Published in Computer Networks and ISDN systems, Proceedings of the 7th International World Wide Web Conference, Elsevier, pp. 85 - 94, 1998

.

WSDM: A User Centered Design Method for Web Sites

O.M.F. De Troyer, C.J. Leune
Tilburg University, Infolab
P.O.Box 90153
5000 LE, Tilburg - The Netherlands
email: detroyer@kub.nl , kees@kub.nl

Abstract

WSDM is a user-centered method for the design of kiosk Web Sites. By explicitly starting from the requirements of the users or visitors, WSDM solves web site problems that are primarily caused by that fact that a site has no underlying design at all, or that the design is mostly data-driven.

Keywords: web site, design, modeling.

1. Introduction

Many companies and organisations already have a site on the internet or maintain an intranet. Web sites are becoming critical for companies and organisations. Either they are seen as part of a marketing strategy (internet sites) or as a tool to enhance the internal communication and as such the efficiency of the organisation (intranets). Therefore effective design of web sites is important.

There is a lot of literature on graphical and user interface aspects of web site design [Mau 96, Nie 95, San 96] but little on a systematic way for designing 'good' web sites. Surfing on the WWW shows that a lot of sites are built in a very ad hoc manner and suffer from a number of problems. These problems can be classified according to two perspectives: the viewpoint of the visitor and the viewpoint of the web manager. The web manager is usually faced with maintenance problems, the visitors with usability problems (we describe these problems into more detail in section 2).

A number of researchers have already recognized the lack of design methods for web sites, or more in general for web-based information systems, and have proposed methods: HDM [GPS 93] and its successors HDM2 [GPM 93] and OOHDM [SR 95], RMM [ISB 95], W3DT [BN 96], and [TL 97]. The older methods (HDM, OOHDM, RMM) were originally designed for hypertext or hypermedia applications and do not deal with web-specific issues. Some methods are very implementation oriented (e.g. W3DT). Some have their origin in database design methods and are heavily data driven. These data-driven methods may be able to solve some maintenance problems but they do not meet the usability problems.

We propose a method which is 'user centered' rather than 'data driven'. In a data driven method the available data is the starting point of the modeling approach. In our approach, the starting point is the set of potential visitors (also called audience or users) of the web site. In our method, these users are classified into user classes and the available data is modeled from the viewpoint of the different user classes. This results in web sites which are better tailored to their users and will therefore have a higher usability and give greater satisfaction.

Additionally, we make a clear distinction between the conceptual design (which is free from any implementation details) and the design of the actual representation, which takes into consideration the implementation language used, the grouping in pages, the use of menus, static and dynamic links, etc. This distinction is similar to the distinction made in database design between the conceptual schema (e.g. an E-R schema [Che 76]) and the logical schema (e.g. a relational schema [Dat 90]). This distinction has proven its usefulness for more than 15 years. It also allows us to propose a method for web site design which is not biased by current implementation limitations. And even more importantly, the conceptual schema can be made available to the users. This may prevent them from searching for information which is not available in the site and it may help reducing the so-called lost-in-hyperspace syndrome [Mau 96]. The conceptual schema may also be exploited by search engines to enhance their search capabilities.

In section 2 of this paper we describe some of the problems web sites suffer, and indicate to what extent existing design methods deal with these problems. Section 3 gives a general overview of the different phases in WSDM. Section 4, 5, and 6 respectively describe the User Modeling phase, the Conceptual Modeling phase, and the Implementation Design. Section 7 deals with implementation aspects. In section 8 we presents conclusions and discuss further research.

2. Designing Web Sites

There are two different kinds of web sites: the kiosk type and the application type [ISB 95]. A kiosk web site mainly provides information and allows users to navigate through that information. An application web site is a kind of interactive information system where the user interface is formed by a set of web pages. In this paper we concentrate on the kiosk type of web sites. The design issue of this type of web sites is the structuring of the (complex) information domain and to present it to the users in a clear and easily accessible way.

If the information is not well structured, **maintenance problems** will occur. These are very similar to the problems which appear in databases: redundancy, inconsistency and incompleteness. This is not surprising as web sites, like databases, provide large amounts of information which needs to be maintained. The same factors also lead to **usability problems**, experienced by the audience of the web site. Information which is repeated over and over again during navigation irritates the user. If some information on the site is found to be inconsistent, the user will probably distrust the whole site. A stale or broken link may be consider as a form of incompleteness, another kind is experienced if the user cannot find the information that, according to his opinion, should be available on the site. Another problem is that of actuality. Organizations and information are often changing so quickly that the information provided on web sites soon becomes out of date. If a web site has not been updated for several months, the confidence of the users in the provided information will probably not be very high. An additional problem, in particular experienced by the users, is related to the medium used for web sites. This medium, Hypertext, requires users to navigate through the information provided. If the navigation process is not well structured or guided, users may easily get lost (the lost-in-hyperspace syndrome [Mau 96]). This makes it more difficult and time consuming to locate the desired information.

