Journal of Web Engineering, Vol. 5, No. 3 (2006) 203–215

A DISCUSSION OF THE ROLE OF USER TRAILS IN WEB APPLICATIONS

ERICH GAMS and SIGI REICH
Salzburg Research - SunTREC
Jakob Haringer Straße 5/III, 5020 Salzburg, Austria
{*egams, sreich*}*@salzburgresearch.at*

Received April 21, 2005
Revised February 16, 2006

Designing, implementing and maintaining Web applications is a challenging task. Moreover, driven by some of the characteristics of Web applications, such as multiculturalism, continuous change, fast pace and competitiveness, there is an increasing need to use mechanisms that automatically adapt Web applications to new environments.
Trails, built from information about the users' browsing paths and activities, are an established approach to assist users in navigating vast information spaces and finding appropriate information. In this article we investigate how Web applications can profit from the integration of the concept of user trails — implemented as navigation pattern — in the Web modelling process. Furthermore, we investigate how trails can be applied to the various categories of Web applications. The results of our research show that trails are particularly suitable for those Web applications which exhibit a high degree of user interaction.

Keywords: User Trails, Navigation Assistance, Navigation Pattern, Web Modelling
Communicated by: S. Christodoulou

1 Web Applications and Web Engineering: Definitions

The World Wide Web has evolved into a global environment delivering all kinds of applications including reservation and booking systems, online shopping or auction sites, games, multimedia applications, calendars, maps, chat applications or data entry/display systems, and many more.

However, there exists no common definition of a "Web application". Baxley [2], for example, stresses the task orientation of Web applications, which he believes is one of the main differences to content-centric Web sites. Users of Web applications have specific goals, tasks and expectations in mind. The purpose of Web applications is to facilitate the completion of these tasks. In addition to these usage oriented definitions there are more technical oriented definitions (e.g. Kappel et al. [16]) which distinguish between Web applications and traditional software applications. In this article we will follow the latter as we focus on modelling activities of Web applications.

Following [16], we define Web applications as "software systems based on technologies and standards of the World Wide Web Consortium (W3C) that provide Web specific resources such as content and services through a user interface, the Web browser." This definition

explicitly includes technologies as well as user interaction; also it excludes pure software such as components implemented as Web services.

Although the complexity of designing, developing, maintaining such Web applications has increased significantly [21], the development process used has been ad hoc, lacking a systematic approach, quality control and assurance procedures.

Web Engineering, an emerging new discipline, promotes a process and a systematic approach to development of high quality Web-based systems [16]. It promotes the establishment and use of scientific, engineering and management principles, and disciplined and systematic approaches to development, deployment and maintenance of Web-based applications [9]. Web developers need a new and sound knowledge of not only development of traditional applications and the methods of software engineering but also concepts and techniques from other relevant areas such as human-machine communication or hypermedia engineering.

Following the definition above, the activity of modelling Web applications is of particular importance: modelling Web applications helps to manage the raising complexity of systems and therefore plays an important role in Web engineering. Especially at the hypertext level, complexity increments the risk of users getting lost in hyperspace or getting exposed to cognitively overload [8]. Navigation patterns [12, 26, 19] used as abstract design solutions provide guidelines for a designer to organise the hyperspace for easier navigation, each of them offering a very particular objective.

Nevertheless, it's up to the Web designer to decide how these facilities are deployed and how the different navigation possibilities are offered. Often they have to make a decision between overview (e.g. a site map or coarse categories) and deepness (i.e. a direct link to content).

In this article we introduce the concept of trails to counterpart these problems of modelling Web applications. Trails enable adaptivity of Web applications to users by considering their past behaviour.

We start with a brief overview of how navigation patterns support modelling of Web applications in Section 2. Then, we give a definition of trails and how they can be described as navigation pattern and thus integrated into the modelling process in Section 3. In Section 4 we enumerate the different categories of Web applications based on an existing classification scheme [16] and investigate the applicability of the trail approach to these types. Finally, Section 5 summarises the results of our investigations and conclusions gained thereof.

