Experiment of Large-Scale Compound Web Services Using a Traffic Information Web Service

Michiko OBA
Software Division,
Hitachi Ltd.
26-2 Minamioi 6-chome, Shinagawa-ku, Tokyo 140-8573
JAPAN

Norikazu MATSUYAMA
Product Development Division,
PFU Active Labs Limited,
Nu-98-2 Unoke, Kahoku, Ishikawa 929-1192
JAPAN

Norihisa KOMODA
Dept. of Multimedia Engineering, Graduate School of Information Science and Technologies,
Osaka University
2-1, Yamada-oka, Suita 565-0871
JAPAN

Abstract: The XML consortium, a committee in Japan, developed a travel planning system that used WS-BPEL to organically combine a traffic information Web service with multiple Web services. The system was a proof-of-concept experiment for compound Web services. The developed large-scale system included more than 10 kinds of content-offering Web services, and various functional components. These were developed in 20 companies. The experiment verified the effectiveness of using Web services, SOA, and XML.

Key-Words: Web services, WS-BPEL, SOA, XML

1 Introduction

This paper reports on a large-scale experiment that integrated a variety of technologies (e.g., Web services, SOA, WS-BPEL, and XML) into a Web-based travel planning system. The experiment enabled the XML consortium to verify ease of system integration and to identify the possibility of new business models.

The experiment involved issues confronting an increasing number of companies as they implement desired features and content as Web services, and enable access over the Internet (companies such as Google and Amazon.Com).

The experiment used traffic information provided by the Japan Road Traffic Information Center (JARTIC) [3]. The JARTIC foundation provides traffic information in Japan. As an experiment, JARTIC provided travel information as XML data via a Web service. JARTIC's goals were to examine whether it could increase usage of the traffic information and whether it could expand commercial businesses related to such information.

The XML consortium [4] developed the travel planning system based on the assumption that the users would be travelers going to the 2005 World Expo in Aichi. In addition to the traffic information provided by JARTIC's Web service, the system combined content (e.g., satellite maps, weather, and tourist information) provided by various other services, together with blogs, knowledge searches, translation, mail, and other features. The system combined all this functionality by using Web service technologies such as WS-BPEL. Twenty companies collaborated in the development of the system.

Two key focuses were on SOA and Web services. SOA (Service Oriented Architecture) is currently receiving a lot of attention as a technology for building IT infrastructures that can handle the dramatic changes occurring in business environments [1]. Web services are acknowledged as a core technology for achieving SOA [2].

2 Traffic Information Web Service

JARTIC was established in January 1971 to unify collection and provision of traffic information (which includes both road and traffic information). Traffic information is consolidated at a traffic mission control center from various sources such as vehicle-detection devices installed on roads, video cameras, calls to emergency centers, and wireless communications from patrol cars. JARTIC collects, arranges, and analyzes such traffic information, and provides it to users via various media such as cellular and ordinary phones, radio, television, and car navigation systems.

JARTIC began experimental XML delivery of traffic information in October 2004, with the goals of expanding the usage of such traffic information and also expanding related commercial business. JARTIC used XML because of the need for inter-system co-operation, and provides the XML data as a Web service. The XML data is delivered by two types of service: push type (transmission at regular intervals) and pull type (transmission at any time).

3 Experiment Details

3.1 Purpose of the Experiment

The actual experiment had the following two goals:

(1) To develop and evaluate applications based on SOA.

In addition to providing traffic information, in the experiment we converted different functionalities, services, and contents into Web services, and built cooperating applications based on SOA. We then evaluated the merits and demerits from the standpoints of both development and functionality.

(2) To verify the possibility of providing traffic information as a commercial business.

We wished to verify, by actual experiment, the possibility of expanding the provision of traffic information as a commercial business. In particular, we wished to provide and use such information in new situations: for example, situations different from the way in which data is sent to existing car navigation systems, where it is used unchanged.

3.2 Outline of the Experimental System

The experiment consisted of developing a travel-planning system. We chose this system for the following reasons:

(1) When making a travel plan, traffic information is required to calculate the time required to travel between sightseeing spots.

