Identification of Medical Service Providers using Web Services with Mobile Device Extensions

Walisa Romsaiyud
Graduate School of Information Technology in Business, Siam University,
Bangkok, Thailand
walisa.r@siam.edu

Wichian Premchaiswadi
Graduate School of Information Technology in Business, Siam University,
Bangkok, Thailand
wichian@siam.edu

Abstract

Web Services are being used increasingly to integrate and access multiple service offerings. In addition, when integrated with mobile devices, they can be used to provide location based services.

The first challenge is to develop an application that requests web services from a UDDI registry center that dynamically invokes the most appropriate medical web services from among a series of services with the same functionality from private hospitals in Thailand.

Having accomplished that the second challenge is the development of a mobile application to find a suitable healthcare service from the UDDI registry to help patients in emergency situations who need immediate medical information. This paper presents a prototype application that illustrates such an application through a case study.

Key Words- Web Services, WSDL, Broker, UDDI, SOA, Mobile Application

1. Introduction

Since 2003, the government has supported a project to promote Thailand as the medical hub of Asia. Toward this end, most private hospitals in Thailand have met international standards and their medical services are ranked among the best in the world. Many Thai hospitals have been approved for Hospital Accreditation by organizations such as ISO 9002, ISO 9001:2000, ISO 14000 and ISO 18000 [1]. Thailand's private hospitals have modern laboratory equipment and all the latest medical technologies such as Gamma Knife Brain Surgery or CABG and Magnetic Resonance Imaging or MRI scans to serve their medical needs. In addition, many provide 24-hour emergency service with ambulances or helicopters to transport patients if necessary. Thailand is now a destination not only for vacationers, but also for those who want to have cost effective medical care. With world-class medical facilities and specialists, in addition to the warm hospitality of hospital staff, patients from all around the world come to Thailand's leading private hospitals to get medical treatment (see Table 1).

Table 1: The Number of Foreign Patients Receiving Medical Treatments in Private Hospitals in Thailand [1]

Rank	Nationality	No. of foreign patients		
		Year		
		2003	2004	2005
1	Japan	162,909	247,238	185,616
2	USA	85,292	118,771	132,239
3	South Asia	69,574	107,627	98,308
4	UK	74,856	95,941	108,156
5	Asian	36,708	93,516	74,178
6	Middle East	34,704	71,051	98,451
7	Taiwan/China	46,624	57,051	57,279
8	Germany	37,055	40,180	42,798
9	Australia	24,228	35,092	40,161
10	France	25,582	32,409	36,175
11	South Korea	19,588	31,303	26,571
12	Scandinavia	19,851	20,990	22,921
13	Canada	12,909	18,144	18,177
14	Others	323,652	133,782	302,834
	Total	973,532	1,103,095	1,249,984

Source: Office of Export Services, Department of Export Promotion.

According to the Office of Export Services, Department of Export Promotion based on a survey of 33 private hospitals that have a high potential to serve foreign patients,, the number of foreign patients receiving medical treatments in those private hospitals increased each year. Some countries show steady growth across the three year period –e.g. the US, the UK, and the Middle East. According to the latest data, 2005, about 12% come from North America, about 12% come from Europe, and 38% come from Asia and Australia.

For international visitors exploring the availability of medical services, web services provide a mechanism that makes it possible for them to easily identify sources of medical treatments at the various private hospitals. Having developed a system that supports this need, it is possible to extend it in such a way that it can be used for patients in emergency situations, it is critical to be able to contact a hospital or doctor in as quickly as possible. Mobile technology coupled with web services allows

people to find and access hospital information. Both of these applications have to be built keeping in mind that web services are loosely coupled [2]. When the service is not available, the service consumer cannot access the service, and an invocation error occurs. We explore dynamic invocation of web services as a means to allow services requestors are able to automatically choose the most appropriate web service from among various web services.

Sections 2 and 3 which follow describe a motivational scenario and a brief description of the involved technologies in order to provide a background for the rest of the paper. Section 4 describes the case studies that provided the data for the application described above. Finally, section 5 introduces general conclusions about the work.

2. Study Motivation

This section illustrates the need for accessing private hospital information by calling medical services using a Pocket PC and wireless technology.

As illustrated in Figure 1, the mobile application is an e-Business application that relies on the Dynamic Invocation of Web Services. Web Services allows heterogeneous applications to invoke published methods through the standard HTTP protocol. Using the application, patients are able to find private hospital information such as the location of the nearest hospital, treatments available and available doctors from a mobile device. The Mobile Device connects to a web services server with wireless technology (Wi-Fi, Bluetooth and GPRS). Currently, in Thailand, many private hospitals have a large investment in the computer technology such as grid computing, wireless technology, database management systems and web services. There are hundreds of medical services provided by each private hospital. Moreover, patients can find a list of web services with the same functionality and these service providers need to register their services in the UDDI [3] (Universal Description, Discovery and Integration) registry so this healthcare portal can find all service providers and their related services so that patients can be guaranteed that they receive the highest standard of medical care from the mobile application.

