

Patterns for Dynamic Websites

Fernando Lyardet, Gustavo Rossi(*),
LIFIA, Fac. De Ciencias Exactas, UNLP, Argentina
fer@sol.info.unlp.edu.ar, gustavo@sol.info.unlp.edu.ar

(*) is also at CONICET and UNLM
and

Daniel Schwabe.
Departamento de Informática. PUC-RIO, Brasil
schwabe@inf.puc-rio.br

Copyright © 1998, Fernando D. Lyardet. Permission is granted to copy for the PLoP-98 conference;
all other rights reserved.

Abstract

In this paper we present two hypermedia patterns for websites to explore solutions that have been adopted to face common problems about the information available on the user and the website sides. One aspect of this problem is analyzed on the News pattern to handle the problem of what do the users know about the changes on a website. Another aspect is analyzed in the Dynamic Configuration pattern from the website perspective, regarding the activities that are carried out by users and how these activities can be validated.

1. The NEWS Pattern

Intent:

How do you tell the users that there is new information available or updates?

Motivation:

Most large websites are tree-structured, which though not perfect, offers a simple mechanism to organize considerable amounts of information. These information spaces tend to be large, and are hardly ever completely navigated by a single user. Rather, users tend to navigate over a reduced subset of web pages, which are related to their interests.

In this context, new information is added or updated. Since most users navigate a website neither thoroughly nor regularly, this becomes an issue to be considered, because the success of a new product may be closely related to the customer's knowledge of its existence. Clearly, vital commercial information like this cannot be left to chance that it will be discovered.

On the other hand, trying to solve this problem poses a design challenge for web designers who must balance between a well-structured website where information is organized in items with sub-items, etc. and, a structure-less, star-shaped navigational structure where all information is reachable from the homepage. The latter approach is clearly not desirable [Nielsen] because the web usability is greatly reduced and it becomes unmanageable as the website becomes bigger. Therefore, how is the user provided with instant feedback of any recent changes or additions to the information available, while maintaining a well-structured website?

Solution:

Provide visual feedback to clearly announce the recent changes to provide immediate feedback to the website visitors

Implementation

Changes might be announced to visitors on a website-wide basis or within a local context. The first approach structures the homepage in such a way a space is devoted to the newest additions, presenting descriptive “headlines” regarding the newest additions, and make them anchors to link them with their related pages. 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 whole website.

On the other hand, the second approach adds visual clues to the pages that are new or that have been updated to make user note that those pages have changed.

Both approaches might be used together. It is interesting to note that neither implementation conflict between them, nor forces the use/implementation of the other.

Applicability

Use the News pattern when:

- It is needed to communicate to the users the newest additions or changes to the information available on a website.

Examples:

There are hundreds of examples of this pattern. In this paper we present three of them: the first two are from companies where the news pattern is used extensively for corporate announcements <http://www.sun.com>, <http://www.inprise.com>, and finally <http://www.nga.gov> where the news pattern is used to announce new collections and the current tours available.

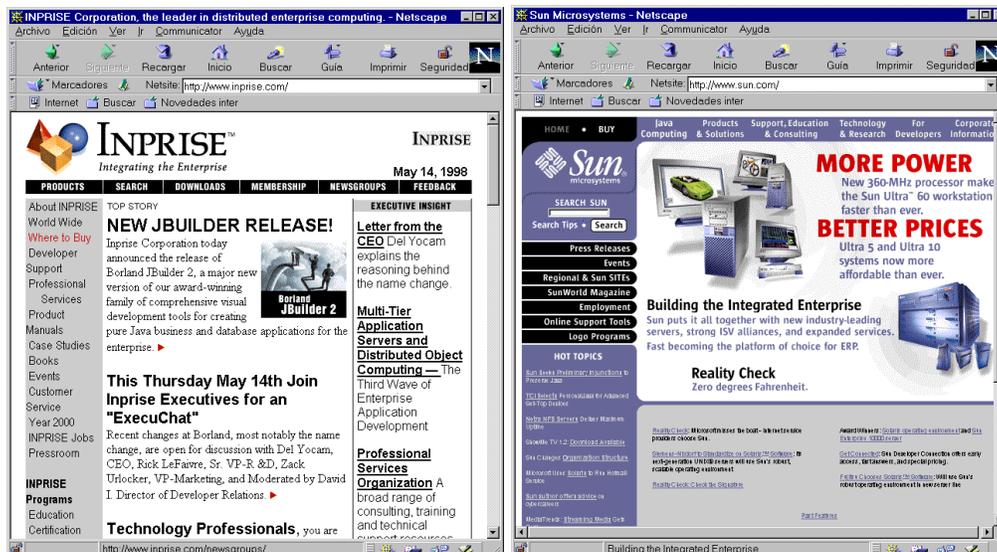


Figure 1: Two examples of websites use the News pattern: www.inprise.com and www.sun.com

Related Patterns:

Information-Interaction Decoupling pattern: this pattern describes the problem of a page of a complex application displaying different contents, and related to many other pages, thus providing many anchors, and proposes dividing of the input communication channels from the output channels, by grouping both sets separately. Allow the "input interaction group" to remain fixed while "the output group" reacts dynamically to the control activation

2. The Dynamic Configuration Pattern

Intent

How to provide the user with the means to perform a selection over a set of options that might be arbitrary large, while keeping track of them and being able to perform the necessary validations over the selections that are being made.

