ISSN : 0976-8580

Vol. 5, Issue 1, January - June 2015

Journal of Engineering and Technology

Online full text @ www.onlinejet.net

Wolters Kluwer Health | Medknow

Multi-Agent Based Cloud Service Composition using Contract Net Protocol for Information Retrieval Purpose

Yu Mon Zaw

Faculty of Information and Communication Technology, University of Technology, Yatanarpon Cyber City, Pyin Oo Lwin, Myanmar

Access this article online

Website: www.onlinejet.net

DOI: 10.4103/0976-8580.149476

Quick Response Code:

ABSTRACT

Information retrieval (IR) systems are important key players for every Internet user and there are many Information Retrieving frameworks and algorithms, which are currently using or under development. As the technology trend is always changing and current industrial and business world is willing to emphasize on not only providing knowledge but also supporting service, cloud computing and Web Services (WSs) become popular. Cloud computing is internet based system development in which large scalable computing resources are provided "as a service" over the internet to users and has attracted more and more attention from industry and research community. WS plays important role in service oriented computing in World-Wide-Web and Cloud environment. Developing a Cloud Wide IRSystem using WSs can fill one blank of Cloud Environment. In this case, retrieving desired specific information from WSs on Cloud Environment cannot be completed by single WS. Hence, there should be a way to build an IRSystem based on a set of related WSs in order to fulfill users' requests and provide desired services. To compose WSs, Multi-Agent System (MAS) can give great help. Agent systems are self-contained software programs embodying domain knowledge and having ability to act as a specific degree of independence to carry out actions needed to accomplish desired goals. Therefore, this paper mainly focuses on building a WSs Based IR MAS framework and Agent Processing Algorithm, which will be running on a Private Cloud Environment. We propose Contract Net Protocol for IR Purpose in order to perform WSs Composition. The implementation and testing of this system utilizes the real datasets of clinics in the Yangon area.

Key words: Cloud computing, contract net protocol for information retrieval purpose, multi-agent system, service oriented computing, web service composition

I. INTRODUCTION

Cloud computing systems provide large-scale infrastructures for high-performance computing that are "elastic" since they can adapt to user and application needs. [1,2]. Clouds are used through a service-oriented interface that implements the "as-a-service" paradigm to offer cloud services on demand. At the same time, multi-agent systems (MAS) represent another distributed computing paradigm based on multiple interacting agents that are capable of intelligent behavior. MASs are often used to solve problems by using a decentralized approach where several agents contribute to the solution by cooperating one another. One key feature of software agents is the intelligence that can be embodied into them according to some collective artificial intelligence approach that needs cooperation among several agents that can run on a parallel or distributed computer to achieve the needed high performance for solving large complex problems keeping execution time low.

When clouds and service oriented computing are popular, Web Service (WS) technology becomes essential, and it supports "as-a-service" nature for cloud to fulfill users' desires. WSs technology is based on the interoperation of many different software applications running on a variety of geographically dispersed systems in a complex, multi-domain environment via the Internet [3]. The definition of W3C (working Group) states: *"A* WS *is a software application identified by a URI, whose interfaces and bindings are capable of being defined, described and discovered as XML artefacts. A* WS *supports direct interactions with other software agents using XML based messages exchanged via internet-based protocols."* Nonetheless, WSs are closely related to the agent programming paradigm. The definition of the WSs architecture states: *"A* WS *is viewed as an abstract notion that must be implemented by a concrete agent. The agent is a concrete entity (a piece of software) that sends and receives messages, while the service is the set of functionality that is provided."* Typical agent architectures have many of the same features as WSs. Agent architectures provide yellow-and white-pages directories, where agents advertize their distinct functionalities and other agents search to locate the agents so they can request those functionalities.