(2) When making a travel plan, more than just travel information is needed. You also need cooperation among different functions and content (e.g., to acquire sightseeing information or weather information).

(3) We could use the results of past experiments carried out by the XML consortium [5][6].

3.3 Configuration of the Experimental System

We developed all the functionalities for the above-mentioned requirements as separate Web service components. Next, we used SOA to enable these services to cooperate, and then we developed the travel planning system.

We developed 13 Web services for this experiment: 11 new services and 2 existing services. In addition to these Web services, code to enable co-operation among Web services was developed for three subsystems related to blogging. This was implemented for the client applications, which had user interfaces. Each Web service was controlled by a client application.

We developed the following two kinds of user interface:

(1) Rich client interface in Java.

(2) Web application client interface in JavaScript

Figure 1 shows the configuration of the experimental system.

This experimental system was based on the assumption that the user would be a traveler going to the 2005 World Expo in Aichi, Japan. Therefore, we developed the system by using content (e.g., satellite images and sightseeing information) relating to the surroundings of Nagoya City, Aichi Prefecture, and the Expo area.

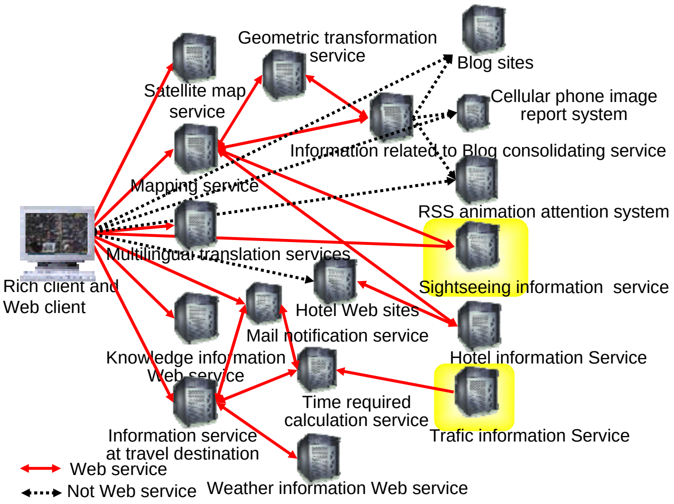

Figure 1: Configuration of the Experimental System

3.4 Implementation of Each Web Service and Subsystem

3.4.1 Client Applications

(1) Rich client

The rich client application used for making a travel plan is an Eclipse-based Java application (Figure 2). Users make a travel plan by selecting an icon on the map, and adding a destination.

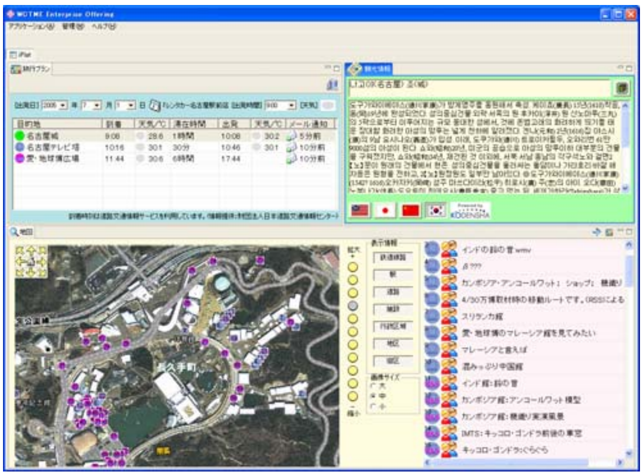

Figure 2: Example of the Rich Client

Key features of this application are as follows:

- It has a highly functional, interactive, user interface.
- The functionalities used to access the different Web services were developed as components, so the functionality was executable as plug-ins.
- The application executes SOAP requests by using asynchronous processing and multithreading.
- Once loaded, images are encrypted, cached, and stored securely.

(2) Web client

The Web client application used for making a travel plan runs on a Web browser (Figure 3). Users make a travel plan by selecting an icon on the map, and adding a destination, in the same way as for the rich client.

