THE INTERNATIONAL JOURNAL OF
SCIENCE & TECHNOLEDGE

Efficient Querying and Streaming of Multimedia Content Using Web Services

S. Gerald Christopher Isaac
PG Student, Department of Computer Science
Arunai Engineering College, Tiruvannamalai, Tamil Nadu, India

Abstract:
Web service standards provides a standard support for interoperability and reusability among various platforms, but in case of streaming multimedia data the current web service standards in terms of data transfer is not well supported. At present mostly Universal Description Discovery and Integration (UDDI) is used for publishing and discovery of multimedia web services providing a key word based search for the user. The web service standards used for delivering the multimedia streaming objects is also not sufficient due to very large object size and also the processing and delivering continuous stream of large scale data is difficult and time consuming to variety of client devices. This paper describes about the Service Oriented Architecture, in using an efficient querying mechanism which is specified in an extension of WSDL using the multimedia content metadata and its descriptions for querying the best service. Transmission of streaming data requires a message exchange protocol (MEP) to be defined and then implement the MEP in their SOAP HTTP bindings of the web service. Processing the demand for non functional requirements such as efficiency and security in streaming increases and the performance of SOAP is impacted by XML processing and communication overheads. To overcome these we go for various compression schemes for SOAP messages, so that the transfer overhead is reduced. By the new querying mechanism the efficiency of finding services can be improved as irrelevant data can be eliminated and by creating the MEP, efficiency is increased in terms of resource consumption.

Key words*: SOA; Web service; Multimedia streaming*

1. Introduction

Service Oriented Architecture (SOA) is a set of principles related to the growth of systems using abilities offered by services in a distributed environment. SOA give emphasis to the reusability, growth and interoperability of solutions developed as a composition of the distributed services. Building multimedia applications using service oriented architectures is done especially for the processing and delivery of continuous multimedia data streams. It has been a controversial topic since years, one of the familiar implementations of the SOA representation is web services [1]. Web services permit machines to communicate over the network on different platforms in a consistent manner, by means of a set of XML-based standards. Data is usually encapsulated and exchanged using SOAP, which is a lightweight protocol for transmitting structured information in an XML based format. In the Web service architecture a specified task can be attained by consuming a set of services from various different service providers. Using SOA concepts in [6] a complex multimedia system can be built from the composition of multiple small reusable components. Web services technology publishes the distributable data to make their services accessible to more other organizations. Web Services provide easily accessible services over a network. They should be simply usable regardless of the underlying network structure or configuration, operating system, communication mechanism or implementing language.

1.1. Objective

In this paper we efficiently find the web services regarding multimedia data and delivering them to the user by streaming the data and improving the performance in a secured and efficient manner. Querying the multimedia data is obtained by successful discovery of web services where only the relevant data is returned to the users according to the client configuration. Existing systems use UDDI and other mechanisms for finding the web services. Here WSDL is used in the keyword based retrieval. The interoperability of services is achieved when the service interfaces are well defined and described inside the WSDL which are published in service registries. These WSDL descriptions are applied to multimedia web services. The XML schema, which is the language used to describe data in WSDL, is not sufficient for the description of multimedia resources [9]. Issues in using multimedia metadata schemes are that the metadata created by these schemes are larger than the XML schema of an ordinary web service, the growth of the size of WSDL will make it unmanageable if all metadata information is stored within WSDL. We propose a new keyword based search mechanism where it uses the multimedia content metadata like the multimedia descriptions etc., as its base for retrieving the correct information thereby avoiding the irrelevant data.

After efficiently discovering the multimedia data, delivering the content to the user involves a process called streaming. A new proposal of creating SOAP based streaming content delivery is invoked where the process occurs two SOAP nodes over a HTTP protocol where the streaming message exchange pattern (MEP) is involved. Multimedia data contains large live data, so streaming is typically used as the delivery method. Here in this process multimedia web services have to transfer multimedia data entirely, which reduces the efficiency of the application because the total size of the data is not known in the beginning. Our proposed design removes the dependency on any external entity by streaming multimedia data encapsulated in SOAP messages. The transmission mechanism is created in a new HTTP SOAP binding [5]. Because the streaming data is transmitted within the web services architecture, the proposed design can be published, composed, and consumed just like any other web services. Finally the efficiency of XML based data communication is improved by using some compression techniques for SOAP messages. SOAP based web services is occupying high resource requirements for processing, which increases the vulnerability of various security attacks [7]. A number of compression schemes [10] such as generic text compressors, XML compressors, SOAP compressors, and binary XML representations have been studied to improve SOAP message processing and performance.