There are some advantages of Agents Computing over WS technology. A WS knows only about itself, but not about its users, clients, or customers. Agents are often self-aware at a meta-level, and through learning and model building gain awareness of other agents and their capabilities as interactions among the agents occur. Without such awareness, a WS could not capitalize on new capabilities in its environment or customize its service to the client, such as by providing improved services to repeat customers. Agents are naturally communicative, whereas WSs are passive until made invocation. A WS, as currently defined and used, is not autonomous. Autonomy is a characteristic of agents, and it is also a characteristic of many envisioned Internet-based applications. Among agents, autonomy generally refers to social autonomy, where an agent is aware of its colleagues and is sociable, but nevertheless exercises its independence in certain circumstances. Agents

are cooperative, and by forming teams and coalitions can provide higher-level and more comprehensive services. Current standards for WSs do not provide for composing functionalities. According to the above facts, MAS can give a good hand for WSs to fill their blanks.

Therefore, for the cloud-wide information retrieval (IR) system based on WSs, it is sure that not a single WS can fulfill the user needs. To get the complete and desired information results, numerous related WSs should be cooperated. By compositing related WSs, we can get a new form of IR system for cloud based system. In this case, multi-agents system can give a great help to make data retrieving more intelligent, effective and time saving. Hence, not only to make composition of cloud services, but also to serve as an intelligent IR system, we can make a combination of Cloud Computing, Agent technology and WSs.

2. APPLYING CONTRACT NET PROTOCOL FOR INFORMATION RETRIEVAL PURPOSE

The contract net protocol (CNP) [4] has been developed to specify problem-solving communication and control for nodes in a distributed problem solver. Task distribution is affected by a negotiation process, a discussion carried on between nodes with tasks to be executed and nodes that may be able to execute those tasks.

Contract net protocol is mainly used in Agent-based Systems that intend to apply in commercial and business area. CNP can also be applied in WS Composition, and Sub-contracting is also needed for commercial service composition. In our system, we use CNP to train Agents for service composition not for business but for IR purpose. Hence, sub-contracting is not needed. But Data composition is essential. It can be called Multi-Contracting because more than one contractor will be participated in service composition and they perform their data retrieving duties respectively. We named it as IR-CNP (CNP for IR Purpose).

Like CNP, IR-CNP is a network of loosely coupled asynchronous nodes. Each node contains a number of distinct knowledge sources. However, unlike CNP, nodes (agents) of IR-CNP are not fully interconnected. The data retrievers (i.e., contractors) live independently and have no interconnection between them. They only have to contact with the main task distributor (i.e., manager) to help it in data retrieving purpose. The main distributor sends a call-for-proposals to resolve the user's request to n data retrievers. In turn, these may reply with a proposal wrapped with retrieved datasets if they accomplish the user's request successfully otherwise they may refuse. The distributor evaluates the received proposals and sends accept-proposal messages to all the selected data retrievers to make contracts.

Task distribution and service composition are performed in six stages:

1. Recognition: An agent recognizes it has a problem that it wants help with. The agent has a goal, and either realizes it cannot achieve the goal in isolation (does not have the capability to fulfill the goal).
2. Announcement: The agent with the task sends out an announcement of the task that includes a specification of the task to be achieved. The specification must encode a description of the task itself, any constraints, and meta-task information.
3. Selection: The agent that takes the role of information searcher selects a collection of Agents who have the ability to solve the problem.
4. Distribution: The main distributor agent sends out the task to the selection of Agents that inside the contract net to make a contract.
5. Retrieving: Agents that receive the task decide themselves whether they can solve the task or not by own knowledge. If it is possible, then they do data retrieving according to the task they received from main distributor agent. They then send back the result respectively.
6. Composition: Main distributor Agent must perform data combining using own knowledge. It composites all the searching results received from other agents and produces the final result dataset.

In the propose system, Cloud nodes will play as distributed nodes. WSs orchestrated Agents (WSAGs) have to handle these distributed nodes. The middle agent, the Information Agent, will take the role of main distributor of the user request (problem) and will do combine (compositing) returned datasets of WSAGs. The very first initiator agent will be Gateway Agent. All Agents in the system will play the initiators and participants alternatively.

