WEB ENGINEERING
An Aspect Oriented Approach

Joumana Dargham, Sukaina Al Nasrawi
American University of Beirut,Bliss Street, Beirut, Lebanon

Keywords: Aspect oriented Programming, OOHDM, Web-design.

Abstract: Web-Engineering has become, nowadays, the main research interest for software developers. With the spreading of use of the World Wide Web and the need for a new category of applications the research community has shifted its interest toward a new era of applications: web-based applications. To parallel the fast growth of the technology and the new needs for general/special web-applications, research should be done to improve the development and standardize it as it is done for non-web applications. Many development and programming tools were implemented to support web-engineering, however studies at the design level are still premature. Frameworks, design methodologies and web-based development tools are at an experimental level and depend on individuals efforts. In this context and considering that overheads for web-development are obstacles more than new methodologies, we are adopting the aspect-oriented approach for the development of web-applications.

1 INTRODUCTION

In the past few years, the web based information systems have become the center of interest of many researchers and business communities. The main challenges related to their development appear when dealing with large complex web applications. They are powered by distributed software implemented using a myriad of languages like HTML, XML, JavaScript, etc. The latter encouraged the researchers who have had a certain experience with software engineering to explore the WWW. Keeping in mind the positive impact that software engineering had on traditional applications; They proposed a new approach called Web Engineering. The approach deals with the establishment and use of scientific, engineering and management principles, disciplined and systematic approaches to the successful development, deployment and maintenance of high quality web based systems and applications (Ginige, A, 2002).

Within this context, many efforts have been addressed towards defining design models for web applications such as the Object Oriented Hypermedia Design Model (OOHDM) or even frameworks (Schwabe, D., 2001) to improve reusability. They all explored the maintainability and scalability of the web application that may be simplified by modularizing the web system. The importance of modularization leads us to the notion of separation of concerns in Aspect Oriented Programming or AOP. In this paper, we tend to apply the concepts of AOP onto the web environment to benefit from the advantages of separation of concerns and end up having an aspect based web design model. Very few works have presented aspect oriented design solutions although we know that distributed enterprise applications, such as web applications, have lots of crosscutting concerns. This scarcity in AOP approaches, especially with web applications, highlights the importance of the new developed approach and sheds the light on the need for the modularity provided by AOP in the web domain. We try to apply the concepts of AOP onto web engineering and design web applications using aspects in order to clearly structure the application in terms of functionalities and to improve reusability in web applications. In what follows, we will first, in section 2, review the major characteristics of web applications including the difficulties that should be taken into consideration when building web systems. Then, in section 3, we explore the key concepts of OOHDM and pinpoint the way it can be used to move into an aspect oriented design model. Section 4 proposes a new approach that adapts the concepts of AOP onto web applications. Finally, section 5 concludes the paper by mentioning the future works and plans.

Dargham J. and Al Nasrawi S. (2005).
WEB ENGINEERING - An Aspect Oriented Approach.
In *Proceedings of the Seventh International Conference on Enterprise Information Systems*, pages 267-270
DOI: 10.5220/0002525302670270

2 WEB APPLICATION CHARACTERISTICS

The quality of both Web and traditional applications is judged according to the application's reliability, usability and maintainability. However, unlike many traditional software vendors, if a new company puts up a new competitive site of a quality that is higher than existing websites, customers will almost immediately shift their business to the new site once they discover it (Offutt, J. 2002). Usually, when dealing with traditional applications, users take their time before shifting to another application however in the web environment, no loyalty is expected from any customer. To gain users' trust in the web domain, many characteristics that assess the quality of the application have to be considered These include: reliability, usability and security in addition to many other important factors such as: availability, scalability and maintainability. In the case of online information systems, the required quality may be higher than many traditional information systems (Holck, J, 2001). Despite the different issues handled in the newly emerging web approaches such as OOHDM, this quality requirement must be clearly taken into consideration.

