Consistency Checking of UML Model Diagrams Using the XML Semantics Approach

Yasser Kotb and Takuya Katayama

Japan Advanced Institute of Science and Technology

School of Information Science

Asahidai 1-1, Nomi, 923-1292, Ishikawa, Japan

{ kotb, katayama }@ jaist.ac.jp

ABSTRACT

A software design is often modeled as a collection of unified Modeling Language (UML) diagrams. There are different aspects of the software system that are covered by many different UML diagrams. This leads for big risk that the overall specification of the system becomes inconsistent and incompleteness. This inherits the necessary to check the consistency between these related UML diagrams. In addition, as the software system gets evolution, those diagrams get modified that leads again to possible inconsistency and incompleteness between the different versions of these diagrams. In this paper, we plan to employ our previous novel XML semantics approach, which proposed for checking the semantic consistency of XML documents using attribute grammar techniques, to check the consistency of UML diagrams. The key idea here is translating the UML diagrams to its equivalent XMI documents. Then checking the consistency of these XMI documents, they are special forms of XML, by employing them to our previous XML semantics approach.

Categories and Subject Descriptors

D.2.2 [**Design Tools and Techniques**] Computer-aided software engineering (CASE), Object-oriented design methods, State diagrams. D.2.4 [**Software/Program Verification**] Formal methods, Model checking, Validation. I.7.2 [**Document Preparation Languages and systems**] Markup languages, Verification.

General Terms: Documentation, Design, Standardization, Languages, Verification.

Keywords: UML, XML, Attribute Grammars, XMI, Model Checking.

1. INTRODUCTION

The aim of our past research was checking the consistencies of the semantics associated with XML documents [1]. XML, like all declarative structures, separate their semantic and syntax definitions, and have their own local description that results in high readability and high maintainability. We have focused mainly on the design of a proper method to add semantics to XML documents by associating semantic with the element tag attributes. By extracting such a semantic description, we are able to notify document writers of semantic errors in XML documents, and/or to automatically correct them. This method uses the advantages of *attribute grammars* (**AGs**), which is introduced by Knuth (1968) [2], in this regard. We have used the positive characteristics of AGs in the sense that they provide a clear description by the functional computation of attributes.

We have proposed successfully a novel technique to add semantics to XML documents by attaching the semantic information to the XML element tag attributes. We called this *XML semantics* [3]. This approach is based on the same concept of AGs, as attaching and checking the static semantics of programming languages through their attributes.

In this paper, we plan to expand our XML semantics to check the consistency of *unified Modeling Language* (**UML**) documents [4]. UML is OMG standard [5]; it is a graphical language for visualizing, specifying, constructing and documenting the artifacts of software systems. One of problems in a UML model, different aspects of a system are covered by different types of diagrams and this bears the risk that an overall system specification becomes inconsistent or incomplete. Hence, it is important to provide means to check the consistency and completeness of a UML model. The key idea in this work is that the UML can be easily exported to other software tools in the software life cycle chain using *XML Metadata Interchange* (**XMI**) [6]. XMI. is a standard interchange mechanism used between various tools, repositories and middleware. By employing these exported XMI documents into the XML semantics approach, we can check the consistency and completeness of UML documents.

2. XML SEMANTICS

In our approach of adding semantics to XML documents to check their consistency [3] [7], we classified the XML element tag attributes into two types: *static attributes* and *dynamic attributes*. The former are considered as *lexical attributes*. The latter are used as *intentional attributes*, which carry the semantics information and in turn can be separated into three subtypes; *evaluated attributes*, *context attributes*, and *copy attributes*. These attributes are evaluated by the evaluation algorithm and then compared with the corresponding static ones. In order to automatically evaluate and check the values of the dynamic (evaluated, context, and copy) attributes. We have specified this semantic specification in a meta (formal) language that is based on XML techniques, i.e. it is another XML document. This specification document plays the role of the attribute evaluation rules that is existed in AGs. This language has been called *Specification Language for XML Semantics* (**SLXS**) [7]. Readers refer to [7] for the outline and details of the employed consistency checking algorithm.

WWW 2005, May 10-14, 2005, Chiba, Japan.
ACM 1-59593-051-5/05/0005.

