Conceptual Navigation Analysis: a device and platform independent navigation specification

CRISTINA CACHERO
Universidad de Alicante. Spain
and
NORA KOCH
Ludwig-Maximilians-Universität München. Germany
and
JAIME GÓMEZ
Universidad de Alicante. Spain
and
OSCAR PASTOR
Universidad Politécnica de Valencia. Spain

Most conceptual modeling approaches for existing Web applications are based on a design concept, the page, on which the navigational specification of the system is usually built. However, their increasing personalization and mobility demands, together with the high pace at which new devices and technologies are appearing, are causing such applications and their related models to become increasingly complex and less reusable. This problem would be lessened by integrating a navigation analysis model exclusively based on the different user requirements. Supporting this idea, in this article we introduce two constructs that increase the level of abstraction at which navigational aspects are captured, namely the Navigation Semantic Unit (NSU) and the Navigation Semantic Link (NSL), and which are in the core of a navigational model independent from users, platforms and access devices. Also, we outline how this model can be integrated in a range of user-oriented hypermedia modeling approaches, and influence the way their navigation design models are constructed.

Categories and Subject Descriptors: D2.1 [**Software Engineering**]: Requirements Specification—*Methodologies*; D2.2 [**Software Engineering**]: Design Tools and Techniques—*Object-oriented Design Methods*; H5.2 [**Information Interfaces and Presentation**]: User Interfaces—*User Centered Design*; H5.4 [**Information Interfaces and Presentation**]: Hypertext / Hypermedia—*Navigation*

General Terms: Design, Human Factors, Standardization

Additional Key Words and Phrases: Web Engineering, Semantic Navigation, Analysis Model, Object-oriented Modeling, Visual Models

---

This paper has been partially supported by the Spain Ministry of Science and Technology (project TIC2001-3530-C02-02), and by the European Union within the IST project AGILE: Architectures for Mobility (IST-2001-32747).
Authors' addresses: Cristina Cachero, ccachero@dlsi.ua.es, Universidad de Alicante, www.dlsi.ua.es; Nora Koch, kochn@informatik.uni-muenchen.de, Ludwig-Maximilians-Universität München, www.pst.informatik.uni-muenchen.de; Jaime Gómez, jgomez@dlsi.ua.es, Universidad de Alicante, www.dlsi.ua.es; Óscar Pastor, opastor@dsic.upv.es, Universidad Politécnica de Valencia, www.dsic.upv.es

1. INTRODUCTION

Since the appearance of the Web as a new software development environment, many research efforts have been directed towards the definition of any kind of navigation model that helped to improve the quality of the application structure. In this sense, first proposals for such design navigation models regarded the Web as a directed graph whose nodes were Web pages and their arcs corresponded to the links between such pages [Florescu et al. 1998; Garzotto and Paolini 1993]. However, the increasing level of complexity of Web applications have caused this page concept to become clearly insufficient.

The resulting evolution of perspective has been reflected in a set of methods and methodologies that cover different aspects of an already well-settled discipline - the so called *Web Engineering* [Murugesan et al. 1999; Ginige and Murugesan 2001; Koch 2001] - that aims at providing the Web-based system development with a systematic approach and a quality control and assurance. In spite of this fact, again the rapid development and high degree of pragmatism that have characterized the Web evolution have promoted a more or less tight coupling of the models defined in the different proposals with design concerns. This feature is a clear drawback in an environment characterized by the burst of new architectures, access devices, protocols, languages, or even the appearance of novel organization demands (such as mobile software, that is, software not bound to one particular machine, [AGILE 2002] or adaptive Web applications [Koch 2001]) and, from our point of view, it is undermining the reuse and adaptation capacity of conceptual proposals. Also, there exists no ontology that allows the communication among researchers and practitioners, what is clearly reflected in the fact that terms such as 'service', 'navigation', or even 'pattern' may have different meanings (even sometimes contradictory) in different contexts, including different hypermedia conceptual modeling proposals.

