

Service Composition Techniques for a Context-Aware Smart Hospital Ward Monitoring System

Ashok Immanuel V, Preethi R

*Department of Computer Science,
Christ University, Bengaluru, India*

Abstract—Ubiquitous computing along with Internet of Things will be the most appreciable and needed a paradigm shift for the next generation healthcare programming. Patient care would involve data aggregation from numerous heterogeneous devices, sensors and actuators which will eventually be coordinated with or without much human intervention through effective service composition. Service composition poses numerous challenges. This paper presents a short survey on service composition techniques suitable for healthcare domain and implementation of services.

Keywords —Context-awareness, Service Composition.

I. INTRODUCTION

Services are modular software systems or applications which are loosely coupled and designed to participate in the device to device information interaction and interchange. They are Discovered, Orchestrated and coupled to form applications. With numerous heterogeneous devices in a hospital environment, it becomes imperative that device to device communication exists and it is indispensable. The healthcare domain is set for a major revolution with the research on Internet of Things (IoT) beaming to greater achievements. Hence, a thorough understanding of service composition is of paramount importance when it comes to programming for the next generation of healthcare applications that are IoT-enabled.

Therefore, an urge for such web service composition has become more prominent and compelling [1] [2]. Various methods have been proposed for service discovery and composition. Graph-Based approach proposes a model constructed using automata as a low-level construct [3] [4]. Based on the research works there are two categories of service composition. The syntactic composition is one in which the system adapts to a changing situation without any human intervention. Compositions based on semantics works on the principle of registering functional information and logical inferring rest on the pre and the post conditions of a service [5]. Workflow based approaches are well suited for syntactic composition while AI method is prescribed for semantic composition [6].

Several research initiatives have been attempted to integrate heterogeneous systems. Universal Description, Discovery and Integration (UDDI) is a platform-independent, XML-based system where web service can be registered for discovery and contains the URI as entry entity for consuming the service [7], Web Services Description Language (WSDL) describes the service. It is a document which contains the descriptions on message

passing formats and the operations [8], Simple Object Access Protocol (SOAP) [9] is a protocol for service discovery, description and composition. Business Process Execution Language for Web Service (BPEL4WS) [10] [11] and DAML-S [12] Service Model, are focused on service compositions where bindings between services are known upfront.

II. WEB SERVICE ARCHITECTURE

Service developers and providers of a certain domain publish the service description in terms of their operations, message passing formats and its physical location in directories. This directory being in the public space can be queried using specific languages for service discovery. Applications which wish to consume the service queries against the directory to find the availability of the service and the means to communicate with the service provider. WSDL is used in creating service description. SOAP protocol has been the most preferred for communication between the service provider and the service consumer. The service consumer software formulates its message in XML and sends it to the service provider. First service consumer queries the directory and gets the service description and gets the details of how to communicate with the service provider and then it sends a message to service provider formulated in XML based on the description. Figure 1 Web Service architecture.

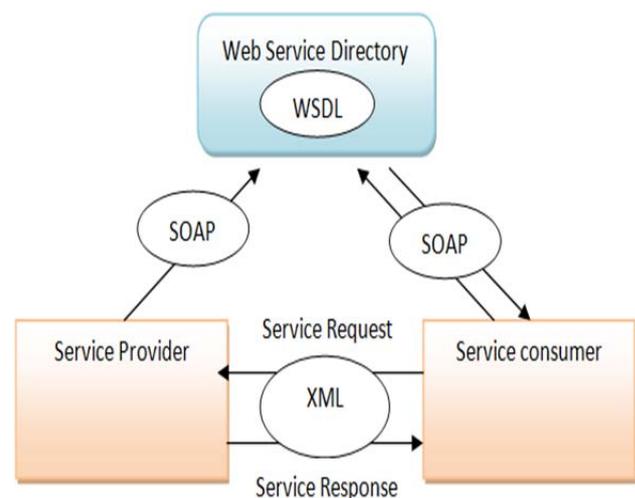


Figure 1 Web Service architecture

Lifecycle of Web services

There are 5 steps included in lifecycle of Web services

1. Advertisement: The service provider publishes its description and endpoints of the web service in the directory called service registry.
2. Discovery: The service consumer or the service requestor locates all the Web services and its description to check if it matches their functional requirements.
3. Selection: The service requestor selects the most suitable web service out of all the web services in the service registry based on application dependent metrics.
4. Composition: After selecting the suitable web services the service requestor integrates all the web services into a complex process.
5. Invocation: In this final step the web services or complex process is invoked and set for execution by providing all the inputs.

III. SEMANTIC WEB SERVICE

The semantic web is a way of having the computers understand more like human beings do. In most languages the syntax is how you say it whereas semantics is the meaning of what we say. In web service, the services are discovered by service consumer using keyword/syntax comparison between the registered service descriptions in the directory and the service request. With this new technology, it enables advanced reasoning functionality over the available data which is at a risk of disposal of application engineers. These semantic web services are developed using DAML-S, OWL-S technologies which help in describing the web service in detail with its capabilities, execution flows, its policies and other information.

The main difference between the normal web services and the semantic web service is that it no longer uses the key-value pair's comparison and uses logic based matchmaking. Three factors mainly affect the service discovery: service description, service consumer description of requirements and service composition algorithm.

Service Provider is an entity or software that provides some services to service consumer based on their request. Service provider tries to promote their services by publishing their description or advertise its services to potential service users in order to maximize its profit. The service provider publishes the service advertisements in service registry which is a public or private directory that is accessible and searchable by third-parties. The service consumer/requestors query against this registry for searching the services. The service requestors might be a human user or enterprise systems, mobile agents or any other systems depending on the application domain.

The previous method of key-word based descriptions did not describe the internal service structure which may be of use during service discovery for service requestors. This new semantic web discovery technology automates the whole service lifecycle. This mainly plays an important

role in service discovery as it affects the architecture, algorithms, tools, and effectiveness of service retrieval and every other aspect of the discovery process. The Semantic Web service follows the same reference architecture used by the other web services but is implemented in different order to enhance the functionality. It also includes several new components such as Service Annotation Ontologies and Domain Ontologies.

A. Service Advertisement

Service Advertisements are the description of the web services stored in the service registry. They help in promoting the services and also helps the service requestor in choosing the best web service according to their request. These are described according to specific Service Annotation Ontologies. The annotation terms used in this service advertisement are already described in vocabularies defined by Domain Ontologies. Hence, these service advertisements are standard documents that comply with specific models and refer to description terms from external terminologies.

B. Service Annotation Ontologies

Service annotation ontologies are service description models which describe the semantics of service functionality, process structure and invocation.

C. Service Request

It is a request sent by service requestor to service provider requesting for a web service. The service request may vary from text to documents according to system implementation and service annotation ontologies. A request should contain relevant information as mentioned by service annotation ontology. After appropriate mediation, the request is transformed to a document which is considered as service advertisement. By using the same domain ontologies by both the service requestors and providers simplifies the matching process.

D. Service Registry

Unlike the traditional web service registries, the semantic web service registry contains references to semantic information annotating the advertised services.

E. Matching Algorithm

These algorithms are designed so as to help in matching the semantics of provided service descriptions and requests and are in general more complex and intelligent than the syntax-based method.

The main difference between the traditional architecture and the semantic web service architecture is the implementation and the integration of the semantic matching engine with the service registry.

Semantic web enables greater access not only to content but also to services on the web. Because of this technology users and software agents are able to discover, invoke, compose and monitor web resources and also can do it with a high degree of automation if desired. OWL-S is an ontology of services that makes these functionalities possible.

IV. IMPLEMENTATION

Consider the following use case in an Intensive Therapy Unit of post cardiac surgery recovery ward. The Pleural tube is a drain which is connected to the chest at the time of surgery. The tube drains the excess fluid collection or the blood which results after the surgery. If the collection of the fluid exceeds 150ml in the first hour after surgery then it calls for an immediate intervention. Based on the various

literature available an implementation of the services was attempted. Figure 2. Illustrates the code written for a drain service. Nusoap is used to create and implement the client services and server services. Figure 3. Illustrates the code for the server. The WSDL file which is used in service composition is also developed using Nusoap. Figure 4 Illustrates the WSDL file needed for discovering the service and use in the composition.

```
require_once "nusoap-0.9.5/lib/nusoap.php";
$lvl=$_REQUEST['id'];
$client = new nusoap_client("server.wsdl", true);
$result = $client->call("drain.putStatus", array("level" => $lvl));
// <Code handler for faulty client>
echo "<h2>Request</h2>";
echo "<pre>" . htmlspecialchars($client->request, ENT_QUOTES) . "</pre>";
echo "<h2>Response</h2>";
echo "<pre>" . htmlspecialchars($client->response, ENT_QUOTES) . "</pre>";
```

Figure 2 PHP code for Drain service implementation

```
$server = new soap_server();
$server->configureWSDL("drainservice", "localhost/Nusoap/drainservice");
$server->register("drain.putStatus",
    array("level" => "xsd:string"),
    array("return" => "xsd:string"),
    "localhost/Nusoap/hospital/drainservice",
    "localhost/Nusoap/hospital/drainservice/#putStatus",
    "rpc",
    "encoded",
    "Get drain by level");
@$server->service($_HTTP_RAW_POST_DATA);
```

Figure 3 PHP code for server implementation

```
1 <definitions xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org
2 <types>
8 <message name="drain.putStatusRequest">
11 <message name="drain.putStatusResponse">
14 <portType name="drainservicePortType">
21 <binding name="drainserviceBinding" type="tns:drainservicePortType">
22 <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
23 <operation name="drain.putStatus">
24 <soap:operation soapAction="localhost/Nusoap/hospital/drainservice/#putStatus" style="rpc"/>
25 <input>
26 <soap:body use="encoded" namespace="localhost/Nusoap/hospital/drainservice" encodingStyle="http://sc
27 </input>
28 <output>
29 <soap:body use="encoded" namespace="localhost/Nusoap/hospital/drainservice" encodingStyle="http://sc
30 </output>
31 </operation>
32 </binding>
33 <service name="drainservice">
34 <port name="drainservicePort" binding="tns:drainserviceBinding">
35 <soap:address location="http://localhost/Nusoap/hospital/server.php"/>
36 </port>
37 </service>
38 </definitions>
39
```

Figure 4. Extracts from the WSDL file

V. CONCLUSION

Service composition has taken a center stage for the next generation programming paradigm. This paper presents a study on the existing techniques in service discovery and composition. Based on the learnings, an implementation of the services required for a scenario which would exist in an Intensive Therapy Unit of a post cardiac surgery recovery wards has been presented. As a future work, it has been planned to develop a complete prototype for a smart ward encompassing all the possible context-aware situations in a hospital environment.

REFERENCES

- [1] P Dourish, "What we talk about when we talk about context," *Personal and Ubiquitous Computing*, vol. 8, no. 1, pp. 19-30, 2004.
- [2] Sato K, "Context-sensitive interactive systems design: A framework for the representation of contexts," in *10th International Conference on Human-Computer Interaction*, vol. 3, 2003, pp. 1323-1327.
- [3] C Orwat et al., "Adopting Pervasive Computing for Routine Use in Healthcare," *Pervasive Computing*, vol. 9, no. 2, pp. 64-71, 2010.
- [4] Seyyed,Vahid Hashemian and Mavaddat Farhad, "A graph-based approach to web services composition." Applications and the Internet, 2005. Proceedings. The 2005 Symposium on. IEEE, 2005.," in *Symposium on Applications and the Internet*, 2005, pp. 183-189.
- [5] S Mitchell, MD Spiteri, J Bates, and J Coulouris, "Context-Aware Multimedia Computing in the Intelligent Hospital," in *9th workshop on ACM SIGOPS European workshop: beyond the PC: new challenges for the operating system*, 2000, pp. 13-18.
- [6] J Favela et al., "Estimating Hospital Work Activities in Context-Aware Healthcare Applications," in *Pervasive Health Conference and Workshops*, 2006, pp. 1-10.
- [7] JE Bardram, "Applications of context-aware computing in hospital work: examples and design principles," in *ACM symposium on Applied computing (SAC '04)*, 2004, pp. 1574-1579.
- [8] M.B Skov and R.Th Høegh, "Supporting information access in a hospital ward by a context-aware mobile electronic patient record," *Personal and Ubiquitous Computing*, vol. 10, no. 4, pp. 205-214, 2006.
- [9] N Baker, "Context-Aware Systems and Implications for Future Internet," *Towards the Future Internet*, pp. 335-344, 2009.
- [10] R Gay, H Mantel, and B Sprick, "Service automata," in *th International Conference on Formal Aspects of Security and Trust (FAST'11)*, 2011, pp. 148-163.
- [11] Wang Pengwei, Ding Zhijun, Jiang Changjun, and Zhou Mengchu, "Automated web service composition supporting conditional branch structures," *Enterprise IS*, vol. 8, pp. 121-146, 2014.
- [12] Z Zhichao, W Li, Z Wu, and W Tan, "Towards an Automata-Based Semantic Web Services Composition Method in Context-Aware Environment," in *9th International Conference on Services Computing (SCC)*, 2012, pp. 320-327.
- [13] Xinlin Zhang, Miao Huaikou, and Zeng Hongwei, "The syntactic and semantic model of web services composition based category," in *International Conference on Advanced Computer Theory and Engineering ICACTE'08*, 2008, pp. 444-449.