MAKING XML FROM HYPERMEDIA MODELS

Luís Carriço, Rui Lopes, Miguel Rodrigues, Amadeu Dias, Pedro Antunes
LaSIGE & Department of Informatics, Faculty of Sciences, University of Lisbon
Campo Grande, Edifício C5, 1749-016 Lisboa, Portugal
{lmc, rlopes, mrod, amadeu, paa}@di.fc.ul.pt

ABSTRACT

This article presents a framework for the development of hypermedia systems, based on systematic modelling approaches. These introduce models at three abstraction levels (conceptual, navigation and presentation). An object-oriented approach, UWE, proposing UML extensions was adopted. The work described in this paper reports a solution for an integrated and consistent translation of those models into XML specifications. The generated applications are derived from the design models and reinforce a clear separation between information structure and presentation, and a strong consistence at the user interface level. An application example is presented.

KEYWORDS

Hypermedia, Modeling, UML, XML.

1. INTRODUCTION

The adoption of specific hypermedia design methodologies, emphasizing conceptual modeling, has emerged to minimize the complexity of creation of hypermedia applications. RMM (Isakowitz et al., 1998), HDM (Fraternali and Paolini, 2000), OOHDM (Schwabe and Rossi, 1998) and UWE (Koch and Kraus, 2002) are but a few of the proposed approaches. UWE, in particular, integrates UML (Booch et al., 1999) through the whole process. These hypermedia design methodologies follow a systematic approach, comparable to that of common software development processes (e.g. Unified Modelling Process (Jackobson et al., 1998)), though emphasizing the richness, flexibility and usability requirements inherent to "hypermedia". They enlarge and organize conceptual modelling with the identification of: (1) the information structure underlying the hypermedia system (conceptual model); (2) the navigational structure (navigational models); and (3) the presentational components and their relationships (presentational models).

A problem motivated by conceptual modelling methodologies is how to translate the conceptual models into concrete hypermedia implementations. The aim of our work is to provide that translation, maintaining as much as possible the underlying methodological notions and the overall coherence of the approach, in the resulting applications. Obviously, the idea is not to burden the coding process with repetitive tasks, or reformulating established patterns when translating the models into hypermedia implementations.

This paper proposes the use of XML (Morrison et al., 1999) for the construction of a logical support that enables the implementation of hypermedia systems designed according to the UWE approach. Document types (DTD) are defined with XML elements corresponding to the concepts used in conceptual, navigational and presentational modelling. These DTD introduce the UML concepts and stereotypes. The resulting vocabularies are then used to automatically generate application models, which in turn are instantiated either as XML specifications or as a set of rules (XLST, CSS, …). The proposed XML production process differs from others (Kraus and Koch, 2002), because it avoids the creation of successive interrelated instances for each model, thus facilitating maintenance and reinforcing coherence and consistency of the applications.

2. MODELLING HYPERMEDIA SYSTEMS

The systematic approach to hypermedia modeling is a recent discipline. Departing from the initial proposals of RMM, HDM and OODHM, several evolutions (Isakowitz et al., 1998; Fraternali and Paolini, 2000;

IADIS International Conference WWW/Internet 2003. Algarve, Portugal, 2003.

Schwabe and Rossi, 1998) have been produced, that, in the case of OOHDM, already suggests some integration with UML. UWE (Koch and Kraus, 2002) proposes a methodological approach similar to OOHDM, but stressing the use of UML throughout the whole modelling process. The methodology presents the following set of phases:

- Conceptual modeling - Reflects the information structure in terms of classes and relations and, in the case where the system is materialized in a database, represents roughly a conceptual schema.
- Navigation modeling - Two steps are defined: (1) navigation space; and (2) navigation structure. The first one introduces the notion of navigational class, a UML stereotype. These represent the information accessible through a specific navigational perspective - relating to a particular use case or actor. A parallel could be drawn with "views" in the database domain. For the navigation structure diagram four basic stereotypes are proposed: (a) Indexes represent computed lists of connections (anchors/links) from instances of one class to a selected set of instances of another class; (b) guided tours are determined similarly, but the navigation process is stepwise; (c) queries represent free selection criteria that result in an index; and (d) menus are predetermined groups of links. The final result is a set of models conveying different perspectives (or use cases), each one corresponding to a couple of diagrams (space and structure).
- Presentation models - The static presentation models include several stereotypes like frames, collections, forms, text, etc.. UWE also defines ways (still evolving) to specify the interrelation between the UI stereotypes and the corresponding navigation classes. This interrelation enforces the coherence between presentation and navigation classes – instances of the latter (e.g. "Dali", "Picasso" from class "NavPainter") will be all represented in similar instances defined by a particular presentation class (e.g. "Text" and "picture" on the left, defined in class "PresPainter"). The same reasoning is applied between conceptual and navigational classes.

