IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Suite 200, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com
ITP4815

Contributions of Web Services-Technology for Improving Resource Management of Real-Time Multimedia on the Internet

Benny B. Nasution, Elizabeth A. Kendall, Asad I. Khan, and Jana Dospisil
School of Network Computing, Faculty of Information Technology, Monash University
McMahons Road, Frankston 3199, Australia
Ph: +61 3 9904 4287; Fax: +61 3 9904 4124
{benny.nasution, elizabeth.kendall, asad.khan, jana.dospisil}@infotech.monash.edu.au

ABSTRACT
Integration of real-time multimedia, telecommunication and computer technologies requires various considerations. It has been observed that a viable and reasonable solution is through improvements in resource management. Furthermore, Web Service Technology seems suitable as the facilitator for the heterogenous Internet community to improve scalability in the Internet. In fact, the main breakthrough in Web services technologies is that disparate applications of both users and providers can both use and be used by each other. This article will outline conceptual implementations for improvements in resource management by using Web Services technology to result in improvements in the resource management of multimedia on the Internet. The concept shows that a number of resource management approaches can be implemented using Web Services technology.

1. INTRODUCTION
Recent research in multimedia [2, 9-11] shows that the integration of multimedia, telecommunication and computer technologies requires various considerations and may in some ways worsen the quality of service and the manageability of resources. Correspondingly, due to the lateness of IP datagram for real-time exchanges, some experts believe that one of the main problems of running multimedia over the Internet is located in the IP layer. To overcome such problems, a number of researchers have increased the utility and the capability of computer caching technology [9]. Some others have advanced the traffic bandwidth, while others have tried to improve resource management [2, 10, 11].

After analysing these three approaches, it has been determined that a viable and reasonable solution, that would not incur significant additional costs, can be found through improvements in resource management. This paper will outline conceptual implementations of some of these ideas using Web Services technology. The aim is to provide improvements in resource management for multimedia on the Internet.

2. MULTIMEDIA ON THE INTERNET
Following rapid improvement in computer technologies and the inclusion of sound, vivid pictures (both on the screen and in print) and motion pictures, in various computing applications, the popularity of multimedia has increased rapidly. As a result, experts and researchers have developed numerous innovations that confirm that real-time multimedia will not only exist as technology for entertainment, but also for various commercially and socially relevant purposes, such as business transactions [5], education, health, government and defence.

However, the demand for innovations in multimedia may create additional problems, particularly if real-time media technologies are coupled with computing, networking and telecommunication technologies. Nevertheless, W3C [8] remarks that the convergence of those technologies already exists. In anticipation of potential obstacles, real-time multimedia applications on the Internet may require better Internet technologies, such as higher traffic bandwidth and faster computers, including specialised operating systems.

Those strategies seem, in contrast, to not be adequate to solve the problems, due to the IP packet bottleneck. Therefore, in this paper methodologies, such as those proposed by Buttazzo, et al. [2], Leis [9], Lienhart, et al. [10] and Shahabi and Banaei-Kashani [11], which propose new and better resource management techniques, have been evaluated. This article will propose outlines for the use of Web Services technologies as the implementations of these approaches.

3. WEB SERVICES
According to Clabby, et al. [4]:

> *Web Services are an evolving distributed computing architecture that uses its own program-to-program interfaces, communications protocols and a registry service to enable similar and disparate applications to communicate and perform "services" for one-another.*

Even though the framework of Web Services is still evolving, some promising Web Services technologies produced by leading software companies, such as IBM and Microsoft, are already on the computer market. Various reports [1, 4, 6, 7] show that the responses of both users and vendors to these quite new technologies are positive. Web service is aimed as the facilitator for a heterogenous Internet community to improve the scalability in the Internet. The main breakthrough in Web Services technologies is that disparate applications of both users and providers can both use and be used by each other.