2 Existing Approaches for Modelling Web Applications

Although modelling Web applications has been proposed as a core activity in developing Web applications, it has not often been put into practice. Unfortunately, in most cases our *ad hoc* approach prevails and causes problems such as missing the requirements, inadequate specification or missing documentation of the application. The models, which are developed during the design process, elaborate static and dynamic aspects of a Web application's content, hypertext and presentation level [15]. Kappel et al. [17] propose a framework of requirements consisting of three dimensions to be considered when modelling Web applications: levels, aspects and phases (see Figure 1).

The latter dimension covers the different activities of a software life cycle, including the basic steps of analysis, design and implementation. Another dimension of Web application

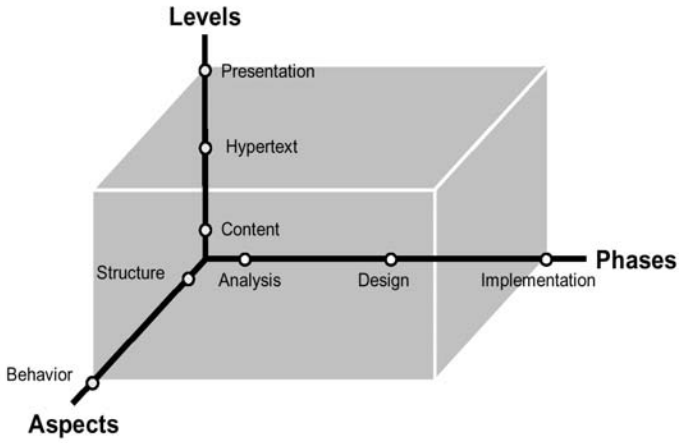

Fig. 1. Modelling Dimensions for Web Applications [17]

modelling comprises the three different levels: content, hypertext and presentation. The content level comprises domain-dependent data stored in a database system. The hypertext level denotes the logical composition of nodes and the navigational structure. The presentation level covers the layout of pages and the user interactions. Finally, the aspects' dimension, which is orthogonal to the three levels, covers structure and behaviour.

Modelling the hypertext level principally aims at modelling navigation or formally describing the path a user can go. Covering the structural aspect, modelling results in a hypertext structure model defining page compositions and navigational relationships (e.g. represented in a UML class diagram [15]).

The *hypertext structure model* describes which classes of the content model can be visited directly, e.g. by links embedded in a document. In contrast to the "static" navigation possibilities described by the hypertext structure model, the *hypertext behavioural model* tries to model the dynamic navigation behaviour, such as computing end points of links at run time or types of access through supplementary navigation structures such as index pages or guided tours.

Modelling and reusing such navigational access methods results in describing abstract *navigational patterns* [26, 25]. Many design methods that have evolved for the hypermedia domain [27, 18], support the integration of patterns.[a]

Different attempts have been made to define and classify patterns in hypertext. The OOHDM [27] framework, for example, classifies hypermedia patterns as navigation and interface patterns according to the design stages. The "Node in Context" pattern, as a navigation pattern example, suggests how to handle nodes which appear in multiple sets, allowing different representations depending on the context. The "Set-based Navigation" pattern organises information in sets of related items. For example, the "Information on Demand" interface pattern describes which nodes should be further described in the context of the same node [12].

[a] Patterns, originated from architectural design in the area of urban planning [1] and were then used in software design [10]; they describe recurring problems and suggest abstract solutions to solve them.

Nanard [22] proposed "Golden rules" and "Constructive Templates" as new types of reuse patterns. Bernstein [5] describes hypertext structures or patterns from the rhetoric point of view observed within hypertext rather than patterns used as a design facility. Some of them may be considered as rhetoric patterns as well as structure patterns (loop). Furthermore, Nanard et al. [22] proposed a classification based on four major dimensions: hypermedia design and development, hypermedia application, hypermedia system, and human factors; Each of these dimensions is further divided into subparts (e.g. hypermedia system is subdivided into architecture, interaction and production). For a catalogue of the different patterns proposed see [12, 26, 19].