An important, but largely neglected aspect of kiosk web sites is the many different types of visitors they may have. Different types of visitors have different needs. Consider, for example, the web site of a typical university department. Typical users of such a web site are candidate students, enrolled students, researchers and university staff.

□ Candidate students are looking for general information about the university and the content of the different curriculums.
□ The enrolled students need detailed information about the different courses, timetables and contact information of the lecturers (telephone extension, room number, contact hours).
□ Researchers look for information on research projects, publications, and general information concerning the researchers (full address, research interest, research activities).

This shows that different types of users have different information requirements. This should be reflected in the web site. For example, for a student it should be possible to follow a navigation path which provides him with the desired course and lecturer information and not much more. Usually, this is not the case. For example, at our university site, all information on a lecturer is concentrated on the home page of the lecturer which contains much more, like his research projects, research interest, publications, university functions, etc. This means that the student has to scan the page(s) in order to find the contact hours (if they are given at all).

Different types of users may also share some information requirements. If this is the case, it may be necessary to present the information in different ways to the different kinds of users. As an example we again consider the university site. Candidate as well as enrolled students may need information on courses.

- □ Candidate students, especially secondary school students are not familiar with the university jargon. By preference, the information should be offered in the local language and university specific issues should be explained thoroughly.
- □ The enrolled students are familiar with the university jargon. They also prefer to have the information in the locally common language. However, the same information is also useful to foreign exchange students. Therefore, the same information should also be given in English.

In HDM [GPS 93] the notion of having different presentations for the same content is captured by the concept of *perspective* . However, perspectives are just a syntactic device to organize information; it has no pre-defined semantics. HDM provides a representation technique to model hypertext applications, but gives little information on how to use it.

OOHDM [SR 95], [SRB 96], a successor of HDM, starts with a 'neutral' conceptual schema (using the object oriented modeling technique OMT [RBP 91]). The intended users are taken into consideration in the Navigational Design phase by means of several different navigational models. Although this method considers different types of users, (as far as we know) it does not specify how to describe or distinguish different types of users, and neither does it describe how to derive the different navigational models.

RMM [ISB 95] is based on the E-R method. It allows to split the information concerning an entity into different *slices* which could be used to support different users. However, the method uses slices only to group information into manageable packages of information.

The method in [TL 97] which focuses on web-based information systems (WBIS) is also based on the E-R method, but in addition scenario analysis is used to determine how potential users will use the WBIS to accomplish their business goals. Scenario analysis is well accepted for modeling the dynamic aspects of a system (see e.g. [Jac 93], [RPM 95]). For a kiosk site, we think the scenario analysis is less suited. In this kind of applications all dynamic behavior is of the same type: searching information. How a potential user is going to access the web's information cannot be described in a general way. Trying to do so would result in a long list of all possible ways to reach the information provided. Although a developer's intent may be to guide a user down a series of pages, the actual use may differ. A developer cannot control the links that are made to the web and so users might enter the web at any arbitrary point. Most developers will omit scenario analysis when designing kiosk sites.

In our method WSDM (Web Site Design Method, also pronounced as Wisdom) we have opted for a 'user-centered' approach. To create effective communication it is not only important to plan 'what' to communicate but also to `whom'. Therefore, we start by identifying the different types of users. Next, we describe their characteristics and their information requirements. This results in so-called *perspectives* . Only after this is done, the conceptual design of the web site starts, taking as input the perspectives and the (possibly already available) business object model of the organization.

Note that a user-centered approach is not the same as a user-driven approach. In a user- driven approach the requirements of the users (acquired by e.g. interviews and scenario analysis) largely guide the design. This is not possible for kiosk web sites on the internet because most of the users are unknown and cannot be interviewed in advance. In addition, the main goal of the site is to provide information. The key issue is to do this in such a way that both the provider and the inquirer benefit from it.

3. The Web Site Design Method (WSDM)

In our paper we present a general overview of WSDM. We describe the different phases, how they relate to each other and how they contribute to web site design. It is not possible to fully describe each phase in detail in this paper. More information can be found in [DL 98].

The method consists of the following phases: User Modeling, Conceptual Design, Implementation Design and the actual Implementation. The User Modeling phase consists of two sub-phases: User Classification and User Class Description. The Conceptual Design phase also consists of two sub-phases: the Object Modeling and the Navigational Design. Software life cycle phases such as Evaluation, Maintenance and Innovation are not considered in this paper. We also do not discuss possible iteration loops. Figure 3.1 shows an overview diagram of WSDM.

