International Journal of Advanced Computer Research (ISSN (print): 2249-7277 ISSN (online): 2277-7970)
Volume-4 Number-3 Issue-16 September-2014

Real Time Messaging Server Using Duplex Web Services

Hitesh K. Mhatre[1], Ketan N. Kanere[2]

Abstract

In this paper of real time messaging server is proposed to support two-way client-server interactions. Here, the role of service broker between the clients and the server is played by the duplex web services. Depending on the client profile, web service registry and run time service status, web service is dynamically selected. The Real Time Messaging Framework will dynamically configure into a Client Proxy to maintain communication, which can be real time text, audio or video. Because of the duplex web services, overhead will be reduced at the server side by using push back services at the same time. This service is the part of two – way interaction of the web service that passes the message to the other client in the network. In the whole process, emphasis will be on increasing the data transfer speed and efficiency. Thus, use of a web service in a chat server will make it work better and faster in distributed environment.

Keywords

Duplex Web Services, HTTP, RTMS, WCF, W3C.

1. Introduction

A Web Service (WS) is a type of communication between two electronic devices over World Wide Web. A web service is a software function used at runtime provided at a network address over the web or the cloud; it is a service that is "always on" and running as in the concept of utility computing.

Basically, Web Services are software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-executable format (specifically Web Services Definition Language) [3].

Manuscript received July 30, 2014.
Hitesh K. Mhatre, Head of Information Technology Department, Pravin Patil College of Diploma Engineering and Technology.
Ketan N. Kanere, Assistant Professor, Computer Engineering Department, Rizvi College of Engineering.

Other systems interact with the Web service in a manner prescribed by its description using Simple Object Access Protocol Messages. These are typically showed using HTTP with an XML serialization along with other Web-related standards.

Duplex Web Services are Two-way Web services, which are inherently asynchronous, since all messages, such as reply and event, initiated from the server are asynchronously sent to the client. In Two-way full duplex WS, interaction involves both client initiated service request and proactive server push for event notification.

Communication between client and servers is done independently through a duplex contract, so that either can initiate calls to the other. The duplex contract is one of the three message patterns available to Windows Communication Foundation (WCF) services. The other types of message patterns are one-way and request-reply. A duplex contract includes two one-way contracts between the client and the server and does not require the method calls to be correlated.

Use of web service as a communication media between applications is a very important aspect. An efficient way is provided by Web Services for facilitating communication between applications. But, Web Services also provide some limitations. The main drawback of using web services is that communication is possible over HTTP only [5]. Another drawback is that it provides only simplex communication and there is no way to have half duplex or full duplex communication using web services.

Windows Communication Foundation enables to run Web Services not only on HTTP, but also on the other protocols and it supports duplex communication. With WCF, one can define his/her service once and then configure it in such a way that it can be used via HTTP, TCP, IPC, and even Message Queues. Also, one can utilize Web Services using server side scripts (ASP.NET), JavaScript Object Notations (JSON), and even REST (Representational State Transfer).

Organization of the rest of the paper is as follows. Section II covers one way operations seen in Web Services. Section III discusses various operation patterns used in Web Services. Section IV talks about the two way interactions in Web Services and various interaction patterns seen between client and server. Section V throws light on the proposed methodology. Section VI presents the architecture of real time messaging server using duplex Web Services followed by Conclusion and Future Work.

2. One Way Operations in Web Services

Sending a request is provided by one way operations with the expectation that a SOAP response will not be returned [7]. The Siebel application provides the ability to publish and utilize Web services that implement one-way operations. For both inbound and outbound scenarios one way operations comes into play.

If any output arguments is not given in Business Service Workflow, then it is a one-way inbound operation. If no output arguments are provided by the service proxy method, then it is one-way outbound operation [2].

Consider a scenario for one-way operations when data loss is tolerable. In such cases, a SOAP request is sent and it won't receive a SOAP response. The SOAP request is received by the provider and then it is processed by it.