In this article we will center on one of the perhaps more controversial concepts in the hypermedia community: the concept of navigation, and we will propose both a definition and a device, platform and user independent conceptual model. Our aim is not to make an authoritative statement but to serve as a starting point to arise a (we hope) fruitful discussion on its implications and limits. We are conscious that the readers' different educational and research background may add very important turns to the concepts here defined, but we think time has come in Web Engineering to reach an agreement on certain 'first-class hypermedia citizens', in order to achieve a greater research synergy that eventually allows the hypermedia community to provide enterprises with the kind of methods and tools they require.
In order to get our aim, the article is structured as follows: first, section 2 and 3 present the concepts of Navigation Semantic Unit, around which the navigation analysis task is performed, and Navigation Semantic Link, that provides access to such units. With these constructs, section 4 sketches a possible definition of navigation analysis. The traceability from the navigation analysis to navigation design concepts is illustrated in section 5. Section 6 discusses relationships between current proposals and the concepts here presented. Last, section 7 presents the conclusions of this work.

2. NAVIGATION SEMANTIC UNITS

A *Navigation Semantic Unit (NSU)* can be defined as a set of information and navigation structures that collaborate in the fulfillment of a subset of related user requirements. This concept is the basis for the concept of *Semantic Navigation* understood as a **voluntary invocation of a NSU**. This invocation must furthermore imply a change in the user interaction aim. Hence, NSU differ from current *node* definitions (e.g OOHDM navigation nodes, defined as views over conceptual objects) in that NSU capture the intention of the user and therefore are not necessarily related to the conceptual objects required to fulfill these objectives. In this article we claim that NSU, together with *Navigation Semantic Links* (NSL, which will be presented in section 3) are enough to define a sound navigation analysis model that can be shared among different implementations (be them for different users, platforms and/or target devices). This analysis model must not be confused with the kind of design-oriented navigation models included in most current conceptual modeling approaches [Ceri et al. 2000; Koch et al. 2001; Gómez et al. 2001; Schwabe et al. 2001; de Troyer and Casteleyn 2001], which can be regarded as refinements on this analysis model and that are further discussed in sections 5 and 6.

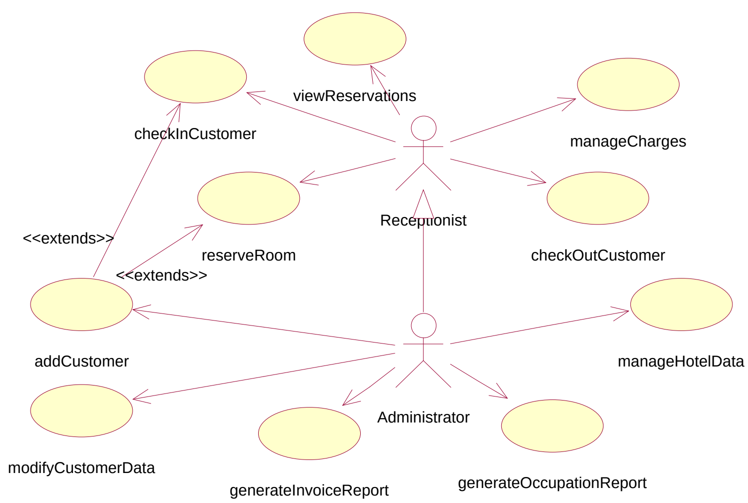

Fig. 1. *Hotel Management System Use Case Diagram*

In order to illustrate the NSU concept, a small example is going to be employed all along the paper: a *Hotel Management System*. In this system, let's imagine we have identified two roles: the *receptionist* and the *hotel administrator*. Let's also suppose that the receptionist is allowed to *view current reservations*, *reserve a room* (adding also the related customer if necessary), register a *customer check-in*, note down the services (laundry, room breakfast, etc.) provided to such customer while staying in the hotel (*manage charges*) and register the customer departure (*check-out customer*). On the other hand the hotel administrator may not only perform the above mentioned tasks, but also generate a set of reports (namely, *generate*

Invoice Report and *generate Occupation Report*), *modify the customer data* and manage any other hotel characteristics, e.g. the kind of services offered to customers or room characteristics, (*manage Hotel Data*). These requirements may be gathered by means of a UML-compliant Use Case Diagram [UML1.4 2001] such as the one depicted in Fig. 1.