3 BACKGROUNDS: OOHDM

The Object Oriented Hypermedia Design Model (OOHDM) uses abstraction and composition mechanisms in an object oriented framework to, on one hand, allow concise description of complex information items, and on the other hand, allow the specification of complex navigation patterns and interface transformations (Schwabe, D., 2002). With OOHDM, we end up having two model levels that are the conceptual and navigational model. The conceptual model shows classes and their relationships specifically related to a domain (Lima, F., 2002). It is concerned with the application's domain semantics instead of the type of users and tasks. On the other hand, the navigational model in OOHDM describes the navigational structure of a hypermedia application in terms of navigational contexts, which are induced from navigation classes such as nodes, links and indices (Schwabe, D., 2002). The navigational model consists of two schemas that are the navigational class schema and the navigation context diagram. The former defines all navigable objects as views over the application domain while the latter defines the main structuring primitives for the navigational space: navigation contexts and links that connect them (Lima, F., 2002). Note that different navigational models may be built for the same conceptual schema to express different views on the same domain. The navigation contexts which are sets of navigation objects may be specified as groups of contexts too (Refer to (Lima, F., 2002) for illustrations and more details). In this paper, we tend to complement the OOHDM by providing an aspectual representation of the different possible navigation contexts through the adaptation of the concepts of AOP onto web applications. The conceptual part stays intact and we adopt the same definition and notation used in OOHDM. The following section details the newly introduced aspect oriented approach aimed at structuring the web application and achieving aspect reusability.

4 ASPECT ORIENTED NAVIGATIONAL MODEL

For web applications to be successful, the navigational structure must be carefully defined (Schwabe, D., 2001). In this paper, we mainly aspectualize the navigational model of a web application; An aspect refers to a specific functionality of the application. Note that any website has a special purpose to achieve. This purpose refers to the main functionality of the web system and it is designated as the **Core Aspect (CA)** of the web application. The other functionalities offered by the web system are called **Supporting Aspects (SA)**. Therefore, according to this approach, any web application can be modeled as a set composed of one core aspect and a number of supporting aspects that interact with each other in order to achieve a fully functional web system. This aspectual decomposition of the web application leads us to a more structured web design model which in its turn simplifies maintainability and scalability since the module where the code must be altered or added can be easily specified. This has a positive impact, for example, on the usability of an online application. Indeed, an easily scalable application will grow without problems and will provide new functionalities to its customers making it more and more usable. Figure 1 models the idea of dividing the web application into CA and SAs. In addition, there exist other types of online applications which have separate profiles and users. In such applications, a new aspect is introduced and it is referred to as the **Authenticity Aspect (AA).** In this case, the application is modeled as a set of AA, CA and SAs. We can not access the CA unless the AA returns correctly. Figure 2 illustrates the usage of AA.

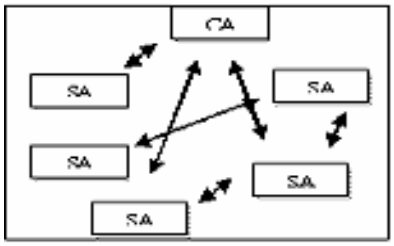

Figure 1: Aspect decomposition of a web application.

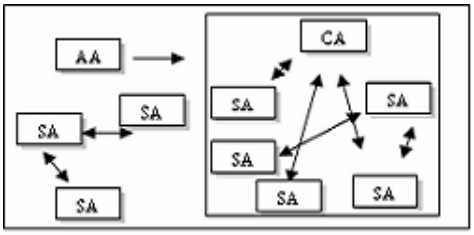

Figure 2: Aspect decomposition of a web application with AA.

If we consider a CD store online application example, The main functionality or the main purpose of the web application would be to buy CDs. So the core aspect of the online application is to "Buy CDs". However, the same website provides a variety of options such as viewing the list of available CDs, sending comments to the webmaster besides many others. These additional functionalities fall into the category of SAs. Figure 3 illustrates the idea. Also, Figure 4 represents a sample aspect model of an online application with AA such as msn hotmail application. Note that in figures 3 and 4 we only include few functions due to scarcity of space.