2. Related Work

Without SOA, processing and delivering continuous stream of large scale multimedia data is difficult, costly and time consuming in the present systems [2]. Difficulty lies in the nature of the data to be exchanged, limited and inadequate support for handling continuous data with web services. Here we go for selection of suitable workflow, then discovers an apt service that fit into the workflow and finally validate the interoperability. Another problem the authors face in [3], is delivering multimedia streaming in continuous form to a variety of client devices where it can be of many configurations such as transcoding constraints, device capabilities etc. Solutions in [4] is implemented to handle this is to adapt to a SOA framework where the extensions of WS are used for multimedia streaming and content adaptation for client devices. Efficiently finding the multimedia web services is one of the major studies involved in the discovery of web service. Relationships among various web services are complicated as users are confused by huge number of irrelevant returned web services where the intentions of the users in the returned web services are ignored. Another resource requirement in existing system is that the streaming multimedia data requiring a special streaming server, in place where the clients are required to have particular software installed to be able to connect to that server. The problem in this technique is that streaming multimedia data is difficult to maintain its interoperability and reusability because of the lack of standardized approach for streaming multimedia due to its size or its streaming nature. The solution to the above problem is that the process of streaming multimedia data through HTTP channel stays within the web service so that there is no need to maintain a separate streaming server. Authors in [8] prescribe another technique to improve streaming efficiency is by placing a proxy server, near the service requestors where the proxy server is capable of proving efficient caching between the requestor and the provider.

SOAP based web services is occupying high resource requirements for processing which increases the vulnerability for DOS attacks as the authors has described in [7]. This vulnerability can be avoided by a secured SOAP message where the WS-Security features are defined in that and it is processed in a streaming manner. It has the ability to detect malicious messages and make security processing for the streaming methods. In order to provide good performance the related work below has been described from various problems where one such problem is that, the performance of SOAP is impacted by XML processing and communication overheads. The solution proposed is to reduce the SOAP processing overheads and message sizes which give the advantage of repeatability between two end points. According to the authors in [10] using SOAP for application integration requires enormous demand for network bandwidth and causes more network traffic.

3. System Architecture

System design follows a general form of SOA where the service consumer or the user continuously searches and discovers the service in the service registry and consumes the service. The service provider publishes the service every time in the registry and provides the best available service needed for the consumer. The service provider holds the descriptions given to the multimedia data and it is used as a metadata for the multimedia content through metadata keyword based query services. The service provider also holds the multimedia web services needed for the consumer.

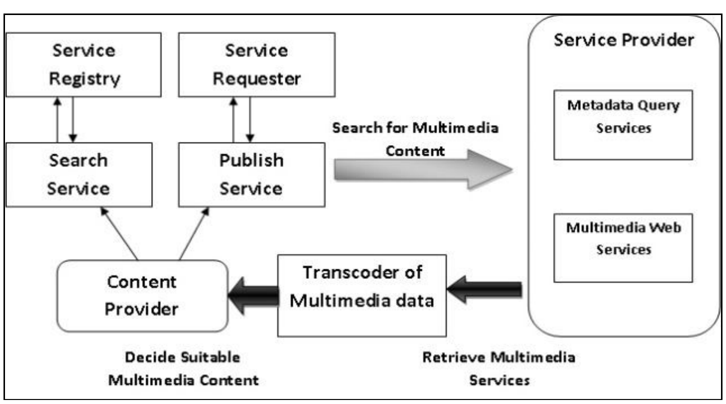

Figure 1. Architecture Diagram

The service registry generates a local identification every time for the generated web service. All web service files need to be registered at the service registry. All the inputs are to be converted to XML format using the transcoder. The transcoder converts or transcodes the given multimedia content from the web service to XML format, so that the transmitting content remains in XML format and the transfer protocol is SOAP HTTP binding. The content provider accepts the multimedia data from transcoder and provides the content to the clients and it also manages the web services between the users and the server. All communications take place in the form of SOAP.

