RetroWeb: A Web Site Reverse Engineering Approach

Sélima Besbes Essanaa and Nadira Lammari

CEDRIC Laboratory, CNAM,
292 rue Saint Martin 75141 Paris Cedex 03, France
Besbes_s@Auditeur.Cnam.fr, Lammari@cnam.fr

Abstract. Most of Web sites are built as a matter of priority. Therefore, to reduce the development time, the conceptualization phase is often put aside and the associated documentation is neglected. Moreover, during the exploitation phase, Web sites suffer the effects of a rapid and unstructured evolution process. Their reconstruction encompasses inevitably a reverse engineering process. In this paper, we propose RetroWeb, a reverse engineering approach of semi-structured Web sites. It aims to provide a description of the site informative content at the physical, logical and conceptual levels. This approach uses, at each level, a meta-model which is instantiated using reverse engineering rules.

1 Introduction

In spite the effort and the generated high costs for their development and maintenance, most enterprise Web sites are not suitable. The traditional principles of the information systems design and documentation are often neglected on behalf of the visual and esthetic aspects. The lack of conceptualization during the development leads to maintenance problems and to a bad structuring of the information over the Web site pages. To reconstruct already existing Web sites that do not respect the development life cycle, the reverse engineering is essential. It aims to extract, in a clear and formal way, at various abstraction levels, all available information in order to understand the site functionalities.

This paper presents an approach, called RetroWeb, to reverse engineer the informative content of semi-structured Web sites. Besides the EER conceptual model, two meta-models are proposed to describe, at three abstraction levels, the semi-structured data coded on the Web site HTML pages. The first one is used to represent the Web site through its physical views. It is instantiated using the semi-structured data extracted from each HTML page. The second one is used to describe the Web site through its logical views. Mapping rules are proposed for the translation of physical views into logical views and then into conceptual ones. The whole site conceptual description is obtained by merging the generated conceptual views.

The remainder of the paper is organized as follows. Section 2 describes RetroWeb. Section 3 discusses the related works. Section 4 concludes and presents future work.

N. Koch, P. Fraternali, and M. Wirsing (Eds.): ICWE 2004, LNCS 3140, pp. 306–310, 2004.

2 RetroWeb Approach

RetroWeb is an approach to reverse engineer the informative content of semi-structured Web sites[1]. It is built on the inversion of the life-cycle design process. Starting from web site HTML pages, it deduces an EER conceptual description of its informative content. It encompasses three steps: the extraction, the conceptualization and the integration steps. The following paragraphs describe each of these steps.

The Extraction Step. It aims to the retrieval of the semi-structured data coded on the site HTML pages and to describe them through physical views (one physical view per page). Its result is the instantiation of the meta-model describing the extracted physical views. For instance, let us consider a Web site for academic journal publications. The left part of Fig. 1 presents a Web page that displays, for each volume of the academic journal, its authors. The right part of the same figure gives its corresponding physical view.

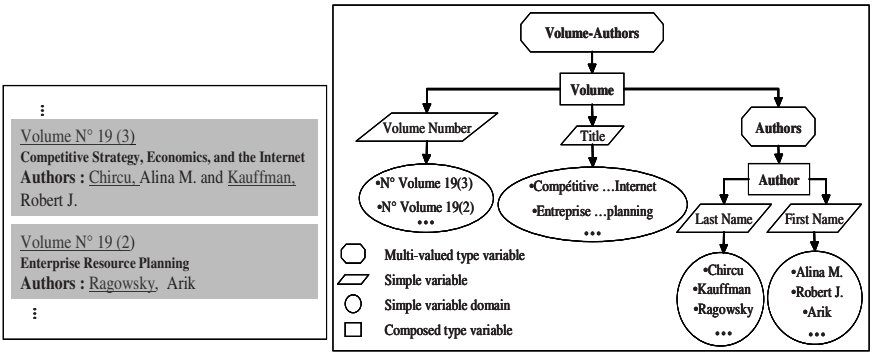

Fig. 1. An academic journal publications Web site page and its corresponding physical view

The concepts used to build physical views are: simple variable, simple variable domain, composed type and multi-valued type variables. A simple variable is an atomic structure that can hold an atomic piece of data. Simple variable domain is the values set (data) that can be hold by a simple variable in a page. A composed type variable is a data record build by one or more simple variables. A multi-valued variable is a set of composed type variables.