Motivation

Often, it is necessary to provide the users with the ability of specifying options or selecting items in order to perform an action such as selecting which kind of news are we interested in, the destination and departure cities for a flight reservation, books in a bookstore or pictures from a collection to build a customized poster.

Many websites have e-commerce capabilities, and therefore being able to select items is required in order to purchase them. Nevertheless, selections can be not only a precondition to be fulfilled, but also an activity in itself. As an example we can explore many websites for travelers where users can select a departure and arrival cities to find out how many flights from which companies are available, and this activity does not imply a purchase.

Buying goods on Internet is now a daily reality, ranging from software and books to flowers and pepperoni; e-commerce has become a new way of interaction among companies and their customers.

Developing a website where customers can make selections brings about both technical and design problems. On the technical side, issues like providing secure transactions, high performance connectivity among web servers and DBMS are addressed, while providing simple and flexible user interaction means are common design concerns.

Some of the design problems involved are:

- How to keep a good web information structure while providing e-commerce capabilities.
- How to provide the ability of creating a order for several different products instead of forcing different transactions for each different product we want to buy.
- How to give the user the chance to change the order at any moment.
- How to cope with users with low bandwidth and unstable Internet connections.

Many websites define their structure on the basis of e-commerce, with pages where you select the product you want out of a list. Although this may appear as a natural approach, it is suitable only in those cases where only a small amount of products are to be offered and it is not likely to grow or change very often.

The simplicity of these design scheme allows existing websites to add e-commerce capabilities by adding a page with a list of products and an anchor on the homepage, leaving the whole website untouched.

Nevertheless, this simplicity leads to new problems since all the products are offered together in a list. In these systems, the user has to navigate over the pages related to the products, write down their names and eventually ISBN or model number and finally go to the product list and select the right one, and repeat this last procedure until all items have been ordered. It is easy to see that this interaction model is not only error-prone but also tedious since it forces the user to write down a model specification and repeat ordering procedures. Since navigating from one page to another in a Web application takes longer than moving from a main window to a dialog box in a traditional application, there's also an inherent cognitive load because users have to keep contextual information in their minds while waiting. It also makes their work last longer than they expect [Shubin].

Web information systems are used not only by people who have access to a full Internet connection but also by those accessing the web with a modem. This means that if a product is available on Internet, it should maximize user access performance and reduce unnecessary navigation. This consideration falls beyond simple graphics fine-tuning and preparation, where design considerations are made to those users accessing via unstable connections to allow, for example, resume an "on-going" transaction after an involuntary connection shutdown.

Solution

Provide the users a different metaphor closer to the activity of navigation, to allow selecting the items as the relevant products are traversed, by providing a mechanism to add the current displayed product to be added to a list of items (the "shopping basket") to be purchased.

This solution adds to the page of every Internet available product, a button to allow the current displayed product be added to a list of products to be purchased, releasing the user from the task of writing down product names, model numbers or ISBN codes, and therefore reducing navigational overhead to a minimum.

Furthermore, it adds a very important capability: it allows the system to perform validation upon the items selected. If a customer is planning is vacation with an Internet travel agent service, the travel agent is capable of doing some checking on the arrival and departure times on hotel reservations and viability of flight connections.

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

Applicability

Use the Dynamic Configuration pattern when:

- The amount of items to be offered is high, for example a bookstore.
- The list of products changes rapidly, as well as their stock.
- Adding/removing new products with minimum effort regardless of the amount of products being offered.
- Need to perform certain validation on the shopping list, as the user adds items to it.

Participants

- Products: the goods that are offered to the customers. Usually, offered presented on one per page basis, though this is not a requirement.
- Dynamic Configuration: a list of items that is being filled by the user selections.

Examples:

There are hundreds of examples of these patterns available on the web. In this paper we present three well-known examples: Amazon Bookstore (<http://www.amazon.com>) is a good example of a huge offer of products.

The second example is Business travel (<http://www.biztravel.com>) where the user adds different destination to a business tour, together with car and hotel reservations, and 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).

