Chapter 11

Efficient and Effective XML Encoding

Christian Werner  
University of Lübeck, Germany

Carsten Buschmann  
University of Lübeck, Germany

Ylva Brandt  
University of Lübeck, Germany

Stefan Fischer  
University of Lübeck, Germany

ABSTRACT

Compared to other middleware approaches like CORBA or Java RMI the protocol overhead of SOAP is very high. This fact is not only disadvantageous for several performance-critical applications, but especially in environments with limited network bandwidth or resource-constrained computing devices. Although recent research work concentrated on more compact, binary representations of XML data only very few approaches account for the special characteristics of SOAP communication. This chapter will discuss the most relevant state-of-the-art technologies for compressing XML data. Furthermore, it will present a novel solution for compacting SOAP messages. In order to achieve significantly better compression rates than current approaches, the compressor described in this chapter utilizes structure information from an XML Schema or WSDL document. With this additional knowledge on the "grammar" of the exchanged messages, this compressor generates a single custom pushdown automaton, which can be used as a highly efficient validating parser as well as a highly effective compressor. The main idea is to tag the transitions of the automaton with short binary identifiers that are then used to encode the path through the automaton during parsing. The authors’ approach leads to extremely compact data representations and is also usable in environments with very limited CPU and memory resources.

DOI: 10.4018/978-1-61520-684-1.ch011
INTRODUCTION

The text-oriented data encoding of XML is the reason for SOAP messages causing significantly more overhead than the binary message formats of Java RMI and CORBA. In an earlier work we compared different approaches for realizing Remote Procedure Calls (RPC) and showed that SOAP over HTTP causes significantly more traffic than similar technologies. Using SOAP the data volume is about three times higher than with Java RMI or CORBA (Werner et al., 2007).

Fortunately, most of today’s wired networks are fast enough to provide sufficient bandwidth for all applications. However, there are still some application domains with tight bandwidth limits: For example, in cellular phone networks it is still common to charge the customer according to the transmitted data volumes. Another very common example is a dial-up connection over older technologies like modem or ISDN links. Although their bandwidth is very limited they are still in use in many enterprise networks. Additional limitations are imposed by foreseen application domains of web services such as Ubiquitous Computing. In such energy-constrained environments the radio interface is usually a main power consumer and therefore tight restrictions apply to the transmitted data volumes on mobile devices.

In order to address the problem of excessive XML message sizes in these domains a lot of research effort went into the development of binary (and therefore more compact) representations of XML data.

In order to preserve the universal compatibility of binary encoded XML, standardization is a very important issue: The W3C founded the W3C XML Binary Characterization Working Group in March 2004. Its members analyzed various application scenarios and created a survey of the existing approaches in this field (W3C, 2005a). Furthermore, this working group has specified a set of requirements that are important for binary XML representations. The most requested features are compactness, the possibility of directly reading and writing the binary format, independence of a certain transport mechanisms, and processing efficiency.

Another major outcome of this working group was a set of 18 typical use cases for binary XML representations with a detailed analysis of their individual requirements. In all use cases the property “compactness”, which is in the focus of this article, was of major importance or has been rated as a nice-to-have feature. In ten of 18 cases it was even rated as mandatory.

The W3C XML Binary Characterization Working Group has finished its work and its successor, the Efficient XML Interchange Working Group (W3C, 2005b), has taken up work in December 2005. It focuses on interoperability aspects of binary XML and has published its current working draft in September 2008. It describes the Efficient XML Interchange (EXI) Format (W3C, 2008) which we discuss in the section on related work.

In this article we elaborate on how to encode SOAP messages efficiently. We exploit the fact that web service messages are described by an XML grammar that is known to both the sender and the receiver (usually in form of a WSDL file). A large part of the information contained in a message can be inferred from this grammar. This “a priori” known part can therefore be omitted during transmission; this leads to very promising compression results.

Although the idea of creating web service-specific compressors from WSDL descriptions is not new and has already been presented by the authors in a previous publication in this book series (Werner et al., 2007), considerable advances are presented here: The original idea was to create a set of “empty” web service messages, called skeletons, containing all XML constructs that reoccur in subsequent service calls. When a service is called only the differences between the message and the corresponding skeleton is transmitted over the network. We could show that this differential
Related Content

Seamlessly Securing Web Services by a Signing Proxy
[www.igi-global.com/article/seamlessly-securing-web-services-signing/3046?camid=4v1a](www.igi-global.com/article/seamlessly-securing-web-services-signing/3046?camid=4v1a)

Web Service Enabled Online Laboratory
[www.igi-global.com/chapter/web-service-enabled-online-laboratory/59931?camid=4v1a](www.igi-global.com/chapter/web-service-enabled-online-laboratory/59931?camid=4v1a)

Interoperability and Functionality of WS-* Implementations
[www.igi-global.com/article/interoperability-functionality-implementations/74704?camid=4v1a](www.igi-global.com/article/interoperability-functionality-implementations/74704?camid=4v1a)

The Development, Testing, and Deployment of a Web Services Infrastructure for Distributed Healthcare Delivery, Research, and Training
[www.igi-global.com/chapter/development-testing-deployment-web-services/26072?camid=4v1a](www.igi-global.com/chapter/development-testing-deployment-web-services/26072?camid=4v1a)