In summary, navigation patterns build more complex navigation architectures based on nodes, links and indexes. During the development process, designers have to decide when and why they should use navigation patterns to best assist user navigation. Thus, navigation patterns that are used reflect the designers view how to describe the navigational space and how to design usable interfaces (and not necessarily the reader's view). The design poses many problems for the designer, such as how to prevent the user from getting lost in hyperspace [8].

We believe that the concept of trails defined in terms of a navigation pattern can enrich Web applications with a navigational element that dynamically adapts to the users' interaction needs.

In the next Section we introduce the trail concept and argue how trails could assist users in managing information spaces more efficiently. Furthermore, we define trails in terms of a navigation pattern.

3 The Concept of User Trails

The notion of trails is an established concept in the field of hypertext navigation [7]. People leave trails by handling information, e.g. by interacting with Web applications. Trails are defined as vectors of trail marks, each consisting of a node (representing a document, e.g. a Web page), the activity performed by the user and other properties (see [23] for a concise definition).

Trail-based systems recommend related items and provide assistance in navigation, and therefore share similarities with recommender systems and browsing advisors [24]. Recommender systems originate from people asking friends for their advice or knowledge on some specific question or area.

The basic intention of trail-based recommendations is to look at other peoples' paths to find similar trails, to find related documents and to find users with the similar interests. Interesting paths can be extracted from other peoples' paths, who have "walked" similar ways when searching for similar things or topics. Documents located at the end of a path or holding specific property values, may indicate high importance. Such properties can be a document's reading or editing time, which can be higher than that of other documents and therefore be an indicator for special interest of the user. The activity performed on a document can also be important for the document's relevance (e.g. "Printing" might indicate special importance [23]).

Having these considerations in mind, we argue trails to be understood in terms of a concept, as they enable recommendation of different types of categories, namely *paths*, *documents* and *persons*.

In terms of patterns, trails could be defined as navigation patterns. Similar to the trail concept, search patterns as suggested by [12] add search functionality to Web applications. All these patterns commonly represent the designer's view of Web applications. This means, once implemented, the structure and the navigation elements will hardly change (only if the site is redesigned). Adaptive hypermedia systems [6], for example, try to dynamically change interfaces and functionality according to the user's needs.

Summarising, trails represent the users' point of how to access and traverse a Web application within the natural navigation borders set by the document space; e.g. recommended paths or documents may change, depending on the trails of the major part of people traversing a Web site. These trails "adapt" dynamically whenever navigation action takes place. They give a hint of how to traverse a site or what could be interesting or where to shorten a path. Thus, trails are able to enhance the richness of navigation. The trail concept defined as a navigation pattern supplies a designer with a navigation structure that can be built into applications during design phase.

Next, we define a trail pattern following the scheme proposed by [1] for the description of abstract patterns. It includes the pattern name, description of the intent, the problem, the solution and references to related patterns.

3.1 A Trail Pattern as a starting point for the analysis

Following [1], we define a trail pattern consisting of the properties *intent*, *problem*, and *solution*.

3.1.1 Intent

Provide the user with information from other users. e.g. find similar trails of users and suggest several next steps to follow, find similar documents or similar users in order to identify the most interesting pages from the user's point of view.

3.1.2 The Problem

Web Designers try to design pre-defined paths (tours) through a site to the best of their knowledge, in order to help a user to comfortably navigate through the site. However, for instance business applications also bring in the vendor's intention to control the users' navigation to a certain extent and to lead them to specific documents or products. When entering a Web site a user should always be able to have an overview (e.g. via a sitemap) and a detailed preview of what is available and what is offered. Furthermore, authors will always focus on specific user groups and therefore will not be able to accommodate personal view points. Navigation design is based on the designers' point of view and will remain static in terms of structure and navigational possibilities.