4. Description for Multimedia Web service

Metadata can be used to describe the input and output of the multimedia content of a multimedia web service. The approach is to insert the metadata inside the description of a WSDL operation. Multimedia content description varies from time to time. If an item of metadata is changed, the corresponding web service has to review and publish its WSDL to all related service registries. This approach focuses on the path to select the correct service for the users. We describe a keyword based search where the keywords are the metadata descriptions to the multimedia content. The results are retrieved according to the user query and it is optimized further by applying some filtering techniques to get the results according to user intentions. A clustering algorithm is designed to eliminate irrelevant services based on the query. A technique called semantic analysis (SA) is utilized to capture the semantics behind the words in a query along with the descriptions in the services. The problem lies on how efficiently, we are going to discover the services on the web. Given a query in keyword search, eliminate the contents which are not compatible with users query by using the clustering algorithm where it is applied to the dataset to extract the semantics from the query.

```
<?xml version="1.0"?>
<wsdl:definitions name="MediaService"
  targetNamespace="http://example.org/media"
  xmlns:media="http://example.org/media"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:mmws="http://schemas.xmlsoap.org/wsdl/mmws"
  xmlns:xlink="http://www.w3.org/1999/xlink">
  . . .
  <wsdl:portType name="GetMediaPortType">
    <wsdl:operation name="GetMediaDesc">
      . . .
    </wsdl:operation>
    <wsdl:operation name="GetMedia">
      <wsdl:input message="media:GetMediaRequest"/>
      <wsdl:output message="media:GetMediaResponse"/>
      <wsdl:fault message="media:GetMediaFault"/>
      <mmws:metadata operation="media:GetMediaDesc" />
    </wsdl:operation>
  </wsdl:portType>
  . . .
</wsdl>
```

Figure 2. An example WSDL description

Clustering semantic approach (CSA) is the combination of the keyword technique and the semantics extracted from the service descriptions. The objective is to match the services at the semantic concept level. Firstly we eliminate irrelevant web services with respect to a query by using a modified clustering algorithm. After getting the dataset, we use semantic analysis to find a common semantic concept between web services and query so that service matching against the query can be carried out. The CSA approach is done to improve the efficiency of finding services if irrelevant data can be eliminated before extracting the semantics. CSA uses a dynamic algorithm that partitions a service dataset into smaller pieces. It includes the two main phases: Eliminating irrelevant services and matching services at semantic concept level. At first phase the irrelevant services are removed from the samples of services to form a working dataset. The main objectives in this phase are to reduce the initial size of service collection and also to manage the cost factor. At second phase semantic analysis approach is applied to the dataset for capturing semantic concepts. Here the web services are clustered into a finite number of semantically related groups where a set of matched services can be returned by comparing the similarity between the query and the related group. Find below the summary of the pseudo code for Clustering algorithm:

- Retrieve initial service samples
- Eliminate irrelevant services to form a dataset
- Apply SA to the dataset
- Semantic matching query with services in related Clustered group
- If the results match the query then go to step 8
- Else choosing next cluster goes to step 4
- End if
- End

5. Streaming Data Transfer

Streaming is the method of delivering the multimedia content over a network in real time. The end user receives the content constantly, with little or no delay. By streaming, a multimedia file is divided into small pieces which can be transmitted by it. The use of web service approach for the transmission of multimedia data is through the HTTP channel. We use the MEP and the SOAP HTTP binding of a MEP.

5.1. SOAP MEP for Streaming

The message exchange pattern (MEP) defines the model for SOAP messages between two SOAP nodes under a SOAP binding, which introduces two message exchange patterns. These are the Request- Response MEP and the Response MEP, the two patterns which are used for the standard implementation in exchanging SOAP messages between two end points. Streaming multimedia data divides the data into smaller data segments. When there is a request for a multimedia stream, the service responds with the multiple segments that are one request may correspond to more than one response. Based on this, we define a new MEP as shown in below Figure 3.

