



XML Data Binding for C++ Using Metadata

Szabolcs Payrits, Eötvös Loránd University, Hungary

Péter Dornbach, Nokia Research Center, USA

István Zólyomi, Eötvös Loránd University, Hungary

ABSTRACT

Mapping XML document schemas and Web Service interfaces to programming languages has an important role in effective creation of quality Web Service implementations. The authors present a novel way to map XML data to the C++ programming language. The proposed solution offers more flexibility and more compact code that makes it ideal for embedded environments. The article describes the concept and the architecture of the solution and compares it with existing solutions. This article is an extended version of the paper from ICWS 2006. The authors include a broader comparison with existing tools on Symbian and Linux platforms and evaluate the code size and performance. [Article copies are available for purchase from InfoSci-on-Demand.com]

Keywords: C++, Metadata, XML, Data Binding, Web Services

INTRODUCTION

Web Service technology builds on simple foundations, like HTTP (Fielding et al., 1999), XML (Bray, Paoli, Sperberg-McQueen, Maler, & Yergeau, 2004), and SOAP (Gudgin, Hadley, Mendelsohn, Moreau, & Nielsen, 2003); yet such systems can easily become quite complex. Hence, most Web Service consumer and provider implementers prefer to use a toolkit that helps the implementation. In fact, good Web Service tools can decrease the implementation effort significantly so that application

developers are able to concentrate on the real work (business logic).

A Web service toolkit typically provides a mapping of the Web service interface *types* and *operations* to the application development environment and programming language. Web service interfaces are commonly described in WSDL (Christensen, Curbera, Meredith, & Weerawarana, 2001), which also implies that XML data types are described in an XML Schema Description (XSD, Biron & Malhotra, 2004; Thompson, Beech, Maloney, & Mendelsohn, 2004). Such a mapping can simplify the work of developers enormously: it provides structures

in a well-known language so that developers do not have to understand WSDL and XSD and it eliminates the need to manipulate XML documents directly and enforces conformance to the schema automatically (White, Kolpackov, Natarajan, & Schmidt, 2005).

Some development environments have a standard mapping. For example, Java defines the Java XML Binding (JAXB, Fialli & Vajjhala, 2003) that is well-supported in the Java environment. Microsoft .NET and, more recently, Windows Communication Foundation (Microsoft, 2006) defines a similar binding that hides XML data completely in most cases. These environments usually exploit the complete type reflection mechanism provided by the underlying virtual machine.

However, there are cases where the use of a virtual machine is inappropriate; this is often the case in mobile and embedded systems where small binary code size and memory footprint is crucial. The C++ programming language is widely used in such environments, for example in the Symbian smartphone platform and in Linux-based devices. This article focuses on creating an efficient and compact binding in C++. The proposed toolkit is ideal for mobile devices and may offer advantages in other C++ environments as well.

The remainder of the article is organized as follows: The next section explains the motivation of this work and investigates related tools. We then describe the concept of the proposed binding; and explain the architecture in detail. Remaining sections take a look at compatibility issues and compare the tool with existing solutions, including XML Schema conformance, code size, and performance.

This article is an extended version of the paper with the same title by Payrits, Dornbach, and Zólyomi (2006). Since the original paper, the tool has been improved significantly to handle more complicated XML Schema constructs. Besides the existing Symbian version, a standard C++ version was created that can be compared more deeply with a wider variety of tools. The "Results" section describes the new

code size and performance results on Linux on Symbian.

MOTIVATION

There are many open source and proprietary XML data binding tools available for C++ but they vary in principles and architecture. Our main goal was to create a layered architecture where the C++ mapping of data types is separated from the underlying XML InfoSet implementation as much as possible.

Another important goal was that we preferred data members in generated C++ structures, because use of accessor methods would have meant a significant increase in code size. In correlation with this, we wanted to minimize the amount of mapping-specific code and move as much logic to a mapping-independent library as possible.

Finally, we also desired our architecture to be independent from the underlying C++ type system as much as possible in order to be able to support multiple mobile application development environments relatively easily, for example Symbian and Linux.

Related Work

According to Sourceforge (2006), gSOAP is an open source XML Schema and WSDL mapping implementation for C and C++. It is specifically advised to be used in embedded and mobile devices (van Engelen, 2004) because of its compact code size and standard-compliance. C code generated by gSOAP can be used in a wide variety of embedded environments including Symbian and PalmOS; gSOAP uses value objects to represent XML data but its serialization architecture is not layered as it is coupled with its own parser and XML InfoSet representation. In fact, gSOAP is the only tool that can be compared directly with the solution proposed in this article.

15 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/article/xml-data-binding-using-metadata/34104

Related Content

Aspect-Oriented Framework for Web Services (AoF4WS): Introduction and Two Example Case Studies

Ghita Kouadri Mostefaoui, Zakaria Maamar and Nanjangud C. Narendra (2009). *Managing Web Service Quality: Measuring Outcomes and Effectiveness* (pp. 245-263).

www.irma-international.org/chapter/aspect-oriented-framework-web-services/26082/

Context Dissemination in Peer-to-Peer Networks

Antje Barth, Michael Kleis, Andreas Klenk, Benoit Radier, Sanaa Elmoumouhi, Georg Carle and Mikael Salaun (2010). *Developing Advanced Web Services through P2P Computing and Autonomous Agents: Trends and Innovations* (pp. 69-90).

www.irma-international.org/chapter/context-dissemination-peer-peer-networks/43648/

Mashup Service Recommendation Based on Usage History and Service Network

Buqing Cao, Jianxun Liu, Mingdong Tang, Zibin Zheng and Guangrong Wang (2013). *International Journal of Web Services Research* (pp. 82-101).

www.irma-international.org/article/mashup-service-recommendation-based-on-usage-history-and-service-network/108883/

Model-Based Development and Spatiotemporal Behavior of Cyber-Physical Systems

Peter Herrmann, Jan Olaf Blech, Fenglin Han and Heinz Schmidt (2019). *Innovative Solutions and Applications of Web Services Technology* (pp. 69-93).

www.irma-international.org/chapter/model-based-development-and-spatiotemporal-behavior-of-cyber-physical-systems/214832/

Ubiquitous Provision of Context-Aware Web Services

Stephen J.H. Yang, Jia Zhang and Irene Y.L. Chen (2007). *International Journal of Web Services Research* (pp. 83-103).

www.irma-international.org/article/ubiquitous-provision-context-aware-web/3110/