The protocol that is currently available to implement this accessibility is called the Simple Object Access Protocol (SOAP). In order to be accessed, services must be first advertised and registered in a registry service, called the Universal Description, Discovery, and Integration (UDDI). The language for defining its specification is described in the Web Services Description Language (WSDL).

All three XML-format standards, SOAP, WSDL and UDDI, have been produced by W3C. Combining the capabilities of these three elements of Web Services, this paper shows that there is a possibility of

Copyright © 2003, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

deploying Web Services as improvements in resource management of real-time multimedia on the Internet.

4. WEB SERVICES' CONTRIBUTION FOR THE IMPROVEMENTS

Despite quite significant improvements in network bandwidth and network/computer devices, limitations and increasingly new demands of the Internet have been driving the need for real-time multimedia applications on the Internet to be handled more effectively and efficiently to support the quality of service (QoS) [5]. As a consequence, recent research related to resource management suggests three main approaches: efficient search/transfer method [10], elastic scheduling [2] and decentralised resources [11].

Efficient Search/Transfer Method

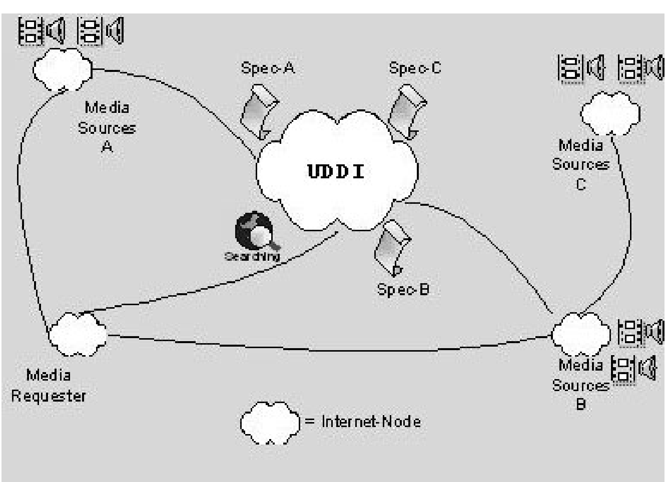
The idea behind this approach is that if the efficiency of services is to be improved, the way in which multimedia is handled should depend on the type of data, and on the characteristics of both the application-users' and application-providers' requirements. Lienhart [10] adds that this approach is necessary to enhance the quality of object-searching. Complete information of the media should then be kept in some kind of "rich content" [10] naming system and should be well distributed. To meet such requirements, Web Services, through its WSDL, may be an alternative solution. Figure 1 illustrates an example of the implementation.

WSDL uses XML format to describe its abstract functionality of services [3]. WSDL is also deployed as the framework for describing the details of the offers of distributed services [3]. A WSDL-standard XML-file may contain descriptions about the type, messages, portType, serviceType, binding, and services [3, 12] of an object. By using those WSDL-descriptions in UDDI as the implementation of "rich content" meta-information, a search on specific objects may be carried out more efficiently than running such a system locally. Therefore, an accurate delivery system may be able to be developed.

The following is an example of a WSDL modified script of Vasudevan [12]:

```
<?xml version="1.0"?>
<definitions name="MediaStock"
            targetNamespace="http://example.com/mediastock.wsdl"
            xmlns:tns="http://example.com/mediastock.wsdl"
            xmlns:xsd1="http://example.com/mediastock.xsd"
            xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
            xmlns="http://schemas.xmlsoap.org/wsdl/">
  <types>
        DELETED
  </types>
  <message name="GetMediumInput">
     <part name="body" element="xsd1:MediumRequest"/>
  </message>
  <message name="GetMediumOutput">
     <part name="body" element="xsd1:Medium"/>
  </message>
  <portType name="MediaStockPortType">
        DELETED
  </portType>
  <binding name="MediaStockSoapBinding"
           type="tns:MediaStockPortType">
        DELETED
  </binding>
  <service name="MediaStockService">
        DELETED
  </service>
  <binding name="MediaStockServiceBinding"
        DELETED
  </binding>
  <service name="MediaStockService">
        DELETED
  </service>
</definitions>
```