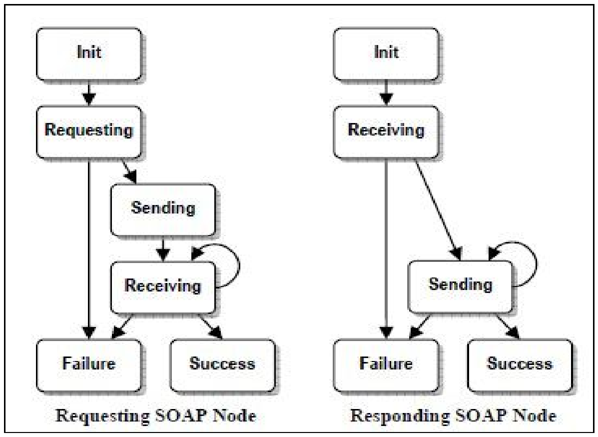

Figure 3. MEP state transition diagram for streaming

The left side of Figure 3 shows the transition lifetime of a requesting SOAP node. After the request is initialized the node sends the request to the responding node. To receive multiple responses it has to remain in the receiving state. The process of message exchange is completed on an exit message. The right side of Figure 3 shows the responding SOAP node to remain in the sending state so that multiple responses can be sent to the requesting node. This approach about the MEP enables the sender to send multiple multimedia data segments for a single request, with the receiver receiving these segments in the same response until the end of the stream.

5.2. SOAP HTTP Binding for Streaming

In SOAP HTTP binding the Response MEP and the Request-Response MEP can be made practical by using the HTTP GET method and the HTTP POST method respectively. In Response MEP a HTTP GET request is sent to the web server, on receiving the request the appropriate SOAP response will be returned inside the body of the HTTP response. In Request-Response MEP the SOAP node sends a SOAP request to the web server by embedding the request inside a HTTP POST request. Once a web server receives the POST request, an appropriate SOAP response will be generated and returned along the HTTP response. SOAP response is generated only from the web server.

```
POST /service HTTP/1.1
Host: lms.edu
Content-Type: application/soap+xml; charset=utf-8
Content-Length: nnnn

<?xml version='1.0'?>
<env:Envelope
  xmlns:env=
    "http://www.w3.org/2003/05/soap-envelope">
 <env:Body>
  <m:getMediaStream xmlns:m="http://lms.edu/">
   <m:sid>http://lms.edu/video.mpg</m:sid>
  </m:getMediaStream>
 </env:Body>
</env:Envelope>
```

Figure 4. An example SOAP request for a multimedia stream

Figure 4 demonstrates a SOAP request for a multimedia stream from a web service using the SOAP HTTP binding. The process of streaming MEP is the ability to put multiple SOAP responses in a single HTTP response. Using the multipart structure in the Multipurpose Internet Mail Extensions (MIME), these responses can be returned by a web server sequentially. Figure 5 shows the corresponding response for a multimedia stream request.

```
HTTP/1.1 200 OK
Content-Type: multipart/related;
              boundary=MIME_boundary

--MIME_boundary

...Data segment 1...
--MIME_boundary

...Data segment 2...
--MIME_boundary

...
```

Figure 5. A SOAP response of a multimedia stream using the extended SOAP HTTP binding

Segments in a multimedia stream are binary data as they cannot be enclosed in SOAP messages. The way to put binary data inside a SOAP message is by encoding the data from binary to text format. Using SOAP with attachment a single multimedia data segment can be transmitted as a SOAP message package. The package can include as many parts as possible like the primary SOAP message that contains the metadata of the data segment and a reference to the binary data and another part like the binary attachment of the data.

6. Soap Message Processing and Compression

Performance of SOAP is impacted by XML processing and communication overheads and the payload of the transmitted messages cause size overhead in time critical applications. We study about various messages processing and compression techniques such as generic text compressor to reduce the size of the SOAP message. A class of technique encodes XML messages using a binary encoded format and aim at minimizing the encoding size and at the same time processing speed of an XML message. We focus on the solution to reduce the SOAP processing overheads and message sizes which gives the advantage of repeatability. Ways to improve the SOAP processing firstly includes accelerating message composition of an outgoing message called the message pattern is built once and merged with the parameters for each new instance of messages. Secondly we reduce parsing overheads where parsing is done to SOAP message on a tree based method which is to identify the operation to be invoked at its parameters. Finally the SOAP message compression is a hop-by-hop optimization and that the enhanced protocol is stateful. This SOAP message processing and compression technique allows the integration of the above three mechanisms to achieve the overhead reductions and message compressions without latencies.

