirements. For example, owing to its changing job requirements, a consumer may request more storage capacity in addition to its current requests contracted to a broker [1][4].
3) Build a tree using Unified Modelling Language (UML) diagrams, such as an activity diagram which improves the web service composition process by offering alternative task sequences [7].

TECHNIQUE USED	IMPROVED FACTOR
DPAWSC	Cost effective, Better scalability, High quality of Service
Context oriented WSC	Improve security
FSCNP & SCT	High efficiency
Acquaintance N/W & CNP	High efficiency
Constraint oriented WSC	Cost effective
WSC in Centralized orchestration	Improve performance at low load
WSC in Decentralize orchestration	Scale better at high load
Template based WSC	Good management
Tree based WSC	Increase efficiency & decrease response time
SLA Based	Provide good Quality of Service

Table 1. Comparative Analysis

5. Conclusion

Main idea here is to give an overview of current progress in automatic web services composition. This paper presents a different approach for web service composition. It also includes language used for web service composition. When dynamic and semantic service composition is exploited together by overcoming the issues with its own, better composition results are obtained. For cloud service composition that aims to overcome the issues caused by the open and flexible nature of Cloud services; by incorporating some trusted third-party entities to govern and optimize the service composition process.

6. References

[1] K. M. Sim "Agent-based Cloud Computing" IEEE Transactions On Services Computing, Issue 99, 2011.
[2] Hongxia Tong, Jian Cao, Shensheng Zhang, and Minglu Li "A Distributed Algorithm for Web Service Composition Based on Service Agent Model" IEEE Transactions On Parallel And Distributed Systems, Vol. 22, No. 12, pp-2008-2021, December 2011.
[3] Domenico Talia "Cloud Computing and Software Agents: Towards Cloud Intelligent Services", in proc. Of CEUR-WS, Vol.741, pp-2-6, 2011.
[4] J. Octavio Gutierrez-Garcia and Kwang-Mong Sim" Self-Organizing Agents for Service Composition in Cloud Computing" 2nd IEEE International Conference on Cloud Computing Technology and Science,pp-59-66,2010.
[5] A.Vishnu Vardhan M.S.Saleem Basha P.Dhavachelvan "An Overview of Web Services Composition Approaches" international Journal of Computer Applications (0975 – 8887) Volume 29– No.8, September 2011.
[6] Wei-Tek Tsai, Peide Zhong1, Janaka Balasooriya, Yinong Chen1, Xiaoying Bai, Jay Elston"An Approach for Service Composition and Testing for Cloud Computing "2011 Tenth International Symposium on Autonomous Decentralized Systems.
[7] Ching-Seh Wu and Ibrahim Khoury "Tree Based Search Algorithm For Web Service Composition In SAAS", 2012 Ninth International Conference on Information Technology- New Generations.
[8] J. Rao and X. Su, "A Survey of Automated Web Service Composition Methods" Proc. First Int'l Workshop Semantic Web Services and Web Process Composition (SWSWPC), pp. 43-54, 2004.
[9] G. B. Chafle, S. Chandra, V. Mann, and M. G. Nanda, "Decentralized orchestration of composite web services" in Proc. of the 13th international World Wide Web Conference on Alternate Track

Papers & Posters, ACM, New York, NY, 2004, pp. 134-143.

[10] R. Aggarwal et al., "Constraint Driven Web Service Composition in METEOR-S" Proc. IEEE Int'l Conf. Services Computing (SCC), pp. 23-30, 2004.

[11] Z.Maamar, S.K.Mostefaoui, and H.Yahyaoui,"Toward an Agent-Based and Context-Oriented Approach for Web Services Composition" IEEE Trans. Knowledge and Data Eng., vol. 17, no. 5, pp. 686-697, May 2005.

[12] J.Yan, J.Zhang, J.Lin, M.B.Chhetri, S. K.Goh, and R.Kowalczyk, "Towards Autonomous Service Level Agreement Negotiation for Adaptive Service Composition" 10th International Conference on Computer Supported Cooperative Work in Design, Nanjing, China, May 2006, 1-6.

[13] E.Pejman, Y.Rastegari, P.Majlesi Esfahani, A.Salajegheh "Web Service Composition Methods:A Survey" proceeding of the international Multi Conference of engineers and computer scientists 2012 Vol I,IMECS 2012,March 14-16,2012,HongKong.