3.1.3 Solution

Rather than author-based, pre-defined tours, automatically established trails of previous users are recorded, stored in a trailbase and provided to end users. A Web server access log contains a complete history of file accesses by clients. For an overview of how to process Web log data see [3]. Thus, recommendations can be produced individually for each session, respectively for each user and each situation in a Web application.

In order to determine the relevance of a trail, trail similarity measures [11] compute the

similarity between trails and — based on their computation — produce the recommendation output for the user.

For designers, trails can provide information on the distance between pages and thus support conclusions about reducing the distance and reorganise the navigational structure, either at runtime or in future by a kind of automatic redesign.

Although trails assume a human-computer interaction, we do not further discuss user interface patterns or how trail patterns can be implemented in the user interface. In any case there are several different ways of implementing trails (e.g. a sitemap overlaid by trails or simply a list of documents or even the more sophisticated hyperbolic tree).

There is also no special guideline at which state of a Web application's execution workflow designers should implement the trail pattern. They can decide individually, also depending on the Web application type. Nevertheless, the earlier a trail recommendation is provided to the user, the more users can benefit from the pattern.

In the following Section we investigate how the trail pattern can be instantiated in different Web application categories, according to a classification suggested by [16] and [20].

4 Categories of Web Applications and Applicability of Trails

In [16] Web applications are classified by complexity and development history. In Section 3 we identified three recommendation types trails are able to support, namely *similar paths*, *similar documents* and *people with similar interests*. In the following we briefly describe the different Web applications' categories and types and investigate how the concept of trails — according to the recommendation categories — can be applied to each of these types. The listing follows the structure of [16].

4.1 Document Centric Web Applications

The class of document centric Web applications comprises Web pages that are stored ready-made on a Web server, without adapting to the user or holding any dynamic component. Examples are Web presentations of companies, communities, or a Web radio, from where users can connect to a radio server.[b]

Path: The recommendations based on trails can help users to orient themselves by providing a useful view and filter of the whole document space, based on past experiences of other users.

Trail data can be extracted from e.g. Web logs by applying different session tracking methods discussed in [3, 28].

The users visiting the site are anonymous and thus their interests are only represented through their trails. In order to find similar trails a trail similarity measure (e.g. the sequence alignment method [14] adapted to trails [11]) is used to compare recently accessed nodes to previous trails of other users and make them navigable.

The larger the number of available Web pages and thus different navigation possibilities (also supported by navigation patterns such as sitemap or index) the more trails can effectively help users to find their way.

[b] Strictly speaking, these are not Web applications in the sense of the definition in [16] as they do not contain any purpose-built software components.

Document: Similar trails may indicate a relationship between the documents comprised, respectively the documents' content located in the trails, even if there is no explicit link between them.

Depending on the maximum depth — the distance of a page to the root — recommendation of Web pages (located in similar trails) can be based on their access frequency or users' average visiting time.

Person: Information about a visitor's Internet address, nationality or even the search terms entered in a search engine would be nice features to have but do not directly assist other users' navigation.

However, persuading users to get registered and give away information about themselves would essentially enhance the possibilities of recommendations e.g. by automatically building groups of similar interest.

Trails can be well applied to document centric Web applications, because the locations of documents and thus also the references to the nodes of a previous trail are hardly changing (Unreachable nodes and thus trails will users soon get frustrated and uninterested). Generally, the applicability of trails to this category depends on the overall size and range, the number of pages and links, of the Web site.

4.2 Interactive Web Applications

Interactive Web applications are a first form of allowing users to interact with a Web application. Radio buttons or selection menus support user navigation and help users to get an overview. News sites (e.g. *www.cnn.com*), virtual exhibitions and others are representatives of this category.

Path: Guided tours planned by an author are popular techniques of how to access an exhibition. However, trails can support visitors by offering different paths from the users' point of view. In other words, trails represent a kind of guided tour composed by users, even grouped according to their interests.