3. RELATED WORK

Chang *et al*. presented an Ontology based Agent Generation for IR on Cloud Environment [5]. While user submitting a flat-text based request for retrieving information on a based on predefined ontology and reasoning rule, and then be translated to a Mobile Information Retrieving Agent Description File (MIRADF) that is formatted in a proposed Mobile Agent Description Language (MADF). A generating agent, named MIRA-GA, is also implemented to generate an MIRA in accordance with MIRADF. In short, this research paper mainly focus on building IR System for specific field (music field in this paper) using related ontology to generate mobile agent moving around cloud nodes.

Vadivelou *et al*.presented an agent and ontology-based approach that supports the semi-automatic composition of WSs [6]. This paper provided the way to select an optimal composition of services and it also propose a framework for Semi-Automatic WSs Composition.

Energy-Saving Information MAS with WSs for Cloud Computing was given by Yang *et al*. from St. John's University in Taiwan [7]. It employs the concept of SQL IC to construct the operational interface of cloud database as a data warehouse. It presented the three-stage intelligent decision processing strategy with four agents: Interface agent, Data Mining agent, Reasoning agent and Web-Service-Based Information Agent System (WIAS).

Vishal Jain proposed the IR practical model through the MAS with data mining in a cloud computing environment [8].

He recommended that users should ensure that the request made to the IaaS is within the scope of integrated data warehouse and is clear and simple. In that research model/ architecture, the use of cloud computing allows the users to retrieve meaningful information from virtually integrated data warehouse that reduces the costs of infrastructure and storage. In short, this research paper mainly focus on building IR System that retrieves information and knowledge from data warehouse by means of data mining concepts supplied MAS.

Our proposed system applies WSs in order to retrieve information instead of mobile agents. By the use of WSs, it does not need to know each Machine's database detailed schema and does not need to handle the whole database so that it can solve database security issue. By the help of agents, desired information can be searched in a set of WSs instead of single WS so that searching will be more effective at one sitting. Moreover, decision making on which web methods to be called can also be solved by agents' knowledge so that system will be more intelligent.

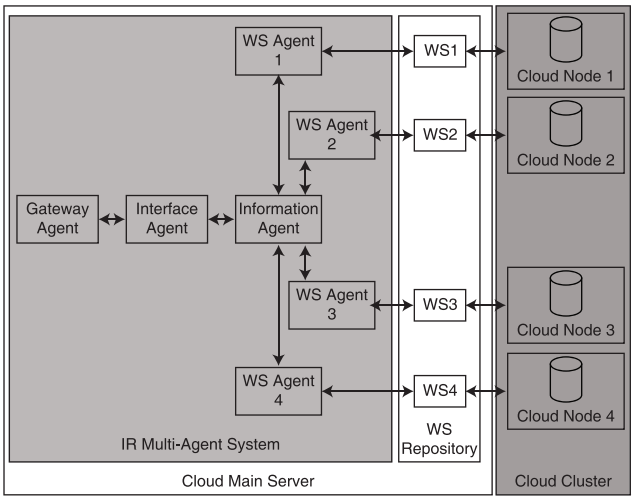

Figure 1: Web services based medical information retrieval multi-agent system architecture

4. PROPOSED SYSTEM FRAMEWORK

We propose a framework for WSs Based IR MAS for Cloud Computing Environment. For Service composition and communication, agents in the system adopt IR-CNP protocol. The proposed system framework is intended to apply in Medical field. We have already assumed that a private Medical Cloud Environment was been founded. In that environment, a number of hospitals, clinics and health care services are hosted and are providing WSs. Each WS of a specific hospital offers specialists (doctors) information worked at that hospital by numerous Web Methods. By using our Medical IR MAS, users (patients) can easily search the desired information by day (Monday, Tuesday,…), by time (1 pm-4pm,…), by doctor's name (Prof: Dr. Nay Win,…), by specific clinic (Asia Royal, SSC,..) and by disease type (Liver, Lung, OG,…) [Figure 1].