Figure 1: Implementation of Object-Information using WSDL-Specification



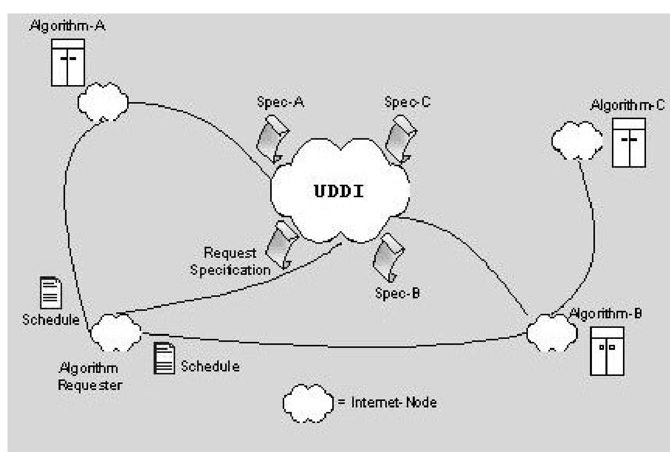
The "rich-content" [10] naming system can be completely specified in between <types> and </types>. The other parts are composed as an interface of the transaction between service providers and service users. The rest of this WSDL script is deleted to give a simple overview of the usage.

Elastic Scheduling

Similar to the previous approach, Buttazzo et al. [2], in their elastic scheduling approach, were inspired by the actuality that not all multimedia constantly require a very good quality of service. For example, entertainment presentations, such as motion pictures or sound, may be disturbed sporadically without causing significant or critical damage. Thus, they believe that, depending on the required QoS, the tasks' execution-schedule may be changed dynamically to accommodate the constraints of other concurrent tasks [2].

On the other hand, this approach needs an algorithm to calculate and to execute dynamic schedules of all tasks inside the local hosts. Since an additional algorithm such as a compressor [2] may also need

Figure 2: Implementation of remote Algorithm



Copyright © 2003, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

CPU resources, it may disturb the state of the computing schedules. Buttazzo, et al. [2] emphasise that such an algorithm must not be executed when there is a task running inside its critical section. Additionally, to analyse all the critical sections of the tasks, the algorithm seamlessly needs another computer resource.

This algorithm and additional supporting tools may be implemented through services in Web Services technology to reduce the complexity of the schedule and to save the usage of resources. Comparing with the time-usage and resource-usage of running the algorithm locally, the task of the local hosts in creating the required parameters in XML-files, and in expecting the value of the optimised calculated schedule back from a service, will require less computer resources and computing time. The kernel on the local hosts can subsequently reallocate the schedule accordingly.

The positive aspect of such a solution is that the implementation service may be written using an efficient programming language and may be executed on a high-speed machine. The following figure presents an example of the logical interaction schema.

Decentralised Resources

The approach to improving multimedia quality by using Web Services technology for decentralised resource management has been inspired by the work of Shahabi and Banaei-Kashani [11]. In their article, they reported that the common approach of hierarchical and centralised resource management might not be sufficient for the globally dispersed users of real-time media. To overcome the problems in minimising communication storage costs [11], they propose that the continuous media servers be distributed.

It is specified by Shahabi and Banaei-Kashani [11] that a type of middleware should be employed for the resource management. For this task, UDDI would be a practical solution for the storage of the service requirements. UDDI can then be accessed by both service providers and service users. In case more than one service exists, some services may act as an efficient service locator, which will always return the best solution. Therefore, this service locator will logically be equipped with a statistical algorithm that can watch the dynamism of services offered in UDDI.

Accordingly, other services may offer resource storage for media providers. Thus, media providers can use a resource-locator first to analyse the best locations for distributing their media efficiently to their potential clients, before undertaking transactions with the resource storage providers. Subsequently, additional consideration, such as storage costs and capacity may need to be addressed. Even in the situation where some resource storage providers have changed its location or address, through changing the specification on the UDDI the media providers can move their media to other resource providers dynamically.

