LiOS: Un Outil de Support à Évaluation au WWW

Renata Pontin de Mattos Fortes
Instituto de Ciências Matemáticas de São Carlos
Universidade de São Paulo (USP)
C.P. 668 - São Carlos - SP - BRASIL - CEP 13560-970
renata@icmsc.sc.usp.br

Maria do Carmo Nicoletti
Departamento de Computação
Universidade Federal de São Carlos (UFSCar)
C.P. 676 - São Carlos - SP - BRASIL - CEP 13565-905
carmo@dc.ufscar.br

RÉSUMÉ. Un outil automatique orienté à links, appelé LiOS, est présenté à travers ses trois modules principaux où sont traités les rôles fonctionnels de chacun. Le LiOS est utilisé pour analyser et évaluer la structure des applications de hypermedia concernant `a consistance et réutilisation de links.

MOTS CLÉS: Outil de propriéte, Évaluation des applications hypermedia à l' INTERNET

1. Introduction

Hypermedia is a new technology which provides both, alternative ways of organising information and a flexible approach to its use. From the hypertext author's point of view, freedom and flexibility can be achieved during the authoring process since there are not strict rules which should be followed. Authors can create a hyperdocument taking into consideration their own goals and their particular views of what information should be made available and also the group of users which will use this information.

However, the large amount of information to be processed makes hypermedia design a complex task. The existence of very few guidelines about what should be taken into consideration when creating a hyperdocument makes the task even harder.

LiOS: A Tool for Supporting Evaluation in the WWW

Renata Pontin de Mattos Fortes
Instituto de Ciências Matemáticas de São Carlos
Universidade de São Paulo (USP)
C.P. 668 - São Carlos - SP - BRASIL - CEP 13560-970
renata@icmsc.sc.usp.br

Maria do Carmo Nicoletti
Departamento de Computação
Universidade Federal de São Carlos (UFSCar)
C.P. 676 - São Carlos - SP - BRASIL - CEP 13565-905
carmo@dc.ufscar.br

ABSTRACT. This paper describes a link-oriented automatic tool named LiOS, which can be used to analyse and evaluate the structure of hypermedia applications, concerning consistency and reuse of links. LiOS is presented through its three main modules and the functional role of each module is discussed.

KEYWORDS: Authoring tool, Evaluating hypermedia applications on the Internet

1. Introduction

Hypermedia is a new technology which provides both, alternative ways of organising information and a flexible approach to its use. From the hypertext author's point of view, freedom and flexibility can be achieved during the authoring process since there are not strict rules which should be followed. Authors can create a hyperdocument taking into consideration their own goals and their particular views of what information should be made available and also the group of users which will use this information.

However, the large amount of information to be processed makes hypermedia design a complex task. The existence of very few guidelines about what should be taken into consideration when creating a hyperdocument makes the task even harder.

Since there is no consensus on which methodology to use in order to produce a "good" hyperdocument, it is very common to find poorly written hyperdocuments, lacking a well organised structure [FOR 97]. It is well known that the explosion of the Web has been a major concern among webmasters and consequently evaluation of hyperdocuments has become a problem of great urgency to be solved. This whole situation has been pushing the development of many different types of tools aiming at the improvement of the quality of hypermedia design, including tools that aid hypertext authors during both activities, authoring and maintenance, such as LiOS.

Many different metrics that can be used for evaluating the quality of a hyperdocument can be found in literature [BRO 90; GAR 95; HAT 95]. Among some design-oriented metrics proposed in [GAR 95] two of them have oriented the design of LiOS since this tool helps to check for consistency and link reuse, supporting authors in both, the evaluation of Web pages and the task of correcting their eventual structural problems.

The WWW is a challenging environment concerning control because it gathers a huge amount of information in a very dynamic environment; it is constantly under maintenance aiming not only at its expansion but also at the consistency of its contents. In fact, the maintenance of Web pages can be thought of as an authoring activity since it generally involves the creation of new pages and/or changes in those which already exist. In order to do that, authors should keep trace of the already existing structures to prevent mismatches with those being created.

The idea underneath the conception of LiOS is to have means of analysing Web documents subsets concerning their link reuse approach [FOR 96]. This tool can give hints of what may be wrong in the hyperdocument structure and consequently, improvements can be carried out by authors.