The extraction step is performed into three phases: pre-processing, extraction and naming phases. The pre-processing phase takes as an entry HTML pages, corrects them, proceed to some cleaning, executes some transformations and then returns, for each page, a coded sequence describing the page structure. In this sequence, the structure tags are codified using the same number of positions. All textual data that are not identified as tags are replaced by a token "Text". The second phase deduces pattern expressions that will be used by the wrapper to extract data from pages. It uses the DeLa system technique described in [1]. The last phase assigns significant names

[1] Semi-structured web sites are in majority data-rich and display data in contiguous blocks. These blocks are ordered and aligned such that they exhibit regularity.

to variables of the physical views. It uses an algorithm that improves the labeling itself by reducing the number of concepts to name. It first defines classes of concepts that may be assigned the same label. Then, it assigns to any concept, not yet labeled, a name and gives this name to its family (i. e. to all concepts sharing the same class). It uses, to that end, some heuristics. More details about this phase can be found in [2].

The Conceptualization Step. It aims to produce the EER schemas associated with the physical views. In order to reach this result, the conceptualization step translates first the physical views into logical ones, constructs for each logical view its corresponding EER schema and then, affects, according to the same naming process used in the precedent step, significant labels to entity-types and relationship-types of the obtained conceptual schemas. Consequently, it generates successively an instance of the meta-model describing the logical views of the web site pages and an instance of an EER model. For our example, the physical view described in Fig. 1 is transformed into the logical view of Fig. 2a and then into the EER schema of Fig. 2b.

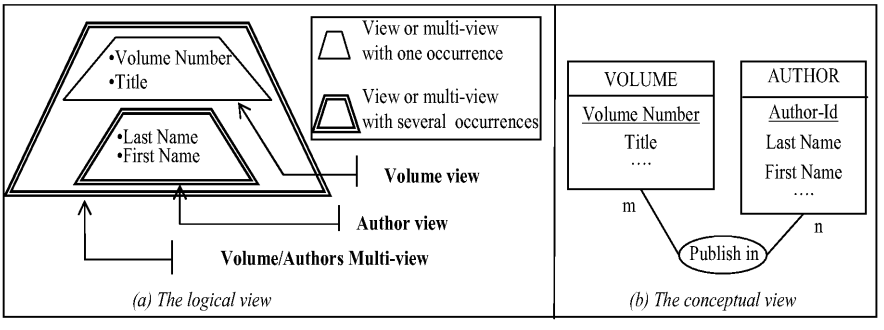

Fig. 2. The logical and conceptual views deduced from the physical view of Fig. 1

A logical view is described through three concepts: the property, the view and the multi-view concepts. A property is a logical representation of a variable. A view is a logical representation of a composed type variable. It groups properties that describe an object represented in a web page. It can have one or several occurrences. A multi-view is obtained by assembling all deduced views. It can also have several occurrences.

The different schema transformations are performed thanks to reverse engineering rules. For example, among rules used to translate logical views into conceptual ones, we can quote:

- Rule 1: every logical view becomes an EER schema
- Rule 2: each view of a multi-view becomes an entity-type of the EER schema
- Rule 3: if two views V1 and V2 belong to the same multi-view then the entities that they represent are linked by a relationship-type. If the two views have a number of instances higher than 1 then the cardinality of this relationship-type is M: N. In contrary, if one of the two views has a number of instances equal to 1, then the cardinality of this relationship-type is 1: N.

The Integration Step. It merges the portions of EER schemas into a global one in order to give a global conceptual description of the whole web site informative content. This step is based on integration techniques well known in the information systems context. The majority of these techniques propose an integration in 4 phases: (i) a pre-integration phase which aims to standardize the schema sources by translating them into a common conceptual model, (ii) a comparison phase whose aim is to identify the relations between schema sources, (iii) a fusion phase which allows the merging of the schema sources in an integrated one according to the results of the precedent phase and to the existing integration rules and finally (iv) a reorganization phase that improves the quality of the integrated schema. We re-use these phases. We choose the EER model as a common conceptual model.

3 Related Works

The evolution of Web sites has been addressed from various ways. Some research works take an interest to the evolution of the presentation [3, 4] and others to the restructuring of the HTML code [5, 6, 7]. [2] proposes an approach that allows small display units, like PDA and WAP, to access to the Web site content. [5] describes a clustering technique to translate static pages into dynamic ones. [6] uses a slicing technique to reduce the site size. [7] applies re-writing rules to the HTML code.