Before defining the NSU derived from this description, we would like to clarify some aspects that might mislead the designer in the NSU definition process.

2.1 Factors interfering in the definition of a NSU

The first confusion derives from the implicit identification of NSU with the concept of 'entity' (in a relational database sense) or 'business class' (in a UML sense). On the contrary, a NSU provides a higher level of abstraction. In fact a NSU is a concept far less elaborated than that of an entity or a business class. Our objective when defining NSU is to base the navigation definition on concepts that are not necessarily related to the underlying domain information structure. For example, and supposing we are in an Object-Oriented environment, concerns such as responsibility assignment for attributes and/or methods, transient or permanent objects or relationships, etc. are irrelevant, from our point of view, at this level of abstraction. Also, information and functionality sets included in different NSU are not necessarily disjoint.

Another controversial concept is the definition of nodes following a given user interaction model. We think that, at this level of abstraction, we shouldn't depend on features such as the number of times that the user interacts with the application (even if that implies a new server access) in order to get the desired results. In fact, this user interaction model may greatly vary depending on target devices or even efficiency considerations that, we think, should't influence any analysis model. Otherwise stated, the information splitting process into (static or dynamic) pages, no matter the criterion applied, is not necessarily related to a change in the user aim.

Back to our example, in Fig. 2 we can observe the Analysis Navigation Model corresponding to the *Hotel Management System*, and how the set of requirements specified in Fig. 1 can be orchestrated around five NSU, each one depicted as a UML class symbol decorated with a ≪*semantic unit*≫ stereotype:

—Reservations: interface related to hotel reservation management.

—Stays: interface related to hotel occupation management.

—HotelAccounts: interface related to hotel departure and payment management.

—Customer: interface related to customer management.

—HotelManagement: interface related to general hotel characteristics management.

For each NSU, a subset of the concepts and service logic interfaces involved in the solution of the corresponding user requirements (relevant from a user perspective and so not necessarily equal neither in number nor in nature to those that will appear in the domain structure model) have been defined . Note how these atoms, as we stated above, may be repeated among different NSU. This repetition does