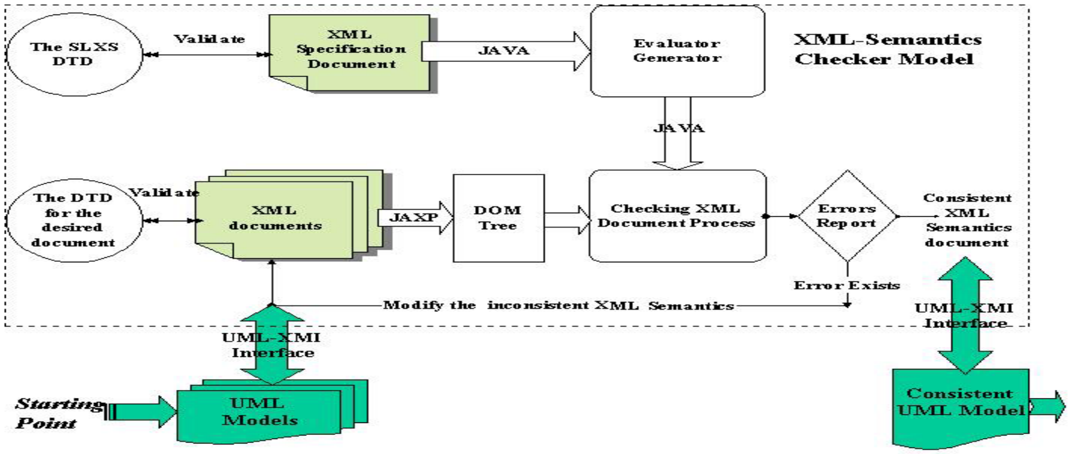

Figure 1. UML Consistency Checker Architecture.

3. UML CONSISTENY CHECKER MODEL ARCHITECTURE

We are working to build a novel system that is able to check the consistency of UML models through XML. This proposed system is based on the XML semantics checker model, as depicted in the Figure 1. For simplicity in this paper, the system is divided into three main parts.

3.1 UML Model Design Part

This part will be our starting point (as shown in left down part of the Figure 1). In this part, the user will start to design his desired UML model, which may contains an inconsistency and incompleteness problems. This part can be done by using any of the existence UML Modeling tools. For example, IBM Rational rose[1] or ArgoUML[2].

3.2 UML Model Checking Part

This part employs our approach for checking the consistency of XML documents. This is done as follows: firstly, we convert the given UML model(s) to its equivalent XMI document(s) using the UML-XMI interface part (see below). Then, we check this XMI document using our XML semantics checker Model. Lastly, the system will output a report by the different inconsistency and incompleteness in our model. By correcting these problems, either automatically or manually, we will obtain a consistent UML model.

3.3 UML-XMI Interface Part

This interface is responsible to convert between the UML models and their corresponding XMI documents. XMI is a standard interchange mechanism used between various tools, repositories and middleware. The main purpose of XMI is to enable easy interchange of metadata between modeling tools (based on the OMG UML) and between tools and metadata repositories (OMG MOF based) in distributed heterogeneous environments. XMI integrates three key industry standards: XML - eXtensible Markup Language, a W3C standard; UML - Unified Modeling Language, an OMG modeling standard; and MOF - Meta Object Facility and OMG modeling and metadata repository standard. The integration of these three standards into XMI marries the best of OMG and W3C metadata and modeling technologies allowing developers of distributed systems share object models and other metadata over the Internet.

[1] http://www-306.ibm.com/software/rational/

[2] http://argouml.tigris.org/

4. CONCLUSTION AND OBJECTIVES

In this paper we propose a novel model to check the consistency and the competence of UML models. This model builds over the previous XML semantic checker model.

Our goals from this research can simplify as follows:

- Build a stiff UML Consistency Checker Model.
- Success to check the different consistency and completeness aspects of UML models.
- Standardize our approach by W3C consortium and/or OMG group.

5. ACKNOWLEDGMENTS

This research is partly supported by JSPS (Japan Society for the Promotion of Science) and the Grant-in-Aid for JSPS Fellows.

6. REFERENCES

[1] Bray, T., et al. Extensible Markup Language (XML) 1.0 (Third Edition), W3C, February. 2004, http://www.w3.org/TR/REC-xml/

[2] Knuth, D.E. Semantics of Context-free Languages. Math. System Theory J.2, 1968, 127-145.

[3] Kotb, Y., Gondow, K. and Katayama, T. XML Semantics, Chapter 8, Book Title: "Web Mining: Applications and Techniques". Idea Group, Inc., 2005, 169-188.

[4] UML Resource Page(UML), Object Management Group (OMG), http://www.omg.org/uml

[5] Object Management Group(OMG), http://www.omg.org/

[6] XML Metadata Interchange (XMI), OMG, http://www.omg.org/technology/documents/formal/xmi.htm

[7] Kotb, Y. Checking Consistency of XML Documents by Attribute Grammars Techniques, Ph.D. dissertation, School of Information Science, Japan Advanced institute of Science and Technology, September 2003.