This work is organised as follows: the next section discusses the strategy adopted for identifying link reuse cases and, based on the way the link reuse components are defined, a taxonomy for grouping links together, which is dependent on different kinds of reuse, is proposed. Section 3 presents the architecture of the tool LiOS (Link-Oriented System) and describes each of its three main modules. Finally, we present our conclusions and point out the next steps to be followed in order to continue this work.

2. Virtual structures - classifying links

When discussing a hypertext engineering environment, is proposed in [NAN 95] the use of tools able to alternate between the bottom-up and top-down approaches as it happens when doing prototyping, in conventional software engineering. The bottom-up approach contemplates the generalisation of models from examples and through the top-down approach, models can be instantiated for constructing and implementing hyperdocuments.

It is consensus that virtual structures are a powerful mechanism for representing abstractions in hypermedia systems although nowadays there are not many systems able to implement them. The proposed link taxonomy implemented by LiOS is a sound way of identifying possible abstractions that can happen when reuse of links occurs. By identifying the reuse of link components, it is possible to identify abstractions and then, have a way to validate a possible conceptual model of the hyperdocument. Besides, by evidencing the different abstractions of link reuse, authors can re-examine their own definitions concerning the organisation, style and uniformity of terms created for anchoring. When creating a hyperdocument, this taxonomy can be of much use during the exam phase: it helps to evidence, in a systematic and hierarchical way, possible abstractions that can be constructed. These abstractions give, as a side effect, an effective feedback for creating heuristics to evaluate hyperdocuments.

The proposed taxonomy arises from a systematic investigation about different ways of defining links between Web pages written in HTML. In order to do that, two possibilities were considered: anchors placed in the same node (links send the user to a different piece of information, which is still part of the same node) and placed in different nodes. It was assumed that the static HTML links, which are pre-defined by the webmasters, are composed by three components: source-node, anchor, destination-node. Based on the different ways these components can be used and having link reuse on mind, eight different configurations, named cases, were detected. By a further analysis, the eight evidenced cases could be divided into three different groups, named group of *general links*, of *unusual links* and of *contextual links*.

Links belong to the *general* group when they can be interpreted as an abstraction of index pages or as abstraction of table-of-contents pages. Links which have the same source-node, same anchor and same destination-node and those which have different source-node, different anchor and different destination-node belong to the *unusual* group because they suggest an unusual way of organising information for a WWW application. Finally, links which can be considered wrongly defined at first sight (for example, links which have different anchors placed in different source-nodes reaching the same destination-node) but which are strongly dependent on the context, are part of the *contextual* group of links; a more detailed exam of its contents and of the region which surrounds the anchor should be carried out in order to qualify *contextual* links more precisely.

Next, two operations named *aggregation* and *degeneration*, were defined. These operations can be applied to link reuse cases and can be considered as abstractions; they were created with the intent of providing an alternative view for constructing virtual structures. By applying these abstractions, sub-cases could be derived. It is important to mention that sub-cases can be considered refinements of its parent-cases; by identifying them, authors can obtain a more precise view of the structure of the hyperdocument and based on that, carry out some changes to improve it.

Virtual structures in hypermedia systems can be approached in the same way the concept of view, in relational databases, is approached. In this sense, the current hypermedia systems which implement abstractions generally use a DBMS (Database Management System). However, when the focus is on WWW applications it is obvious that a DBMS will not be able (and is not designed for) to deal with and to control, in a plausible and efficient way, the huge amount of information that can be part of a WWW application. So, the proposal which underlies this work is that the existing DBMS should only have the auxiliary role of helping the extraction of those links which are instantiations of link reuse cases identified before. LiOS will carry out the extraction process guided by queries expressed in the DBMS's query language.

Currently, the LiOS tool is limited in the sense that it only accepts one Web site as input (or one file directory and its sub-directories), at a time. In order to make it a more general tool, only small changes need to be done.

3. The architecture of LiOS

One of the main goals of LiOS is to provide means to present, analyse and classify the links of a hyperdocument, in order to gather data which can help authors to get a closer and more refined exam of its structure. As LiOS focuses on the structure of a hyperdocument through its links, it is important here, to define what exactly a link means to this tool. Figure 1 presents a diagram of the object link and its components. It is important to add that a link, by being considered as a composite object of the kind *aggregation of objects*, is uniquely identified by LiOS, as well as are its components. This means that LiOS could, besides analysing the terminology of anchors, part of links, analyse and/or verify any other attribute which could eventually characterise a link component. This is not implemented in LiOS yet, but it will be soon, in a next future.