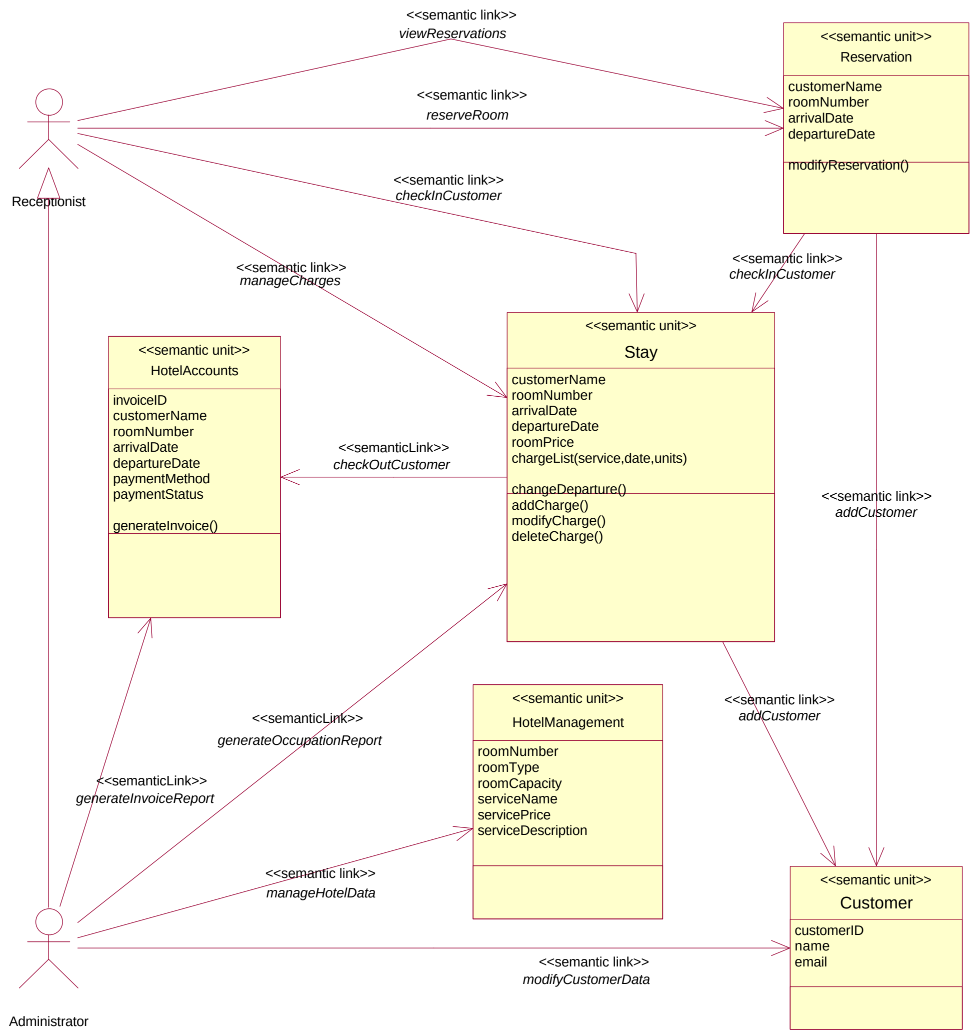

Fig. 2. *Hotel Management System Navigation Analysis*

not necessarily imply a common mapping rule to any underlying domain concept. For example, we may be interested in different attribute *perspectives* [Garzotto and Paolini 1993] appearing in different NSU, which in turn may imply different structural attributes providing the contents to that navigation analysis attribute. Also, the way operation calls are performed may differ from one NSU to another. The NSU provides a context to such call, what may imply for example that certain parameters are implicitly provided by such context in some cases and explicitly introduced by the user in others.

Also from our example we can derive another advantage of performing this NSU analysis process: the consideration of the system as a whole allows an early identification of reusable views. For example, in Fig. 1 we observe how the hotel administrator is allowed to add new customers. Also we observe how the use case *reserveRoom* may also make use of such functional requirement. This fact has suggested the convenience of defining an independent NSU to deal with customer data, providing in this way an homogeneous view that may be included in any (present or future) user requirement that required from such kind of customer management. Acting the other way round, that is, departing from detailed analysis of each user and/or requirement and merging the sub-models afterwards may cause an extra modeling effort, due on one hand to this merging process and on the other hand to the more difficult process of coming up with such redundant views, due to a lack of a bird-eye perspective such as the one that the navigation analysis provides.

As a final remark, it is true that the final set of NSU depends on the designer criterion, even if s/he should bear in mind the user perception of the application subsystems. Nevertheless, this arbitrariness exists in every modeling process: eventually it is the experience and individual skills of the designer what determines the quality of the model, including an analysis navigation model.

3. NAVIGATION SEMANTIC LINKS

Once the NSU have been defined, *Navigation Semantic Links* (NSL) provide the materialization of a change in the navigation intention of the user. NSL can either connect different NSU or provide the user with a direct access path to fulfill a given requirement. In Fig. 2 the different NSL are depicted by means of arrows labeled with the corresponding requirement. It is important to note how more than one NSL can provide a path for the fulfillment of each requirement (see e.g. the two NSL providing a path to fulfill the *Check-in Customer* user requirement in Fig. 2.)

The most important characteristic of this construct is that **the existence and homogeneity of semantic links must be preserved in subsequent design models, and thus be warranted in every final implementation, regardless of user, platform or access device**. With homogeneity we refer to a preservation of certain design characteristics and behaviour among the different design models. The number and value of such characteristics may vary depending on the specific design navigation model provided by each hypermedia modeling proposal. In this way we assure a coherence among different implementations of the same application, what on its turn facilitates the adaptation of previous users (or users with other roles), who will find a familiar structure in the new implementation. We would like to stress the fact that those links should be independent from designer punctual decisions, such as the inclusion of a new index to further organize the information, the application of certain interface patterns (e.g. addition of a header and a footer to every page) or the pagination of the view. Furthermore, with this concept we are introducing a hierarchy [Yoo and Bieber 2000] in the importance of the links, which in turn may influence several design features:

—Visually speaking, the definition of a NSL may imply its automatic inclusion in direct access menus. They could be seen as functional landmarks, that is, main gates to the functionality offered by the applications. Additionally, special colours, fonts etc. might be defined to set them off from 'design' links.

