Automatic XML Schema Generation UML Application Profile

Dr. Eda Marchetti
Istituto di Scienze e Tecnologie dell’Informazione “A. Faedo” - CNR - Pisa
Area della Ricerca di Pisa
via Moruzzi, 1, 56124, Pisa (Italy)
Phone: +39 050 315 3467
Fax: +39 050 315 2924
e-mail: eda.marchetti@isti.cnr.it

Automatic XML Schema Generation UML Application Profile

Eda Marchetti
ISTI - CNR - Pisa
Via Moruzzi, 1, 56124,
Pisa (Italy)
eda.marchetti@isti.cnr.it

Summary

Several authors have recently proposed to represent XML metadata by means of the widely used UML, which provides a more intuitive and expressive description, which can then be automatically mapped to XML Schema. Following this trend, and considering the e-learning context a prototyped tool developed for this purpose is presented.

Zusammenfassung

Verschiedene Autoren schlugen unlängst vor, XML Metadata mitden Mitteln der weithin genutzten UML darzustellen, welche eine intuitivere und ausdrucksstärkere Beschreibung ermöglicht. Diese kann wiederum automatisch auf XML Schema abgebildet werden.Diesem Trend folgend, und den e-Learning Kontext berücksichtigend, wird der Prototyp eines Tools vorgestellt, welches diesem Zweckedient.

Keyworkds

Interoperability, Application profile, XML Schema, UML profile

Introduction

Recent years have seen a growing interest into the e-Learning domain, thanks also to the rapid advances of the Information and Communication Technology (ICT), which have made available new powerful and low-cost means to provide on-line and interactive training services. At the same path with this growth a vivid and urgent need of interoperability between diverse e-learning tools, resources and educational system arose. This implies the capability of independently developed software components to exchange information, or the ability — anywhere, anytime is the current slogan— to discover, query and access resources drawn from different sources. To ensure exchange of information across applications, and common interpretations between users, standardization of the terminology and encoding and agreement on the format and vocabulary of data exchanged are needed.

Thus accredited organizations like IEEE LTSC (LTCS, 2005), W3C (W3C, 2005), IMS (IMS, 2005) and ADL (ADL, 2005), are currently publishing technical specifications, with the intent of satisfing the requirements of the largest number of users. However these standard specifications often remain at a very abstract level, so developers from different sectors have always been using diverse metadata elements sets, adding elements to existing sets and redefining the semantics of existing elements in the context of specific applications.

Recently for better satisfing a community exigencies the notion of an Application Profile (AP) has been defined. In literature diverse definitions of AP exist, but all refer mainly to the adaptation, constraining and/or augmentation of one or more reference metadata schemas, forming the so-called base specification. Thus the usage of APs increases level of interoperability, because changes and extensions to the reference metadata schemas are established in advance and made explicit, and facilitates and supports the selection and reuse of elements inside existing profiles, rather than their development from scratch. Specifically the modifications allowed by an AP to the base schema involved: change of cardinality of elements in the base schema; modification of attributes occurrence/multiplicity and cardinality required, optional or prohibited, or even supplying a default value; substitution of simple value types of attributes or elements with newly defined types or replacing them with fixed values; addition of elements or attributes at locations where the base schema provides appropriate extension points.

Generally the schema language adopted for expressing APs is the XML (eXtensible Markup Language) Schema (Fallside and Walmsley, 2004),which is used for defining the rules that map the profile elements onto lower-level objects, with their types, attributes and relations. However XML Schema is a quite low level abstraction of managed resources useful to tools for recognizing structured XML data, but hardly manageable and readable by the involved stakeholder .

In this direction, Routledge, (2002) have pioneered work towards allowing developers of metadata to use the more intuitive and expressive UML notation, and then providing them with an automated translation to the XML schema. The Unified Modeling Language (UML, 2005) is in fact a general and widely used notation, with many supporting tools available for editing and analysis. UML is increasingly being used in a variety of applicative domains. Also IMS, which represents a large community inside the e-Learning domain, has recognized the popularity and power of the UML, and in fact it has recently moved to expressing its specifications in UML. The idea then is that the XML binding can be automatically generated from a UML representation of the Base Schema information.

Following this trend, and recognizing the importance of ensuring the isomorphic capability to go back and forth between the UML and the XML Schema representations, this paper focuses on automating the translation of APs from UML into XML. For this a prototype tool, called U2X APTLY ((UML to XML-Schema Application Profile TransLation with stYle sheet), has been developed and is later outlined.