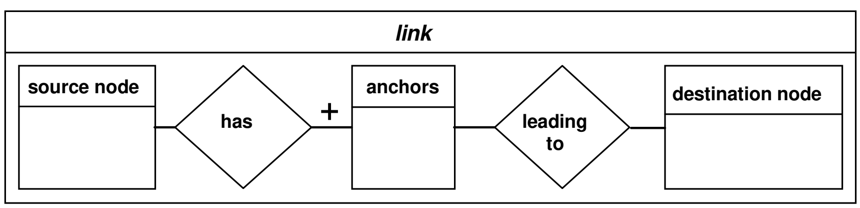

Figure 1. *The object **link** and its components*

Currently LiOS starts by identifying and recovering links, more specifically, their components, namely source-nodes, anchors and destination-nodes from a Web site, given its URL location (its Home Page). Its architecture is defined by three main interacting modules (the grey boxes on the right side), as shows Figure 2.

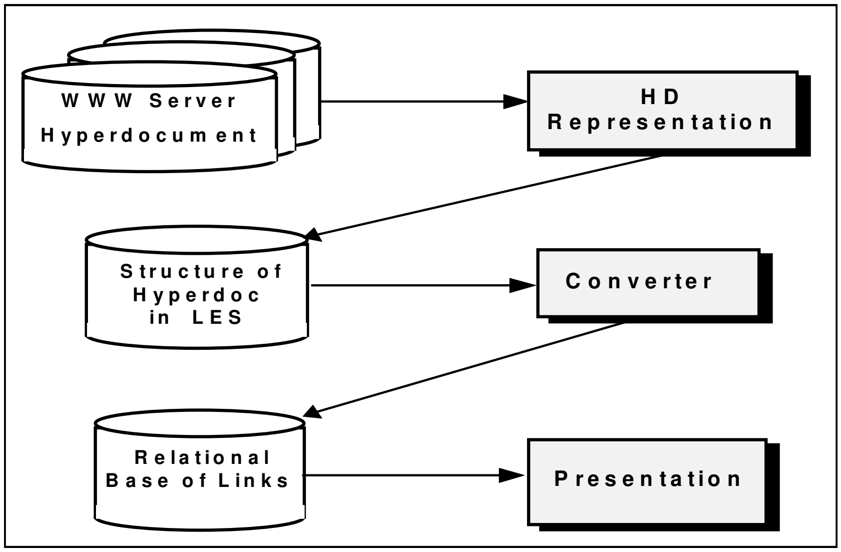

Figure 2. *Modular architecture of LiOS*

3.1. The HD Representation module

The *HD Representation module* is the module responsible for scanning the hyperdocument-input by following its local links, in order to generate its link-based structure described in LES[1]. By default, LiOS assumes that hyperdocuments are written in HTML, a language which only allows the definition of unidirectional links. Nevertheless, since the implementation of this assumption is confined into one piece of software, it can be easily changed.

The output of this first module is an ASCII file containing the LES specifications of the hyperdocument links. The description of links in LES is based on the fact that the anchors presented to the reader at the current node (source node) correspond to possible links to be followed by the reader. It is well known that elements in an HTML document are denoted by tags. In HTML, the string of characters that is delimited by <*a* > </*a*> has been adopted in this work as being the source-anchor. Due to the fact that the HTML language only supports the definition of unidirectional links, we mapped the source-node of a link into the origin of a transition and its destination-node into the destination of the transition. It should be observed that it is very important for the functionality of the *HD Representation module* that the resolution of the URL addresses is conducted in a uniform way. In order to make possible a real identification of reuse of link components, the address

[1] LES (Statechart Language Specification) allows the textual description of the structural components of a hyperdocument, according to the Statechart syntax and semantics.

hidden in the reference of the links should be decoded and uniformly coded back again.

As an optional feature the LiOS environment can be branched to another environment, named StatSim. This allows the output of the *HD Representation module*, i.e., the ASCII file, to be analysed by the StatSim LES Analyser, and to simulate the link structure of the Web site, aiming to recursively analyse all of its embedded paths.