We suppose that the *mission statement* for the web site has already been formulated. The mission statement states the purpose of the web site for the organization. Without giving due consideration to the purpose of the web site, there is no proper basis for making decisions, or for evaluating the effectiveness of the web site and the users will have little idea what the web site is for (i.e. they can only guess and assume).

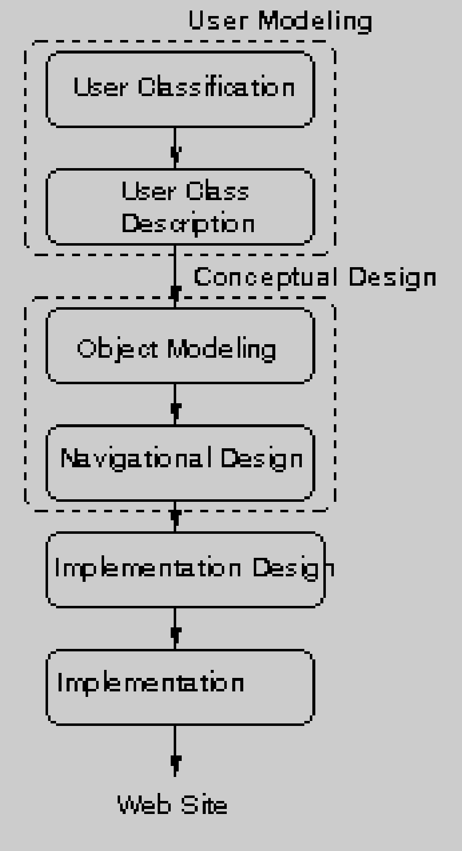

Figure 3.1: Overview of the WSDM phases

4. User Modeling

Users usually visit web sites with questions in mind. The web site should anticipate the user's questions and answer them. Therefore, in the first phase of our method we concentrate on the potential users of the web site. This is a considerably different approach then the one usually adopted. Often, designers first try to identify as much information as possible and then try to present it as fancy as possible.

4.1 User Classification

In this step we identify and classify the audience of the web site. One way of doing this is by looking at the organization or the process for which the web site will be built (this should be stated in the mission statement for the web site). Each organization or process can be divided in a number of activities. Each activity involves people. These people are potential users of the site. The activities and parties involved can be represented in a schema.

If we apply such an environment analysis to our (simplified) department example, we can distinguish the following activities:

□ Provide Education
□ Perform Research
□ Advise Companies
□ Publicize

The activity 'Publicize' is seen as a separate activity because not all publications are the result of research. If we identify the types of people involved in each activity, we arrive at the following *user classes* :

□ Candidate Students
□ Enrolled Students (of the university)
□ Researchers (departmental researchers as well as others)
□ Staff Members (of the university)
□ Companies

Activies and related user classes are represented in figure 4.1.

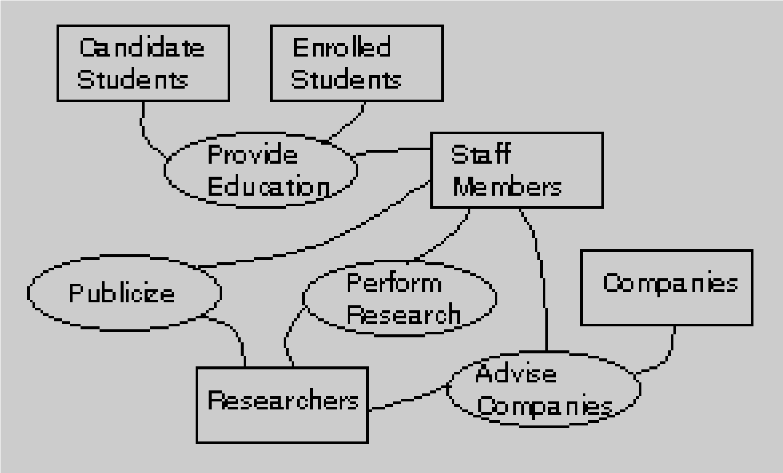

Figure 4.1: Environment schema for university department

In our method, a user class is a subset of the all potential users who are similar in terms of their information requirements. Users from the same user class have the same information requirements. User classes need not be disjoint. The same person may be in different user classes depending on the different roles he plays in the organizational environment. For example, a person can be an enrolled student as well as a staff member.

4.2 User Class Description

Next, the user classes are analyzed in more detail. The analysis has two different focuses. The first focus is on the *information requirements* of the different user class requirements (the conceptual 'what'). All users from a user class potentially have the same information requirements. However, the users within a certain user class may diverge on how the information should be presented to them. For example, as already explained, all enrolled students will require information on courses. However, the local students prefer to have this information in the national language while for foreign students the same information should be offered in English. Therefore, in this phase we also analyze the *characteristics* of the user classes. These characteristics will provide information on how the information should be presented for a particular user class (the conceptual 'who'). Some examples of a user's characteristics are: levels of experience with web sites in general, frequency of use of this web site, mandatory/discretionary use of this web site, motivation for using the web site, language issues, education/intellectual abilities, age.