At the basis of the automatic translation of the AP into XML is the concept of UML profile (UML, 2005). Using this facility the UML elements and extension mechanisms for representing all the aspects of an AP and a Base Schema are provided[1]. Without going deep into specific details, the UML Profile considered has been defined considering the work of Carlson (2001) and Routledge (2002), extended for including those aspects of XML Schema specifically for the e-learning context.(A complete description of stereotypes used can be found in Bertolino (2004)). Thus from the Class Diagrams of Base Schema and Application Profile an automatic binding to XML Schema can be obtained. The adoption of an automatic mapping mechanism represents an interesting facility for translating the UML model into a XML Schema, which can be generalized and specialized for any purpose and for different environments. Any possible modification in fact can be introduced and implemented at UML level, avoiding the complexity of managing huge number of XML lines of code.

[1] an AP which requires no modification/extension to the base specification represents the specification itself, thus UML profile defined can be used straightforwardly to also represent Base Schemas.

Automatically Deriving XML Schema from UML Profile

A prototype tool, called U2XAptly (**U**ML **to** **X**ML-Schema **A**pplication **P**rofile **T**rans**L**ation with st**Y**le sheet) (Bertolino, 2004), has been developed for the automatic generation of XML Schema from the UML Application Profile. As shown in Figure 1, this tool manipulates the XML Metadata Interchange (XMI) (XMI, 2005) exported model by means of Extensible Stylesheet Language (XSLT) (XSLT, 2005) for deriving the XML Schema model. More precisely, taking in input the UML-based AP, exported into the XMI 1.2 format by the Poseidon UML case tool (Poseidon, 2005), U2XAptly maps the XMI model into the XML Schema model via a XSLT processor.

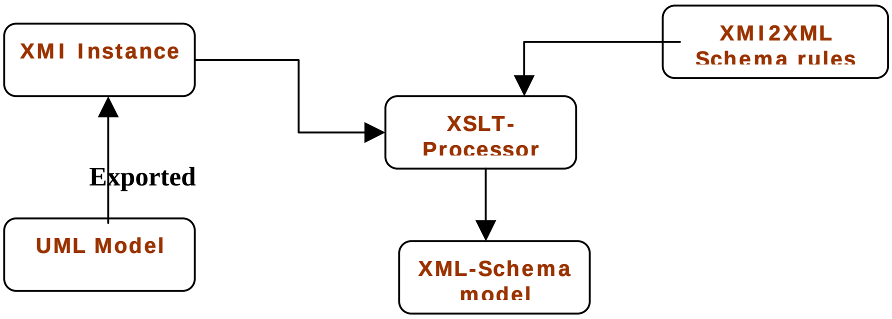

Figure 1 U2XAptly tool structure

Different global structures of APs can be defined. Without loose of generality, this paper explicitly considers that used inside the TELCERT project (TELCERT, 2005)) and currently under consideration by IMS. Within the TELCERT project different kind of XMI information and constraints must be managed for correctly mapping the derived XMI documentation into the XML-Schema model, like stereotypes, dependences, generalizations, multiplicity, constraints.

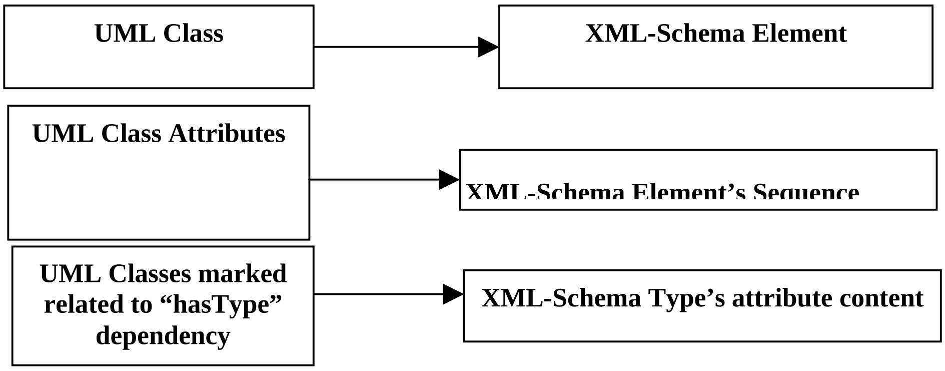
UML Class
XML-Schema Element
UML Class Attributes
XML-Schema Element's Sequence
UML Classes marked related to "hasType" dependency
XML-Schema Type's attribute content

Figure 2 Simple mapping mechanism

Ad Hoc UML features (for instance classes' stereotypes, parent associations, "Has Type" relations), have been introduced for making the mapping process simple and quick to implement. These reduce exponentially the complexity related to XML-Schema element generation. So, starting from a graphic view (UML) the XML Schema is realized by transforming the UML class into a <*xs:element*>, and each attribute into a xs:element's *sequence*, *union* or *group*. The obtained XML Schema model is then visualized. Figure 2 resumes the main details of this mapping process, further information are in Bertolino (2004).