Figure 3: Example of the Web Client

Key features of this application, which uses Ajax technology, are as follows [7]:

- The application can asynchronously send and receive SOAP messages, and render satellite images.
- Unlike previous Web applications that redraw the whole window, this application redraws the necessary parts only.
- All the window parts are re-usable components.

3.4.2 Web Services

We developed the following as Web services:

(1) Traffic information service

Provides the JARTIC traffic information.

(2) Satellite map service

Provides satellite maps in response to a request, which can include a variety of information (e.g., UTM coordinates and map size). An existing satellite map application was converted to this Web service.

(3) Sightseeing information service
Handles sightseeing information provided by public and private organizations.
(4) Hotel information service
Provides information about accommodation (e.g., in hotels or traditional inns).
(5) Knowledge information service
Enables users to enter keywords to search sightseeing information, and provides links to explanations of keywords. We converted an existing knowledge search engine into a Web service.
(6) Multilingual translation service
Translates Japanese into English, Chinese, and Korean. We converted an existing machine translation engine into a Web service.
(7) Blog-related information consolidation service
Searches blogs for keywords. The blogs use RSS.
(8) Mapping service
Performs keyword searches of various content (e.g., sightseeing information, accommodation information, and blogs) relating to a specified area. The client application uses the mapping service to display icons on a currently displayed satellite map.
(9) Geometric transformation service
Converts between the different coordinate systems used in the experiment.
(10) Time-required calculation service
Calculates the time required to go between two points, based on traffic information.
(11) Weather information service
Retrieves weather forecast information for a specified location.
(12) Email notification service
Uses email to notify users of the departure time calculated from the time required.
(13) Travel-destination information provision service
Enables two or more Web services to be called as one service from a client application. We implemented this service using WS-BPEL [8]. Figure 4 shows this method.

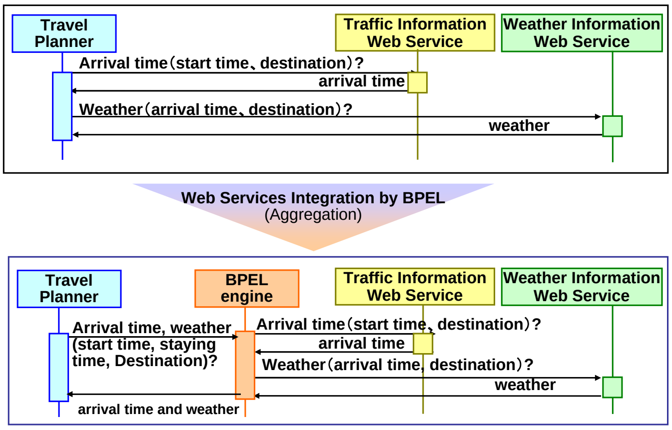

Figure 4: Simplified Web Service Calls

3.4.3 Other Subsystems

(1) Blog sites
We developed blog sites for this experiment. Local information on the sightseeing spot was registered at each site. The sites were displayed as icons on satellite maps.
(2) Cellular phone image report system
This system can easily make a report by adding a simple comment to the image taken by a cellular phone's digital camera. This generates ATOM web feeds. These are displayed as icons on a satellite map. Coordinates on the displayed map use GPS coordinates included in the portable image.
(3) RSS video-feed system
This Web application gives explanations in RSS format for the time that a video was taken, and enables the explanation to be played from any location. The explanations can be located by coordinate information displayed as icons on a satellite map.

4 Findings

The following findings were obtained from this experiment.

4.1 Findings Related to Development

(1) Development schedule, participating organizations, and usable products:
This was a very large-scale product, with 29 participating organizations. Figure 5 shows the development schedule. The development period was about four months. However, the actual mounting and connection tests were less than half of this period because this experiment was in addition to participants' main work.
Eighteen Web services and products were used for development. Co-operating systems included 13 Web

services and three subsystems. In addition, we developed two types of portal systems that called these systems. In the past the XML consortium has developed experimental systems that used many Web services. This experiment, however, enabled co-operation among the largest number of Web services so far. From examining the literature, we believe this is one of the largest projects for enabling co-operation among services.