—Logically speaking, the definition of a NSL may imply a physical navigation step. Otherwise stated, information giving an answer to two different user requirements should not appear in the same 'design page'.

—Also the kind and amount of contextual information that can be transmitted along the link should be limited. If NSL imply a semantic step (a change in the user intention), they should be as light as possible, in order to make this cognitive change easy to be performed.

4. THE SEMANTIC NAVIGATION CONCEPT

Once the NSU and NSL concepts have been introduced, we can redefine the concept of semantic navigation as a *the materialization of a user change of intention or, otherwise stated, the act that the user performs when s/he traverses a NSL*. This statement includes two important assumptions. The first one is that, due to the fact that *the user is the only one that can navigate*, every link activated *de motu propio* by the application does not imply navigation. The second assumption is that, due to the fact that *user requirements are the basis on which NSU are defined*, they are in fact defining what is or is not navigation. Therefore this concept substantially increases the weight of a correct requirements definition process in the quality of the final interface.

5. TRACEABILITY FROM NAVIGATION ANALYSIS TO NAVIGATION DESIGN

The main characteristic of the NSU and NSL definition presented above is that they are independent from design considerations. Back to our example, let's focus on the receptionist interface. In Fig. 2 we saw how the five requirements corresponding to the receptionist were covered by means of four out of the five possible NSU: *Reservation*, *Stay*, *Customer* and *HotelAccounts*. This analysis may be used during the design navigation process either in a direct or indirect way:

(1) Used in a direct way, the designer departs from the navigation analysis model and, getting help from the domain model, defines both inner navigation structures and how analysis concepts are mapped to the underlying domain structure, in order to achieve a full traceability of the models.

(2) Used in an indirect way, the designer departs from the domain model to construct the design navigation model, and uses the analysis navigation model to take decisions about certain design navigation features.

The first approach generates very intuitive diagrams, where the domain structure may remain hidden if desired. However, this approach requires the inclusion of a new *mapping activity* where any kind of derivation rule or even new diagrams showing the domain elements collaborating in the definition of each NSU must be defined. The second approach, on the contrary, implicitly provides the 'gathering' paths through the domain model that make up the information encapsulated within the boundaries of each NSU. The main disadvantage of this approach is however that diagrams become more complex. Whatever the case, we think that the mapping process from navigation structures to domain structures should be dealt with at design time, and not before. In this way, the navigation analysis remains exclusively focused on a user-oriented *what*, while the navigation design phase includes, among other aspects, the definition of the equivalence between analysis elements and underlying domain structures, as well as an enrichment process by means of inner navigation constructs (a domain-related *how*).

5.1 Navigation Semantic Units and the page concept

Also, as the reader might have already inferred, the fact that NSUs may include inner navigation structures implies that, in our example, more than five 'abstract pages' (the number of NSU, see Fig. 2) regarding information structures (that is, without counting the pages generated due to the addition of access constructs) can be derived from the analysis navigation model. For example, and centering on the interface corresponding to the *viewReservation* requirement, the first step of such use case might imply on a traditional browser the visualization of a single page with the whole set of reservations (customer name, arrival date, departure date, room number) and a form to modify any reservation. However, on a PDA perhaps splitting this set in two, that is, presenting on one 'page' the reservation data, and on a different one the form to modify a given reservation, and providing different access structures (e.g. grouping the *reservations* in groups of five, with the help of a *guided tour*) could be more appropriate. Other platforms and/or access devices might suggest the use of an index that provided access to detailed information about a given *reservation*. Moreover, implementation restrictions (e.g. some underlying data structures having costly ordering algorithms associated), might suggest the inconvenience of such indexes being defined. Does the *viewReservation* requirement change? Obviously not, what from our point of view implies that the analysis navigation model should remain invariant. This example shows how this two-step process, consisting on developing a navigation design model based on a navigation analysis model, allows the abstraction during the analysis process from characteristics of each target device type, what makes it specially suited for the design of mobile software.

We can already advance a consequence of this definition: *access structures (indexes, guided tours etc.) do not make up independent NSU and therefore do not imply navigation and should not be part of the analysis model.* Also, *NSU do not restrict the maximum number of 'core' pages included in an application interface.* We claim these two assertions are fundamental if we are to split up the concept of navigation from the final presentation of such navigation.

On the other hand, the view the user has on the analysis model (and the way s/he navigates through it) certainly changes. It is precisely this variation what constitutes, from our point of view, the design navigation concern. We do think design models showing particular navigation views that take into account any kind of implementation consideration are needed, and have in fact proven useful when used in the context of a number of modeling proposals. However the definition of such models without an additional reflection on a general navigation structure is dangerous, and can derive models exclusively driven by underlying data structures instead of being based on actual user needs.

Due to space constraints, next we are just briefly discussing the consequences of including an analysis navigation model such as the one presented in this paper in two well known hypermedia design methods, namely OO-H (Object-Oriented Hypermedia method) [Gómez et al. 2001; Cachero et al. 2001] and UWE (UML-based Web Engineering) [Koch et al. 2001]. Interested readers may find a detailed description of this inclusion process, corresponding to the *Hotel Management System*, at $http://www.pst.informatik.uni-muenchen.de/publications/$

5.2 Navigation analysis implications on navigation design models

The navigation design modeling phase of OO-H is reflected in a set of Navigation Access Diagrams (NAD) that are partially based on the class diagram that represents the domain structure of the system. In this method, the consideration of a navigation analysis model such as that depicted in Fig. 2 as an activity prior to NAD design has caused a set of restrictions to be applied on the NAD models. For the sake of this article, the most relevant is that NAD links corresponding to NSL will always imply a user change of view (which is reflected in its corresponding *Show-in* tagged value set to *destination*) and a demand for the user to explicitly start such navigation step (*Activation=manual*). Furthermore, the parallelism drawn between design links and NSL provide a powerful mechanism to make sure that the final interface is complete, that is, provides at least one way to answer every user requirement. Many more implications may arise with the enrichment of semantic links with new semantic attributes, as will be suggested in section 7.

UWE (UML-based Web Engineering) defines an iterative UML-based process for Web applications that includes a navigation task involving two models: a *navigation space model* and a *navigation structure model*. If we change the UWE process to include the navigation analysis modeling this will only influence the construction of the navigation space model, i.e. the guidelines given for the modeling of the navigation structure will remain unchanged. In UWE the navigation space model is a model at analysis level while the navigation structure model implies design decisions. The navigation space model has some similarities with navigation analysis model. Both are based on the identification of nodes and links. Both introduce two concepts and define two stereotypes for class and association, respectively. The concepts used in the navigation analysis model are the navigation semantic unit and semantic link, while the concepts used in the navigation space model are the navigation class and the direct navigability. However, the semantics of such constructs is different. While the node definition process in UWE is based on a domain structural model, nodes in the analysis navigation model defined in this paper are only based on use cases and user navigation intentions, what causes the results of the node definition to be different.