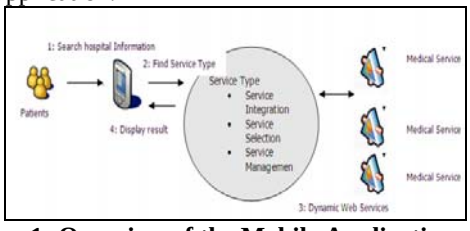


Figure 1: Overview of the Mobile Application on dynamic selection from medical services in private hospital, Thailand.

It will be useful if get the hospital information from 3 medical service providers can be obtained with mobile application.

3. Background

In this section the technologies involved in the application and the proposed architecture are briefly described, followed by a review of other approaches that are closely related to this work.

3.1 Related Technology

The two technologies involved in this proposed application architecture are the Dynamic Invocation Framework and SOA of Web services which are described in detail in the following:

3.1.1 Dynamic Invocation Framework [3] [4] [5]

The UDDI v3 protocol introduces the function of service subscription, which enables a service requestor to subscribe to interested service information and receive notification of additions, updates and deletions to a service. Figure 2 shows that the UDDI Proxy is added in SOA and acts as a proxy for Web Service, which is located on the client side.

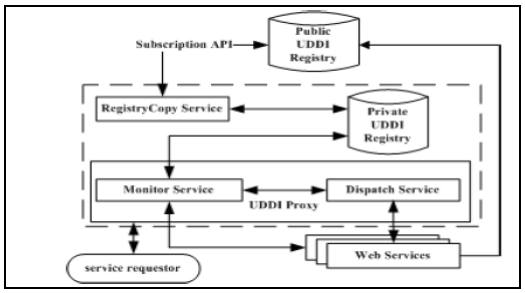


Figure 2: An Overview of the web services dynamic invocation framework based on the UDDI v3 protocol

The UDDI Proxy obtains a list of currently usable Web Services from a private service registry and performs the work of finding, testing, verifying and managing the Web Services through the Monitor Service and Dispatch Service inside the UDDI Proxy component.

1) RegistryCopy Service is closely related to the private service registry. It subscribes Web Service registry information with specific functions from the public service registry through the Subscription API. When the public service registry adds updates or deletes relevant Web Service information, the private service registry will receive all of these changes through notification and make changes to the corresponding registry information inside itself.

Monitor Service helps to acquire currently usable Web Service information from the private service registry, and supervises its status (e.g. the loading and activity status) periodically.

2) Monitor Service will pick out a list of active services, and then forward it to Dispatch Service after considering the load status of each Web Service.
3) Dispatch Service will choose the most suitable service and send the request to the Web Service.

For example, a service provider transfers a Web Service program to another server, and changes information in the public service registry. The Private Service registry will get a notification immediately and change the corresponding information. After being informed of the change of address, Monitor Service immediately changes the access point and then forwards the request to the new Web Service.

3.1.2 SOA of Web Services [6]

The Service-oriented architecture (SOA) of Web services has three distinct components (Actors), namely: Service Requestor, Service Provider, Registry and Broker.

1) Service Requestor: For a business to identify with this SOA role, it must find some commonality between their business activity and the actions of a requestor.
2) Service Provider: For a business to identify with this SOA role, it must view itself as performing some form of electronic service. Whether that service is defined as the processing of data or the act of carrying out a specific task, the business entity must believe it is performing work for others as a vocation or a business. Since almost anything can be a service, it would be hard to itemize an exhaustive list of applicable businesses services.
3) Registry: If a business entity finds itself collecting and cataloging data about other businesses and then selling that data to others, it may identify well with a registry, a form of SOA Broker. Usually, a registry would collect data such as business name, description, and contact information. In UDDI terms, this SOA role is often referred to as the White Pages.
4) Broker: Building on the concept of a registry, business entities may also be able to identify with the notion of a broker, which in UDDI terms is often referred to as the Yellow Pages. Brokers usually extend the value proposition of a registry by offering intelligent search capability and business classification or taxonomy data.

4. Cases Study

The mobile application, in this case study, is an e-Business application that demonstrates the Dynamic Invocation of Web Services. Web Services allow for heterogeneous applications to invoke published methods through the standard HTTP protocol.

This mobile application acts as a medical services portal as shown on Figure 3. Since there are hundreds of medical service providers, patients will always be interested in knowing if they are getting the best deal in medical services. This application is used to get information from three vendors who provide medical services that are published in the UDDI registry. The only restriction on participating service providers is that they must host one Web Services that implements the "MedicalService" interface. These service providers must register their services in the UDDI registry so that this portal can find all service providers. This application browses through the UDDI registry and maintains a list of all medical services vendors who provide the "MedicalService" interface. When patients request any information, this portal generates the stubs necessary for service invocation dynamically and retrieves the data. This data may be reformatted or sorted so that users have up-to-date information about the medical services they are searching for.