There are six main components:

1. Gateway agent — Gateway Agent is the connection point between proposed IR MAS and JSP Servlet; in other words, it is the gateway of JADE Agent Platform to the Cloud environment.
2. Interface agent — Interface Agent takes the duty of accepting queries from end users to search required information from distributed cloud nodes and showing back the queries result. The behaviors of Interface Agent are to receive queries from end users, prepare the queries into a format match for Information Agent's working style, pass the well formatted data to Information Agent and to show back the queries results.
3. Information agent — Information Agent performs three main service composition functions: Service discovery, service selection, and service combination. It searches available WSAGs, makes selection WSAGs, distributes user input query to a set of WSAGs depending on conditions and then combines the returned result of WSAGs and sends the final result to Interface Agent.
4. Web Service Agent — There are a set of WSAGs which invoke specific WS associated with them. One WSAGs orchestrates with one WS. In that case, one WS may contain more than one web methods. The duty of WSAGs is to make the best choice in selecting the web method according to the data parameters sent by Information Agent. They will send back a message to Information Agent whether they get the desired data or not.
5. Web Service repository — WS Repository contains a number of WSs and WS Description files (WSDLs) published by various web applications hosted on the cloud nodes. In our system, to make comfortable for WSAGs in sending message and carrying data, we generate each WSDL into a set of client classes and so WSAGs can handle the retrieved results in object and can carry inside the messages.
6. Cloud nodes — There are four cloud nodes in the same cluster. Several web application systems from Cloud Nodes support services for Information Searching and Retrieving purpose. One Node represents one hospital. Each node contains a web application (dynamic web projects) for each hospital that desires to coordinate with our Medical IR MAS. They support WS with many web methods. They possess databases with different schemas. Each web method contains SQL statement to access the database according to the received parameter values. We use MySQL database for data storage.

5. PROPOSED ALGORITHM FOR INFORMATION RETRIEVAL MULTI-AGENT SYSTEM

- Request is defined as Req.
- Refined Request is defined as RefineReq.
- Final Result is defined as FinResult.
- The Returned Result from each Web Service is defined as ResWS.
- ResWS1, ResWS2,…, ResWSn ∈ ResWS.
- WS is the set of Web Services published in WSDL Repository.
- WS1, WS2,…, WSn ∈ WS.
- AG is the set of all agents in the System.
- InterfaceAG, InfoAG,WSAG1,WSAG2,…,WSAGn ∈ AG
- Interface Agent is defined as IfAG.
- Information Agent is defined as InfoAG.
- Web Service Agent is defined as WSAG.

Begin

Fin Result←NULL.
Selected WSAG←NULL.
User sends Req to IfAG.
Refine Req←Req refined by IfAG.
IfAG sends Refine Req to Info AG.
Selected WSAG←Info AG determines which WSAGs to be called.
While not receiving the Fin Result from Info AG
InfoAG sends RefineReq to WSAG1, WSAG2, …, WSAGn ∈ SelectedWSAG.
For All WSAG1, WSAG2, …,WSAGi…, WSAGn.

WSAGi determine which web method of WSi to invoke according to the received RefineReq parameters.

```
        If WSAGi can solve RefineReq
                It will return ResWSi to InfoAG.
        Else
                Return NULL.
        End If
        FinResult←FinResult+ResWSi.
    End For
End While
End
```

6. SELECTING THE MOST APPROPRIATE WEB METHOD

Selecting the most suitable web method is the special duty of WSAGs. There will be more than one web methods are resided in one WSDL with various parameter types and values. For example, WSDL of Hospital1 possesses searchByDoctorName (String DoctorName) method. The return type is Class Hospital1DoctorSchedule Array. Some part of Hospital1 WSDL related with this web method is shown as below.

```
<element name="searchByDoctorName">
    <complexType>
    <sequence>
    <element name="docName" type="xsd:string"/>
    </sequence>
    </complexType>
    </element>
    <element name="searchByDoctorNameResponse">
    <complexType>
    <sequence>
    <element maxOccurs="unbounded" name=
    "searchByDoctorNameReturn" type="tns1:Hospital1Do
    ctorSchedule"/>
    </sequence>
    </complexType>
    </element>
```

To accomplish it, every WSAG requires three main components. The first is Input Data Format File (IDFF) to learn the input parameter contained in the message sent by InformationAgent. The second is Web Methods Description File (WMDF) which states what web methods possess how much parameter passing values, describes what that values type are (String, Integer, etc), describes the return type and states their purpose. The mappings between IDFF and WMDF are described in Mapping Parameter and Methods File (MPMF) in which what web methods should be called according to the received parameter values are stated.