If within one user class, users have different characteristics which result into different usability requirements, the user class is divide into so-called *perspectives* , e.g. in our case the user class 'Enrolled Students' may be divided into the 'Local-Student' perspective and the 'Foreign-Student' perspective.

In short, we can say that the user classes describe the information requirements and the perspectives describe the usability requirements. As an example we give the (informal) user class descriptions for the user classes Candidate Students and Enrolled Students.

Candidate Students

These users are looking for general information about the different programs and courses that may help them to make up their mind about where and what to study. Usually they are young (between 18 and 24 years), not familiar with the university jargon and speak the local language. The level of experience with the WWW may range from 'no experience' to 'a lot of experience'.

Enrolled Students

These students need specific information about the courses provided by the department, such as contents, time tables, lecturers, exam dates, preconditions, etc. They need flexible ways to search for this information. We distinguish two perspectives: the Local-Student and the Foreign-Student.

□ The **Local Students** are young (between 18 and 28), are familiar with the university jargon, the university rules and customs. They have a good level of experience with the WWW. Th prefer the local language for communication, but have in general a good understanding of English.

□ All communication with the **Foreign Students** is done in English. We may not presume that they are familiar with the university rules and customs, or with the WWW.

5. Conceptual Design

This phase consists of two sub-phases, the Object Modeling and the Navigational Design. During Object Modeling the information requirements of the different user classes and their perspectives are formally described. In the Navigational Design phase we describe how the different users can navigate through the web site. Each perspective will have its own *navigation track* .

5.1 Object Modeling

The purpose of this phase is to formally model the information requirements expressed in the user class descriptions. This is done by building (conceptual) object models for the different user classes. These models are called *User Object Models* .

A User Object Model (UOM) is a model of the business objects in the organization from the viewpoint of the users in a user class. Since a user class usually has only a limited view on the organization, a UOM will only cover part of the total set of business objects of the organization.

In conceptual modeling in general, object models describe the different types of objects, the relationships between these object types (OTs), and rules or constraints on the OTs and relationships. Object-Oriented models also describe the behavior of the objects. For our purpose, kiosk web sites, behavior description is not (yet) needed. Several methods for conceptual modeling exist, e.g. object-oriented methods like OMT [RBP 91], or the more traditional methods like E-R [Che 76] and the Object-Role Model [Hal 95]. These methods are all suitable. Figure 5.1 shows the UOM for the user class Enrolled Student (in OMT notation).

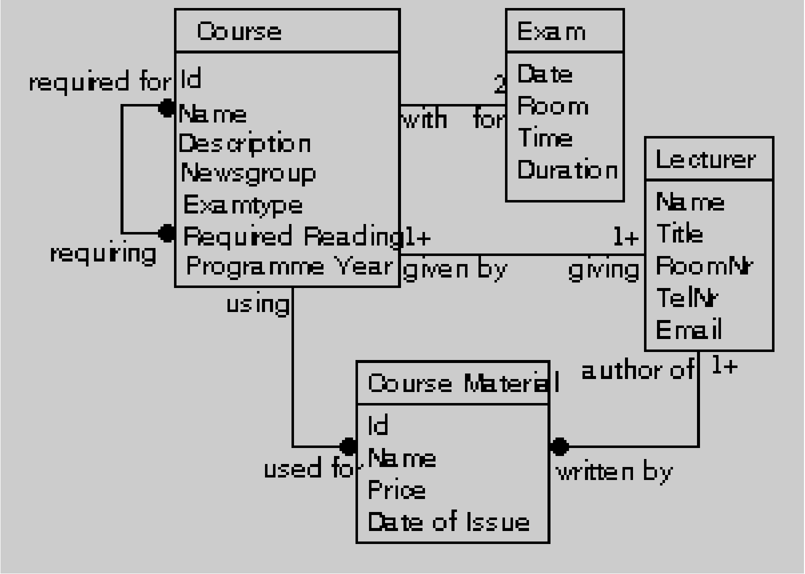

Figure 5.1: User Object Model for Enrolled Students

As explained, a user class may have different perspectives expressing different usability requirements. It is possible that this results in slightly different information requirements. For example, the Foreign-Students perspective requires the same course information as the Local-Students perspective but the information should be given in English. This can be modeled by means of *perspective variants* . A perspective variant can be compared to a subtype. For each OT in the UOM and for each Perspective, a perspective variant may be given which is a variant of the original OT. For example, for the User OT 'Course' two perspective variants may be defined, one for the Local- Students perspective and one for the Foreign-Students perspective. This is illustrated in figure 5.2. For the attributes affected by the perspective, the name of the attribute is followed by the name of the attribute variant, e.g. 'Name/English-Name' and 'Name/Local-Name'. The attribute 'Name' in the User OT 'Course' is called an *abstract attribute* , i.e. it has no representation, only its variants do.