3. One Way Operation Patterns

In this section, One-way operation patterns used in Web Services are covered. These patterns are one way and notification operations, Request/Reply operation.

A. One way notification operation: This pattern provides a high level of decoupling between the client and service provider by having two datagrams, one for the request and other for the response as shown in Figure 1.

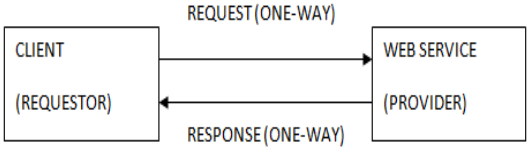

Figure 1: One-way and notification operations

Here, the client creates the correlation ID and sends it to the service via some specified mechanism. This mechanism is accepted by both the parties for communication between the client and the web service.

Acceptable mechanisms can be HTTP, SOAP, etc.

B. Request/Reply operation: This pattern has two messages, request and response and they are defined in a single request/reply operation. However, both messages are sent as two separate transmissions. The request/reply is applicable for general purpose services where the request results in a single response. Examples of such services include the persistence or retrieval of data, or the initiation of a business process which may include a single unit of work, such as an electronic payment.

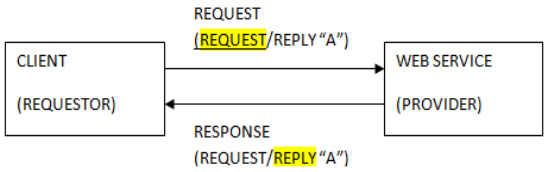

Figure 2: Request/Reply operations

In Figure 2, the client creates the correlation ID and passes it to the service provider via HTTP or SOAP. Both, client and the service provider should have the same mechanism.

Client defines the reply-to address in which it denotes where to send the response. Due to the single operation used for this pattern, a reference to the address or the explicit address itself must be provided as a parameter on the request. For example, if the client has published an asynchronous response listener service supporting one-way operations, a reference to the service's WSDL can be provided on the initial request [1].

4. Two Way Web Services Interaction

In two-way Web services interaction, at each end point of Web service, both the client and server are required. Therefore, an interaction requires two appropriate WSDLs, one at each end point . These two WSDL interfaces must be correlated in a certain way to make two-way interaction possible. From the perspective of an end point, two-way full duplex Web services interaction involves both client-initiated

service requests and proactive server pushes for event notification.

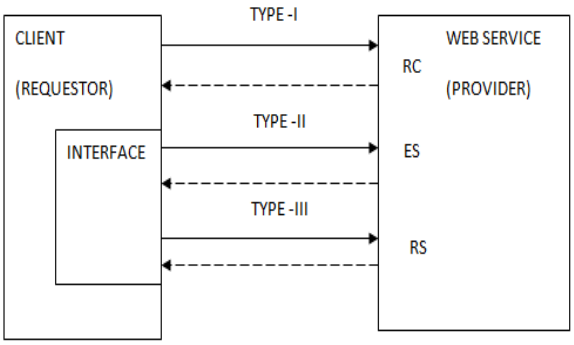

Figure 3: Two-way Interaction patterns between client and server

A generic client and server are used to separate two WS endpoints, although both of them act as a client as well as server [6]. From service interaction point of view, three types of generic Web service interaction patterns are seen. They are Type I (RC), Type II (ES) and Type III (RS).

Type I (RC): Request is initiated by the client, with or without response.
Type II (ES): Event reports from the server, with acknowledgment (solicitation) or without (notification).
Type III (RS): Request is initiated by the server, with or without response.

Type I is the conventional one-way WS interaction pattern, where the client makes a one-way WS request to the server. Type II is the interaction pattern of asynchronous reply and event notification from the server to the client. In practice, an asynchronous reply is often modeled as event notification with some correlation information with which the client can associate the reply to its request. The type II interaction pattern is separated into two sub-classes, depending on the response pattern. If the server solicits acknowledgement from the client when the client receives the event report from the server, then the Type II interaction pattern is called "solicitation." If no acknowledgement is solicited after receiving the event report, then the Type II interaction pattern is called "notification."