Finally, we present another use of the Dynamic Configuration pattern at PublishersDepot (<http://www.publishersdepot.com/>), an image bank where the users can search for different kinds of images (textures, backgrounds, etc) and put them on a list. Furthermore, the users are able to create different lists of selected pictures that are persistent, that is, the website keeps those lists of selections made by a user and every time the user logs in, new additions or deletions can be made to these selections lists

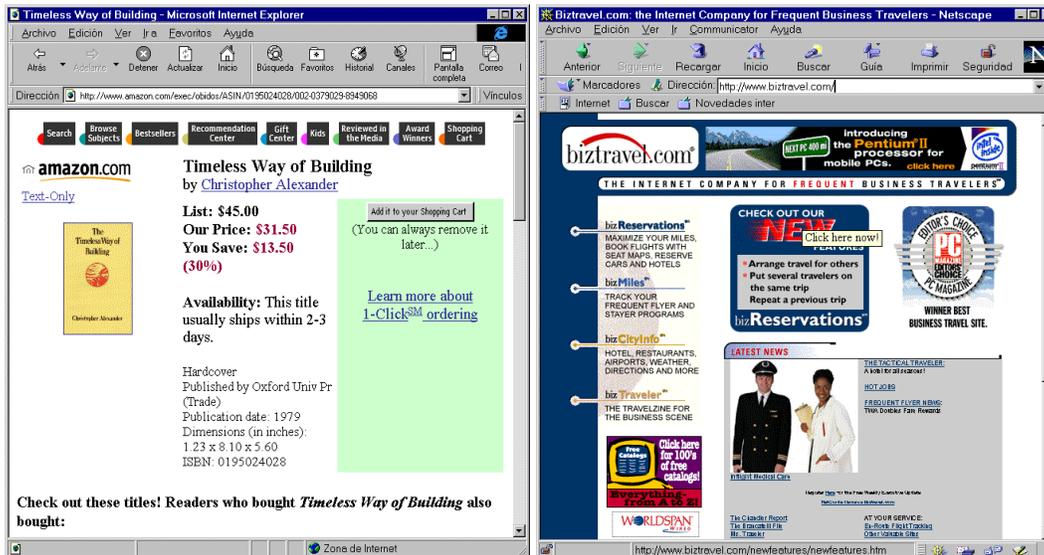


Figure 2 and 3: Two examples of the Dynamic Configuration pattern: www.amazon.com and www.biztravel.com



Figure 4: another example of dynamic Configuration available in www.publishersdepot.com where the users can configure different lists of images, and those lists are persistent between different user sessions

Implementation:

The easiest implementation of this is an arbitrary set of bookmarks made by the user, but since it is not possible to relate them and therefore treat them as a whole within transactions with a webserver, is therefore a very poor and expressiveness option.

Most implementations rely on thin clients (plain html) and the server creates a new temporary transaction for every visitor. These transactions are named in a many ways: "Shopping cart", "My Bag", etc. And the transactions' lifetime depends on which policy is adopted. There are many variations that can be found, and we can group them in persistent and non-persistent. In the first group, user selections are stored in a database and ask the visitor for a login name and a password to access those previously made selections, while in the second group the selections made by users are volatile, and they provide different policies of discarding selections such as like time aging or user reconnection.

Bibliography

- [Garrido 97] A. Garrido, G. Rossi and D. Schwabe. "*Pattern Systems for Hypermedia*" In Proceedings of PloP'97, Pattern Language of Programming, 1997.
- [GOF] "Design Patterns. Elements of Reusable Object Oriented Software." Gamma E., Helm R., Johnson R., Vlissides J.
- [Lyardet98] Fernando Lyardet, Gustavo Rossi and Daniel Schwabe. "Using Design Patterns in Educational Multimedia applications". To be presented at the AACE EDMEDIA Conference, Freiburg Germany.
- [Nielsen Alert Box] "The Alertbox: Current Issues in Web Usability". <http://www.useit.com/alertbox/>
- [Rossi 97] G. Rossi, D. Shwabe and A. Garrido: *Design Reuse in Hypermedia Design Applications Development* Proceedings of ACM International Conference on Hypertext (Hypertext'97), Southampton, UK, 1997, ACM Press.
- [Schwabe 96] D. Schwabe, G. Rossi and S. D. J. Barbosa. "*Systematic Hypermedia Application Design with OOHDM*". Proceedings of Hypertext'96 (HT96). Washington, March 1996.
- [Schwabe96] D. Schwabe, G. Rossi and S. D. J. Barbosa. "[Systematic Hypermedia Application Design with OOHDM](#)". Proceedings of Hypertext'96 (HT96). Washington, March 1996.
- [Shubin] Hal Shubin and Margarrt M. Meehen: "Navigation in Web Applications". 1997, ACM Interactions. Volume iv number 6.
- [Nielsen] Jakob Nielsen, "Hypertext and Hypermedia"