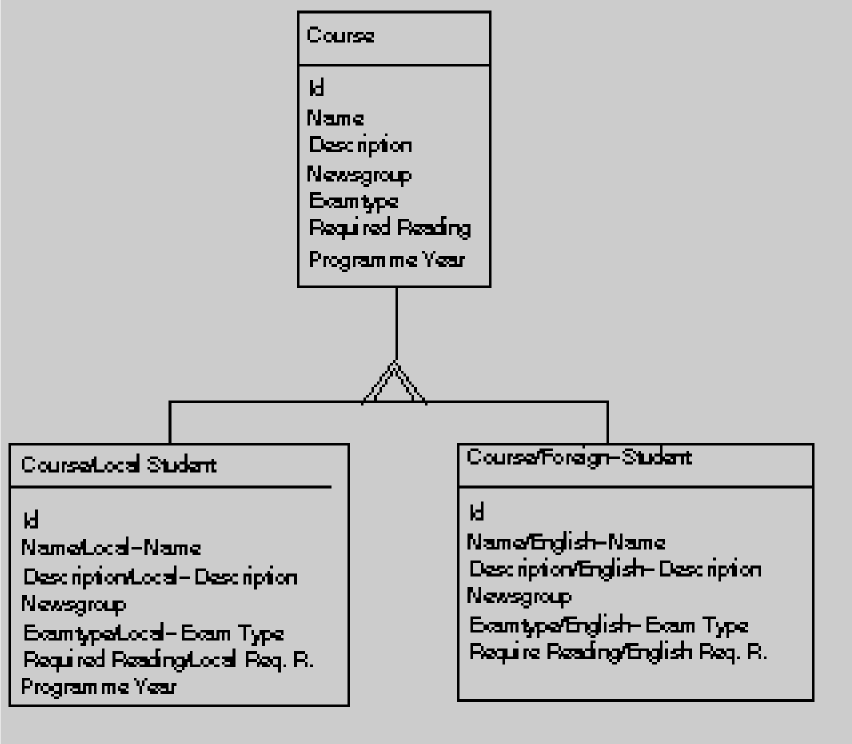

Figure 5.2: Perspective variants for Course

In this way so-called *Perspective Object Models* can be built. The Perspective Object Model (POM) for a given perspective is obtained by replacing the User OTs in its User Object Model by the corresponding perspective variants. If a User OT has no perspective variants, the original OT is kept. If a user class does not have multiple perspectives, the UOM has only one POM which is equal to the UOM. In what follows the Object Types of a POM are called Perspective OTs (POTs).

To avoid redundancy when storing the information, and related maintenance problems, the different POMs will be described as views on a single object model. This object model is in fact the *Business Object Model* (BOM), a conceptual description of the information (business objects) available in the organization. Such a business object model may already exist in the organization. If not, or if it is not specified in a format fit for our purpose, it must be (re-)developed. The classical information analysis methods mentioned earlier could be used for this. POMs are expressed as (possibly complex) views on the BOM.

5.2 Navigational Design

In this phase a *Conceptual Navigation Model* is constructed. The Navigation Model consists of a number of *navigation tracks* , one for each perspective (POM). A navigation track expresses how users of a particular perspective can navigate through the available information. This is described in terms of *components* and *links* . We distinguish *information components* , *navigation components* and *external components* . (see figure 5.3).

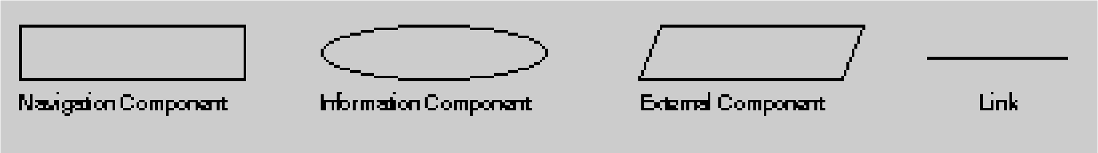

Figure 5.3: graphical representation of the navigation concepts

Each information component corresponds with a Perspective OT (POT). As a POT may be associated to other POTs (by means of relationships), an information component may contain links to other components to allow navigation to associated information. A navigation component can be seen as a grouping of links. An external component is actually a reference to a component in another site.