Type III is the reversal of the Type I interaction pattern, where the server issues the service request to the client. The patterns of these interactions are illustrated in the Figure 3, where solid arrows indicate initial messages, and dotted arrows indicate optional response messages. Type I based WS interaction pattern requires the server to expose a proper service WSDL interface for client to utilize services from the server through SOAP messages [4]. Type II and Type III interaction patterns require the client to expose an appropriate WSDL interface in order to receive the asynchronous event notification from the server and allow the server to utilize services from the client.

5. Proposed Methodology

The classic object and component-oriented programming models offer only a single way for clients to call a method. In this model, the client issues a call, blocks while the call is in progress, and then continues executing once the method returns.

The proposed system will be designed using duplex web services for communication between the client and the server. Due to the duplex web service, the messages coming from the client machine will be given to a web service, which will be responsible for responding back the message to other multiple clients in case of multi-client system.

The transfer of message will be faster than other messaging systems as the web service will be using a callback operation, which will reply to the message coming from the client machine and send an acknowledgement to the client machine.

The client sending messages to the remote server is shown in the Figure 4.

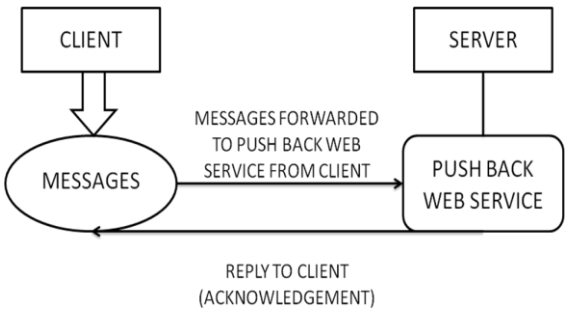

Figure 4: Client sending message to remote server

The working of the model is as follows: First, the client will send messages to the server, which will be operated by a web service from client to the server. This web service will be responsible for handing the datagram over http protocol, as well as event

handling. The authorization will be carried out by the web service, which is responsible for transferring the message from the client machine to the server.

The pushback web service will be an important factor in responding to the messages coming from the client. Now, the pushback web service will gather the message coming from the client in a queue. These messages will be stored further to send response to the client. In this way, web service is used to carry out interaction between server and client. Here, the pushback web service is actually responsible for sending the messages to the client.

An event notice will be triggered after some due course of time to notify the server that it will receive the messages from the client. The pushback web service will handle the task of notifying the server about the interactions that is going to happen with the client in some fixed intervals of time. This time interval can be changed according to the requirement of the cache memory and the duration between the interactions.

Now, let us see the mechanism for pushback web service sending messages to multiple clients. As seen in Figure 5, the client will send messages to the server which will be routed through a web service from client to the server. The messages coming from the client machine will send to the pushback web service. For handling the response of the server, the pushback web service uses the callback operation to respond to the client machine.

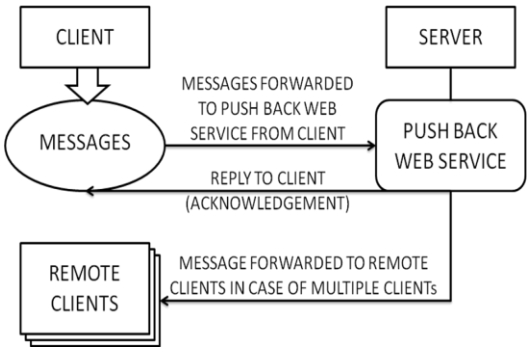

Figure 5: Client sending message to multiple clients

Now, consider an instance where the multiple clients are receiving messages from single client. In such cases, the message coming from the single client will be forwarded by the pushback web service to the multiple clients thus making it possible to send messages to all the remote clients at the same time. This is a classical scenario where interaction between one client and the other remote multiple clients through the event created by the pushback web service for multiple response to multiple client machines.