Document: References to news articles accessed most often can help the user to find interesting articles more quickly and easily (e.g. *Most Popular* Button at *www.cnn.com*).

Person: Equipped with a login function, such applications can provide accurate information about its users and their ways of navigation. In order to guarantee privacy of the individual user, persons could be grouped together based on their profiles or interests. Groups and their interests could then be made available to new users by the Web application and the users can decide which group best meets their requirements.

In what way the trail pattern can be instantiated in interactive Web applications depends very much on the type of application. Virtual exhibitions are most suitable for path recommendations, whereas news sites could be enhanced with documents, or news messages that users have been interested in.

4.3 Transactional Web Applications

Transactional Web applications put the focus on business activities. Online banking systems, online shopping (e.g. *www.amazon.com*) or booking systems (e.g. *www.expedia.com*) permit the user to not only read but also modify data, e.g. by querying databases, retrieving and storing information. Login access is a prerequisite.

Path: So far most online booking and shopping applications support users by offering a "search by category" or a "search in taxonomy" functionality to find the product, respectively the description of the product they are looking for. These search and navigation functionalities are based on a previously specified once and for all categorisation scheme.

Nevertheless, items that belong to different categories may also be related depending on the context. In terms of trails the context is represented by the previous steps of a path. Thus, trails cannot assist users to orient themselves by providing "next steps to go" but are used to detect relationships between items, that are not explicitly defined.

Document: In addition to the recommendation of related items, the most popular hotels or trips e.g. based on access frequency, can be provided.

Person: Persons with similar interests book similar trips and hotels. Similar to interactive Web applications, groups can be formed and their interests be recommended. Additionally, trails can help to find coherence between users and categories even if the users' interests are not stored in a profile.

Transactional Web applications offer a wide range of possibilities for the applicability of trails to assist users in the choice of navigational possibilities or selection of documents. In particular shopping applications are most suited for recommendations in general. Gathering consumers' data enhances the variety and accuracy of producing recommendations (Amazon as one of the most popular shopping applications already leverages some different recommendation methods based on stored user data). Due to security and privacy issues online banking systems have limited applicability of recommendation functionality.

4.4 Workflow-Based Web Applications

Examples of workflow-based Web applications are B2B solutions in the e-commerce or e-government domains, that provide services distributed across different locations. The purpose of e-government applications is to offer a predefined workflow of forms, which can only be accessed in a determined order. Therefore, the applicability of trails is rather limited.

B2B systems are mainly based on Web services, to be machine processable, and therefore do not fulfill the requirements of human-computer interaction via a user interface.

4.5 Collaborative Web Applications

Collaborative Web applications are needed whenever communication between cooperating users is high. Generally speaking, these applications support generating, editing and managing information (e.g. commonly edited documents, message exchange) in a shared workspace. There are various fields of applications ranging from systems supporting collaborative work or e-learning platforms to simple chat or scheduling systems.

Path: When participating in different discussion groups (e.g. threads in newsgroups) or editing documents in a shared space users leave a full range of different trails. Collected trails of other peoples' trails and activities in collaborative working environments can provide a kind of workflow of not only where to go next but also what to do next. Similar paths of other users or the trail most people took might be of interest to learning novices in e-learning systems to find other or more information on a topic.

Document: Collaborative Web applications offer much more different activities than e.g. "opening" a document which can be carried out. Considering different activities can help to indicate the importance of a document [23] (e.g. editing, downloading a document indicates more importance than just accessing it). These investigations may be used to rank documents.[c]

Person: In this category trails are well suited to find persons with similar interests. Trails of users belonging to different groups, editing documents might improve intergroup communication. Users discussing different threads on the same topic might share the same interests.

It is incontestable that there are many possibilities to apply the trail concept to various collaborative Web applications. In any case, the usefulness of trails recommended to individual users still depends on the number of previously existing trails e.g. the number of users participating in the same discussions or editing the same documents. Otherwise trail support will be limited.