For mapping the XMI to XML Schema a style-sheet collecting XSLT Schema rules is defined. This is in particular organized in different templates, including: a testing template for verifying the XMI version, a XML schema preamble template used for generating the XML

Schema and the namespaces declaration; auxiliary templates for: generating XML-Schema element related to attributes (-less) classes; creating element details, (e.g.: for stripping undesired characters, testing which attributes to print and so on).

Case Study

U2XAptly has been used for XML binding of an example AP in which a set of simple modifications are specified with respect to the IMS Question & Test Interoperability QTILite (QTILite 2002) Information Model.

Very briefly, IMS Question & Test Interoperability (QTI, 2004) specification has been designed for facilitating interoperability between different e-learning systems, and describes a datamodel for representing questions and test data and their corresponding results. QTILite is a subset of the QTI, which does not support all the features of the full QTI. However it provides a well-documented content format and a valid development support with enable also the report of test results. From a practical point of view the QTILite (as the QTI) is a datamodel for representing both the question and test data so information exchange between content providers, content management tools, assessment delivery systems and learning system is enabled (QTI, 2004).

The IMS does not provide a UML specification of the Base Schema, but only its XML Schema description. Thus, on the basis of the developed UML Profile, two UML Class diagrams representing the QTILite Information Model (Base Schema) and the AP, respectively, were derived. Using the information of these Class Diagrams, the tool automatically derives the respective XML binding, highlighting the specific changes/extensions of UML AP with respect to the UML Base Schema.

Conclusions

In many application domains where information resources are shared and routinely exchanged in the form of standard metadata, Application Profile is emerging as an important means to adapt the generic metadata schemas to the specific exigencies of a community. The XML Schema definition language is becoming the universally adopted notation for specifying the metadata, but it does it at a very detailed level, hence the exigency of expressing above it the metadata Base Schema in a more readable and manageable format arises. The UML is currently proposed by many as a convenient notation for this purpose, due to its graphical flavour and its high expressiveness power, and several authors have already proposed mappings from XML Schemas to UML models. In this context, the contribution of this paper is a prototype tool called U2Xaptly for translating directly the UML description of Application Profile into XML Schema. The usage of this tool has been illustrated in a simple case study of profiling the QTILite (QTILite 2002) Information Model.

Acknowledgements

This work has been financially supported in part by the European project TELCERT (FP6 STREP 507128) Technology Enhanced Learning Conformance - European Requirements and Testing. The author would like to thank the partners of the above project for their useful hints and feedback, and in particular Prof. Ingo Dahn for his very fruitful suggestions. Federica Ciotti and Sara Santoponte contributed to the UML profile definition and to the U2XAptly tool development, respectively. Antonia Bertolino and Andrea Polini have been a valid and irreplaceable help throughout the process

References

Bertolino, A. (2004, November) *Initial Recommendations on Advantage Testing Technologies. D09.* Retrieved April 30, 2005, from http://www.imsglobal.org/telcert.html

ADL (2005) *Advanced Distributed Learning*. Retrieved April 30, 2005, from http://www.adlnet.org/index.cfm

Carlson D. (2001) Modeling XML Applications with UML: Practical e-Business Applications. Addison Wesley

Fallside D.C, & Walmsley P. (2004, October) *XML Schema Part 0: Primer Second Edition*, Retrieved 28 October 2004 from http://www.w3.org/TR/xmlschema-0/

IMS (2005) IMS Global Learning Consortium Retrieved April 30, 2005, from http://www.imsglobal.org/

LTCS (2005) IEEE Learning Technology Standards Committee. Retrieved April 30, 2005, from http://ltsc.ieee.org/

Poseidon (2005) Retrieved April 30, 2005, from http://www.gentleware.com/

QTI (2004, June*) IMS Question & Test Interoperability*. Retrieved April 30, 2005, from http://www.imsglobal.org/question/index.cfm

QTILite (2002, February*) IMS Question & Test Interoperability QTILite Specification Final Specification Version 1.2.* Retrieved April 30, 2005, from http://www.imsglobal.org/question/qtiv1p2/imsqti_litev1p2.html

Routledge N. (2002.) *XML-Schema Profile Definition for UML Honours thesis extract*. Retreived April 30, 2005, from http://titanium.dstc.edu.au/papers/xml-schema-profile.pdf

TELCERT Technology Enhanced Learning Conformance - European Requirements and Testing Retrieved July 15, 2005 from http://www.opengroup.org/telcert/

UML (2005), *Unified Modeling Language (UML).* Retrieved April 30, 2005 from http://www.uml.org/..

XSLT (2005) *XSLT The Extensible Stylesheet Language* Retrieved April 30, 2005 from www.w3.org/TR/xslt

XMI (2005) *XML Metadata Interchange (XMI).* Retrieved April 30, 2005 from http://www.omg.org/technology/documents/formal/xmi.htm

W3C (2005) World Wide Web Consortium (W3C). Retrieved April 30, 2005, from http://www.w3.org/