A scenario where multiple remote clients responding back to the host client is shown in Figure 6.

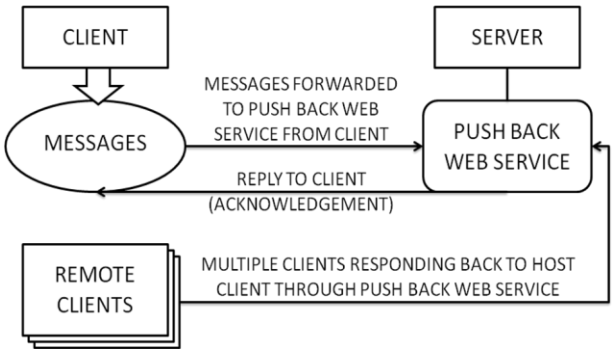

Figure 6: Multiple clients responding back

The messages coming from the remote client will be sent to the pushback web service first. This web service will be responsible for handling the response of the host client. As soon as the message is reached at the pushback web service, it will send the acknowledgement to the remote clients by sending the message.

As soon as the host client receives the message, it sends back the acknowledgement to the pushback web service. The pushback web service will forward the same along with message if any to all remote clients at the same time. This scenario represents the interaction between one host client and the multiple remote clients with the help of pushback web service.

6. Architecture of Real Time Messaging Server

To implement the duplex communication as discussed in the previous section an altogether new architecture is proposed to handle real time messaging server. The architecture of real time messaging server consists of five modules: Text module (TM), White Board module (WBM), File Transfer Module (FTM), Audio Module (AM), Video Module (VM) as shown in Figure 7.

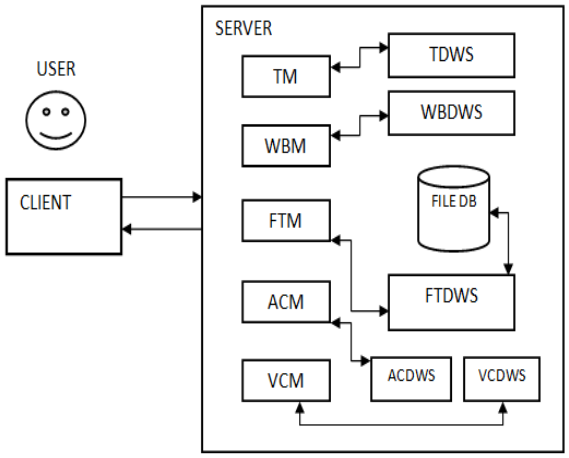

Figure 7: RTMS Architecture

Each module will use duplex web services for the client-server communication. Text Module(TM): Text module will be having a text message duplex web service, which will pass text messages between the communicating clients. Messages can be transferred to selected users or multiple users based on the selection.

White Board Module (WBM): White board module will have a White Board Duplex Web Service, which will allow drawing of images or diagrams such that the drawing or pattern made can be send to multiple users at the same time. This feature will enable users to share their idea regarding any concept or problem discussion.

File Transfer Module (FTM): File Transfer module will have a File Transfer Duplex Web Service, which will allow uploading and downloading of files between multiple users. Sharing of files will become easier and faster using this feature.

Audio Conferencing Module (ACM): Audio Conferencing Module will have an audio conferencing Duplex Web Service, which will enable audio conferencing between different nodes connected in the network.

Video Conferencing Module (VCM): Video Conferencing Module will have a video conferencing Duplex Web Service. This module will be used for video conferencing between different clients connected in the network.
Text Duplex Web Service (TDWS): Text Duplex Web Service will be interfaced with the text module to send the response back to the client in the form of a text message.

White Board Duplex Web Service (WBDWS): White Board Duplex Web Service will be responsible to send the content drawn on White board through a duplex callback service.