In IDFF, there are five input field: DoctorName (String), Disease (String), Day (String), StartTime (Integer), EndTime (Integer). These five input type format and received message parameters have to perform miss or match value checking. If the received message contains only one parameter that is for DoctorName field, the related web method must be with one String parameter passing type. So, according to the predefined rules and knowledge in MPMF, WSAG decides the method "searchByDoctorName (String DoctorName) to be called.

7. IMPLEMENTATION AND TESTING

We implemented this proposed system based on J2EE and JADE platforms. We used apache tomcat server, axis for web development and MySQL database. The system has run on 5 machines and host OS are Ubuntu 12.04 LTS because of its private cloud building facilities. One machine is for main Metal-As-A-Service (MAAS) cloud server and others are represented as nodes in the same cluster.

Figure 2 represents the MAAS main cloud server API opened by browser. There are total of 4 nodes in this MAAS.

Each cloud node status, FQDN and MAC are described in Figure 3.

Figure 4 is the sample Search Page for Medical IR MAS. Users can find their desired doctors' schedule by name, by doctor's specialized field (e.g. Cardiologist), by day (e.g. Tuesday…), by Time (e.g. 11 am to 8 pm). Users can search using at least one criterion or all. The example searching result will be in the form of Table 1.

8. EVALUATING PERFORMANCE RESULT

In the proposed Medical IR MAS, there are four main agent types: Interface Agent, Information Agent and WSAGs. The number of WSAGs depends on the number of WSs they must handle in the system. So, the more WSs exist, the more WSAG