The StatSim environment is an automatic graphical supporting tool for Statechart specifications of systems. Its purpose is to validate specifications of systems written in Statechart and its main module is a simulator of Statechart [FOR 91]. The StatSim environment can be operated in three different ways: interactive, batch and programmed. It is worth mentioning that in order to make possible the interaction LiOS-StatSim, the LES Analyser was adapted for dealing with a bigger volume of information and had its interface customised for accepting longer strings of data.

3.2. The Converter module

The *Converter module* is the responsible for converting the ASCII file generated by the HD Representation module, into a relational data base. In the reference model of Dexter [GRO 94], this database corresponds to the Storage Layer: the functional module of a hypertext model which supports the resolution of links. *The Converter module* creates unique identifiers for each and all links and for their components. Microsoft Visual FoxPro version 3.0, which already has an internal automatic converter for translating databases written in ASCII into databases in relational form (.dbf), was used for implementing the *Converter module*. A link is mapped into a 4-uple having the form:

(<u>id-link</u>, source-anchor, source-node, destination-node.destination-anchor)

where:

id-link = link-identifier (primary-key);
source-anchor = event;
source-node = source;
destination-node.destination-anchor = destination

For example, the LES description of a link:

Evento “About the Institute”
origem “/httpd/docs/index.html”
destino “/httpd/docs/icmsc/index.html”

is translated into the following record:

(<u>id-lk</u>, “About the Institute”, “/httpd/docs/index.html”, “/httpd/docs/icmsc/index.html”) and for each of the above components, another identifier is created. This was done in order to provide different names each time the same component appears among all the records.

3.3. The Presentation module

The *Presentation module* is responsible for presenting the user with a list of all the existing links in the hyperdocument, so that they can be chosen for further examination. This module focuses first on links and then on the nodes they connect. Its main window shows a list of all the anchor names used in the hyperdocument and makes them available for selection.

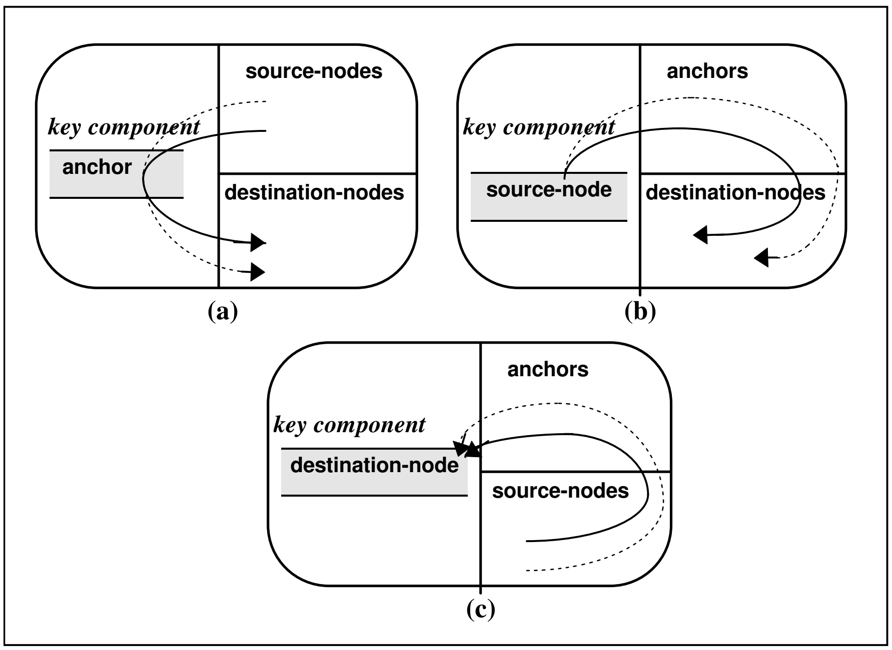

Figure 4. *Link based presentations of the structure of a hyperdocument*

Selecting any of these anchor names gives the user access to two new lists: the first containing all the source-nodes which have as starting point, the selected anchor and the second, containing all the destination-nodes which can be reached from that same anchor.

The *Presentation module* was prototyped using the TCL language version 7.0 with the graphical toolkit TK, version 4.0. A remotely installed server SQL "Mini-SQL", version 1.0.6 was used for storing the data.