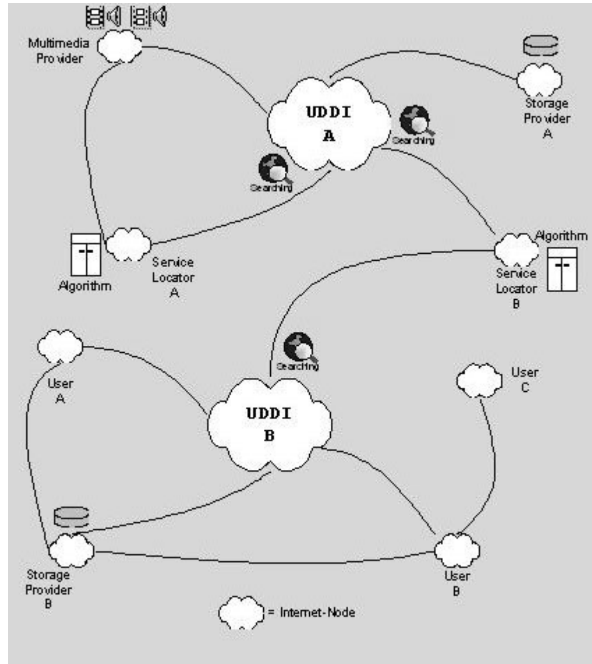
Figure 3 shows that a multimedia provider may obtain statistical information about potential active users A, B and C from Service Locator B. Since the multimedia provider does not have a good (direct) link to those users, it will get additional information from Service Locator B, and the result may say that the best position to put the media is on storage provider B. At the end of the transaction, there will be a new specification registered on UDDI B about the media service located at Storage Provider B which provides direct connection to Users A and B and a two hop connection to User C.

People may argue that the meeting point (directory service) can become the bottleneck of the whole transaction. However, due to the capability of Web services to improve scalability, dynamism and diversity, Web services will attract ongoing service participants who will offer various services and even UDDI platforms. The more services involved in the whole system of Web services, the more scalable and robust the multimedia transactions from the Web services will be.

Other Approach

Unlike the previously mentioned approaches to resource managements, the approach related to caching technology [9] may also be leveraged by using Web Services technology. For example, a service which is advertised by a P2P virtual-disk may implement Leis' approach [9] of pre-fetching forthcoming pages. This service may be capable of doing such tasks, due to the effectiveness of P2P characteristics, for example small online latency [9] and simpler overhead structure. Furthermore, the latency will be much smaller if the P2P host is near to the user or both hosts are connected through a high-speed link.

Figure 3: Implementation of Decentralised Resource Management



5. SUMMARY

Web services offer a new kind of flexibility in implementing distributed systems on the Internet. Since each and every service needs only to describe its services in a highly structured XML-format, and the real implementation can be written in any format and on any platform, its contribution in some network management problems can become significant. To solve efficient searching for appropriate objects, WSDL not only offers the service name but also has a number of detailed specifications of all services, which may give more accurate information about an object.

Similarly, to use computer resources efficiently, an application can transfer some tasks to services only through detailed specifications packed in an XML file. After finishing the task, the service just sends the detailed result in an XML format to the requester. Sending and receiving two XML files, instead of running the task-algorithm locally, can reduce computing time and, therefore, increase the QoS.

The more services that are involved in Web services, the more possibilities will exist for solving decentralised resource management. As a result, object placement and object delivery [11] requirements can be distributed to disparate services. The characteristic of Web services' loose-coupling connection makes for a straightforward implementation.

6. FUTURE WORK

As the real transactions running through Web Services are using XML format only, additional consideration will need to be given to security requirements. This especially applies to the confidentiality of objects for critical applications, such as confidential company conferences or defence purposes.

