ABSTRACT

A web service is a software application published on the web and accessible through standard Internet protocols. Multimedia web services generally integrate multimedia contents over web services. Core techniques of web services need to be enhanced accordingly in order to facilitate multimedia transportation and handling. Seamlessly integrated with Simple Object Access Protocol (SOAP), this paper presents a model for network router supporting dynamic selection and binding of most efficient protocol serving for Quality of Service (QoS) based multimedia web services.

1. INTRODUCTION

In broad sense, a web service is a software application published on the web and accessible through standard Internet protocols [6][10]. This emerging paradigm enables web applications to easily integrate other independently published web-based software components to conduct new business transactions. Multimedia web services generally involve transportation of multimedia contents over web services, and management of composite devices for multimedia contents. Multimedia content here refers to the content that seamlessly integrates multiple media types in a synchronized and interactive presentation [5]. Core techniques of web services need to be enhanced accordingly in order to facilitate multimedia transportation and handling. For example, Simple Object Access Protocol (SOAP) [8] is a lightweight protocol for exchanging structured and typed information specifically for web services [1], and is rapidly becoming a de facto standard. However, SOAP was not designed particularly to support multimedia transportation. In addition, multimedia service generally requires Quality of Service (QoS) due to the nature of multimedia data [3], such as synchronization within and among different multimedia data streams and real-time delivery requirement.

In our previous work, we presented a SOAP-oriented component-based framework supporting device-independent multimedia web services [10]. The framework consists of two intelligent agents: Multimedia Web Service Server Agent (MWSSAgent) and Multimedia Web Service Agent (MWSAgent) residing on web service provider and local proxy server respectively, as illustrated in Figure 1. SOAP messages are adopted to transfer requests and responses through Internet between MWSSAgent and MWSAgent. However, transportation of SOAP messages is considered as end-to-end transportation only. In the real world, a SOAP message sent by the MWSSAgent may have to travel through heterogeneous intermediate networks before it finally reaches MWSAgent. As illustrated in Figure 2, a message sent by the MWSSAgent needs to pass through three routers (router 1, router 2, router 3) and two different types of networks (network type 1: router 0->1, router 2->router 3; and network type 2: router 1->router 2, router 3->destination node) before hitting the MWSAgent. In addition, a SOAP message in this paper refers to a SOAP message containing multimedia information that normally possesses QoS requirements. Therefore, routers [4] at the gate of every network may need to select the most appropriate network protocol so as to satisfy the protocol requirement specifications contained. For example, a router may choose NACK reliable multicast protocol [2] on ATM network to meet performance requirement. The
challenge is to ensure timely delivery of digitalized audio-video SOAP messages.

The purpose of this paper is to propose a router framework in order to support QoS-based multimedia web service transportation. This research concentrates on protocol selection and dynamic binding based on QoS requirements of multimedia information to be conveyed. Further enhancement of SOAP to facilitate multimedia web services is presented as well. The rest of this paper is organized as follows. In Section 2, related work is discussed. In Section 3, we discuss the enhancement of SOAP protocol supporting multimedia web services. In Section 4, we present a framework for router. In Section 5, we discuss an example. In Section 6, we summarize the contributions and innovations, assess limitations and discuss future work directions.

2. RELATED WORK

A variety of research has been conducted in the literature in order to provide QoS guarantees for multimedia transportation over networks. Among the variety of efforts, Wu and colleagues [9] present an additive-increase and multiplicative-decrease transport protocol to support quality streaming video to proxy server. Floyd and colleagues [2] present an IP multicast group delivery model to support a scalable and reliable multicast framework, which guarantees minimal reliability of delivering data to a group of members, while deferring more advanced functionality to particular applications. Some other researchers focus on particular QoS features of multimedia transportation, such as synchronization requirement among different media streams. For instance, Khan and colleagues [3] introduce parameters to measure QoS performance of protocols for multimedia transportation in terms of synchronization. Some other researchers concentrate on QoS routing algorithms to select best paths based on connection traffic parameters and link load information [7]. For example, Lui and colleagues [4] present a new format of network QoS presentation and propose a corresponding hierarchical QoS routing in delay-bandwidth sensitive networks. However, research on multimedia QoS transportation is not oriented toward web services, where SOAP messages are the primary messages passing protocol on Internet.

3. SOAP ENHANCEMENT

SOAP is rapidly becoming a de facto standard supporting web services. However, since it was not designed particularly for multimedia web services, some enhancements is needed to facilitate multimedia transportation. The original SOAP specifications do not define how SOAP messages are routed between nodes, nor the manner by which the route is determined [8]. However, different types of multimedia content may require different transportation QoS. In addition, some multimedia content may require real-time transportation. Therefore, we suggest several enhancements as follows.

Several global attributes are introduced to the SOAP definition: reliable, realTime, unicast, multicast, and secure. Each attribute MUST appear in the presence in order to be effective. The reliable attribute can be used to indicate whether the SOAP message requires reliable transportation or not. The value of the reliable attribute is either “1” or “0”. The absence of the reliable attribute is semantically equivalent to its presence with the value “0”. If a header element is tagged with a reliable attribute with a value of “1”, the recipient of that header entry either MUST find a reliable transportation protocol, or MUST fail processing the message.

The realTime attribute can be used to indicate whether the SOAP message requires real-time transportation or not. The value of the realTime attribute is either “1” or “0”. The absence of the realTime attribute is semantically equivalent to its presence with the value “0”. If a header element is tagged with a realTime attribute with a value of “1”, the recipient of that header entry either MUST find enough resources to transfer the message right away, or MUST fail processing the message.