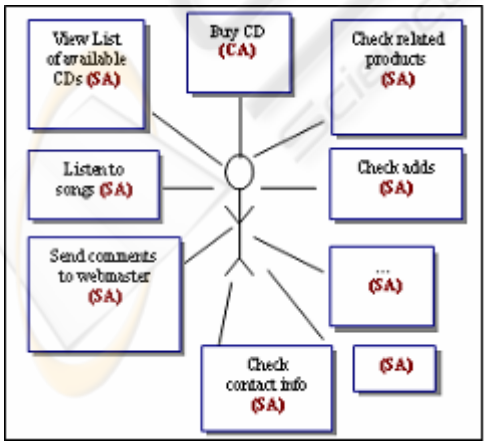

Figure 3: Aspect model of an online CD store.

Also, we distinguish two different types of SAs, mainly the static and generic supporting aspects. This differentiation simplifies the developer's task when it comes to updating an online application. From the

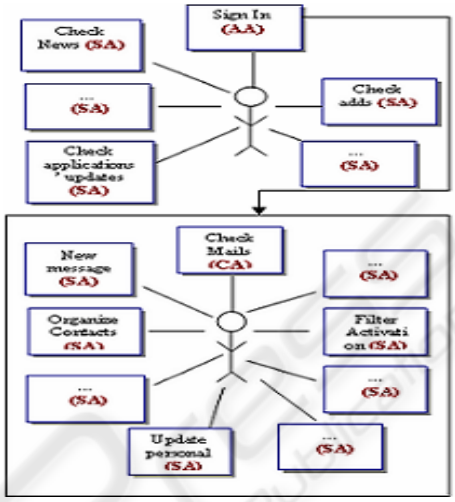

Figure 4: Aspect model of a web application with AA.

SA's type the developer can limit the changes he/she needs to perform. In a static aspect, the update is mostly informational but in dynamic aspects the update can affect more than one entity and thus consistency has to considered. For example, if we were to send an e-mail to the webmaster, an interaction between the user and the application is required thus we categorize the "Send Comments" aspect as a Generic Supporting Aspect (GSA). However, the "Check Contact Info" SA provides static information about the website and can thus be categorized as Static Supporting Aspect (SSA). The interaction between the different identified aspects of an application mainly models the navigation aspect of a web application. Aspects are independent from each other, however in the CD store example, the user can view the list of available CDs in order to buy one. The "View List of available CDs" SA can lead us to the "Buy CD" CA. The latter will be studied and detailed in further studies.

5 CONCLUSION AND FUTURE WORK

In conclusion, by aspectualizing a web application, we are mainly separating its different concerns. The designer, when adding a new functionality, can categorize it, add its corresponding aspect and then

weave it into the built application. The light has been shed on the importance of AOP concepts in web environment and especially in web engineering. We plan to categorize these aspects and explore the common aspects between different web applications belonging to the same domain such as e-commerce and implement them.

REFERENCES

Offutt, J., 2002. *"Quality Attributes of Web Software Applications"*. George Mason University. IEEE Software.

Schwabe, D., Rossi, G., Esmeraldo, L., Lyardet, F., 2001. *"Web Design Frameworks: An approach to improve reuse in web applications*". Epartmento de Informatica, Brazil; LIFIA Facultad de Informatica, Argentina.IEEE.

Lima, F., Schwabe, D., 2002. "*Exploring Semantic Web Modeling Approaches for Web Application Design*". . 2nd International Workshop on Web Oriented Software Technology.

Schwabe, D., Rossi, G., 2002. *"The Object Oriented Hypermedia Design Model"*.www.telemedia.puc-rio.br/oohdm/oohdm.html

Holck, J., Clemmensen, T., 2001. *"What makes Web-development different",* Electronic Proceedings of the 24th Information Systems Research Seminar in Scandinavia, Ulvik in Hardanger, Norway, 11-14 August.

Ginige, A.,2002. "*Web Engineering: Managing the Complexity of Web Systems Development*". University of Western Sydney, Australia. Proceedings of the 14th international conference on Software engineering and knowledge engineering.