To arrive at a Navigational Model, we construct a navigation track for each POM. Such a navigation track consist of three layers. The top layer is the *context layer* and is used later on to connect the different navigation tracks. The middle layer, *the navigation layer* provides different ways to access the information. The lowest layer is the *information layer* and contains the actual information by means of the information components.

We now briefly describe an algorithm to construct a navigational track for a POM. A more detailed description can be found in [DL 98].

1. **The context layer**. The navigation track starts with a navigation component with the same name as the perspective. This forms the top level of the navigation track.
2. **The information layer**.
   - □ Each POT becomes an information component or an external component in the navigation track. The choice between information component and external component is determined by the availability of information.
   - □ For each relationship between POTs a link is created.
3. **The navigation layer**. This layer connects the context layer and the information layer and has to be designed according to the needs of the users of the perspective. The following procedures may be used for this.
   - □ Each information component should be within reach of the context layer. Therefore, for each POT a navigation component named after it is constructed and linked to the context level.
   - □ If it is necessary to access the same information component in different ways, intermediat navigation components and links are created.

Figure 5.4 and figure 5.5 show the navigation tracks for respectively the POMs 'Foreign-Students' and 'Local-Students'. Notice that the navigation layer for Foreign-Students contains less navigation components than the navigation layer for Local-Students. This is because foreign-students do not follow a programme, they only follow courses. Therefore, they do not need to search exams and courses by programme year but only by the course name.