Conceptually, designers of the hypermedia system should only define the models, their interrelations (some implicit) and the UI details (colours, fonts, etc.). Later, the system implementation (apart from architectural issues) should only require the insertion of data, preferably structured according to the conceptual model.

3. SPECIFYING AND GENERATING XML

Among the proposals for relating XML and UML, XMI, should be mentioned (OMG, 2001). XMI defines rules for the generation of: (1) DTDs (Document Type Definition) or XML Schema, from class diagrams; (2) XML documents from object diagrams (instances of class diagrams). The translation of UML-based hypermedia models to XML differs from the plain XMI approach because it must cope with:

- The interrelations between the elements of a model and those of other models (e.g. conceptual and navigational related attributes and classes);
- The rules and procedures associated with stereotypes (e.g. an index implies a query to the navigation class where it ends, based on the instance of the class from where it departs);
- The closeness that the presentation models (and the navigation ones, to some extend) have with the their implementation in the hypermedia system.

Some frameworks were proposed to solve the implementation of hypermedia systems from high level hypermedia models, using RMM (Balasubramanian et al., 1997) and HDM (Fraternali and Paolini, 2000) as modelling methodologies. Both use HTML as a destination language - recurring to CSS and databases access. Recently, and in the line of the work reported in this paper, Kraus and Koch (2002) propose an approach for the publication of XML, as Web applications, from models defined according to UWE.

According with the authors, the approach uses a tool for the specification of UML models and integrates it with a XML publication framework. The UML models produced by the first are induced in successive phases of the framework. The framework is fed with XSLT specifications and generates an XML schema for each model. Then, for each possible instance, a XML document describing it is generated, for each step of the methodology (each model). A set of enriched, but redundant, specifications are built, from the data description (conceptual model level) to the UI (after the presentation model). As a consequence, maintenance of contents and structure becomes inefficient, as it requires to rebuild the whole system.

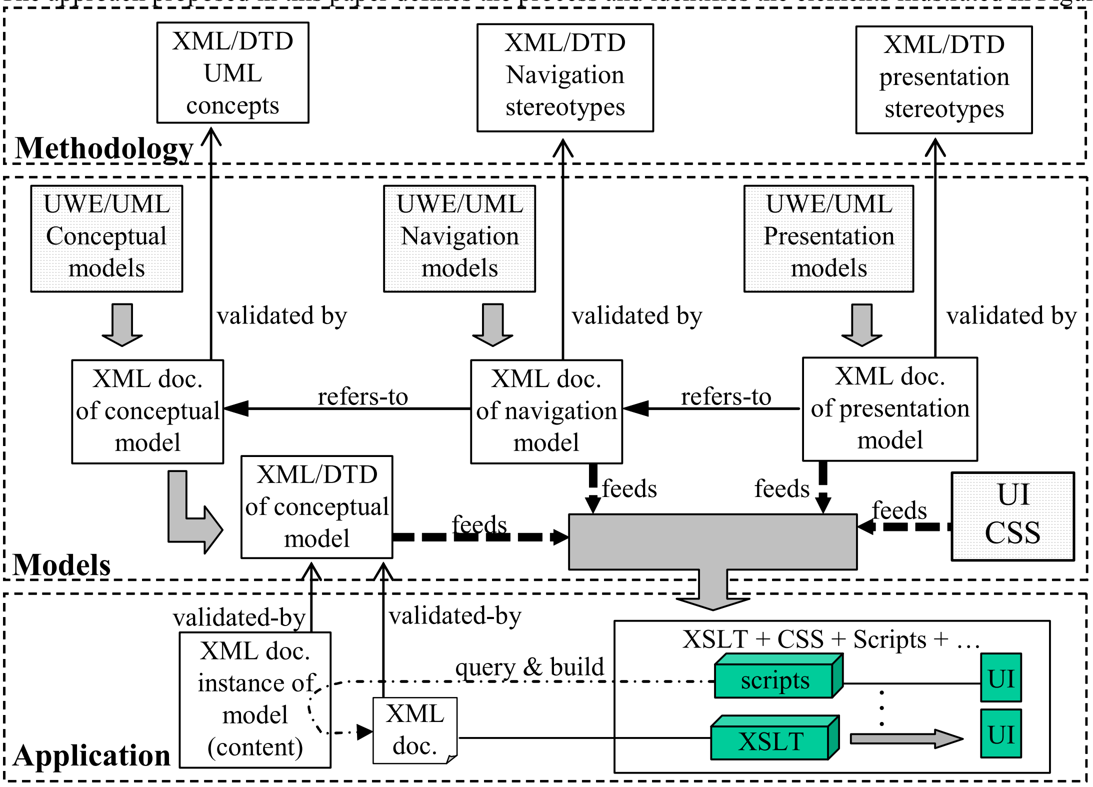
In order to cope with these problems the created hypermedia applications should:

- Maintain a clear separation between information and the mechanisms for presentation and interaction, in order to cope with modularity, reuse and adaptation to situations, devices or users;
- Maintain the coherence from the presentation and interaction to the information structure that is the baseline of the modeling methodologies for hypermedia systems.

In this sense it is rather important that the XML specification reinforces the notions of the methodologies and establishes rules instead of redundant specification of content.

4. TRANSLATING MODELS TO XML APPLICATIONS

The approach proposed in this paper defines the process and identifies the elements illustrated in Figure 1.



XML/DTD UML concepts
XML/DTD Navigation stereotypes
XML/DTD presentation stereotypes
Methodology
UWE/UML Conceptual models
UWE/UML Navigation models
UWE/UML Presentation models
validated by
validated by
validated by
XML doc. of conceptual model
refers-to
XML doc. of navigation model
refers-to
XML doc. of presentation model
XML/DTD of conceptual model
feeds
feeds
feeds
feeds
UI CSS
Models
validated-by
validated-by
XML doc. instance of model (content)
XML doc.
query & build
XSLT + CSS + Scripts + …
scripts
XSLT
UI
UI
Application

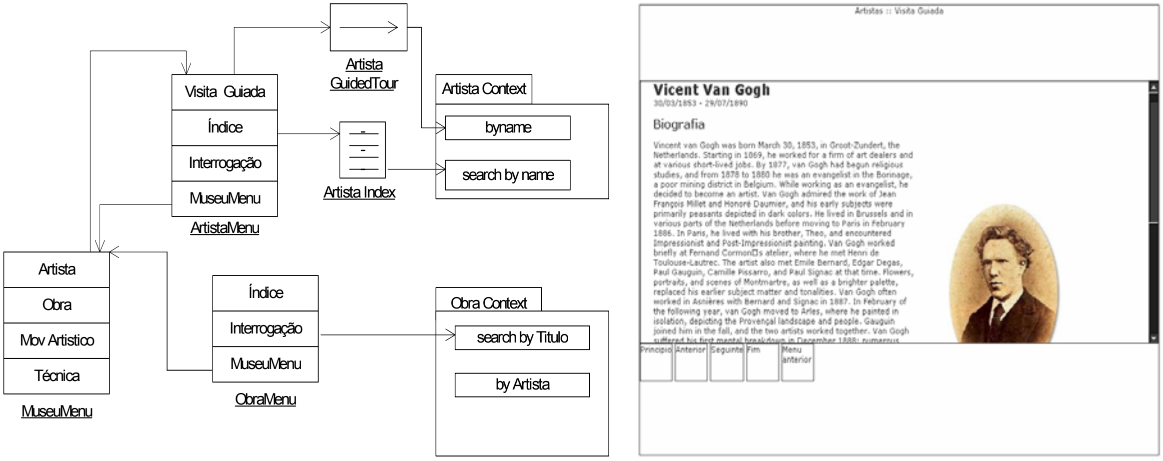
Figure 1 – Process for the generation of XML applications from hypermedia models

The elements identified at the top (under the box “Methodology”) constitute DTDs and introduce the basic constructs of: (1) UML; (2) the stereotypes proposed to extend UML, that are used in the navigation models; (3) the UML extensions that are used in the presentation models. These DTDs depend only on the used methodology. In the central box (named “Models”) are represented the documents that result from the modeling process of a specific hypermedia system. From the designer point of view, only the models labeled UWE/UML and the UI specifications (CSS, …) are required. The remainder documents are automatically generated by the framework. A XML DTD representation of the conceptual model (thus the vocabulary for the description of the systems data) is also created. Finally, in the “Application” box, the actual data of the hypermedia system is bound to the models previously defined. At this level, XML documents, validated by the XML DTD of the conceptual model, are built in order to structurally describe the data. XSLT specifications and a set of scripts are generated from the set of XML documents. They contain the code and rules to get the content and present it to the user, as specified by the models.

Contrary to the Kraus and Koch (2002) approach, no bound to specific publication frameworks is made. Also, the generation of XML documents, instances of the navigation and presentation models are avoided. Scripts and XSLT specifications merges all these models, defining a set of rules, that build and present XML views as interaction goes.

5. APPLICATION EXAMPLE

An “Interactive Art Museum” prototype was built. The development followed the modeling steps and the process previously defined. The result is a prototype running on Internet Explorer 6.



Vicent Van Gogh
30/03/1853 - 29/07/1890

Biografia

Vincent van Gogh was born March 30, 1853, in Groot-Zundert, the Netherlands. Starting in 1869, he worked for a firm of art dealers and at various short-lived jobs. By 1877, van Gogh had begun religious studies, and from 1878 to 1880 he was an evangelist in the Borinage, a poor mining district in Belgium. While working as an evangelist, he decided to become an artist. Van Gogh admired the work of Jean François Millet and Honoré Daumier, and his early subjects were primarily peasants depicted in dark colors. He lived in Brussels and in various parts of the Netherlands before moving to Paris in February 1886. In Paris, he lived with his brother, Theo, and encountered Impressionist and Post-Impressionist painting. Van Gogh worked briefly at Fernand Cormon□s atelier, where he met Henri de Toulouse-Lautrec. The artist also met Emile Bernard, Edgar Degas, Paul Gauguin, Camille Pissarro, and Paul Signac at that time. Flowers, portraits, and scenes of Montmartre, as well as a brighter palette, replaced his earlier subject matter and tonalities. Van Gogh often worked in Asnières with Bernard and Signac in 1887. In February of the following year, van Gogh moved to Arles, where he painted in isolation, depicting the Provençal landscape and people. Gauguin joined him in the fall, and the two artists worked together. Van Gogh suffered his first mental breakdown in December 1888; numerous

Figure 2 – Excerpt of a navigation structure model (left) and of the generated application (right).

Figure 2 shows a navigation model (derived from the respective conceptual model) and an instance of an “Artist”. The latter derives from the pattern defined by a presentation model, framed within another (a guided tour) defined in another presentation model (both build over the navigation model). Conceptual, navigation and presentation models where consolidated in scripts and XSLT, combined with CSS specifications.

6. CONCLUSION

The work reported in this paper allowed the construction of a XML platform enabling the automated generation of hypermedia systems based on hypermedia modeling methodologies. It builds upon XML in order to assure the coherence required by the approach and uses XSLT and scripts in order to avoid redundancy and provide flexibility of use. As future work, we plan the introduction of dynamic aspects of modeling, particularly in the presentation side and referring to the presence of time based active media.

REFERENCES

Balasubramanian, V. and Bashian, A. and Porcher D., 1997. A large-scale hypermedia application using document management and Web technologies. *Proceedings of the eighth ACM conference on Hypertext*, pp. 134-145.

Booch, G. and Rumbaugh, J. and Jacobson, I., 1999. *The Unified Modelling Language User Guide*. Addison-Wesley.

Bulterman, D., 2002. "SMIL 2.0 Part 1: Examples and Comparisons", *IEEE MultiMedia*, vol. 9, n.º 1.

Fraternali, P. and Paolini, P., 2000. Model-Driven Development of Web Applications: The Autoweb System. *ACM Transactions on Information Systems*, Vol. 28, Nº 4.

Morrison, M. and Boumphrey, F. and Brownell, B., 1999. *XML Unleashed*. Sams Publishing.

Isakowitz, T. and Kamis, A. and Koufaris, M. 1998. The Extended RMM Methodology for Web Publishing. *Working Paper IS98 -18*, Center for Research on Information Systems.

Jackobson, I. and Booch, G. and Rumbaugh, J., 1998. *The Unified Software Development Process.* Addison-Wesley.

Koch, N. and Kraus, A., 2002. The expressive Power of UML-based Web Engineering. *In Second International Workshop on Web-oriented Software Technology*. D. Schwabe, O. Pastor, G. Rossi, and L. Olsina, editors.

Kraus, A. and Koch, N., 2002. Generation of Web Applications from UML Models using an XML Publishing Framework. *In 6th World Conference on Integrated Design and Process Technology (IDPT).*

OMG, 2001. *XML Metadata Interchange (XMI)*, OMG Document ad/2001-06-12.

Schwabe, D. and Rossi, G., 1998. Developing Hypermedia Applications Using OOHDM. Proceedings of the Workshop on Hypermedia Development, Pittsburgh, USA.