File Transfer Duplex Web Service (FTDWS): File Transfer Duplex Web Service will be interfaced with the FTM to share files among clients. The content of FTM will be stored in a FILE DB as shown in Figure 7. Audio Conferencing Duplex Web Service (ACDWS): This Web Service will give the audio callback between different clients.

This Web Service will be interfaced with ACM to enable audio conferencing between clients.

VCDWS (Video Conferencing Duplex Web Service): Video Conferencing Web Service will enable webcam to be shared between multiple clients and hence give a video callback. This Web Service will be interfaced with VCM to enable video conferencing between clients.

The duplex web services, which give a callback to the request coming from the client is designed for each module according to the requirement of the module. The duplex web service will then be responsible for the message to be transferred to multiple clients or selected clients at the same time. The architecture implies that the services generated for the different modules will be published to the client via the modules generated for each of the task assigned according to the module. This will give us a communication system, which will be having different facilities and better output than the existing systems.

7. Conclusion and Future Scope

A Web Service is a mean for communication between two clients over World Wide Web. One way operations are used in inbound and also in outbound scenarios. The one way SOAP response is expected not to be returned. Request Response can be one type of one way operation. In two way web services, endpoints will be present at both client and the server. Here, duplex web services are found to be suitable to implement the architecture of RTMS. The callback operation facility in Duplex Web Services helps in increasing the speed and efficiency of the system. The

message communication speed between client and server is expected to increase by 10% using RTMS architecture. Here, the messaging server communicates using duplex web service.

In coming days, it is decided to realize the proposed architecture by carrying out the implementation of it and later on analyzing and testing the working of realized system to carry out improvement if any in it. The present system architecture can be scaled up to mobile clients. The mobile clients can communicate to standalone systems, i.e., client nodes using duplex web services.

References

[1] OASIS. (2007, April.), "Web Services Business Activity (WSBusinessActivity) Version 1.1," Available: http://docs.oasisopen. org/ws-tx/wstx-wsba-1.1-spec-os.pdf [Aug. 23, 2013].

[2] N. Srirajun, P. Bhattarakosol, P. Tantasanawong, and S. Han, "Trustable Web Services with Dynamic Confidence Time Interval," in Proc. of the 4th Int. Conf. on New Trends in Information Science and Service Science (NISS), 2010, pp.11-13.

[3] WSDL 2.0: A Pragmatic Analysis and an Interoperation Framework, 2008 SYS-CONMedia.

[4] W. Dong, "Multi-agent Test Environment for BPEL-based Web Service Composition", the IEEE Int. Conf. on Cybernetics and Intelligent Systems, 2008, pp.855-860.

[5] C. -H. Liu, S. -L. Chen, and X. -Y. Li, "A WS-BPEL Based Structural Testing Approach for Web Service Compositions," the IEEE International Symposium on Service-Oriented System Engineering, 2008, pp.135-141.

[6] W3C. (2007, June), "Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language," Available: http://www.w3.org/TR/wsdl20/ [Aug. 10, 2013].

[7] N. Parimala, and A. Saini, "Web Service with Criteria: Extending WSDL," in Proc. of the 6th Int. Conf. on Digital Information Management, 2011, pp. 205-210.

Hitesh Mhatre: Currently working as Head of Information Technology Department at Pravin Patil College of Diploma Engineering and Technology. Has 4 years of teaching experience. Completed Bachelor's of Engineering from Mumbai University in 2010 and Master's of Engineering from Mumbai University in 2014. Has published 4 International Papers (3 in IEEE) and 1 National Paper and also published book on "Computer Simulation and Modeling".

Ketan Kanere: Currently working as Assistant Professor in Computer Engineering Department at Rizvi College of Engineering. Has 4 years of teaching experience. Completed Bachelor's of Engineering from Pune University in 2010 and Pursuing Master's of Engineering from Mumbai University. Has published 3 International Papers (2 in IEEE) and 1 National Paper.