Chapter V

Advanced Data Compression Techniques for SOAP Web Services

Christian Werner, University Lübeck, Germany
Carsten Buschmann, University Lübeck, Germany
Stefan Fischer, University Lübeck, Germany

Abstract

A major drawback of using SOAP for application integration is its enormous demand for network bandwidth. Compared to classical approaches, like Java-RMI and Corba, SOAP messages typically cause more than three times the network traffic. In this chapter we will explore compression strategies and give a detailed survey and evaluation of state of the art binary encoding techniques for SOAP. We also introduce a new experimental concept for SOAP compression based on differential encoding, which makes use of the commonly available WSDL description of a SOAP Web service. We not only conduct a detailed evaluation of compression effectiveness, but also provide the results of execution time measurements.
Introduction

Like all other XML protocols, SOAP suffers from the fact that only a very small part of the transmitted message contains real payload. The rest of it is XML markup and protocol overhead. Comparisons on different approaches for realizing remote procedure calls (RPC) have shown that SOAP, over HTTP, uses significantly more bandwidth than competitive technologies (Marahrens, 2003; Tian et al., 2003). For our experiments, we implemented a simple RPC server and client on different platforms (MS .Net, Apache Axis, Corba, JavaRMI, RMIIOP). Then, we measured the resulting network traffic for each case using the Ethereal network analyzing utility.

Figure 1 summarizes the results. For all implementations the number of transmitted bytes increases with the number of transmitted RPC messages in an almost linear way (all values do not include overhead for protocols on network layer and below). There is virtually no difference in the number of transmitted bytes between Microsoft’s SOAP Implementation and Apache Axis. Both cause three times more network traffic than JavaRMI and Corba.

For the case of only one message \((n = 1)\) the SOAP implementations cause the smallest amount of traffic: 1,972 bytes (Java), 1,976 bytes (SOAP .Net), 2,626 bytes (RMI), and 2,887 bytes (Corba). For all cases with more than one message the two SOAP implementations perform worse than RMI, RMIIOP and Corba. The reason for this is, that unlike SOAP, these implementations exchange information about naming service before sending the first RPC message, causing a relative high traffic offset.

Figure 1. Transmission of random strings \((l = 250\) Bytes)