LiOS allows three alternative ways of presenting links, each of them dependent on the key component chosen, as shown in Figure 4. An expression which formalises the relation among the components source-node (*SN*), anchor (*A*) and destination-node (*DN*), and gives all the possible presentation views can be written as $SN(A) = DN$. This expression is read as: an instatiation of an anchor *A* in a certain source-node *SN* takes the user to a certain destination-node *DN*. So, in order to obtain all the possible views given by this relation, it is enough to replace the elements in the expression with a free variable. For example, the expression $SN(?) = ?$ gives all the anchors placed in a certain *SN* and all the destination-nodes which can be reached from them.

The views given by the *Presentation module,* besides allowing the author a complete exam of all the values of anchors used in the hyperdocument, can also help to identify:

— *All the source-nodes which have a certain anchor A in their links and all the destination-nodes reached by these links.*

This is a way of identifying links which possess the same source-anchor as shown in Figure 4 (a). The two other diagrams, Figure 4 (b) and (c), represent respectively the identification of:

— *All the anchors which are part of a certain source-node SN and all the destination-nodes reached by the links established by these anchors.*

— *All the anchors which are part of links arriving at a certain destination-node DN and all the source-nodes from which these links leave.*

By the identification of all the above situations, authors can have a broader view of the whole hyperdocument structure, and can detect some of its vulnerabilities.

4. Conclusions

In this paper we present LiOS, a tool for supporting both, the authoring and maintenance activities of Web documents. Its three main modules, *HD Representation module, Converter module* and *Presentation module*, are functionally described. LiOS was designed assuming the structure of a hyperdocument represented as a set of links. Three groups of link reuse cases, *general, unusual* and *contextual* are identified and the paper proposes that they can be used for diagnosing the quality of a hyperdocument, concerning link reuse, using the proposed tool. LiOS can be guided aiming to identify all different cases of link reuse and, based on its results, authors can evaluate and improve the quality of hyperdocuments.

We intend next to enlarge the link definition adopted in this work by allowing the attachment of attributes to link components. We expect with this to obtain a more precise description of links so that a more refined analyses of a hyperdocument structure can be carried out by LiOS.

5. Bibliography

[BRO 90] BROWN P.J., "Assessing the Quality of Hypertext Documents" In: A. RIZS; N. STRITZ & J. ANDRÉ, eds. *Hypertext: Concepts, Systems and Applications.* Cambridge, Great Britain, Cambridge University Press, 1990. /*Proceedings of* ECHT'90 - European Conference on Hypertext, Versailles, France, November 1990./ pp.1-12

[FOR 91] FORTES R.P.M., "A Supporting Tool for Using Statecharts when Specifying the Behaviour of Complex Real-Time Systems" (in Portuguese), *Master Dissertation*, ICMSC-USP, São Carlos, 1991

[FOR 96] FORTES R.P.M., "Analysis and Evaluation of Hyperdocuments: an Approach Based on the Structural Representation" (in Portuguese), *Ph.D. Thesis*, IFSC-USP, São Carlos, SP, Brazil, August 1996

[FOR 97] FORTES R.P.M.; NICOLETTI M.C.; GARCIA NETO A. "A Formal Approach to Consistency and Reuse of Links in World-Wide Web Applications" (to appear in: Paternó, F.; Palanque, P. *Formal Methods in Computer Human Interaction.* New York, Springer, 1997)

[GAR 95] GARZOTTO F.; MAINETTI L.; PAOLINI P., "Hypermedia Design, Analysis and Evaluation Issues" *CACM* Vol. 38, No. 8, August 1995, pp.74-86

[GRO 94] GRONBAEK K.; TRIGG R.H., "Design issues for a Dexter-based Hypermedia System" *CACM*, Vol.37, No.2, February 1994, pp.41-49

[HAT 95] HATZIMANIKATIS A.E. *et al.*, "Measuring the Readability and Maintainability of Hyperdocuments" *Software Maintenance: Research and Practice*, Vol.7, 1995, pp.77-90

[NAN 95] NANARD J.; NANARD M., "Hypertext Design Environments and the Hypertext Design Process" *CACM* Vol. 38, No. 8, August 1995, pp.49-56