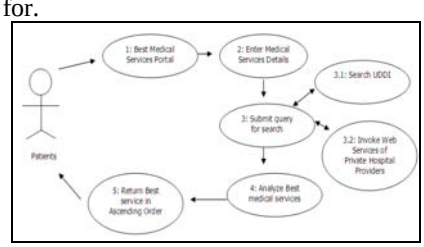


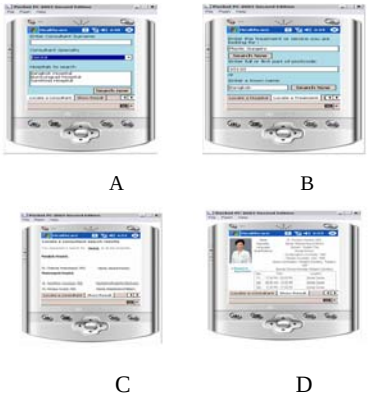
Figure 3: Our Application UseCase Diagram

As part of its Web Services product stack, Oracle provides its own UDDI (Universal Description, Discovery and Integration) registry. Oracle's UDDI implementation runs on top of Oracle9iAS. The UDDI registry allows service providers to publish their Web Services and allows patients to search and find these Web Services along with the associated WSDL url's. Once these locations are known, patients can generate the necessary stubs and utilize these Web Services.

When the experimental application is installed, it creates 3 patient databases which correspond to 3 private hospital providers namely: Hospital1, Hospital2 and Hospital3 etc. Every private hospital provider has a "MedicalService" implementation, so, there are 3 "MedicalService" providers. The WSDL url and Service url of each of these Web Services is then added to the UDDI registry along with relevant information. This application is ready to run once the UDDI inquiry url for the UDDI registry is supplied to the MedicalAnalyzer component of the application.

Figure 4: Find a Hospital

Figure 4 illustrates how a patient can find some medical services that the application provides such as locate a hospital, locate a treatment and locate a consultant from the mobile application. When a patient pushes the button "Search now, will request a medical service through the wireless technology and connect to the UDDI registry that consists of the medical services from the three private hospitals in the study that provide there services.



A B

C D

Figure 5: Our mobile application

Figures 5A and 5B shows how patients can request the medical services that they interested in by using the "Search now" button. UDDI v3, was used to build the public service registry running on the server. The Math Service that performs mathematical calculations is a web service programmed in Microsoft Visual Studio.NET 2005 and deployed using IIS (Internet Information Service) based on Microsoft's .NET framework 2.0. Math Service1 and Math Service2 register service information in the public service registry. The Private Service registry center, the UDDI Proxy and the service applicant run within the same LAN.

The RegistryCopy Service in conjunction with the private service registry, using the subscriptionListener, tModel and the BusinessEntity information of the Math Service [2] [7]. Thus, the subscriptionListener will receive any changes to these data structures from the public service registry, and will synchronize the information with the private service registry. The UDDI Proxy forwards requests from the service requestor and supervises the status of the Math Service. The mobile application sets up three step dynamic invocation scenes:

1) Invoke Math Service1; shut down the Service Container after running for a while, which will result in a malfunction of Math Service1.
2) Invoke Math Service1; transfer the execution of Math Service1 to run at the host computer without changing registry information, which results in a malfunction of the original Math Service1.
3) Invoke Math Service1; transfer the execution of Math Service1 to run at the host computer and change the registry information.

From the results as shown on figure 5C and 5D of the experiment, the mobile application can display the result of a patients query or request from the host computer. The patients can make an appointment with the doctor using email or can contact the hospital.

5. Conclusion

In this paper, a mobile application using dynamic selection of web services focused on medical services at Thailand is introduced. Further research needs to be conducted on the web services management layer (WSML) as a middleware platform for web services in client applications. The next step will be to provide the web services application to cover all private hospital in Thailand.

6. References

[1] Thailand: Centre of Excellent Healthcare of Asia, http://www.thaiwaysmagazine.com/activities/medical_service_and_spa/activities_medical_service.html

[2] JianJun Yu and Gang Zhou, "Dynamic Web Servicce Invocation Based on UDDI", Proceedings of the IEEE International Conference on E-Commerce Technology for Dynamic E-Business(CEC-East'04), 2004.

[3] Uddi Business Registry node urls, *http://www.uddi.org/find.html*.

[4] JianJun Yu and Gang Zhou, "Web Service Discovery and Dynamic Invocation Based on UDDI/OWL-S", 978-3-540-32595-6, Pages 47-55, 2006.

[5] UDDI spec technical committee specification, UDDI version 3.0., http://uddi.org/pubs/uddi-v3.00-published-20020719.htm, 19 July 2002.

[6] IBM web services architecture, Part2: Models for dynamic e-business, *http://www.ibm.com/developerworks/webservices/library/ws-arc2.html*

[7] ORACLE web services Dynamic Invocation Application, http://www.oracle.com/technology/sample_code/tech/java/web_services/wsdinvoke/index.html