A WWW Software Development Environment to Support Cooperative and Spread Working Groups

C. Maidantchik [a,1], A. R. C. Rocha [a], G. B. Xexéo [a]

[a] *Federal University of Rio de Janeiro - COPPE*
Caixa Postal 68.511
21945-970 - Rio de Janeiro - Brazil

This article presents a software development environment based on hypertext techniques for object-oriented software construction performed by cooperative working groups spread all over the world. It uses the WWW to support distributed software development.

Key words: WWW; Software development environments.

1 The Concept

The ATLAS (**A** **T**oroidal **L**HC **A**pparatu**S**) software teams develop, maintain, modify and use programs and related documents produced and stored at various sites. Added to the problem of world wide cooperation, the technology of the new detectors and the huge amount of data make the ATLAS software a highly complex structure. The software development process must lead to the production of reliable programs. Also, adding new capabilities to an existing piece of software should be an easy task. To deal with these problems, we introduce a software development environment based on hypertext techniques and the World-Wide Web [1] to support object-oriented software construction performed by cooperative working groups spread all over the world.

Software engineers are simultaneously integrating into their software development practice the advances of three research areas: object-oriented technology, hypertext systems and software project management. Object-oriented methods provide a unifying paradigm to tie together the software development process phases [2]. However, the resulting documents of one phase contain

[1] ATLAS Experiment - CERN/UFRJ Collaboration

references to classes and objects that are mentioned in other documents. Hypertext provides a powerful method of implementing cross references among documents through hypertext links [3]. However, like a text editor cannot guide an author on which words to write in a document, a hypertext system cannot guide the software developer on which documents should be linked for cross reference. Software project management defines all activities to be performed throughout the phases and identifies intermediate products that should be generated at each phase [4]. The progress and quality of these products have to be monitored continuously. Nevertheless, the task of organizing, scanning and understanding large amounts of software documents is not trivial.

The benefits of these research areas can significantly advance the current state of the art in software development, but the benefit is more effective when the areas are integrated. In our proposed system, the *World-Wide Web Software Development Environment – WWW SDE*, the activities of the software lifecycle and the resulting products are defined through a hypertext guideline. Within the framework, the object representation can be textual or graphical, horizontal (same level of abstraction) or vertical (from the concept to its implementation). The system designer can use cross reference information to make sure that the interactions at a particular level of abstraction meet the requirements and constraints imposed by the higher levels of abstraction. The system maintainer also can use the information to see how a change in representing or implementing a concept will interact with other concept's representations and implementations. The software project manager can use cross reference information to certify that all requirements were respected and not forgotten.

2 The Architecture

Figure 1 presents the *WWW SDE* architecture. Underlying the system there is a set of templates that guides the designer throughout the software development process, defining all activities and products that should be produced at each phase. Documentation generated through CASE tools or text editors can be easily inserted in the system through the *Index of Documents* module. The system objects are stored in the *Object* database. Decisions taken, foreseeable modifications, and constraints are placed in the *Design Rationale* database. The *Document Link Builder* filters and cross links software documents and the information from the *Object* and *Design Rationale* databases generating the hypertext o f the software development process.

Interface In distributed environments, where software and related documents are produced at various sites, the working groups need a well-defined

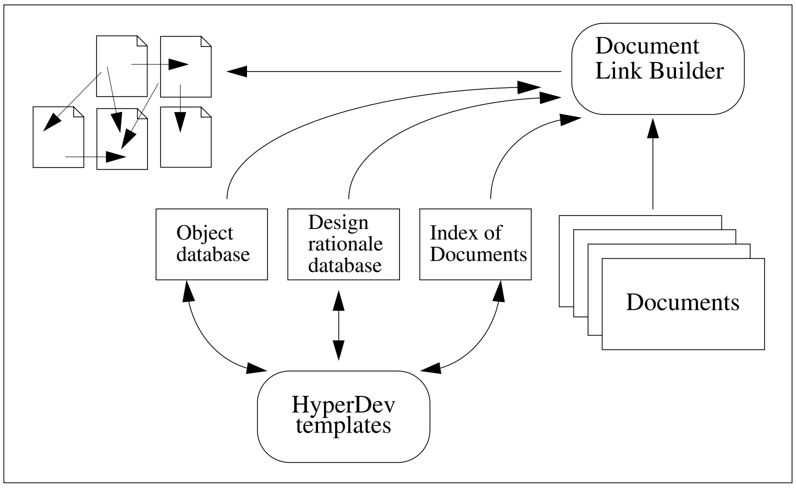

Fig. 1. *WWW SDE* architecture

process to guide the software construction. In order to assist with this, the interface is composed of a set of templates, implemented as hypertext pages. During a software development, the developer first defines an outline of ideas. Then, the requirements, analysis, and design are generated and, the developers can repeat the cycle until a satisfying outcome is reached.

Databases Navigation or browsing is effective only for small hypertext systems. For large hypertext, information retrieval through databases queries becomes crucial. The system incorporates an information retrieval mechanism focus on keyword based automatic searching.

Generation of Hypertext Hypertext applications are usually static information systems. Users can create, modify or remove links and nodes but the update of the remaining nodes are not automatic. In a software development environment, the coherence of the information should be guaranteed. When a modification occurs, the *WWW SDE* identifies it and starts the reorganization of the hypertext, converting only the documents that have changed.

Version Management The *WWW SDE* provides a flexible model to handle multi-user concurrent activities. The system organizes the generated hypertext files in a hierarchical collection of directories. These directories and files can be combined together to form a version release. In order to satisfy the many needs of source management in distributed environments, a single copy of the *master* sources is kept in the *master* repository. Developers can work on personal copies concurrently.

Configuration The templates can be considered as a composite object comprised of other objects such as nodes and links. There is only one configuration file for each project. When modifying the master templates, the developer can name and save the new version for future use. In following projects, users can load and reuse the suitable templates.

3 Conclusions

Hypertext systems are emerging as a new class of complex information management systems. These systems allow people to create, annotate, link together, and share information from a variety of media such as text, graphics, audio, video, animation, and programs [5]. Hypertext systems provide a non-sequential and new method of accessing information unlike traditional information systems which are primarily sequential in nature.

Object-oriented technology provides the ability to model, develop and manage complex systems that cannot be easily implemented with the current technology [6]. Hypermedia systems facilitate distributed memory, distributed processing, and distributed knowledge on a network. An object-oriented approach to hypermedia may help us achieve the object-oriented benefits much better than current approaches. Decisions in a software development group are made through discussions, but a lot of effort is often lost, or locked in individuals' heads. Hypertext templates facilitate the elaboration, organization, and presentation of the software implicit knowledge.

A prototype of *WWW SDE* runs on UNIX and provides a guideline for building software, which leads to the establishment of a common process across the software production and maintenance phases.

References

[1] *World Wide Web Consortium*
http://www.w3.org/pub/WWW/TheProject

[2] T. Korson, J. McGregor, *Understanding Object-Oriented: A Unifying Paradigm*, Communications of the ACM, Sept./1990.

[3] M. Bieber, C. Kacmar, *Designing Hypertext Support for Computational Applications*, Communications of the ACM, Aug./1995.

[4] European Space Agency, *ESA Software Engineering Standards PSS-05-0 Issue 2*, Feb./1991.

[5] J. Cybulski, K. Reed, *A Hypertext-Based Software Engineering Environment*, IEEE Software, pp. 62-68, March 1992.

[6] *HyperMedia Multimedia Reference*
http://heart.engr.csulb.edu/~ slobo/authoring/Project2/reference/mainContent.html