6. RELATED WORK

The navigation concept is not exclusive of the hypermedia interfaces. On the contrary, we agree with the Hypermedia Design Philosophy [Yoo and Bieber 2000] in that navigation analysis and design is important for any kind of interface model. However, its relevance inside the Web application design process lifts it to a key position for the deployment success of the system. In this sense, and to our knowledge extent, all hypermedia modeling methods studied so far (see e.g. [Atzeni and Parente 2001; Garzotto et al. 1999; Ceri et al. 2000; Schwabe et al. 2001; de Troyer and Casteleyn 2001], to name a few) recognize the need to include a navigational model in their design process. Such models, usually defined at a logic level of abstraction, share some characteristics. Perhaps the most important one is that, due to the influence of concepts appearing in first generation proposals such as HDM [Garzotto and Paolini 1993] or RMM [Isakowitz et al. 1995], they regard the *navigation nodes* as the minimum information grouping unit, inside which no further navigation structures nor information splitting processes can be defined. Our contribution in this sense has been to rule out the implicit assumption that nodes must correspond to domain concepts (see e.g. the *HotelAccounts* NSU in Fig. 2). Our proposal regards a NSU as a super-navigation-node/entity that comprises all the information the user requires to fulfill a subset of his functional goals. This conception differs from other functional modeling approaches such as the User Interaction Diagrams of [Vilain et al. 2000] (UID) in that, while the latter describes the user-interface interaction topology in terms of server accesses, we center on the topology and interrelations among user requirements in terms of just semantic relationships. Therefore, our proposal increases the level of independence of existing navigation models. Furthermore, we have provided this node concept with inner navigation/access capabilities that, defined during the navigation design phase, permit the node to be adapted to new user interaction models, new platforms or new access devices without changing its nature.

Also, most hypermedia modeling methods share the identification of user navigation with the activation of an anchor (physical link) that changes the user view. From our point of view, semantic navigation does not need to be associated with a change in the user view, but with a change in the user intention, and so other interaction modes (perhaps the most relevant being the invocation of services) may also imply navigation. In fact, most of the navigation structures included as part of current navigation modeling proposals are related to this *Intra-node navigation*, which is the reason why we consider they are artifacts of a navigation design phase. On the contrary, at analysis level we center on *Inter-node navigation*, represented in our proposal by means of semantic links, that are partially based on the philosophy that drives the *Relationship Navigation Analysis (RNA)* process presented in [Yoo and Bieber 2000]. Although we are much more restrictive in the identification of semantic links (most of the RNA-derived links might still fall from our point of view in the above mentioned intra-node navigation category), we agree with [Bieber 1999] in that the discovery of additional semantic relationships, and its categorization by means of additional metadata may greatly improve the Web application success. As an example, it might be advisable to define that the activation of a given semantic link is giving a monetary benefit to the enterprise (imagine for example semantic links related to selling products or services in an e-commerce application). This semantic categorization (which has nothing to do with how nor through which information structures the user is going to navigate in order to perform such purchase) may become a basis on which to define different application criteria: menu ordering, global availability of links, etc.

Summarizing, the definition of a navigation node as a semantic super-entity causes its different views (defined for different user interaction models, platforms and devices) to restrict or subdivide, but never change, such node concept. This fact allows the reflection and the reuse of global navigation structures among different implementations of the same application, what in turn increases the level of comfortability of users facing new roles and/or new application implementations. We do believe that such change of perspective, based on a higher abstraction effort, would be of benefit for the hypermedia community. Additionally, we expect this change of perspective to diminish controversy when trying to classify many other interaction modes that may not be so directly associated with anchors and/or a change of 'abstract page' process.

7. CONCLUSIONS

In this article we have defined the concept of 'semantic navigation' as a *voluntary invocation of a NSU*. We regard a *NSU* as a super-navigation-node/entity that comprises all the information (both content, functional and navigation related) that the user requires to fulfill a subset of their functional goals. Once the set of Navigation Semantic Units has been defined, *Navigation Semantic Links* (NSL) provide the materialization of a change in the navigation intention of the user, by means of connections between the user and the NSU and/or between two NSU.

The navigation analysis model we are proposing is an attempt to provide a model that should be device, platform and audience independent. Therefore this model can be used as basis for mobile and portable applications as well as for applications that can be expanded to support different personalization schemes. The proposed navigation model can be integrated in well-known object-oriented hypermedia/Web design methods, and provides a common ground for the definition of their respective navigation design models. Our final objective is to make a clear difference between navigation analysis and navigation design concerns and to provide for both diagrammatic tools that support the modeling process of Web applications. Interested readers on the results of our first efforts towards the definition of a (non-exhaustive) list of navigation analysis and navigation design concerns are referred to $http://www.pst.informatik.uni-muenchen.de/publications/$.

This work is just 'a drop in the navigation ocean'. Open research lines include, just to name a few, the enrichment of semantic links with attributes that improve their effectiveness, integration of the model in other well-known hypermedia proposals and tool support for both the model and its traceability to the different navigation design models that it may derive.

