Master Thesis

THE SMARTAG FRAMEWORK FOR THE DYNAMIC RECONSTRUCTION OF ADAPTIVE WEB CONTENT

Mario Belk

UNIVERSITY OF CYPRUS

DEPARTMENT OF COMPUTER SCIENCE

June 2009

ABSTRACT

Mass customization should be more than just configuring a specific component (hardware or software), but should be seen as the co-design of an entire system, including services, experiences and human satisfaction at the individual as well as at the community level.

The main objective of this thesis is to implement and evaluate a dynamic Web-based framework, called smarTag, for achieving mass customization on the Web based on human factors. SmarTag is an easy to use framework that enables any entity, Web designer and / or developer to enhance their Web services (technology and language independent) with adaptive Web objects that adapt according to the users' cognitive factors. More specifically, given the users' individual differences, the same service content provided by an organization will be reconstructed and delivered differently based on the users' profile typologies. This way, we will increase information assimilation, accuracy on cognitive targets' searching activities and consequently enhance acceptability of the On-line services. In order to achieve this we extended the well known html model with a new set of tags; <csl> (cognitive style list) and <csli> (cognitive style list item). A Web Browser (Mozilla Firefox) Extension has been therefore developed in order for the browser to recognize and implement the set of tags for the dynamic reconstruction and adaptation of Web content to the individual characteristics of the users.

Therefore, the main objective of this thesis is to describe the smarTag architecture and its components as well as the involved theoretical implications. Towards this point, an overview of Web Personalization techniques and methods is presented and ways on how they can be integrated with Mass Customization of Web services and products are suggested. A high-level analysis of major Web services / sites with regards to the degree of customization based on a given cognitive framework is also outlined, as well as a comprehensive review of current Web Development Frameworks. