Use of Strategy Design Pattern in Service Oriented Architecture for Healthcare System

UMESH BANODHA

Department of Computer Applications, Samrat Ashok Technological Institute, Vidisha (M.P.), INDIA
ub.pub.2011@gmail.com

KANAK SAXENA

Department of Computer Applications, Samrat Ashok Technological Institute, Vidisha (M.P.), INDIA
ks.pub.2011@gmail.com

Abstract

Healthcare system is one of the most important and uncertain era for the people or society. The patients need to improve very fast with fewer complexities. Now a day, technologies play great role to reduce the complexities in specific era. Service oriented architecture (SOA) provides the web-services in integrated and distributed environment. Service oriented architecture combines the design pattern to improve usability and recurrence solution of the problems. The Service oriented architecture with design pattern improve the solution (improve health of the patients) of patients problem at any instance of time in global environment. The Strategy design pattern strengthens the selection of suitable method among the number of methods as per the circumstances. The services of SOA are inexpensive, easy to access, reducing in operating cost and improve medical protocols.

Key Words - **choreography, Healthcare System, orchestration, Service oriented Architecture, Strategy design pattern.**

1. Introduction

"Health is wealth" – Healthcare system is very important, critical and very large space area for improvement. Last few decay; there are many technological advancement and research work done in this era. Although, there are a lots of works need to be done. The integration of resources of healthcare and their services required an appropriate and suitable framework. As healthcare system is service oriented system so a distributive and dynamically global supportive framework needed for integration of resource components of the healthcare system.

The framework architecture needed for sharing and communication the medical messages (web services) using SOA. There is specific order for communication between service requestor (clients or patients) and service providers (components or healthcare system components). Services are always available universal so client may use at any time without much efforts. The services are fast, easily available and used number of times without change. Thus, this framework is cost saving and instantaneous responsive. Even, much kind of services like prediction of diseases, severity of diseases, classification of diseases, criticalness of diseases etc. will be available as services on the basis of the sign and symptoms and laboratory report of the patients. These services are the product of the SOA.

The SOA is a framework architecture which provides standard methods for demand facilities from distributed modules and maintains the results. As the client request for facilities, the available modules provide the suitable services with the help of multiple message protocols and networking protocols need to deliver messages. The response of components may vary widely. The functions of SOA are divided into many layers. The translation and management layers remove the obstacles of a requestor to find required services.SOA support the clients and components may be written in separate languages. SOA provides easy accessible, ready-made, efficient, optimized, and modular, speedup and sharable components that may reduce software developer and infrastructure operating costs. SOA provide loose coupling between clients and components [1].

2. Service-Oriented Architecture

The service oriented architecture provides services as shown in the figure 1. A medical task is a combination of multiple services. Services are repeatable task in a healthcare system. The SOA is language independent services in any of the distributed application. SOA is a message passing architecture between components of the architecture and client. When the client needs the components or services interact with the components. The component used for receive / response message to the same client. The request from client and reply from components are in metadata standard format.
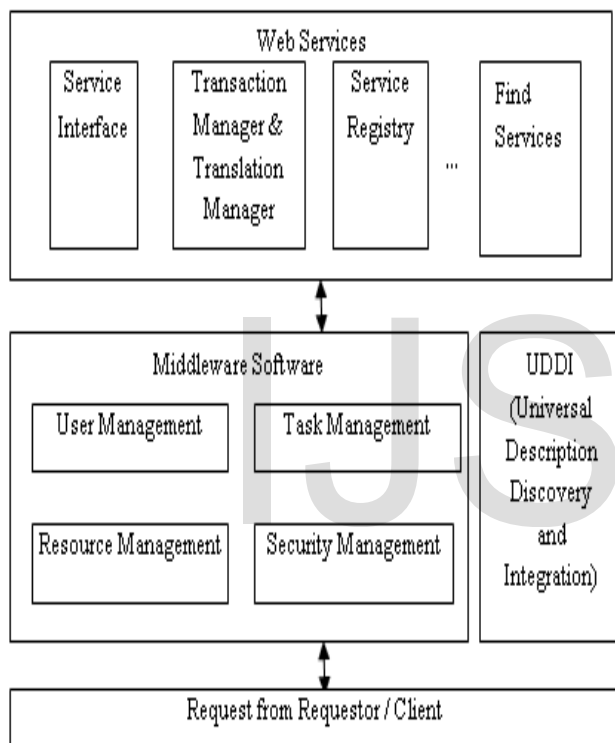

Figure 1 Service oriented Architecture

The Service provider and service consumer are not able to communicate directly with each other. The transaction manager and translator are available between service provider and consumer as middleware software. The transaction manager is responsible for transaction of sender and translator is responsible to understand the language of service requestor and components. The service interface provides the interface between service requestor and service provider. This middleware software prepares the available list; discover the list of services and potential service consumer and provider.

The middleware software is the logic of communication where it resides in the form of service registry. As, the SOA is based on distributed

manner which provides general application as well as specific application like industry, business, private and public services.

Middleware manage the resources with user request and corresponding replies in context of required services or tasks. The Universal description discovery and integration (UDDI) protocol is used to discover the web service and broadcast the web services in the standard format of e-messages using extensible Markup Language documents (eXML). The architecture does not support any executable links of any API. Thus, SOA works in integrated process which is set of linked services [1].

The most popular extensible markup language document is need for message passing format with the help of simple object access protocol (SOAP) as shown in figure2. A web services description language (WSDL) is used to provide interface to services, how to integrate the services, how to bind services and types of services provided by the components and their nature. The web services security available at next level which provides various types of security like reliability, security of services, trustiness of service, transaction, management of the services. Medical processes execution language for web services is available at next level of relationship diagram of components in SOA (Figure 2). The (SCDL) service component definition language is used to specify the service of specific component that provide the services and service information of components which is not part of web services description language.
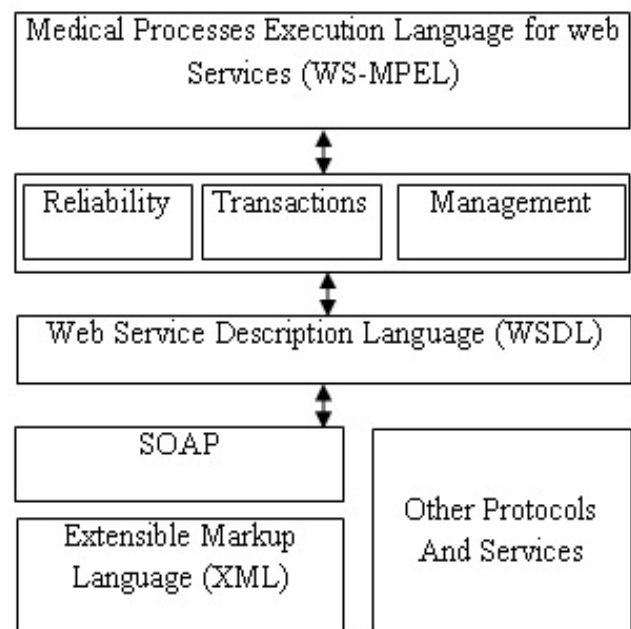

Figure 2 relationship of protocols in SOA

The contract term is used as a transaction in SOA. The term contract specifies a level of services which is available for requestor. The box labeled other protocols and services in figure2 indicates CORBA protocols and services, representational state transfer, RPC, DCOM, Data distribution services, windows communication foundations etc. The level of flexibility and support build SOA strong enough in design complex applications or problems.

The service component architecture is a technology oriented design specification. It uses the services of components which are inscribed in medical processes execution language and other languages like java, C#, C^{++}, FORTRAN, python, ruby and PHP etc. The SCA supports to components form for medical process. The integration of multiple web services to create to medical process should be arranged and managed. The orchestration and Choreography are two methods to combine web services [1].

Orchestration: The SOA's middleware software synchronizes centrally all the various web services (receiving / sending messages) from the orchestrator [1].

Choreography: This method is a collaborative effort. In this method, the middleware software of SOA does not have central coordination function for all web services. The basic logic of web services is passed out to the members (components of middleware software) whom are responsible to find and provide operation for execution. The member also responsible for other factors like structure of the messages, messages passing within the given time frame etc [1].
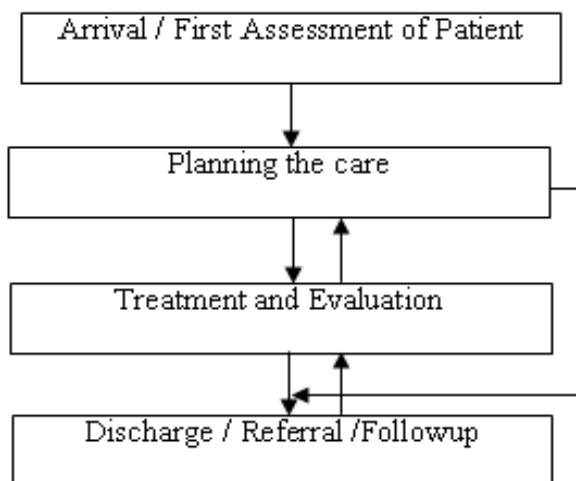
3. Healthcare System (HCS)

Figure 3 Healthcare System

The general healthcare system (HCS) is considered which consist of four steps at the abstract level. Each

step is part of healthcare system and each step is further divided into sub services of the healthcare system. The steps and their relationships are presented in figure 3. This healthcare system begins when patients come to reception or doctor and completes if the patient is discharged or stop to take the services from healthcare system. The second step of HCS is planning the care where HCS provides the caring of patient like treatment plan, inpatient medication, drug (medicine), add or remove drug, in patient medication list, and instruction for treatment etc. the third step is treatment and evaluation where the complete treatment and their evaluation is carried out. In figure 3 the step (ii) and step (iii) is bidirectional communication between HCS components. The bidirectional shows that the direction of planning the treatment decides the direction of treatment and the treatment evaluation provides strengthen to the planning the care. Similarly, bidirectional communication between step (iii) and (iv) The evaluation decides the discharge of patient or more follow up the plan and again referral decides the more treatment. Similarly if the patients are satisfied with the inpatient medication then patient will leave the medication unit, in such case no advancement are required [5, 6, 7, 8, 9, 10].

Orchestration: The middleware software of the SOA is the central part in orchestration. There is a central coordination system which receives the request from patients and appropriate medical services are provided to them. The orchestration use standard operations to sends and receives the medical services from different medical components. The orchestration individually establishes the communication with each medical component as shown in figure 4. The orchestrator keeps management of all individual medical components with their service along with patient requests.

Choreography: There is no centrally coordinate system to manage communication between the request sender and receiver. There is no centrally coordinate system to keep the record of medical services provided by medical components. The medical services of medical components aware with information for example when to process a message, what is the client, and need of component interaction etc.

The logic of this method is that medical process is passed on to the member of middleware software who is responsible to find the medical services from medical components and provide interaction with other medical service of same and different medical component as shown in figure 5. It shows that the medical service1 can communicate to the medical process 2 and so on.

The member of middleware software is responsible to find and provide operation for execution. The member also responsible for other factors like structure of the messages, messages passing within the given time frame etc.
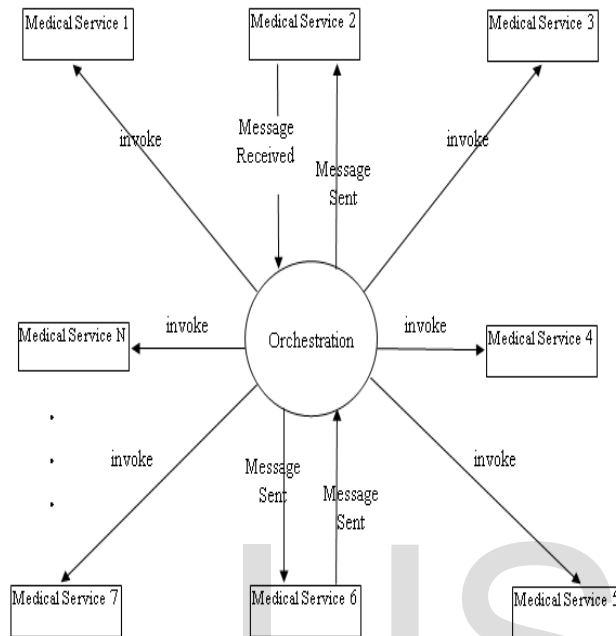

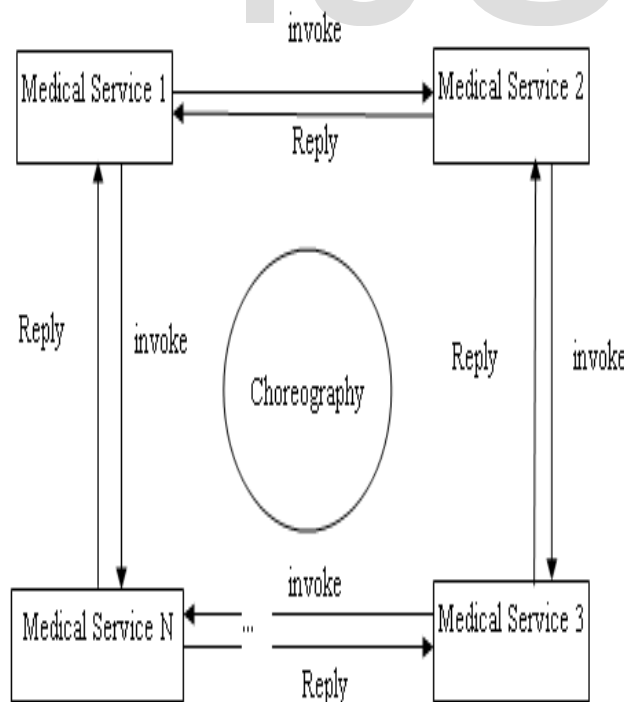Figure 4 an orchestrated Medical Process

Figure 5 a choreography medical process

4. Strategy Design Pattern

The strategy pattern has a number of methods/ algorithms encapsulated within classes. The client can select one out of the available methods. The strategy design pattern provides the option for selection of method or algorithms. It does not provide any kind of details of method like how the algorithm / method would work. Depending on the selection made by the user, the object of that particular class is instantiated, and the operation is carried out. The methods should be encapsulated within a common method implemented by different classes.

The strategy pattern can be utilized in cases where a client program needs a task to be performed, and the task can be done in many ways. The method to be used is selected by the user; it can be selected by other factors such as the efficiency and performance of computation if that particular method is implemented [4].
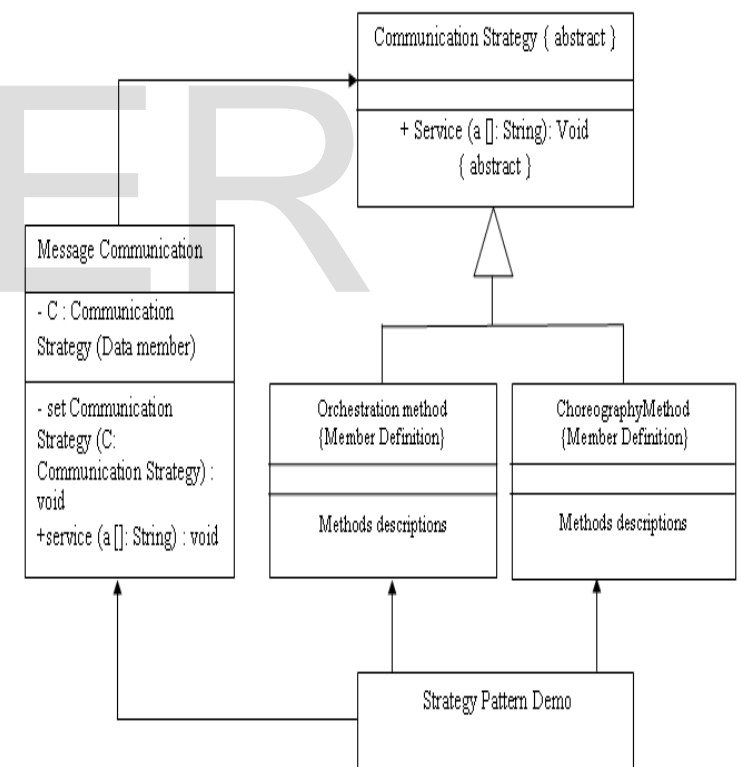
5. Strategy DP in SOA for HCS

Figure 6 UML diagram of Strategy pattern in SOA for HCS

The problem is to encapsulate various strategies of message communication (messages combine in web service) with in the classes. The selection made by user, the object of that particular class (orchestration and choreography) is instantiated and the operation is carried out.

In our example, the task to be performed is communication of web services and communication (messages) can be done by using the methods orchestration and choreography as illustrated in the UML diagram (figure 6).

The method is selected by the user or it can be selected by other factors such as the efficiency of computation, availability of components at specific time, easy way of communication at specific operation, way of communication (sequential or parallel), order of communication etc.

The following two cases/ conditions specify the selection of appropriate method.

The function of medical doctor is just like orchestration that keeps the management of medical services. He works centrally manage the communication between patients and medical components. The doctor has manages medical history of the patients and provide corresponding services of medical components (Laboratory tests, medical operations, medical drug etc.).

If the patient is well aware of the diseases due to medical history and symptoms then directly use the service of medical laboratory test. In such case patient can start the medication as per the knowledge of medical history. In this example one medical service is directly interact with the other medical service without interaction of the centrally coordinate system.

Although it's hard to predict which method is best but SOA specifies the advantages of orchestration over the choreography due to the following reasons.

The central coordination system has basic logic of communication if changes occur in the logic then changes done at one point. The integration of services into SOA is simple and easier compare to choreography because service does not know anything about the medical service. It is easy to place error handling mechanisms in orchestration. It is easy to manage and analyze the event that occurs in and outside the medical process and that associate to part of the service.

6. Conclusion

The paper describes the meta-model of healthcare system with the use of strategy pattern in service oriented architecture model. The paper helps the software developer to provide the appropriate selection of method on the priority of the patients. The concept of strategy design pattern provides the option for solution of the problem occurs frequently in real world applications. The outcomes of the proposed model is that provide the flexibility to the patient in service oriented architecture to select the way of communication between the service requestor and service provider with or without centrally coordinated system.

References

[1] Barrie sosinsky, "*cloud computing Bible*", Wiley india, isbn : 978-81-265-2980-3, 2013.

[2] Bass Clements, and Kazman, *Software Architecture in Practice* 2nd Ed, Addison-Wesley 2003.

[3] http://hercules.gcsu.edu/~adahanay/mmis6 298/softwarearchitecture/chapter02.ppt

[4] Mahesh P. Matha, "*object-oriented analysis and design using Uml*", PHI learning Pvt. Ltd., 2010

[5] U.Banodha , K.saxena, "*Impact of Pipe and Filter Style on Medical Process Re-engineering*", International Journal of Engineering Sciences, October 2011.

[6] U.Banodha , K.saxena,"*Usability of Software Architecture design pattern in Medical process reengineering model*",International Journal of Application or Innovation in Engineering & Management (IJAIEM), Volume 2, Issue 6, June 2013, ISSN 2319 – 4847.

[7] U.Banodha, K.Saxena, "*Comparison of Software Architecture Styles in Medical Process Re-engineering Model*", International Journal of Wisdom Based Computing, Vol. 2(1), April 2012.

[8] U.Banodha, K.Saxena, "*Dynamic Software Architecture for Medical Domain Using Pop Counts*",International Journal of Advanced Computer Science and Applications, Vol. 5, No.2, 2014

[9] U.Banodha, K.Saxena, "*An Artifact on Prediction of Software Architecture Style in Medical Domain*",International Journal of Computer Applications (0975 – 8887), Volume 98– No.18, July 2014.

[10] U.Banodha, K.Saxena, "*A Software Architecture Style for Medical process Re-engineering*", World Congress on Engineering and Computer Science, San Francisco, USA, October 2011, pp.135-140

IJSER