7. Conclusion

This paper presented an efficient web service framework for multimedia web services. A simple mechanism for publishing and discovering multimedia web services is described. A metadata keyword based query service is created to facilitate the description for the multimedia content. This querying technique must make sure that timely replacement of failing services is done. High cost in generating semantics in keyword search is a limitation to this approach. The streaming of multimedia data is the end result of two innovative streaming data MEPs and the implementations of the MEPs with SOAP HTTP. The HTTP protocol follows a simple request response pattern. Advantages in this method is memory resource consumption for streaming is lower and the newly created SOAP MEP the requesting node displays the process of streaming while it still receives the segments. Lack of support for continuous data streams in present web service infrastructure is a disadvantage. As a part of future work the QOS parameters also can be measured and controlled on a broader scale over the network. The new MEPs and SOAP bindings cause transfer overhead in the transmission of multimedia streaming data. To reduce the effect of this overhead, various SOAP performance techniques for the packaging of SOAP messages are discussed.

8. References

1. K. Nahrstedt and W.T. Balke, “A Taxonomy for Multimedia Service Composition,” Proc. 12th ACM Int’l Conf. Multimedia (MULTIMEDIA ’04), pp. 88-95, 2004.
2. I. Brunkhorst, S. Tonnies, and W.T. Balke, “Multimedia Content Provisioning Using Service Oriented Architectures,” Proc. IEEE Int’l Conf. Web Services, pp. 262-269, 2008.
3. S. Tonnies, B. Kohncke, P. Hennig, and W.T. Balke, “A Service Oriented Architecture for Personalized Rich Media Delivery,” Proc. IEEE Int’l Conf. Services Computing, pp. 340-347, 2009.
4. S. Decneut, F. Hendrickx, L. Nachtergaele, and S. Van Assche, “Targeting Heterogeneous Multimedia Environments with Web Services,” Proc. IEEE Int’l Conf. Web Services, pp. 682-689, 2004.
5. M. Gudgin et al., “SOAP Version 1.2 Part 1: Messaging Framework,”second ed., http://www.w3.org/TR/soap12-part1/, 2007.
6. J. Zhang and J.Y. Chung, “An Open Framework Supporting Multimedia Web Services,” Multimedia Tools and Applications, vol. 30, no. 2, pp. 149-164, Aug. 2006.
7. Gruschka, N ; Jensen, M. ; Iacono, L.L. ; Luttenberger, N, "Server-Side Streaming Processing of WS-Security", IEEE transactions on service computing, vol. 4, no. 4, October- December 2011.
8. M. Ruth, F. Lin, and S. Tu, “Adapting Single- Request/Multiple- Response Messaging to Web Services,” Proc. 29th Int’l Conf. Computer Software and Applications, pp. 287-292, 2005.

9. J.R. Smith and P. Schirling, “Metadata Standards Roundup,” IEEE Multimedia, vol. 13, no. 2, pp. 84-88, Apr.-June 2006.
10. C. Werner, C. Buschmann, and S. Fischer, “Compressing SOAP Messages by Using Differential Encoding,” Proc. IEEE Int’l Conf. Web Services, pp. 540-547, 2004.
11. Tekli, J.M ; Damiani, E. ; Chbeir, R. ; Gianini, G, "SOAP Processing Performance and Enhancement", IEEE transactions on service computing, vol. 5, no. 3, July-September 2012.
12. G. Lam and D. Rossiter, “A SOAP-Based Streaming Content Delivery Framework for Multimedia Web Services,” Proc. IEEE Asia-Pacific Conf. Services Computing, pp. 1097-1102, 2008.
13. Khoi Anh Phan; Tari, Z.; Bertok, P, "Similarity-Based SOAP Multicast Protocol to Reduce Bandwith and Latency in Web Services", IEEE transactions on service computing, vol. 1, no.2, April-June2008