Since most multimedia applications do not need a live connection to the media resource, the latency during setup/transaction time compared to the long duration of the real media, and the data header-rate

Copyright © 2003, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

compared to data content-rate, is generally insignificant [11]. For mission critical purposes, such as a surgery, however, defining efficient WSDL interactions to reduce overheads and negotiation-time is necessary.

7. REFERENCES

[1] Atkinson, B., et al., *Web Services Security (WS-Security)*. 2002. p. 1-22.

[2] Buttazzo, G.C., et al., *Elastic scheduling for flexible workload management*. Computers, IEEE Transactions on, 2002. **51**(3): p. 289-302.

[3] Chinnici, R., et al., *Web Services Description Language (WSDL) Version 1.2*, in *http://dev.w3.org/cvsweb/~checkout~/2002/ws/desc/wsdl12/wsdl12.html*. 2002. p. 1-31.

[4] Clabby, J., et al., *Web Services Gotchas*. 2002, Bloor Research N.A.: Yarmouth, ME. p. 1-109.

[5] Conti, M., et al., *Quality of service issues in Internet Web services*. Computers, IEEE Transactions on, 2002. **51**(6): p. 593-594.

[6] Hondo, M., et al., *Security in a Web Services World: A Proposed Architecture and Roadmap*. 2002. p. 1-20.

[7] IONA-Technology, *Enterprise Security in Web services*. 2002. p. 1-32.

[8] Jacobs, I., *About the World Wide Web Consortium (W3C)*. 2000.

[9] Leis, J., *Rapid display of Web content: a simple method for prefetching Web files*. Computing & Control Engineering Journal, 2002. **13**(3): p. 149-152.

[10] Lienhart, R., et al., *Improving media services on P2P networks*. Internet Computing, IEEE, 2002. **6**(1): p. 73-77.

[11] Shahabi, C. and F. Banaei-Kashani, *Decentralized resource management for a distributed continuous media server*. Parallel and Distributed Systems, IEEE Transactions on, 2002. **13**(7): p. 710-727.

[12] Vasudevan, V., *A Web Services Primer*. 2001, XML.com.

Copyright © 2003, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

0 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:
www.igi-global.com/proceeding-paper/contributions-web-services-technology-improving/32040

Related Content

An Empirical Analysis of Antecedents to the Assimilation of Sensor Information Systems in Data Centers
Adel Alaraifi, Alemayehu Mollaand Hepu Deng (2013). *International Journal of Information Technologies and Systems Approach (pp. 57-77).*
www.irma-international.org/article/empirical-analysis-antecedents-assimilation-sensor/75787

Research on the Clustering of Emotional Elements in Art and Design Based on Visual Language Communication
Xia Duand Shahbaz Ahmad (2024). *International Journal of Information Technologies and Systems Approach (pp. 1-15).*
www.irma-international.org/article/research-on-the-clustering-of-emotional-elements-in-art-and-design-based-on-visual-language-communication/352040

An Adaptive Curvelet Based Semi-Fragile Watermarking Scheme for Effective and Intelligent Tampering Classification and Recovery of Digital Images
K R. Chetanand S Nirmala (2018). *International Journal of Rough Sets and Data Analysis (pp. 69-94).*
www.irma-international.org/article/an-adaptive-curvelet-based-semi-fragile-watermarking-scheme-for-effective-and-intelligent-tampering-classification-and-recovery-of-digital-images/197381

Fuzzy Logic in Medicine
Michelle LaBrundaand Andrew LaBrunda (2010). *Breakthrough Discoveries in Information Technology Research: Advancing Trends (pp. 218-224).*
www.irma-international.org/chapter/fuzzy-logic-medicine/39583

Big Data Time Series Stream Data Segmentation Methods
Dima Alberg (2018). *Encyclopedia of Information Science and Technology, Fourth Edition (pp. 364-372).*
www.irma-international.org/chapter/big-data-time-series-stream-data-segmentation-methods/183750