ACKNOWLEDGMENTS

We would like to thank Daniel Schwabe for his useful comments on this work.

REFERENCES

AGILE 2002. AGILE project. http://www.pst.informatik.uni-muenchen.de/projekte/agile/.

Atzeni, P. and Parente, A. 2001. Specification of Web applications with ADM-2. In *Proceedings of First International Workshop on Web-Oriented Software Technology IWWOST'01*. Valencia University of Technology, 99–129.

Bieber, M. 1999. Hypermedia: A Design Philosophy. In *ACM Computing Surveys*. ACM.

Cachero, C., Gómez, J., Párraga, A., and Pastor, O. 2001. Conference Review System: A Case of Study. In *Proceedings of First International Workshop on Web-Oriented Software Technology IWWOST'01*. Valencia University of Technology, 195–227.

Ceri, S., Fraternali, P., and Bongio, A. 2000. Web Modeling Language (WebML): a modeling language for designing Web sites WWW9 Conference. In *First ICSE Workshop on Web Engineering, International Conference on Software Engineering*.

de Troyer, O. and Casteleyn, S. 2001. The Conference Review System with WSDM. In *Proceedings of First International Workshop on Web-Oriented Software Technology IWWOST'01*. Valencia University of Technology, 30–98.

Florescu, D., Levy, A., and Mendelzon, A. 1998. Database Techniques for the World Wide Web: a Survey. In *Sigmod Record*. Vol. 27. 59–74.

Garzotto, F. and Paolini, P. 1993. HDM A Model-Based Approach to Hypertext Application Design. *ACM Transactions on Information Systems (TOIS) 11,* 1 (01), 1–26.

Garzotto, F., Paolini, P., Bolchini, D., and Valenti, S. 1999. Modeling-by-Patterns of Web Applications. In *Proc. International Workshop on the World Wide Web and Conceptual Modeling WWWCM'99*.

Ginige, A. and Murugesan, S. 2001. Web Engineering: an Introduction. *IEEE Multimedia Special Issue on Web Engineering*, 14–18.

Gómez, J., Cachero, C., and Pastor, O. 2001. Conceptual Modelling of Device-Independent Web Applications. *IEEE Multimedia Special Issue on Web Engineering*, 26–39.

Isakowitz, T., Stohr, E. A., and Balasubramanian, V. 1995. RMM: A Methodology for Structured Hypermedia Design. *CACM: Communications of the ACM.*, 34–44.

Koch, N. 2001. *Software Engineering for Adaptive Hypermedia Systems: Reference Model, Modeling Techniques and Development Process*. UNI-DRUCK Verlag. PhD. Thesis, Ludwig-Maximilians-Universität München.

Koch, N., Kraus, A., and Hennicker, R. 2001. The Authoring Process of the UML-based Web Engineering Approach. In *First International Workshop on Web-oriented Software Technology*, D. Schwabe, Ed. To appear.

Murugesan, S., Deshpande, Y., Hansen, S., and Ginige, A. 1999. Web Engineering: A New Discipline for Development of Web-based Systems. In *First ICSE Workshop on Web Engineering, International Conference on Software Engineering*.

Schwabe, D., Esmeraldo, L., Rossi, G., and Lyardet, F. 2001. Engineering Web Applications for Reuse. *IEEE Multimedia. Special Issue on Web Engineering*, 20–31.

UML1.4 2001. OMG Unified Modelling Language Specification. http://www.omg.org/technology/documents/formal/uml.htm.

Vilain, P., Schwabe, D., and de Souza, C. S. 2000. A diagrammatic tool for representing user interaction in UML. *UML 2000 - The Unified Modeling Language. Advancing the Standard. Third International Conference, York, UK, October 2000, Proceedings 1939*, 133–147.

Yoo, J. and Bieber, M. 2000. Towards a Relationship Navigation Analysis. In *Prooceedings of the 33rd Hawaii International Conference on System Sciences*.

Received April 2002; Revised May 2002; accepted May 2002.