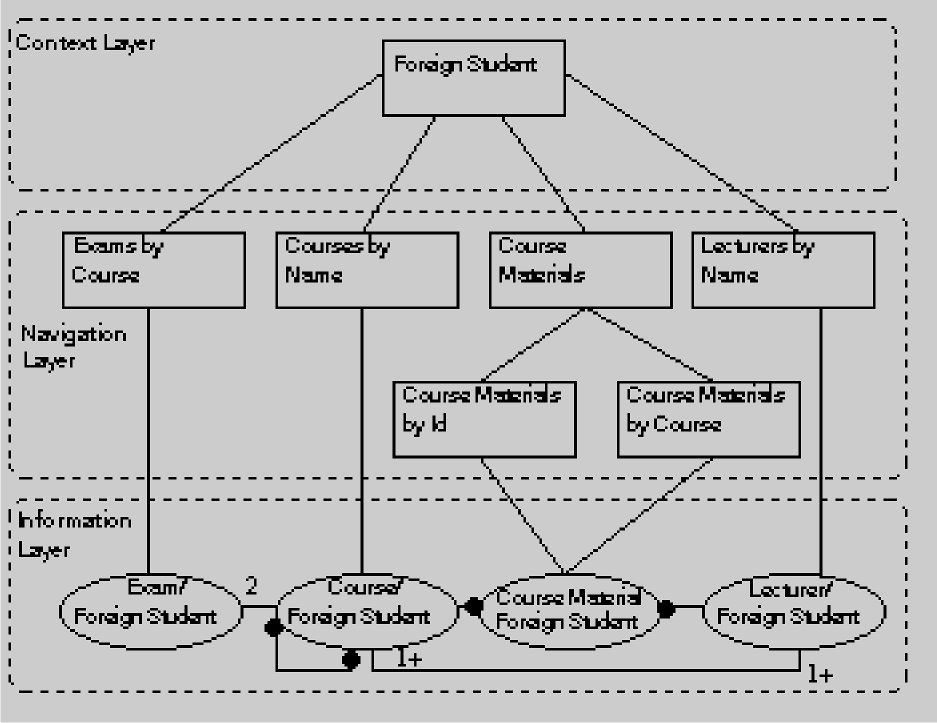

Figure 5.4: Navigation track for Foreign-Students

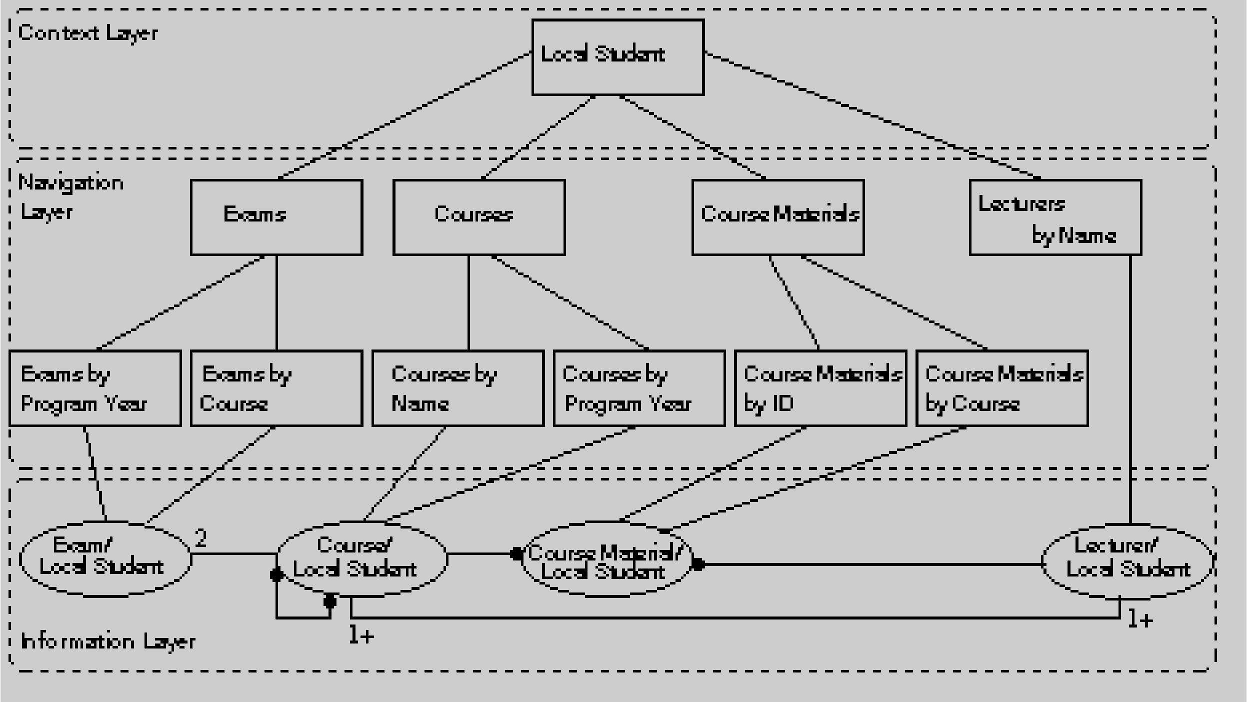

Figure 5.5: Navigation track for Local-Students

6. Implementation Design

In this step we essentially design the 'look and feel' of the web site. The aim is to create a consistent, pleasing and efficient look and feel for the conceptual design made in the previous phase. There is a lot of literature on these aspects of web site design, see e.g. [DG 95]. Due to space limitations, we only mention how some of the well known guidelines fit into WSDM. The result of the implementation design is an *Implementation Model* .

- □ **Package information in right-sized chunks**. Humans can process only a limited amount of information at a time. Therefore, the amount of information on any page should not overwhelm the user. In addition it should also not take a long download time. In WSDM, the web developer can decide to cluster components connected by links and to represent them on a single page. Navigation components can be implemented by means of (ordered or unordered) lists.
- □ **Use of index page**. An index page links to every page of the entire web site, it provides a central point for the user to locate a page in the web site. This index page may be seen as a simplified version of our conceptual model. We suggest to replace the index page by a representation of the conceptual model of the site, which not only allows the user to locate information but also helps the user to build a mental model of the site.
- □ **Use context and information cues**. Explicit statements about the target audience as well as on the purpose of the page can help the user to see immediately whether the information on that page is of interest. We should avoid creating pages that have no cues. If a page is publicly available, provide enough cues so that a user can place the information in the global context of the WWW (e.g. organizational information, copyright notices). This guideline especially suits WSDM. A web site constructed with WSDM consists of a number of navigational tracks, one for each perspective. Each perspective has a different audience and possibly a different purpose. Indicating the target audience and the purpose links up nicely with the concept of perspective.
- □ **Use navigation cues**. Web sites designed with WSDM are centered around the concept of (a user's) perspective. Users which follow the links from the start of a navigational track usually stay within one perspective. However, the pages may contain external links and inter-perspective links (links between pages from different navigational tracks). It is good practice to use different navigation cues for links to external pages and inter-perspective links. In this way, the user will be aware of his leaving the perspective or the local site.

It should be noted that the design of the implementation may depend on the chosen implementation environment. Limitations of the implementation language (such as HTML) may interfere with this process.

7. The Implementation

The last phase is the actual realization of the web site using the chosen implementation environment. For example, for an HTML implementation this means that the implementation model must be converted into a set of files containing HTML source code. This step can be largely automated using available tools and environments for assisting in HTML implementations.

To improve maintainability, more and more web sites have their underlying information (or parts of it) stored in a database. Pages are generated (usually, but not necessarily, at runtime) by extracting the necessary information from the database. This information may be extracted from existing databases or a new database may be implemented for this purpose. In our method, the basis for such a database design is found in the BOM. The BOM is actually the conceptual schema of the database as it describes the available business objects at a conceptual level. From the conceptual schema, a logical database schema can be generated (using appropriate database development tools), or manually built. The queries needed to extract the information needed for the pages should be easily derivable from the POMs because these are expressed as views on the BOM. More information on linking relational databases to web sites can be found e.g. in [Gre 97].

