

Systematic Generation of XML Instances to Test Complex Software Applications*

Antonia Bertolino, Jinghua Gao, Eda Marchetti, and Andrea Polini

Istituto di Scienza e Tecnologie della Informazione “Alessandro Faedo”
Consiglio Nazionale delle Ricerche
Via Moruzzi, 1 – 56124 Pisa, Italy
{antonia.bertolino, jinghua.gao, eda.marchetti,
andrea.polini}@isti.cnr.it

Abstract. We introduce the XPT approach for the automated systematic generation of XML instances which conform to a given XML Schema, and its implementation into the proof-of-concept tool TAXI. XPT can be used to automatize the black-box testing of any general application that expects in input the XML instances. We generate a comprehensive set of instances by sampling all the possible combinations of elements within the schema, applying and adapting the well known Category-Partition strategy for functional testing. Originally, XPT has been conceived for application to the e-Learning domain, within which we briefly discuss some examples.

1 Introduction

Increasingly today complex software systems are developed according to a modular architecture, within which precise features can be identified and separately implemented. Main objective of this “componentization” trend is to permit the development of the different features of a complex composite application by diverse stakeholders while maintaining the possibility of integrating the subsystems into a unique working system. Nevertheless the integration clearly presupposes the definition of a rigorous and checkable format of the data exchanged between the components.

One of the most important innovations that strongly contributes to solve this issue has been the introduction of the eXtensible Markup Language (XML) [1]. In few years this language has established itself as the *de facto* standard format for specifying and exchanging data and documents between almost any software application. Immediately following, the XML Schema [2] has then spread up as the notation for formally describing what constitutes an agreed valid XML document within an application domain. Thus, XML Schemas are used for expressing the basic structure of data and parameters that remote components exchange with each other, and restrictions over them, while XML instances, formatted

* This work has been supported by the European Project TELCERT (FP6 STREP 507128).

according to the rules of the referred XML Schema, represent the allowed naming and structure of data for components interaction and for service requests.

The introduction of XML for specifying standard format of exchanged data is certainly fundamental and strongly increases the possibility of correct interactions, nevertheless XML related technologies do not solve the interoperability problem per se. No information concerning the interpretation of data can be associated to an XML description, leaving the room for different interpretations by the various developers. Trying to make a further step toward guaranteeing interoperability, our proposal here is to combine the great potential of XML Schema in describing input data in open and standard form, with testing activity to assess the common understanding of interacting e-Learning systems. In doing this, our intention is to take advantage of the special characteristic of the data representation suitable for automated processing, which is clearly a big advantage for testing.

We find that the adoption of the XML Schema leads quite naturally to the application of *partition testing*, a widely studied subject within the testing community, since it provides an accurate representation of the input domain into a format suitable for automated processing. The subdivision of the input domain into subdomains, according to the basic principle of partition testing, can be done automatically by analyzing the XML Schema elements: from the diverse subdomains identified, the application of partition testing amounts to the *systematic derivation* of a set of XML instances. Systematic generation of XML instances, differently from a random based approach, clearly has important consequences on the effectiveness of the generated test suite permitting to derive meaningful statistics on the kind of instances generated, and then on the covered features.

This paper introduces our proposed XML-based Partition Testing (XPT) approach for the systematic generation of XML instances. Also a short overview on a proof-of-concept tool, called TAXI (Testing by Automatically generated XML Instances), is provided. Such tool inputs an XML Schema and automatically generates a set of XML instances for the black box testing of a component, whose expected input conforms to the taken schema. At the same time the paper reports a preliminary qualitative evaluation of the approach to the generation of instances for the IMS Learning Information Package specification.

In the remainder of this paper we discuss some related work in Section 2, and summarise the well known Category Partition method in Section 3; then we provide a description of the proposed strategy in Section 4 and of the tool implementing it in Section 5. Section 6 finally reports some preliminary considerations on application of the methodology; in particular in 6.1 we provide quantitative motivations to the application of a systematic approach, then in 6.2 a simple qualitative comparison of TAXI with another existing tool (XMLSpy) is presented. Some conclusions are finally drawn in Section 7.

2 Related Work

Our research is aimed at automatically generating a comprehensive suite of XML instances from a given XML Schema. The generated XML instances can then be