The unicast attribute can be used to indicate whether the SOAP message is to be unicast to a specific end point. The value of the unicast attribute is either “1” or “0”. The absence of the unicast attribute is semantically equivalent to its presence with the value “0”. If a header element is tagged with a unicast attribute with a value of “1”, the recipient of that header entry either MUST find a available path to the specified end point node, or MUST fail processing the message.

The multicast attribute can be used to indicate whether the SOAP message is to be multicast to multiple users. The value of the multicast attribute is either “1” or “0”. The absence of the multicast attribute is semantically equivalent to its presence with the value “0”. If a header element is tagged with a multicast attribute with a value of “1”, the recipient of that header entry either MUST find
available paths to all specified end-point users, or MUST fail processing the message.

The secure attribute can be used to indicate whether the SOAP message has to be kept secure in the process of transportation. The value of the secure attribute is either “1” or “0”. The absence of the secure attribute is semantically equivalent to its presence with the value “0”. If a header element is tagged with a secure attribute with a value of “1”, the recipient of that header entry either MUST find secure paths to the destination, or MUST fail processing the message.

Figure 3 is an example of a SOAP message containing some multimedia information. The message needs to be transferred reliably, real-time, and only transferred to one specific end user. These enhancement provides a way for SOAP message to delineate its QoS requirements; therefore routers can utilize the information as a guidance to choose different paths and transportation protocols so as to increase the performance.

4. ROUTER FRAMEWORK

According to our enhancement to SOAP protocol, SOAP messages carry multimedia QoS requirements in their envelopes. Therefore, we design a router infrastructure in order to support dynamic protocol selection and binding in terms of SOAP QoS requirements. We only consider selection of protocols to increase performance; while selection of paths normally known as routing algorithm is not considered here. The structure of a router is composed of four components, as illustrated in Figure 4. SOAP parser is in charge of parsing incoming SOAP messages, figuring out contained QoS requirements and passing it to Protocol Manager. Protocol Manager (PM) receives QoS requirements, searches Protocol Pool, and finds out appropriate protocol to use. Protocol Pool (PP) is the interface encapsulating all protocols registered in the router. Protocol Registration Manager (PRM) handles the registration of protocols to the router. Under PP are two sub-components: a database of registered protocols, and an abstract matrix that records the protocols and their QoS characteristics, such as reliability, real-time, security, etc. PRM maintains the abstract matrix when a new protocol is registered to the PP or removed from PP.

When PRM registers a new protocol to the PP, it not only records its QoS properties, but also their costs. The cost here refers to the efficiency of the protocol. The more reliable a protocol is, the more costly it is. For instance, TCP is more costly than UDP because of acknowledgment (ACK). Figure 5 is an example of the abstract protocol matrix. Three protocols are registered in the protocol pool: TCP, UDP, and NACKRMP. TCP and NACKRMP are reliable protocols; NACKRMP is a real-time protocol; TCP and UDP can be used for uni-casting; TCP, UDP, and NACKRMP can all be adopted for the multicasting purpose. From Figure 5 we can also conclude that TCP is more costly than UDP; and NACKRMP costs the most.

5. EXAMPLE

We utilize the example we constructed in our previous work [10], which is a distance-learning environment, as the basis to build up an experimental system applying our
router model. In our previous example, the proxy server directly communicates with the server machine through Ethernet. In our new example, we added three routers (router 1, router 2, and router 3) as intermediate nodes between the proxy server and the server machine, as shown in Figure 2. The server machine itself can be considered as router 0 since it is the sender of SOAP messages and has to make the first selection. We introduce two types of networks into our example: Ethernet and ATM. Ethernet is set up between the server machine and router 1, and router 2 and 3; and ATM network is the connection between router 1 and 2, and router 3 and the proxy server. Router 0 and router 2 possess protocol pool containing TCP and NACK RMP [2]; router 1 and 3 possess protocol pool containing UDP and NACK RMP. Suppose we require real-time QoS multimedia transportation. The routers would check the abstract protocol matrix for the best match with least cost. As a result, router 0 and 2 will choose TCP, and router 1 and 3 will choose UDP to increase performance. This example shows that the router model with enhanced SOAP protocol is efficient; and the enhanced SOAP facilitates multimedia web service.

6. ASSESSMENT, CONTRIBUTIONS, AND FUTURE WORK

In this paper we present a network router model, which provides dynamic selection and binding of most efficient protocol to support QoS-based multimedia web services, while increase communication flexibility as well. The modularity of the model separates protocol registration from protocol binding. New protocols can be easily added to router. The novelty in this research is that we enhance SOAP definition so that a SOAP message could carry QoS requirements, which can be utilized by a router as a guidance to choose different paths and transportation protocols so as to increase performance. Our router model seamlessly integrates with SOAP technology therefore it is web service oriented. This router model further enhances our previous SOAP-oriented component-based framework supporting device-independent multimedia web service [10].

Our model requires routers to perform SOAP parsing to identify QoS requirements. However, in our experience, the benefits of dynamic intellectual protocol binding justify this additional effort. In addition, this paper concentrates on protocol selection and dynamic binding based on SOAP message QoS requirements on router to support multimedia QoS requirements. To guarantee multimedia QoS requests, efficient routing algorithms are inevitable.

We would like to continue our research work in the following directions. First, we would investigate QoS routing algorithms on routers to further support SOAP QoS requirements. Second, we would like to investigate caching and pre-fetch techniques to improve QoS performance. Third, we would like to pursue a formal description language to facilitate protocols to be published on the web and dynamically registered to routers. Finally, we would like to build up a mechanism to monitor multimedia QoS performances over different networks.

7. REFERENCES