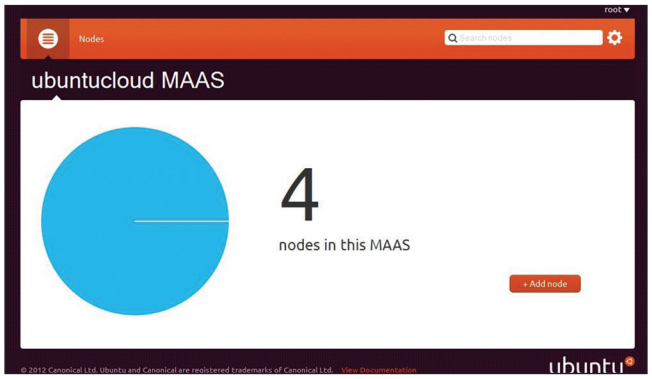

Figure 2: Cloud main page

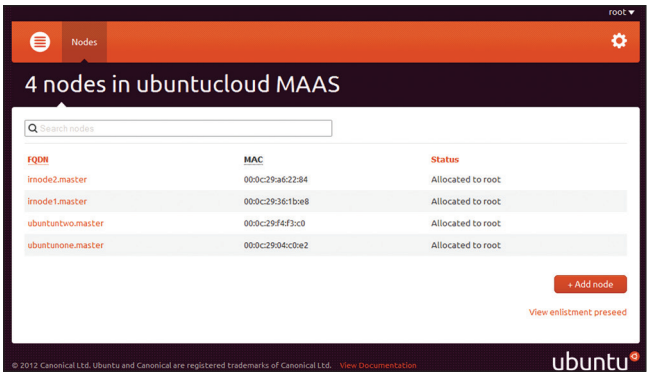

Figure 3: Cloud nodes status and descriptions

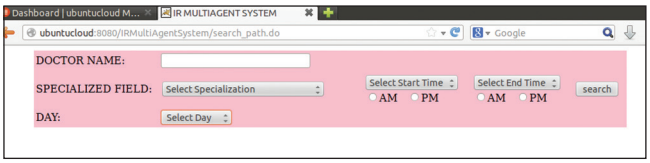

Figure 4: Information search page

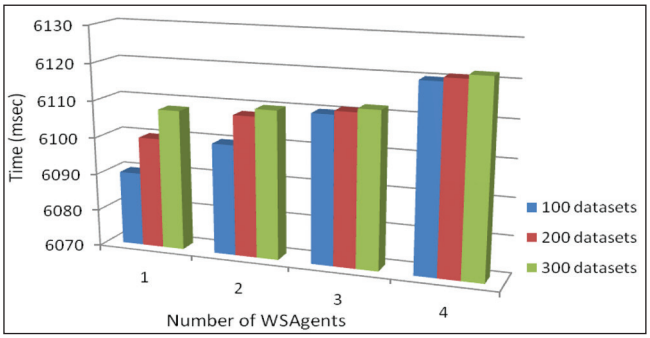

Figure 5: Comparison of average processing time with different datasets on increasing Web Service Agent

Table 1: Sample testing result

Name	Specialist field	Hospital	Day	From	To
Prof. Dr. Cho Lay Mar	Cardiologist	Bahosi	Tuesday	1 PM	3 PM
Dr. Ni Ni Win	Cardiologist	Asia Royal	Tuesday	5 PM	7 PM

the system require and the more complex the system will be. Therefore, more time will be consuming. But according to our testing, the processing time difference between increasing WSAG usage and increasing retrieving data size (datasets) is quite small and is acceptable [Figure 5].

9. CONCLUSION

Multi-Agent based Cloud Service Composition using IR-CNP is proposed. Efficiently composed cloud WSs using multi-agents features can give new form for cloud wide IR systems. The proposed system will become an intelligent way for searching or retrieving information from Cloud environment. By implementing the propose system, it can give a good hand for the public to get the desired specialists' schedule completely and perfectly at one sitting and can make the right choice with their current situations. Moreover, this framework can be applied in other domain area efficiently.

REFERENCES

1. D. Talia, "Cloud Computing and Software Agents," Towards Cloud Intelligent Services; Proceedings of CEUR -WS, vol.741, 2011, pp. 2-6.
2. T. Dillon, C. Wu, and E. Chang, "Cloud Computing Issues and Challenges," 24th IEEE International Conference on Advanced Information Networking and Applications, 2010.
3. F. Curbera, R. Khalaf, N. Mukhi, S. Tai, and S. Weerawarana, "The Next Step in Web Services," Communications ACM, Vol. 46, no. 10, 2003.
4. G. S. Reid, "The contract net protocol: High-level communication and control in a distributed problem solver," IEEE Transactions on Computers, Vol. C-29, no. 12, 1980. p. 1104-1113.
5. Y. S. Chang, C. T. Yang, and Y. C. Luo, "An ontology based agent generation for information retrieval on cloud environment," Journal of Universal Computer Science, Vol. 17, no. 8, 2011.
6. G. Vadivelou, E. Ilavarasan, and M. S. Yasmeen, "QoS based semi-automatic web service composition using multi-agents systems," International Journal of Computer Science Engineering and Technology, Vol. 1, no. 7, 2011, pp. 381-386.
7. S. Y. Yang, D. L. Lee, K. Y. Chen, and C. L. Hsu, "Energy-saving information multi-agent system with web services for cloud computing, Security-Enriched Urban Computing and Smart Grid Communications in Computer and Information Science, Vol. 223, 2011, pp 222-233.
8. V. Jain, "Information retrieval through multi-agent system with data mining in cloud computing," International Journal of Computer Technology and Application, Vol. 3 no. 1, 2012, pp. 62-66.

Authors' Biography

Dr. Yu Mon Zaw Yu Mon Zaw was born in Yangon, Myanmar. She got the Bachelor of Computer Science Degree (B.C.Sc) from University of Computer Studies (Yangon), Myanmar in 2006. She achieved the Master of Computer Science Degree (M.C.Sc) from the same University in 2009 especially studied and finished the thesis by Agent Based Searching model. in October, 2014, She got Ph.D (IT) degree from University of Technology, Yatanarpon Cyber City (UTYCC), Pyin Oo Lwin, Myanmar. Her field of interest is Cloud Computing and Multi-Agent. She works as a tutor at Computer University (Tha-Hton) since 2009.

E-mail: mayumonzaw@gmail.com