8. Discussion and Further Work

In this paper, we presented a method for designing kiosk web sites. The method is based on the principle that the web site should be designed for and adapted to its users. Therefore, the emphasis in this method is on user modeling and conceptual modeling rather than on layout and presentation aspects. In principle, an implementation can be generated once the conceptual design is completed.

Ongoing research aims at refining and elaborating this approach:

- The algorithms to construct the navigation model and the implementation model need to be refined to offer more flexibility. Other algorithms may be needed that result in different types of web sites like e.g. a guided tour site.
- User Classes are not always completely different. We may also consider to use User Subclasses which may have an impact on the rest of the method.
- Web sites are dynamic in nature. Designing them is an ongoing continuous process. We claim that the method is flexible enough to support this incremental and evolutionary design process.
- Issues of multiple site developers and of existing (sub-)sites should also taken into consideration.

In addition to the issues already mentioned it is of course important that the method is extended towards application types of web sites.

References

[BN 96] M. Bichler, S. Nusser
W3DT - The Structural Way of Developing WWW-sites, Proceedings of ECIS'96, 1996

[Che 76] P.P. Chen
The Entity-Relationship Model: Towards a Unified View of Data
ACM Transactions on Database Systems, Vol 1 no 1, 1976, 471-522.

[Dat 90] C.J. Date
An Introduction to Database Systems
Volume I, fifth edition, Addison-Wesley, 1990.

[DG 95] J. December, M. Ginsberg
HTML and CGI unleashed
Sams.net Publishing, 1995.

[DL 98] O.M.F. De Troyer, C.J. Leune
WSDM: the Web Site Design Method WiSDoM
Internal Report in preparation,Tilburg University, 1998.

[Gre 97] F. Greenspun
F. Greenspun's, Database Backed Web Sites:The Thinking Person's Guide to Web Publishing
ZD Press, 1997

[GPM 93] F. Garzotto, P. Paolini, L. Mainetti
Navigation paaterns in hypermedia databases
Proceedings of the 26th Hawaii International Conference on System Science, IEEE Computer Society Press, pp. 370- 379, 1993

[GPS 93] F. Garzotto, P. Paolini, D. Schwabe
HDM - A Model-Based Approach to Hypertext Application Design
ACM Transactions on Information Systems, Vol 11, No 1, pp. 1-26, 1993

[Hal 95] T. Halpin
Conceptual Schema and Relational Database Design
Prentice Hall Australia, second Edition, 1995

[ISB 95] T. Isakowitz, E. A. Stohr, P. Balasubramanian
RMM: A Methodology for Structured Hypermedia Design
Communications of the ACM, Vol 38, No 8, pp. 34-43, 1995

[Jac 93] I. Jacobson et al.
Object-Oriented Software Engineering
A Use Case Driven Approach, Addison-Wesley Publishing Company, 1993.

[Mau 96] H. Maurer
Hyper-G - The Next Generation Web Solution
Addison-Wesley 1996

[Nie 95] J. Nielsen
Multimedia and Hypertext the Internet and Beyond
Academic Press, 1995

[RPM 95] D. Redmond-Pyle, A. Moore
Graphical User Interface Design and Evaluation, A practial Process
Prentice Hall, 1995

[RBP 91] J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy and W. Lorensen
Object Oriented Modeling and Design
Prentice Hall Inc., 1991.

[San 96] D. Sano
Designing Large-scale Web Sites

Wiley Computer Publishing, 1996

[SR 95] D. Schwabe, G. Rossi
The Object-Oriented Hypermedia Design Model
Communications of the ACM, Vol 38, No 8, 1995

URLs

[SRB 96] D. Schwabe, G. Rossi, S.D.J. Barbosa
Systematic Hypermedia Application Design with OOHDM
http://www.cs.unc.edu/barman/HT96/P52/section1.html

[TL 97] K. Takahashi, E. Liang, Analysis and Design of Web-based Information Systems
Sixth International World Wide Web Conference, 1997
http://www6.nttlabs.com/papers/PAPER245/Paper245.html

Vitae

Olga De Troyer received her Ph.D. in 1993 from Tilburg University, The Netherlands. She has held research positions at Control Data's Research Lab in Brussels and at the University of Hasselt (Belgium). Currently, she is an assistant professor at Tilburg University and a visiting professor at the Free University of Brussels. Her research interests include conceptual modeling, object oriented modeling, schema transformations, CASE tools, user interface design and web site design.

Kees Leune graduated from Tilburg University in 1997. Currently he is working as a researcher at the Infolab at Tilburg University. His research interest include object oriented modeling, conceptual modeling and all kind of Internet related activities.

ey

d

t

e