4.6 Portal-Oriented Web applications

The intent of portal-oriented Web applications is to combine various resources of information and services in a single point of access. Many different types of portals exist for many different purposes. Portals for the general public, such as those provided by Netscape, Microsoft, newspapers or search engines, or portals targeting specific communities, shopping, auction or marketplaces.

Path: Every interaction with a portal Web application can be used profitably for the provider as well as the user.

Trails can provide interesting paths to where people go and from where they have come. Because the start page acts as an access point, it is the most interesting part for a new user to get recommendations.

Document: Recommendation possibilities of documents are similar to document centric or interactive Web applications (only based on a bigger amount and variety of documents).

Person: Some portals also provide enhanced services for registered users and thus are able to gather more valuable information about its users. Otherwise, recommendations are limited to additional information already enumerated in Section *Document Centric Web Applications*.

[c] Our considerations always assume that documents are accessable to all users.

4.7 Ubiquitous Web Applications

Ubiquitous Web applications provide access to services anytime, from anywhere — enabling multi-platform delivery, customisation or location-dependency. PDAs, handhelds and smartphones have promoted this category of Web applications by being able to provide e.g. up-to-date information on specific location, buildings featuring services for tourists.

Path: Trails applied to ubiquitous Web applications would connect physical and electronic worlds in heterogenous trails [13] e.g. georeferenced trails. Trails are recorded from people moving through a physical world visiting locations and simultaneously "visiting" electronic documents.
Questions such as — where have people physically been and which electronic information they have accessed or vice versa — could be answered by the system. Thus, trails guide a user through physical objects as well as electronic documents.

Document: Nodes of a trail could not only comprise documents but also buildings or places. Users get recommendations on when visiting a particular location, most people have read a certain document (e.g. information about a famous building).

Person: Several users with the same interest could constitute groups. To give an example: Tourists of the same nationality could be unified into a group and thus could evoke the interest of users to visit specific places.

Ubiquitous Web applications are the most interesting ones to enhance the notion of trails from where they originate — the physical world. Documents can be represented by physical objects (linked to the virtual world by e.g. via radio frequency tags or 2-D barcodes) e.g. buildings, places, as well as electronic documents. Trails are represented by paths people follow in real life to get to a destination.

4.8 Semantic Web Applications

Estimating the applicability of trails to the emerging trend of the Semantic Web is quite challenging, since this domain is still evolving. One objective of the Semantic Web approach is to provide information on the Web not only in human-readable form but also for machines supporting e.g. automatic information gathering services or recommendation engines [4]. So far we have assumed that trails are only constituted by humans and not by machines. If machines do provide specific information about their goal and their functionality, they could be set equal to human actors with special interests; thus agents would be able to find agents doing similar work, i.e. they may exchange the best way to go or search result, etc.

Nevertheless, a machine's strategy or procedure how to gather information is always calculated by some kind of computer program and is therefore controlled (as opposed to human navigation).

5 Summary and Conclusion

In this article we discussed the importance of Web modelling activities for Web engineering. Navigation patterns take an important part in modelling the hypertext layer of a Web application and represent a designer's view on a navigation space. We introduced the concept of trails — enabling recommendations of paths, documents and persons — and defined a trail

navigation pattern that is able to express the users' view on how to navigate a document space. The trail pattern can be inserted in a hypertext behavioural model in the Web application modelling process. The main part of our work focuses on a survey of different Web application classes and an analysis on the applicability of the trail pattern.

The following Table provides an overview of the applicability of trail recommendation types to the different Web application types; where two crosses mark a full, unlimited applicability and enrichment of Web applications, one cross applicability but with limitation, i.e. only for certain subtypes, with requirements, and no cross marks no applicability or usefulness.

Table 1. Overview of applicability of trail recommendation types to different Web application types

Trail Recommendation Type	**Path**	**Document**	**Person**
Document Centric Web Applications	XX	XX	-
Interactive Web Applications	X	X	XX
Transactional Web Applications	X	X	X
Workflow-Based Web Applications	-	-	-
Collaborative Web Applications	XX	XX	XX
Portal-Oriented Web applications	X	X	X
Ubiquitous Web applications	XX	XX	XX
Semantic Web applications	-	-	-

Our investigation has shown that Web applications can benefit from the fact that the trail pattern allows to recommend *users* as well as documents.

- From the development point of view designers are supplied with a navigation structure that enhances not only interaction between Web application users but also is able to gain the interest of individuals as well as communities to their application and content (if other people are interested in that application, it may also be of interest to me.). This is most important for applications whose purpose is to attract users (e.g. Web presentations of companies, communities or virtual exhibitions) or, advertising for some product (e.g. shopping applications). The latter category is already intensively using similar concepts (e.g. Amazon).

- The purpose of a Web application strongly influences the degree of usage of the trail concept. There are Web applications where recommendations are generally not desirable (e.g. banking systems) but also applications where communication between users is highly desirable and the trail concept is well suited to support collaborative working or decision making (e.g. newgroups).

- From the implementation point of view recommendations based on trails are restricted due to the statelessness of the HTTP protocol. It does not enable the identification of user sessions. However, different ways of how to identify a trail and a user have been used [3, 28] but do not fully exploit the whole range of trail recommendation possibilities (e.g. How to identify unique persons that use a Web application from different Internet addresses?).

- From a usage point of view, the trail concept enables communication between user's of a Web application and provides the users point of view of navigating a Web application.

In conclusion, the concept of trails is generally well suited for those Web applications with a high number of interacting users, and mostly static, i.e. non personalised content. In future, ubiquitous Web applications provide the most interesting field for trail based navigation and recommendation support due to the integration of physical and virtual environments and their high location independence.

References

1. C. Alexander, S. Ishikawa, M. Silverstein, M. Jacobson, I. Fiksdahl-King, and S. Angel (1977). *A Pattern Language: Towns, Buildings, Construction.* Oxford University Press.
2. Bob Baxley (2004). What is a web application? [last visit: 2006-01-15] http://www.boxesandarrows.com/view/what_is_a_web_application_.
3. B. Berendt, B. Mobasher, M. Spiliopoulou, and J. Wiltshire (2001). Measuring the accuracy of sessionizers for web usage analysis. In *Proceedings of the Web Mining Workshop at the First SIAM International Conference on Data Mining, SDM*, Chicago, IL, USA, April 4-7, 2001, pp. 7-14.
4. T. Berners-Lee, J. Hendler, and O. Lassila (2001). The Semantic Web, May 2001. Scientific American.
5. Mark Bernstein (1998). Patterns of hypertext. In *Proceedings of the ninth ACM conference on Hypertext and hypermedia: links, objects, time and space—structure in hypermedia systems*, Pittsburgh, Pennsylvania, United States, pages 21–29. ACM Press.
6. Peter Brusilovsky (1996). Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction*, 6(2-3):87–129.
7. Vannevar Bush (1945). As we may think. *The Atlantic Monthly*, 176(1):101–108.
8. Jeff Conklin (1987). Hypertext: An Introduction and Survey. *IEEE Computer*, 20(9):17–41, IEEE Computer Society Press, Los Alamitos, CA, USA.
9. Yogesh Deshpande and San Murugesan (2001). Summary of the second icse workshop on web engineering. *SIGSOFT Softw. Eng. Notes*, 26(1):76–77, ACM Press.
10. E. Gamma, R. Helm, R. Johnson, and J. Vlissides (1995). *Design Patterns: Elements of Reusable Object-Oriented Software.* Addison-Wesley.
11. Gams, E. and Reich, S. (2004). Following your colleagues' footprints: navigation support with trails in shared directories. In *Proceedings of the fifteenth ACM conference on Hypertext & hypermedia*, pages 89–90, Santa Cruz, CA, USA.
12. D. German and D. Cowan (2000). Towards a unified catalog of hypermedia design patterns. In *33rd Hawaii International Conference on System Sciences (HICSS)*, IEEE Computer Society, Washington, DC, USA.
13. Kaj Grønbæk, Peter Posselt Vestergaard, and Peter Ørbæk (2002). Towards geo-spatial hypermedia: Concepts and prototype implementation. In *Proceedings of the thirteenth ACM conference on Hypertext and hypermedia*, pages 117–126, College Park, Maryland, USA.
14. Dan Gusfield (1997). *Algorithms on strings, trees, and sequences: computer science and computational biology.* Cambridge University Press.
15. Rolf Hennicker and Nora Koch (2000). A UML-based methodology for hypermedia design. In Andy Evans, Stuart Kent, and Bran Selic, editors, *UML 2000 - The Unified Modeling Language. Advancing the Standard. Third International Conference, York, UK, October 2000, Proceedings*, volume 1939 of *LNCS*, pages 410–424. Springer.
16. Gerti Kappel, Birgit Pröll, Siegfried Reich, and Werner Retschitzegger, editors (2003). *Web Engineering — Systematische Entwicklung von Web-Anwendungen.* dpunkt Verlag, Heidelberg. In German.
17. Gerti Kappel, Werner Retschitzegger, and Wieland Schwinger (2000). Modeling customizable web applications - a requirement's perspective. In *Proceedings of the Kyoto International Conference on Digital Libraries: Research and Practice, November 13 - 16, 2000, Kyoto University, British Library*, pages 168–179.

18. Nora Koch and Andreas Kraus (2002). The expressive power of uml-based web engineering. In *In D. Schwabe, O. Pastor, G. Rossi, and L. Olsina, editors, Second International Workshop on Web-oriented Software Technology (IWWOST02)*, Malaga, Spain.
19. F. Lyardet, G. Rossi, and D. Schwabe (1999). Patterns for adding search capabilities to web information systems. In *Proceedings of the 4th European Conference on Pattern Languages of Programms*, 322–326, UVK Universitätsverlag Konstanz.
20. San Murugesan (2000). Web engineering for successful software development. In *Tutorial, Asia Pacific Web Conference (APWeb2000)*, October 25-28, Xi'an, China.
21. San Murugesan and Yogesh Deshpande (2002). Meeting the challenges of web application development: the web engineering approach. In *Proceedings of the 24th international conference on Software engineering*, pages 687–688, Orlando, Florida, USA.
22. Marc Nanard, Jocelyne Nanard, and Paul Kahn (1998). Pushing reuse in hypermedia design: golden rules, design patterns and constructive templates. In *Proceedings of the ninth ACM conference on Hypertext and hypermedia : links, objects, time and space—structure in hypermedia systems*, pages 11–20, Pittsburgh, Pennsylvania, USA.
23. Siegfried Reich and Erich Gams (2001). Trailist - focusing on document activity for assisting navigation. In *Procs. of the Twelfth ACM Conference of Hypertext and Hypermedia*, pages 29–30, Denmark.
24. Paul Resnick and Hal R. Varian (1997). Recommender systems. *Communications of the ACM*, 40(3):56–58, March 1997.
25. G. Rossi, D. Schwabe, and F. Lyardet (1999). *Patterns for Designing Navigable Information Spaces*. Addison-Wesley, Reading, Massachusetts.
26. Gustavo Rossi, Daniel Schwabe, and Fernando Lyardet (2000). User interface patterns for hypermedia applications. In *Proceedings of the working conference on Advanced visual interfaces*, pages 136–142, Palermo, Italy.
27. Daniel Schwabe, Gustavo Rossi, and Simone D. J. Barbosa (1996). Systematic hypermedia application design with oohdm. In *Proceedings of the the seventh ACM conference on Hypertext*, pages 116–128. ACM Press, 1996.
28. Jaideep Srivastava, Robert Cooley, Mukund Deshpande, and Pang-Ning Tan (2000). Web usage mining: discovery and applications of usage patterns from web data. *SIGKDD Explor. Newsl.*, 1(2):12–23.