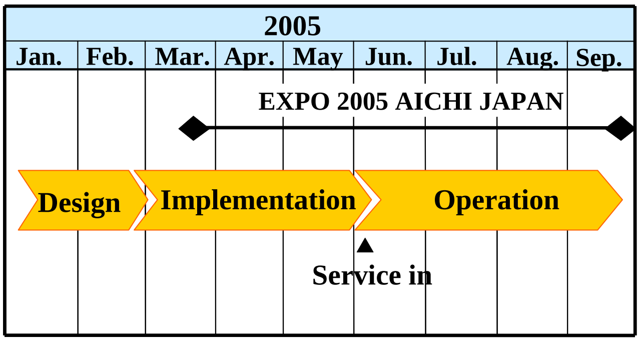

Figure 5: Development Schedule

(2) Findings related to SOA

By using Web service technology, we could extract content and convert existing services into re-usable services. By combining these services, we could develop business systems rapidly. Moreover, in the experiment we were able to create new value that was not achievable from individual services alone.

By using WS-BPEL we could combine services visually, without programming. This decreased the work hours required for logic development. The greater the number of cooperating services, the greater the number of work hours required for service integration. However, using WS-BPEL reduced the work hours required for mounting and testing connections.

(3) Enabling cooperation among business logic procedures, and user interface development

We used Web service technology and XML to separate the GUI programs and logic procedures, and we could then develop them separately. As a result, each business logic procedure was usable from the two types of client applications. Because different functionalities could be developed independently of each other, they could be developed in a short time.

The time required to develop the user interface increased in proportion to the number of business logic procedures that had to cooperate with each other, and revisions were always required when a business logic procedure was added or the interface was modified. As a result, a lot of time was required for these tasks.

This experiment made it clear that we need to enhance the tools and techniques to enable efficient development of the user interface.

4.2 Findings Related to Business

By using Web service technology to convert existing functionality, contents, and services to independent Web services, we can improve reuse and can expect new business opportunities to arise from combining the Web services.

We achieved real-time business cooperation based on Web-service technology. By using blogs, we could provide the public with content in real-time. In addition, we could identify the applicability of different technologies (e.g., blogging and Web services). This in turn made us aware of the possibility of new businesses based on co-operation of technologies and public and private information.

5 Conclusion

In this experiment, we developed a travel planning system. The system used WS-BPEL to organically combine the traffic information offered by a Web service with multiple Web services. This was a large-scale system, composed of 13 Web services and three subsystems. By developing this system, we were able to verify the effectiveness of using SOA and XML to develop new business models.

References:

[1] D. Krafzig, K. Banke, D.Slama, Enterprise SOA: Service-Oriented Architecture Best Practices (The Coad Series), Prentice Hall (2004.11)

[2] Gustavo Alonso, Fabio Casati, Harumi Kuno, Vijay Machiraju; "Web Services: Concepts, Architectures and Applications (Data-Centric Systems and Applications)", Springer (2003.6)

[3] Japan Road Traffic Information Center, http://www.jartic.or.jp/ (in Japanese)

[4] XML Consortium, http://www.xmlconsortium.org/introduce/index-e.htm

[5] Michiko Oba, Norikazu Matsuyama, Kojiro Nakayama, and Norihisa Komoda; Web Services Experiment Using Standard XML: TravelXML for Travel Industry Electronic Commerce, in Proc. of 1st Int. Workshop on Enterprise and Networked Enterprises Interoperability (ENEI'05) (co-located

with the 3rd Int. Conf. on Business Process Modelling (BPM2005)), FP-035, pp.152-156 (2005.9.5, Nancy, France)

[6] Michiko Oba, Norikazu Matsuyama, Kojiro Nakayama, and Norihisa Komoda; Web Services Experiment Using TravelXML: Standard XML for Electronic Commerce in the Travel Industry, in C.Bussler and A. Haller (eds); "Business Process Management Workshops", Lecture Notes in Computer Science, Vol. 3812, pp.385-389, Springer (2006.2)

[7] Nicholas C. Zakas, Jeremy Mcpeak, Joe Fawcett; Professional Ajax (Programmer to Programmer), Wrox Pr Inc (2006.2)

[8] OASIS WS-BPEL TC; http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel