COMPUTER NETWORKS

Computer Networks 31 (1999) 1667–1678

Improving Web information systems with navigational patterns

Gustavo Rossi [b,1,2], Daniel Schwabe [a,*], Fernando Lyardet [b,1]

[a] *Depto de Informática, PUC-Rio, Rua Marquês de São Vicente, 225 Cep: 22453-900 Gávea, Rio de Janeiro, RJ, Brazil*
[b] *LIFIA, Depto de Informática, Fac. de Cs. Exactas, UNLP, Calle 50 y 115, 1er Piso, (1900) La Plata, Buenos Aires, Argentina*

Abstract

In this paper we show how to improve the architecture of Web information systems (WIS) using design patterns, in particular navigational patterns. We first present a framework to reason about the process of designing and implementing these applications. Then we introduce navigational patterns and show some prototypical patterns. We next show how these patterns have been used in some successful WIS. Finally, we explain how patterns are integrated into the development process of WIS. © 1999 Published by Elsevier Science B.V. All rights reserved.

Keywords: Web information systems; Hypermedia design; Design patterns; Navigation design

1. Introduction

Web information systems (WIS), i.e. information systems that are constructed using Web technology, involve a myriad of problems. Those problems are related with different design concerns such as:

- Defining the overall architecture of the application, e.g. the kind of interface (communication, formatting, etc....) with legacy applications (systems or databases).
- How can navigation give appropriate support to user tasks, e.g., how to organize the information items in such a way that the user can navigate them easily.
- How to present both navigation and application functionality in an organized, coherent way.

Many of these issues are often managed in an environment-centric way (i.e. taking into account the details of the implementation platform). However, we claim that since WIS are, after all, software systems, we must use solid software engineering techniques for dealing with the design complexity. There are many design methods that can be used to model different aspects of WIS such as OOHDM [14,15], RMM [7], W3DT [16], etc. In those methods there is a clear separation of concerns among architectural (conceptual) design, navigation design, user interface aspects and implementation. Separating navigational design from the other concerns, especially from the user interface aspects, allows us to concentrate on what we think is the distinguishing feature of WIS — its hypermedia metaphor, which is exercised through navigation.

Unfortunately, many products targeted to develop WIS tend to ignore hypermedia information design, and its attending navigation operations, by treating it as another user interface operation. For example, in Visual Wave [17], designers implement Smalltalk applications, and their interfaces contain controls that allow the opening of another window; if this is

[*] Corresponding author. E-mail: schwabe@inf.puc-rio.br
[1] Also at UNLM and COINCET.
[2] E-mail: {gustavo,fer}@sol.info.unlp.edu.ar

1389-1286/99/$ – see front matter © 1999 Published by Elsevier Science B.V. All rights reserved.
PII: S1389-1286(99)00015-8

a result of a navigation operation, it is treated as if they were just designing a conventional transactional application. In this case, the power of navigation is lost, not because of the features of the environment but mainly because designers do not take into account the navigational dimension of their applications. Typically, they just develop the application model with objects and then build the user interface. As we will show in this paper there are many design problems that can only be solved when thinking about navigation as a separate design issue.

The hypermedia aspects of WISs bring on a slew of new design problems. It has been repeatedly shown in the hypermedia literature (see for example [9,10]) that building a solid navigation architecture is a complex enterprise; we must insure that information is easy to find and that the user will not experience cognitive overhead while exploring the information space.

The purpose of this paper is twofold; first we will present some patterns that appear recurrently in WIS design and show that systematically applying them we can complement our abstract vision of the system (as expressed, for example, with the primitives of a design method). Second, we want to motivate the World Wide Web community to record its collective design experience in the form of patterns, as has been done by other software communities (see for example http://hillside.net/patterns/).

The structure of the paper is as follows: we first introduce navigational patterns and show some examples; then we analyze a well-known WIS (www.amazon.com) and analyze design issues related with the patterns we discuss in Section 2; finally we discuss different ways of integrating patterns into the design process.

2. Navigational patterns: rationale and examples

Though the idea of patterns itself originated in urban architecture [1], design patterns are being increasingly used in software systems [2]. Patterns record design experience by explaining and evaluating recurrent problems and proven solutions. They describe those problems in an abstract way and the core of the solution in such a way that it can used with different cases of the same problem.

Design patterns complement design methods as they show solutions that go beyond the use of primitives of a method. For example, a naive object-oriented designer will tend to follow closely the main concepts of the object paradigm, encapsulating structure and algorithms in the same object. However, complex problems require more ‘advanced’ solutions, like the ones appearing in patterns such as Bridge, Strategy or State [2]. In those cases either the representation, the algorithms or the state of the object are further ‘objectified’ and defined outside the object in the context of a separate hierarchy. We can define patterns at different abstraction levels and they act as micro-architectures in a system; they usually help to improve system modularity and ease of extension.

When we document a pattern, we describe which kinds of problems originate this micro-architecture, when it should be used and what is the trade-off when using it. As explained in [2], patterns are neither original nor new solutions to software problems; instead they are well proven and used strategies employed in successful developments. As such, patterns are not invented but discovered or ‘mined’. Consequently, to expert designers these patterns may appear to be obvious, since they will reflect design choices they usually employ. But, for less experienced designers, these patterns will constitute a valuable source of design expertise, to be tapped when creating a new design.

We have been mining and using patterns in the field of hypermedia for the past three years and have studied how to use them to record the experience of hypermedia designers [13]. Patterns in our catalogue are organized in a taxonomy that includes architectural patterns, navigational patterns and user interface patterns. In this paper we will focus on a sub-set of the navigational pattern catalogue that applies to WISs.

Navigational patterns are in some sense similar to those in [1]. They show how to build usable navigational architectures by pushing the hypermedia paradigm one step further. However, they are more like the original urban architecture Alexandrian patterns, as they describe the organization of a navigable space, the roads you can follow to reach different homes, the kind of orientation signs you will find, the short-cuts, etc. In contrast, traditional object-oriented patterns focus instead on obtaining high quality object interactions in order to make the system easy to evolve and maintain. Architectural hypermedia patterns meanwhile resemble those in [2].

Since WIS are a particular kind of hypermedia application, many of our navigational patterns apply directly to Web applications. However, as the Web is the host of many kinds of applications, we envision the discovery of many specific patterns in the Web; some of them are presented in this paper.

In the next sub-section we briefly describe four navigational patterns using a style that combines the [2] one with the original Alexandrian notation. For each pattern, we describe the motivating problem; we show the naive solution and then how the pattern goes beyond that solution. We do not emphasize implementation issues since they are strongly dependent on rapidly changing technology. It should be clear, however, that each pattern can be easily implemented in the World Wide Web. In Section 3 we show how these patterns are currently used in a successful electronic online shop. In Fig. 1 we present an outline of some navigational patterns. They are part of a catalogue of around twenty architectural, navigational and user interface patterns (see [3,11,12]). In Section 5 we briefly comment which actions are being performed in the hypermedia community to make this catalogue grow.

2.1. Set-based navigation

2.1.1. The problem

WIS usually involve dealing with collections of objects (e.g., paintings, cities, books, CDs etc.). These collections may be explored in different ways, according to the task the user is performing. In an electronic shop we may want to explore all Rolling Stones' CDs. We could also want to navigate to all CDs in which Mick Jagger participated (including Rolling Stones ones). Notice that the same situation appears in different domains, such as institutional or entertainment sites. We should help the user by identifying meaningful sets for the targeted tasks, and ease the traversal of these sets easily by providing fast access to different members of the set.

2.1.2. Solution

The usual naive strategy followed by designers consists in providing an index to set members (see for example the results of a Netscape or Altavista Search); users must then go back to the index to navigate to the next member of the set. As a consequence, the usual simplicity of the hypertext navigation metaphor is lost. Even if we closely follow a systematic design method (like OOHDM or RMM), we may obtain the same solution. The reason is that there are no additional semantic relationships among

Pattern	**Intent**
Set-Based Navigation	Organize the information in Sets of related information items. Provide intra-set navigation capabilities
News	Allow easy access to new information items as the WIS grows
Landmark	Provide direct access to critical sub-systems in the WIS
Shopping Basket	Keep track of user selections during navigation, making these selections persistent to process them when the user decides to. Decouple product selection from product consumption and/or processing
Active Reference (See [13])	Define indexes as active orientation tools in sub-areas of the whole hyperspace. Make these indexes co-exist with target objects.
Node in Context (See [13])	Customize representation of objects according to the sets whithin which they are being accessed

Fig. 1. Navigational patterns.

different members of a set (two Rolling Stones' CDs, for example), besides being members of the set itself, so one may not realize that they should be linked explicitly (at least in the node-and-link hypertext model of the WWW).

Beginner designers tend to worry about how the reader reaches a given node, leading to an index. But, these designers fail to consider what the reader should do next (once the node has been reached) — typically, the only option left is to back up to the index. This leads to a purely hierarchical organization of the information, failing to exploit the richness of 'full' hypertext.

The 'expert' solution is to consider set-based navigation as a 'first-class' navigation strategy. We group nodes in meaningful sets and provide inter and intra-set navigation facilities, such as indexes and links for letting the user navigate to the 'next' and 'previous' elements of the current one in his traversal. Notice that a node may belong to more than one set, so it is often useful to allow the user to 'switch sets'. We call such sets ' Navigational contexts' in OOHDM; they are also referred to as 'Collections' in HDM2 [4].

Navigation inside contexts complements conventional semantic links, such as, for example, those connecting a node about a CD with the node about artist himself or CDs to comments about them. In other words, the reader can browse through the set or leave it to explore other nodes (or eventually other sets).

2.1.3. Known uses

'Sets' appear in almost all Web-sites; for example, every index (menu) automatically defines a set, the nodes it points to. This is also true of every Web-site where we can perform searches (the results form a set). In spite of this, set-based navigation itself is not as widely provided. As an exception, many sites provide guided tours through collections; see for example http://www.nga.gov. In Section 3 we show how to improve navigation through sets in a well-known Web-site.

2.1.4. Related patterns

'Nodes in Context' is a pattern that complements 'Set-Based Navigation' showing how to solve the problem that appears when a node appears in more than one set (context); see [3,13]. 'Shopping Basket' is a special kind of set that is built dynamically, as the result of users' selections during navigation [8].

2.2. News

2.2.1. The problem

Most WIS are in constant growth; one of the key differences with conventional hypermedia applications is that the navigable space is not fixed over time (as in CD-ROMs), but tends to evolve as new content (e.g., products and services) is added over time. Suppose that we are building an electronic shop with books, CDs and videos. How can we be sure that users will know about new books, CDs or even new services we are offering as the WIS evolves?

Clearly, vital selling opportunities like these cannot be left to the chance that the user will discover the information. On the other hand, trying to solve this problem poses a design challenge for Web designers, who must balance between a well-structured Web-site where information is organized in items with sub-items, etc., and a star-shaped navigational structure where all information is reachable from the home page. The latter approach is clearly not desirable because the site's usability is greatly reduced and it may become unmanageable as it grows. Therefore, how is the user provided with instant feedback of any recent changes or additions to the information available, while maintaining a well-structured Web-site?

2.2.2. Solution

If we follow the usual guidelines of good hypermedia (and Web) applications design, we will tend to maintain consistency of the structure of the entire navigation space. In this way, a new information item will be added to the correct place of the hypermedia graph and users will reach it through the appropriate paths or through search tools. This solution maintains the site well structured but it might be lacking from a 'business marketing' perspective.

The solution of the 'News' pattern consists in structuring the home page in such a way that a space is devoted to the newest additions, including a summary and a link to the information object. This approach allows the designer to preserve a good organization of the information, while giving users feedback of the changes that take place within the WIS.

‘News’ implement shortcuts to information that may be located in the leaves of a tree-structured site, without compromising the underlying structure. Notice that the navigational structure of the application is slightly affected by the addition of (temporary) links from one node to others. Users can get the new information also by browsing the site in the usual way. In fact, as new headlines appear, old ones are be removed. Notice that when using ‘News’, design documents showing instance relationships will tend to change over time. In Section 4, we discuss how to cope with this problem when documenting the navigational structure of a WIS.

2.2.3. Known uses

‘News’ is used in hundreds of Web-sites and applications such as http://www.nga.gov where it is used to announce new collections and the current tours available. In http://www.inprise.com and http://www.sun.com the ‘News’ navigational pattern is used to present corporate announcements. Not surprisingly, ‘News’ is also the basis for all online publications, as the home page always includes links to the latest stories, as can be seen in http://www.news.com or http://www.cnn.com.

2.2.4. Related patterns

All news can be viewed as a set and could be traversed using ‘Set-Based Navigation’. As the same object can be accessed in different contexts (see Section 3.2), the Nodes in Context pattern should be also applied.

2.3. Landmark

2.3.1. The problem

Suppose we are building a Web information system for an electronic shopping such as http://www.amazon.com. When we describe the navigational schema (i.e. the network of nodes and links types), we try to closely follow existing relationships in the underlying domain model — for example we can navigate from an author to his books, from a CD to the list of songs it includes. We can also go from a book to some comments readers made about it, or read about related books, etc. Building a domain model and making navigation reflect that domain model is a key approach for achieving a solid navigation architecture. However, there may be singular ‘entry-points’ that should be easy to reach such as the ‘checkout stand’.

We should build the navigational schema linking every navigational class (such as book, comment, news, songs, etc) to the ‘checkout stand’; however we will end with a spaghetti-like and difficult to understand schema, since those links are clearly outside the domain model. Again, we have to struggle between structure, usability and maintaining the design documents concise.

2.3.2. Solution

Define a set of landmarks and make them easy to access from every node in the network. Make the interface of links to a landmark look uniform. In this way users will have a consistent visual cue about the landmark. We may have different levels of landmarks according to the WIS area we are visiting. As we explained in the ‘News’ navigation pattern, ‘Landmarks’ pose a challenge to the designer; namely how to express ‘Landmarks’ without making the navigational schema look complex or saturated with links.

2.3.3. Known uses

Landmarks can be found in different Web applications such as http://www.inprise.com and http://www.3com.com, where they provide access to the company’s products and downloads; or http://www.mercedes.com where they are also used to access Help and Dialogue (contact) pages. Finally in CNET’s News.com, landmarks are used to access several subsystems: **Special Reports** | **News** | **Downloads** | **Hardware** | **Shopping** | **Gadgets** | **Games** | **Web Building** | **TV**.

2.4. Shopping basket

2.4.1. The problem

Electronic-commerce is now a reality in the Web. However, users often want to navigate through the e-market to decide what they will buy and in when. In http://www.amazon.com for example, a user can browse through hundreds of books or CDs and choose a sub-set of them to be bought. The naive solution would be either to ask the user to buy the product in the moment it is found or force bookmarking all the desired products and buy them in different navigation sessions.

It is clear that these approaches are not suited to cases in which we want to buy dozens of different products, as it is impractical for the user (and the shop) to require one transaction for each product. This alternative has another drawback, as we need to navigate to the ‘check-out’ page many times, wasting time.

On the other hand, we should keep in mind that these e-commerce restrictions should not interfere with the overall Web-site navigational structure.

2.4.2. Solution

Provide the users with a metaphor similar to bookmarks, by allowing the users select the products to buy as they are traversed. Provide a ‘persistent’ store for those items (a ‘shopping basket’) that can be accessed as another navigation object and associate processing operations to the basket, such as eliminating an item, changing quantities, computing totals, making an order, etc.

This solution is easy to implement by adding an interface object (usually a button) to every available product in a shopping site. When the user selects this product, it is added to his shopping list. Later the user can navigate to the shopping basket where he will find either a detailed description of each product or a summary with an anchor to the product page.

In many Web-sites the shopping basket facility can be enriched with validation (consistency checking) operations. For example, if a customer is planning his vacation with an Internet travel agent service, the travel agent may be capable of doing some checking on the arrival and departure times on hotel reservations and viability of flight connections. In fact, such operations may also be invoked each time an object is added to the basket.

The result is a very natural approach since it resembles the way people buy at the supermarket, adding products to their shopping cart as they walk around.

2.4.3. Known uses

There are many examples of this pattern available on the Web. One of them is **Business travel**[3] where the user adds different destinations to a business tour, together with car and hotel reservations. The system checks, among other things, the dates of departure and arrival from the different destinations (you can’t make a hotel reservation if you are leaving before arriving).

A completely different example is **PublishersDepot**[4], an image bank where the users can search for different kinds of images (based on textures, backgrounds, etc.) and put selected items in a persistent (between sessions) list. Users are able to create different such lists of selected pictures, and the Web-site keeps those lists of selections made by a user. Every time the user logs in, he may retrieve any of the lists, and additions or deletions can be made to them. These lists may also be used to purchase images, so they can function as conventional shopping baskets as well.

3. Obtaining usable navigation architectures with patterns

While validating our patterns, we have evaluated dozens of WIS to discover recurrent design problems and their solutions. We have found that sensible navigational structures are prevalent in well-designed, usable sites. The purpose of this section is to show, through an example, how patterns described in this paper appear in a popular electronic bookstore: Amazon.com. We can also give insights on possible improvements to the navigational structure of this site, by applying one of these patterns as it is defined here.

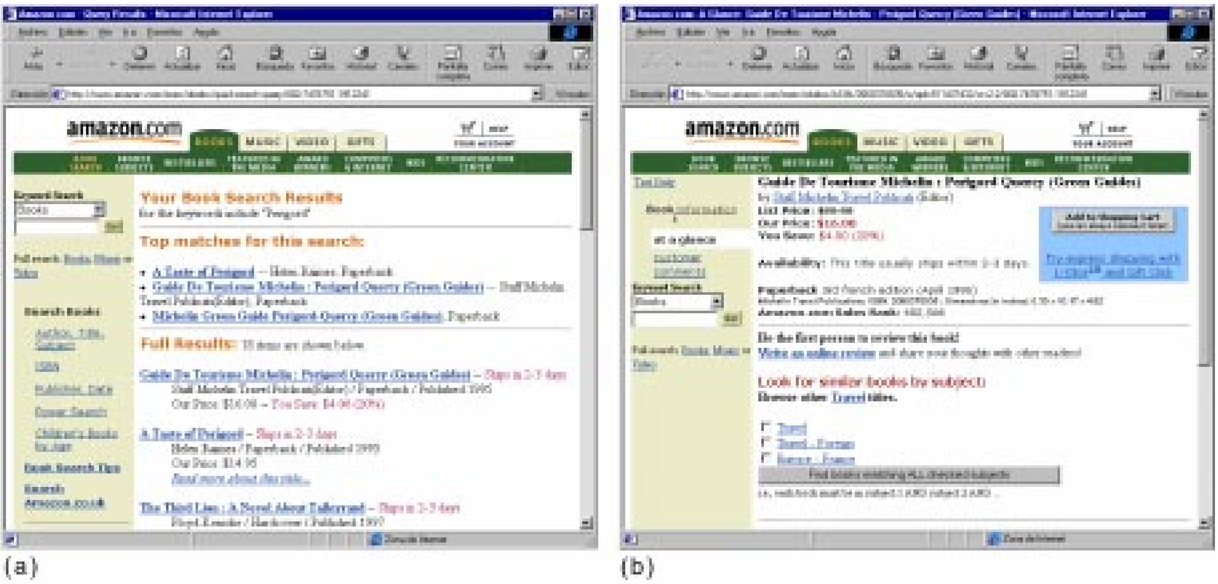
Amazon customers can browse through thousands of books and CDs by querying the bookstore, using keywords for titles, authors, etc. The navigational structure of the application is quite simple and it clearly reflects the application domain. It should be clear that this exercise is not meant as a criticism, especially since we are not fully aware of the designers’ objectives when selecting one particular solution for the set of problems they have faced. We next summarize the use of navigational patterns in this WIS.

3.1. Search results are sets

The use of sets is prevalent in Amazon; all search operations return the set of information objects that match the query. However, search-based navigation

[3] http://www.biztravel.com

[4] http://www.publishersdepot.com/



Your Book Search Results

for the keywords include "Perigord"

Top matches for this search:

- A Taste of Perigord — Helen Raines; Paperback
- Guide De Tourisme Michelin : Perigord Quercy (Green Guides) — Staff Michelin Travel Publications(Editor); Paperback
- Michelin Green Guide Perigord-Quercy (Green Guides), Paperback

Full Results: 18 items are shown below

Guide De Tourisme Michelin : Perigord Quercy (Green Guides) – Ships in 2-3 days
Staff Michelin Travel Publications(Editor) / Paperback / Published 1995
Our Price: $16.00 – You Save: $4.00 (20%)

A Taste of Perigord – Ships in 2-3 days
Helen Raines / Paperback / Published 1995
Our Price: $14.95
Read more about this title...

The Third Lion : A Novel About Talleyrand – Ships in 2-3 days
Floyd Kemske / Hardcover / Published 1997

Guide De Tourisme Michelin : Perigord Quercy (Green Guides)
by Staff Michelin Travel Publications (Editor)
List Price: $20.00
Our Price: $16.00
You Save: $4.00 (20%)

Availability: This title usually ships within 2-3 days.

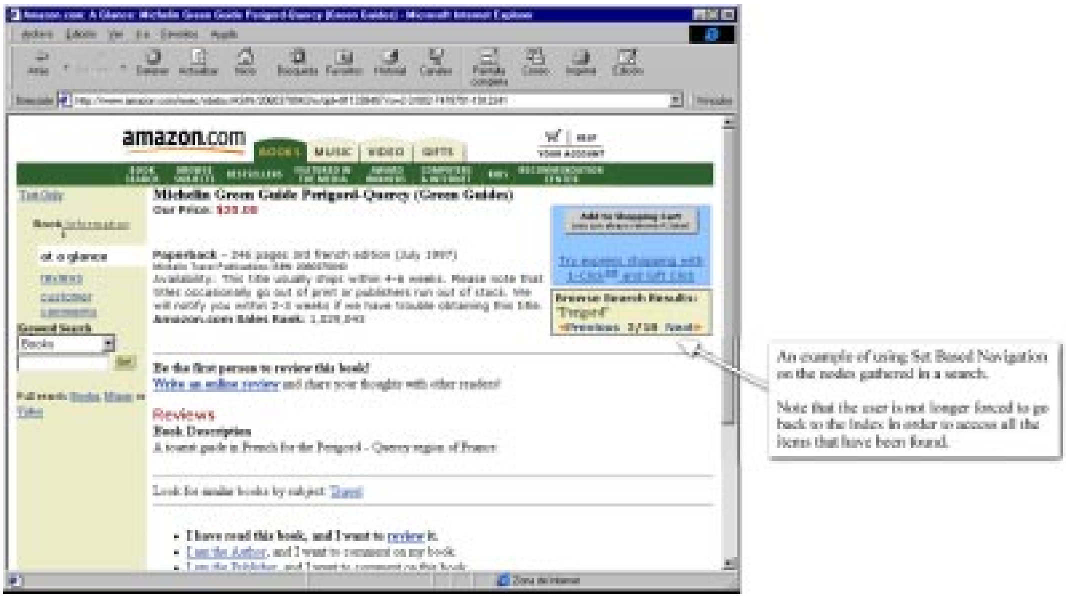
Be the first person to review this book!
Write an online review and share your thoughts with other readers!

Look for similar books by subject:
Browse other Travel titles.

Fig. 2. (a) A set index in Amazon. (b) A set member.

is not improved with intra-set navigation operations. In Fig. 2a we show the result of a search and in Fig. 2b one of the members of that set. The problems of backtracking to the index to navigate to another member of a set are well-known, as we have to scroll to the last navigated index element before proceeding to the next result.

We could solve this problem either by treating the index as an Active Reference [13], thereby allowing the set index to co-exist with set members on the desktop, or to improve navigation in the set as we show in Fig. 3. It should be observed that the two added buttons must have their destination assigned dynamically, according to the set we are traversing.



Michelin Green Guide Perigord-Quercy (Green Guides)
Our Price: $20.00

Paperback – 246 pages 3rd french edition (July 1997)
Availability: This title usually ships within 4-6 weeks. Please note that titles occasionally go out of print or publishers run out of stock. We will notify you within 2-3 weeks if we have trouble obtaining this title.

Browse Search Results:
"Perigord"
<Previous 2/18 Next>

Be the first person to review this book!
Write an online review and share your thoughts with other readers!

Reviews
Book Description
A tourist guide in French for the Perigord - Quercy region of France.

Look for similar books by subject: Travel

- I have read this book, and I want to review it.
- I am the Author, and I want to comment on my book.
- I am the Publisher, and I want to comment on this book.

An example of using Set Based Navigation on the nodes gathered in a search.

Note that the user is not longer forced to go back to the Index in order to access all the items that have been found.

Fig. 3. Improving 'Set-Based Navigation' in Amazon.

Notice that the site provides, for each book, a list of other books that the reader is likely to be interested in; this list is probably built based on other reader's choices. However, if the reader decides to look at the information of one related book, the context in which the navigation was taking place is lost. This problem could also be solved by applying ‘Set-Based Navigation’ to the set of related books of any given one.

The ‘Node in Context’ pattern can be applied here to add information to each book node, when the reader reaches the node via one of the ‘sets’ just mentioned. For example, reader's comments comparing to related books could be shown.

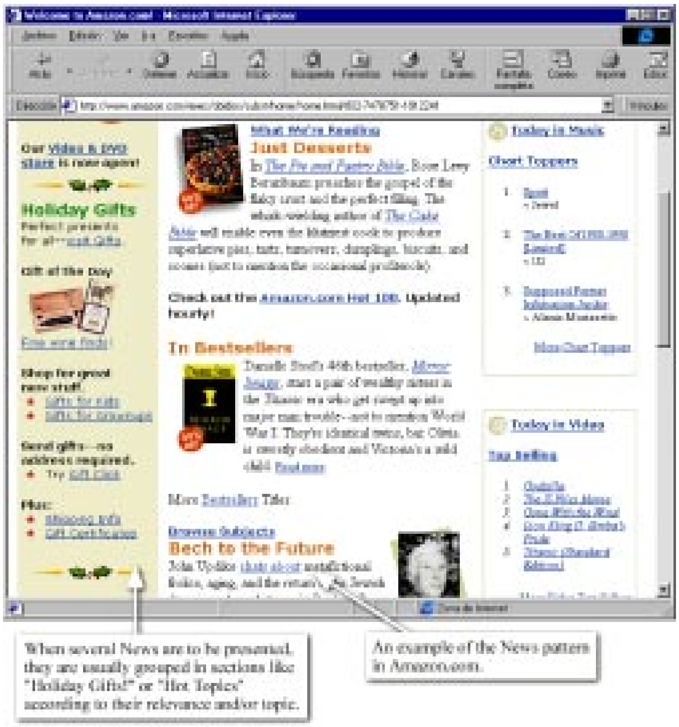
3.2. News

‘News’ is a critical design pattern in dynamic WIS; in Amazon it is widely used to keep customers informed about new releases. An interesting design decision in this WIS is that links to new additions are complemented with comments about the new product as we show in Fig. 4.

Clearly, the use of this pattern requires an additional maintenance effort since the home page must be periodically re-designed. However, semi-automatic tools or server templates such as ASP [6] may be used to simplify this process. Here ‘Set-Based Navigation’ and the ‘Node in Context’ pattern could be employed; for example when we navigate from a ‘News’ link we could explore other new additions by following intra-set links. If the reader navigates from a ‘News’ link in the home page to an item, its description does not have to emphasize it is a new item. On the other hand, if the reader reaches this node through another path, this information should be more prominent.

3.3. Landmarks

Landmarks are critical in this kind of WIS because they simplify the user's task while navigating. In Amazon there are many Landmarks providing access to the three different stores (books, CDs, gifts), to the user's shopping basket and to the user's account. The interface is kept uniform and the current



Our Video & DVD Store is now open!

Holiday Gifts

Perfect presents for all—visit Gifts

Gift of the Day

Fine wine finds!

Shop for great new stuff

- Gifts for kids
- Gifts for [illegible]

Send gifts—no address required.

- Try Gift Click

Plus:

- Shipping info
- Gift Certificates

What We're Reading

Just Desserts

In The Pie and Pastry Bible, Rose Levy Beranbaum preaches the gospel of the flaky crust and the perfect filling. The whisk-wielding author of The Cake Bible will enable even the klutziest cook to produce superlative pies, tarts, turnovers, dumplings, biscuits, and scones (not to mention the occasional profiterole).

Check out the Amazon.com Hot 100. Updated hourly!

In Bestsellers

Danielle Steel's 46th bestseller, Mirror Image, stars a pair of wealthy sisters in the Titanic era who get swept up into major man trouble—not to mention World War I. They're identical twins, but Olivia is sweetly obedient and Victoria's a wild child. Read more

More Bestsellers Titles

Browse Subjects

Bech to the Future

John Updike chats about metafictional fiction, aging, and the return of his Jewish

Today in Music

Chart Toppers

1 Spirit – Jewel
2 [illegible] – [illegible]
3 Supposed Former Infatuation Junkie – Alanis Morissette

More Chart Toppers

Today in Video

Top Selling

1 Godzilla
2 The X-Files Movie
3 Gone With the Wind
4 Lion King II: Simba's Pride
5 Titanic (Standard Edition)

When several News are to be presented, they are usually grouped in sections like "Holiday Gifts!" or "Hot Topics" according to their relevance and/or topic.

An example of the News pattern in Amazon.com.

Fig. 4. News in Amazon.

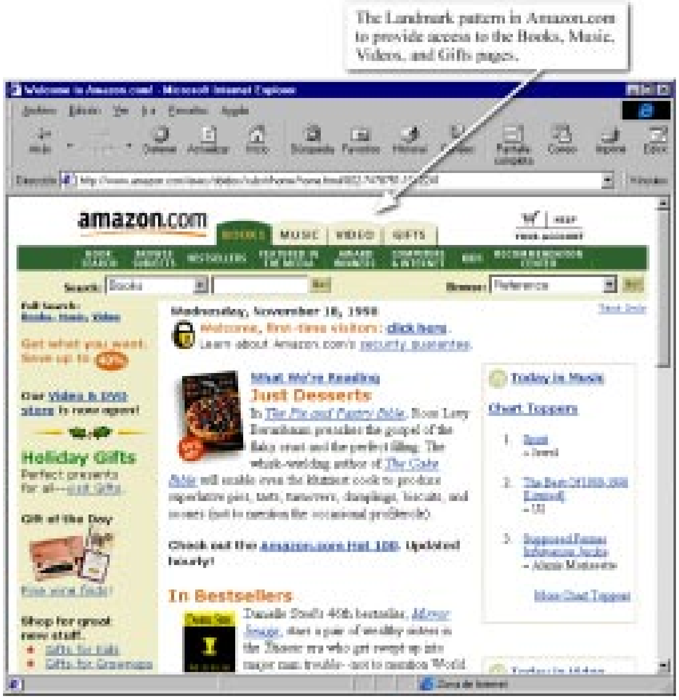

The Landmark pattern in Amazon.com to provide access to the Books, Music, Videos, and Gifts pages.

Fig. 5. Landmarks in Amazon.

sub-system is highlighted. Landmarks generalize the idea of Home page in early Web-sites by enriching it as new requirements emerge in complex applications like electronic commerce. In Fig. 5 we show how landmarks are used in Amazon.

3.4. Shopping basket

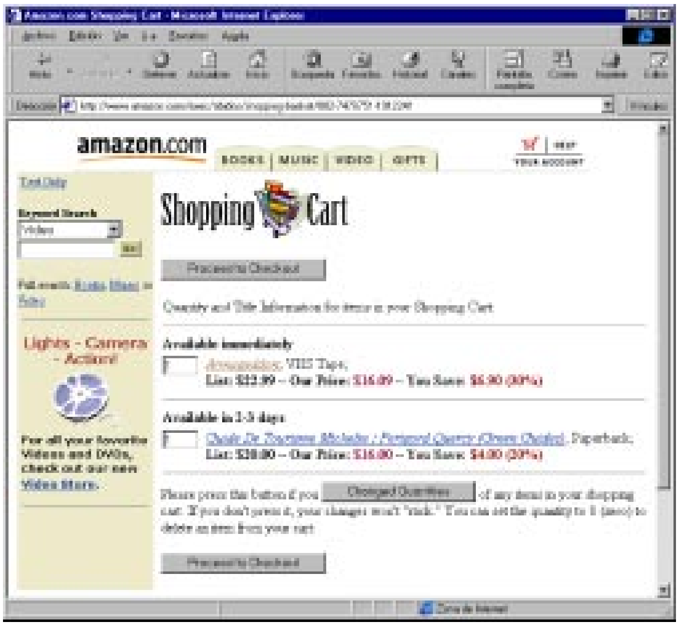
The shopping basket design pattern is used in Amazon to decouple the selection process from the checkout of products. The shopping basket is persistent across navigation sessions; the user can easily add products to the basket and the user can manipulate it to modify the number of items to buy. In Fig. 6 we show the shopping basket implementation in Amazon WIS.

Notice that ‘Set-Based Navigation’ and ‘Node in Context’ can be employed here as well. For example, if the reader is examining his shopping basket, he may wish to double-check the information about a selected book by clicking a link in the order itself. When he reaches the page describing the book, the page could contain, in addition to the usual information, links to the other items in the shopping basket. Therefore it is not necessary to go back to the order itself to select another item, as well as information (e.g., quantity and total cost) regarding this particular book as appearing in the order.

4. Introducing patterns into the development process

Though using a design method (like OOHDM or RMM) is a key approach for dealing with development complexity, good designers go beyond methods by applying their expertise to recurrent problems. This is the point in which patterns appear.

For instance, consider the problem of a node containing information related to the Rolling Stones, which can be accessed from an index of the latest rock albums released. Later, it could be necessary to access this node from the list of the oldest rock bands still in activity. In this example, a page has to be reached from different categories, which are disjoint: as most Web-sites structure is tree-shaped, where information is organized in categories and subcategories, nodes can be leafs of a


amazon.com
BOOKS | MUSIC | VIDEO | GIFTS
YOUR ACCOUNT
Shopping Cart
Proceed to Checkout
Changed Quantities
Lights - Camera - Action!
For all your favorite Videos and DVDs, check out our new Video Store.

Quantity and Title Information for items in your Shopping Cart

Available immediately

VHS Tape,
List: $22.99 – Our Price: $16.09 – You Save: $6.90 (30%)

Available in 1-3 days

Paperback,
List: $20.00 – Our Price: $16.00 – You Save: $4.00 (20%)

Please press this button if you Changed Quantities of any items in your shopping cart. If you don't press it, your changes won't "stick." You can set the quantity to 0 (zero) to delete an item from your cart.

Fig. 6. Shopping basket in Amazon.

single ‘branch’ or ‘category’. As a result of this, the upgraded Web-site may present the following problems:

- **Repetition of information**: the usual approach is to solve this problem at the implementation stage rather than during design, by either having the same information duplicated on a different node or not providing a link to a node that already belongs to other category.
- **Star-shaped navigation**: Since the information is tied-up to a hierarchical organization, it is usually not possible to browse all relevant items one after another. Rather, the user is forced to go back to a general index to look for the next anchor thus duplicating the navigation required to access the same amount of information. Examples of this ‘pathology’ can be found in many commercial Web-sites, when the user wants to browse over the services or products.

In the example above, the original problem is the lack of set-based navigation facilities. ‘Set-Based Navigation’ is a pattern for organizing information that is usually applied by experienced designers and is not trivial to discover for a novel designer, who may fail to address the real problem and solve it as an implementation one.

Design patterns have an invaluable importance in this context, since the design knowledge they present is not solution-oriented but problem-oriented. This a unique property of patterns: they describe problems that are often encountered at the moment of design, together with the forces involved and other related design problems. Patterns go beyond guidelines because they are applied in a context, have certain trade-off and usually appear in combinations (for example ‘Set-Based Navigation’ and ‘Nodes in Context’).

As we previously explained, design patterns are useful to record recurrent design themes in software systems. Studying successful applications and abstracting the problems and their solutions help to discover new patterns. This activity helps us to increase our understanding on the design process and therefore we can find commonalties among similar applications, establishing the context in which these patterns appear.

Design methods should evolve to incorporate some patterns into their repertoire of primitives or abstraction constructs. As an example OOHDM already includes some basic design patterns such as the ‘Set-Based Navigation’ and ‘Nodes in Context’ [14,15]. Augmenting a method with patterns eases its use, leveraging the representation effort by pro-

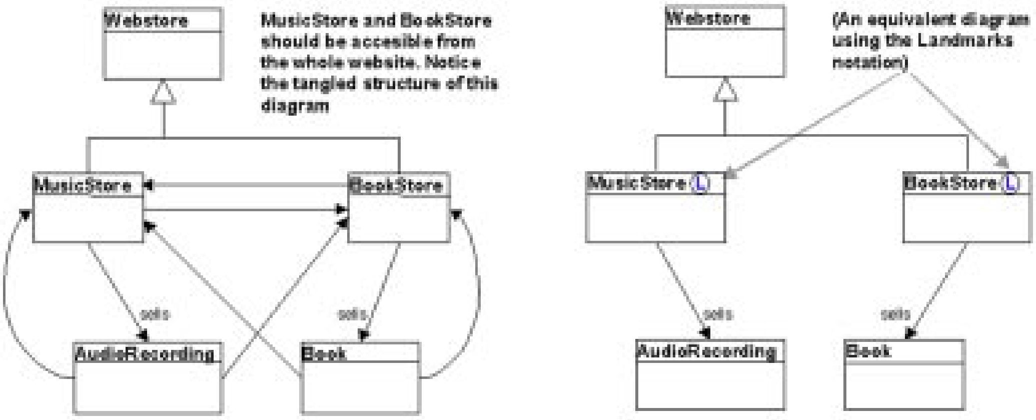

Webstore
MusicStore and BookStore should be accesible from the whole website. Notice the tangled structure of this diagram
MusicStore
BookStore
sells
sells
AudioRecording
Book
Webstore
(An equivalent diagram using the Landmarks notation)
MusicStore L
BookStore L
sells
sells
AudioRecording
Book

Fig. 7. Augmenting navigational diagrams with the Landmark pattern notation.

viding higher level design constructs. The addition of newer notations addresses two distinct aspects of the evolution of a method: the first one is to improve its expressive power and the second one is to simplify the recording of design decisions. An example of this is the Landmark pattern (Sections 2.3 and 3.3). In Fig. 7 we show an example of how the addition of higher abstraction primitives may help to simplify a navigation diagram. In this case we have represented Landmark nodes with a special graphical symbol. Though this notation may be more complex when dealing with nested or context landmarks, it clearly shows the improvement in the notation.

5. Concluding remarks

In this paper we have discussed how to use hypermedia design patterns to improve the navigational structure of Web information systems. We have presented some simple but effective navigational patterns addressing recurrent problems in WIS design. We have found those patterns in many successful applications and we have illustrated this paper with one of them: Amazon.com. We have also shown that the structure of this site could be substantially improved by applying two of these patterns: ‘Set-Based Navigation’ and ‘Nodes in Context’. This and other WIS also use many user interface design patterns (not presented here for the sake of conciseness).

The purpose of this paper is neither to introduce novel ideas nor to analyze existing problems while navigating WIS. Our aim is to motivate the World Wide Web community to record its collective experience in the form of design patterns, as has been done with success in other software design areas (see hillside.net/patterns for example). We are constantly facing similar problems; many of them have been already solved by other researchers or practitioners in related areas such as user interface design and hypertext. It is clear that designing WIS is a novel activity but we should learn from other software engineering areas. We have found patterns, both in the Alexandrian sense and in the object-oriented style of [2], a powerful and simple approach to record and convey development experience; we have used them in many new developments.

We have also discussed how to integrate patterns into the software development process. We envision two ways in which these and other patterns will improve the development enterprise: first, they can help designers to communicate their experience in a more systematic way, thus improving the collective knowledge of the World Wide Web community. Besides, and as new patterns are incorporated as design primitives (whenever this is feasible), they will reduce the amount of design documentation that is necessary to make a design understandable. We are working in this direction by adding different navigational and interface patterns to the OOHDM. We are also studying the way in which these patterns may be incorporated in early stages of the development process as for example during requirement elicitation. We have found that there is an interesting interleave among certain strategies as for example ‘Set-Based Navigation’ and existing formalisms for requirement engineering such as use scenarios. We are now exploring how to derive navigational contexts from scenarios or use cases. We are also studying how to integrate patterns into the Web Composition Markup Language (WCML) [5]. Finally, the hypermedia community has begun a collective effort to collect and catalogue these and others’ hypermedia patterns by building a repository containing comprehensive descriptions, examples and comments for each submitted patterns (see http://ise.ee.uts.edu.au/hypdev/ht99w/). We believe that as we build a dense catalogue of navigation and interface patterns we will be able to improve the process of Web information systems design.

References

[1] C. Alexander, S. Ishikawa, M. Silverstein, M. Jacobson, I. Fiksdahl-King and S. Angel, A Pattern Language, Oxford University Press, New York, 1977.

[2] E. Gamma, R. Helm, R. Johnson and J. Vlissides, Design Patterns: Elements of Reusable Object-oriented Software, Addison-Wesley, Reading, MA, 1995.

[3] A. Garrido, G. Rossi and D. Schwabe, Patterns systems for hypermedia, in: Proc. of PloP 97, Pattern Language of Program, 1997, http://st-www.cs.uiuc.edu/users/hanmer/PLoP-97.html.

[4] F. Garzotto, L. Mainetti and P. Paolini, Hypermedia design, analysis and evaluation issues, Comm. ACM, August 1995, pp. 74–86.

[5] H. Gellersen, R. Wicke, M. Gaedke, WebComposition: An object-oriented support system for the Web engineering lifecycle, in: Electronic Proc. of The 6th International WWW Conference, Santa Clara, USA, April, 1997.

[6] Internet Information Server and Active Server Pages. © Microsoft Corporation (see http://www.microsoft.com).

[7] T. Izakowitz, E. Stohr and P. Balasubramaniam, RMM: A methodology for structured hypermedia design, Comm. ACM, October 1995, pp. 34–44.

[8] F. Lyardet, G. Rossi and D. Schwabe, Patterns for dynamic websites, in: Proc. of PloP ’98, Allerton, USA, 1998, http://jerry.cs.uiuc.edu/~plop/plop98/.

[9] J. Nielsen, Multimedia and Hypertext. The Internet and Beyond, Academic Press, New York, 1995.

[10] J. Nielsen, The Alertbox; Current issues on Web usability; www.alertbox.com.

[11] G. Rossi, A. Garrido and S. Carvalho, Design patterns for object-oriented hypermedia applications, in: Vlissides, Coplien and Kerth (Eds.), Pattern Languages of Programs 2, Addison-Wesley, Reading, MA, 1996.

[12] G. Rossi, D. Schwabe and A. Garrido, Towards a pattern language for hypermedia applications, in: Proc. of the 3rd Annual Conference on Pattern Languages of Programs, Monticello, IL, September 1996, http://www.cs.wustl.edu/~schmidt/PLoP-96/program.html.

[13] G. Rossi, D. Schwabe and A. Garrido, Design reuse in hypermedia applications development, in: Proc. of ACM International Conference on Hypertext (Hypertext ’97), Southampton, April 7–11, 1997, ACM Press.

[14] D. Schwabe, G. Rossi and S. Barbosa, Systematic hypermedia design with OOHDM, in: Proc. of the ACM International Conference on Hypertext (Hypertext ’96), Washington, March 1996.

[15] D. Schwabe and G. Rossi, An Object Oriented Approach to Web-Based Application Design TAPOS (Theory and Practice of Object Systems), Wiley, New York, 1998.

[16] K. Takahashi and E. Liang, Analysis and design of Web-based information systems, in: Electronic Proc. of The 6th International WWW Conference, Santa Clara, USA, 1997.

[17] The VisualWave Programming Environment, Parc Place Systems, http://www.parcplace.com/products/vwave/vwv_prod.htm.

Gustavo Rossi (Ph.D., Puc-Rio, 1996) is full professor at La Plata University, Argentina. His research interests include design patterns and frameworks for hypermedia. He is one of the authors of the OOHDM methodology and is working on the application of design patterns in the hypermedia and Web field. He is the head of LIFIA, a computer science research lab in Argentina.

Daniel Schwabe (Ph.D., UCLA, 1981) has been working on hypermedia design methods for the past 10 years. He is one of the authors of HDM, the first authoring method for hypermedia, and of OOHDM, one of the mature methods currently in use by academia and industry for hypermedia application design. Dr. Schwabe is associate professor at the Department of Informatics of the Catholic University (PUC) in Rio de Janeiro.

Fernando Lyardet is an advanced undergraduate student at La Plata University, Argentina. He has been working in patterns and pattern languages for hypermedia and Web applications and in CASE tools for Web Information Systems. He is a research and teaching assistant at LIFIA.