Towards the Definition of Semantic Hyperstructures to allow Reader-defined Instantiation of Hypertext Systems

Daniel M. Germán
dmg@csg.uwaterloo.ca

D. D. Cowan
dcowan@csg.uwaterloo.ca

Department of Computer Science
University of Waterloo
Waterloo, Ont. N2L 3G1
Canada

Abstract

Hypertext systems are, usually, static entities in which the author defines the way they can be perused by the reader. The structure of a hypertext system can be separated into semantic, navigational and presentational structure. The first is concerned with the domain-specific, inherent structure of the system, the second with the way it will be navigated and the latter with the way it will be split into pieces in order to be presented to the reader. In general, hypertext browsers do not let the readers change any of these structures. In order to allow readers to create, dynamically, their own browsable hypertext structures it is necessary that each of these structures is clearly separated from the others and so readers have the chance to modify or create their own navigational and presentational structure given the semantic structure (and contents) of the hypertext system. Our research is moving towards the definition of a model and definition of an architecture to accomplish it.

Introduction

Readers commonly feel frustrated when they encounter hypertext systems that were designed the *wrong* way, i.e., in a manner different to what they perceive as optimal. Obviously, the decision of what constitutes a *good* hyperstructure depends on many factors: application domain, familiarity of the reader with the information presented, whether she is sequentially reading it or searching for particular information; even bandwidth can play a role in her decision. Authors often question themselves whether they are using the *best* structure to present their hypertext systems.

Hypermedia design methodologies, such as Object Oriented Hypermedia Design Methodology (OOHDM) [SR95] and Relationship Management Methodology (RRM) [ISB95] stress the separation of information content from its presentation. Both methodologies propose that a hypermedia system should be developed in three stages: a description of the information it will contain; a description of how such information is to be presented to the reader; and finally, a description of how such information will be linked together for navigational purposes.

The author concentrates in the content and semantic structure of the hypertext. At a different stage, the other characteristics of the system can be defined. This separates the actual information from its presentation. The

challenge is to allow the reader to participate in the middle of this process and define her own *view* of the hypertext system.

The semantic structure of a hypertext system is normally domain-specific and defines the inherent structure of the hypertext, while the presentational structure is related to the way such hypertext is presented to the reader. For instance, an article has a semantic structure (title, authors, sections, notes, etc.); when presented in paper, the notes could become footnotes or endnotes, nevertheless their concept has not changed, only the way they are presented to the reader.

Reader-defined hyper-structures

If the structure of a hypertext is made available, the reader would have, at least in theory, the ability to decide how to peruse it. For instance, assume that the hypertext system is a collection of articles about a given topic. A typical reader is well acquainted with the structure of articles. Then she can decide how she would like to see them:

- For each article, list its title, followed by authors, and the first 5 lines of the abstract.
- Each article as a single node; or as a collection of nodes, each representing a subsection.
- Hyperlinks (essentially references to other articles) could be follow in a variety of forms:
    - Link to the end of the bibliography entry in the article,
    - Link to the precise line where the idea being used by the former article's author is defined.
    - Present the bibliographic entry and the abstract of the article.

This list can grow as we could imagine better and better ways to present a collection of articles. The challenge is to define an architecture that would allow such hypertext systems. Such architecture could be implemented in different ways (each with different degrees of effectivity), that we enumerated below (in order of difficulty or complexity to be implemented).

- Style sheets. In many cases, the reader would be satisfied with a more flexible way to see the same node. In other words, readers would like to change the way some parts of a node look like, either to make them more or less noticeable or to hide them altogether. Such hypertext system would require that the author supplies one or more styles that could be replaced (partially or totally) replaced by the reader. In this case the hypertext system is totally inflexible, the flexibility is on the browsing tools. CSS and DSSSL respond to this need.
- For every application domain, there are solutions that work better than others; e.g., authors have noted that some types of hypertext systems are better than others. We can build a collection of *good hypertext models* that can be used as a guideline for future applications. A good architecture would allow a reader to create different views from the same semantic hypertext; each of these views would follow each of these models. Publishing all them will give the reader the chance to choose the way she would prefer to see the information, including to jump from view to view as she requires. Each of these views would be recognized in its domain as useful. We called these models *hypermedia design patterns* and will further discuss them below.
- On demand. It the structure of a given hypertext application follows a commonly agreed one (at least for its given application domain) then the reader could devise in advance her preferred way to peruse it. The reader would either download the information and peruse it at will, or issue *commands* to the hypertext system that will allow her to traverse the information in her own personal manner. These commands could be in the form of style-sheets, queries, or programs. Its main drawback is the complexity of such commands and a step learning curve. Such problems could be lessen if the author creates

recommended browsing styles and their corresponding styles. An interesting solution would be the use of an algebra for querying of structured documents such as the one defined in [Ray96].

Hypermedia Design Patterns

Because creating hypermedia applications in a specific environment contains repetitive creative tasks, efforts have focused on methodologies for multimedia development (e.g., OOHDM, RRM, HRM); nonetheless, most current applications are built without the assistance of such methodologies. Authors are now acquiring experience to build good hypermedia systems and assess their quality, and this experience should be compiled and organized. In software development where similar kinds of creativity are prevalent design patterns are being used to capture problem statements and potential solutions [GHJV95]. We propose analogous hypermedia design patterns for hypermedia publishing. These patterns attempt to gather the experience of authors in so that this experience can be reused by others. For instance, in a hypermedia system, an article could be presented in several forms; experience, however, has shown that some forms are better than others. Our research is oriented towards the definition of a pattern's language for hypermedia applications [GC87].

Structural Markup and the need for standards

In order to make our hypertext systems as flexible as possible, we should try to use established standards whenever possible. This is specially true in distributed systems, such as the World Wide Web, where the reader does not necessarily have the same working environment as the author. The hypertext system should be published in ways such that the probability that the intended reader will be able to peruse it is almost one.

It is widely recognized the need to structurally markup information. Nonetheless, it is not always used for hypertext systems. For instance, the ubiquitous HTML is more a presentational language than a structural language since it does not reflect the application domain of the given information. The same is true for other kinds of hypertext system, in which the information has to be stored in the given hypertext environment representation. Wise authors prefer to work with their system in a more flexible format (such as SGML) and then instantiate the required hypertext system. Structural markup simplifies the instantiation of one or more views from the same semantic hypertext; it also increases portability and the potential for reuse.

- SGML (Standard Generalized Markup) [Gol90]. SGML is becoming more common and there are more tools that support it. XML (Extensible Markup Language [BS96]) is a restricted version of SGML. XML is intended to be easier to define and to work with and it is intended for deployment on the World Wide Web. The W3 Consortium is actively pursuing this goal.
- HyTime (Information Technology-Hypermedia/Time-based Structuring Language [ISO92]). Links across hypertext should obey to it. Again, that will guarantee interplatform and interproduct compatibility.
- CSS (Cascading Style Sheets [BRL96]) and DSSSL (Document Style Semantics and Specification Language [ISO94]). The first is a style-sheets language for HTML while the latter is to be used on SGML documents. Both allow the reader to define how a document should look typeset. An author could propose several styles to the reader. For instance, one could be a fish-eye view of the entire document. Other would be a document in which only section names and first paragraphs appear.

Our Research

Our current research interests include:

- The definition a flexible model for semantic structure of a hypertext system, and its implementation using open standards such as SGML.
- The definition of instantiation processes to convert the semantic hypertext system into the final browsable hypertext.
- The creation of an architecture to allow the reader to create her own hypertext instantiation given a semantic one [GC96]
- The discovery and cataloging of *good* hypermedia design patterns [GC97].

Conclusions

A separation of semantic structure from presentation structure is fundamental for flexible hypertext systems in which the readers can customize them according to their needs. An adequate architecture requires the use of established standards to maximize its success and portability, specially in distributed systems.

Bibliography

[BRL96] B. Bos, D. Raggett and H. Lie, *HTML3 and Style Sheets,*W3C Working Draft Jul. 10 1996.

[BS96] T. Bray and C. Spelberg-McQueen, *Extensible Markup Language,* W3C Working Draft Nov. 14, 1996.

[GC96] D. Germán and D. Cowan. *A Federated Database for Hypermedia Development for the WWW* Codas 96. Kyoto, Japan. Dec. 1996.

[GC97] D. Germán and D. Cowan. *Hypermedia Design Patterns* EuroConference 97. To be published.

[GHJV95] E. Gamma, R. Helm, R. Johnson and J. Vlissides. *Design Patterns: Elements of Reusable Object Oriented Software*. Addison-Wesley, 1995.

[Gol96] Goldfarb, Charles, *SGML Handbook*, Oxford University Press. 1990

[ISB95] T. Isakowitz, E.A. Stohr, and P. Balasubramanian. *RMM: A methodology for structured hypermedia design.* Communications of the ACM, 38(8):34--44, August 1995.

[ISO92] International Organization for Standardization, *ISO/IEC 10774:1992, Information Technology-Hypermedia/Time-based Structuring Language (HyTime)*, 1992.

[ISO94] International Organization for Standardization, *ISO/IEC DIS 10179.2:1994. Information Technology - Text and Office Systems - Document Style Semantics and Specification Language (DSSSL)*, Edited by Sharon Adler. Jan. 1994.

[Ray96] D. Raymond *An Algebra for Partial Relations*. Ph.D. Thesis, University of Waterloo. 1996.

[SR95] Daniel Schwabe and Gustavo Rossi. *The object-oriented hypermedia design model.* Communications of the ACM, 38(8):45-46, August 1995.