The literature also supplies approaches that aim to obtain Web sites abstract representation [8, 9, 10, 11, 12]. [8] presents a framework to deduce, from XML pages, the corresponding DTD. [9] analyzes web site code in order to automatically reconstruct the underlying logical interaction design. [10] translates the visual layout of HTML forms into a semantic model. [11] produces HTML UIs by integrating data of several web pages. [12] uses UML diagrams to model views of web applications, at different abstraction levels.

Other related works concern data extraction from HTML code. Their principal concern is the retrieval of data concealed in semi-structured data-rich pages. The way in which these data will be displayed, and the models to which they will be mapped, are left to the user.

Through RetroWeb, we wish to recover the informative content of the whole site. Thanks to the proposed meta-models, RetroWeb supplies, at physical, logical and conceptual levels, a clear and semi-formal description of the web site informative content. This extracted description is useful for its re-documentation, re-structuring or integration with other Web sites.

4 Conclusion

In this paper we have proposed a reverse engineering approach of semi-structured and undocumented Web sites, called RetroWeb. RetroWeb gives a description of the informative content of the site at various abstraction levels: physical, logical and conceptual levels. Reverse engineering rules are defined to map the physical description into the logical one and then into the conceptual one.

According to the type of needed evolution, the web site maintainer can execute totaly or partially the reverse engineering process. For instance, RetroWeb can be used either for the integration of Web sites or for the translation of HTML sites into XML sites. In the firts case, the conceptual description of the informative content of the two sites must be retreived. In the second case, the retrieval of physical views can be enough.

Our current work involves implementing RetroWeb. Further works will mainly concern the enrichment of the set of heuristics used for the naming of concepts. We also expect to enrich the reverse engineering rules set in order to exhibit, for example, generalization-specialization links at the conceptual level. Finally, we wish to extend our process to other aspects like the site navigational structure.

References

1. Wang, J., Lochovsky, F.: Data extraction and label assignment for Web databases. Proc. of the 12th International Conference on World Wide Web, Hungary (2003) 187–196
2. Essanaa, S., Lammari, N.: Improving the Naming Process for Web Site Reverse Engineering. Proceedings of the 9th International Conference on Application of Natural Language to Information Systems, Manchester June (2004)
3. Vanderdonckt, J., Bouillon, L., Souchon, N.: Flexible Reverse Engineering of Web Pages with Vaquista. Proceedings of the 8th Working Conference on Reverse Engineering (WCRE'01), October (2001) 241–248
4. Lopez, J. F., Szekely, P.: Web page adaptation for universal access. In Stephanidis, C. (ed.) Universal Access in HCI: Towards an Information Society for All. In proceedings of the 1st International Conference on Universal Access in Human-Computer Interaction, New Orleans August (2001). Mahwah, N. J.: Lawrence Erlbaum Associates 690-694
5. Ricca, F., Tonella, P.: Using Clustering to Support the Migration from Static to Dynamic Web Pages. Proceedings of the 11th International Workshop on Program Comprehension, Portland Oregon USA May (2003) 207–216
6. Ricca, F., Tonella, P.: Construction of the System Dependence Graph for Web Application Slicing. Proceedings of SCAM'2002, Workshop on Source Code Analysis and Manipulation, Montreal Canada, October (2002) 123–132
7. Ricca, F., Tonella, P., Baxter, I., D.: Web Application Transformations based on Rewrite Rules. Information and Software Technology Volume 44(13) (2002) 811–825
8. Chuang-Hue, M., Ee-Peng, L., Wee-Keong, N.: Re-engineering from Web Documents. Proceedings of the International Conference on digital Libraries (2000) 148–157
9. Paganelli, L., Paterno, F.: Automatic Reconstruction of the Underlying Interaction Design of Web Applications. Proceedings of the 14th International Conference on Software Engineering and Knowledge Engineering (SEKE 02), Ishia Italy July (2002)
10. Gaeremynck, Y., Bergman, L. D., Lau, T.: MORE for less: model recovery from visual interfaces for multi-device application design. Proc. of the Int. Conf. on Intelligent user interfaces, Miami Florida USA January (2003), ACM Press, New York USA (2003) 69-76
11. Stroulia, E., Thomson, J., Situ, Q.: Constructing XML-speaking Wrappers for Web Applications: Towards an Interoperating Web. Proc. of the 7th Working Conference on Reverse Engineering (WCRE'2000), Queensland Australia (2000), IEEE Computer Society
12. Di Lucca, G. A., Fasolino, A. R., Pace, F., Tramontana, P., De Carlini, U.: WARE: a tool for the Reverse Engineering of Web Applications. Proc. of the European Conference on Software Maintenance and Reengineering (CSMR2002), Budapest March (2002)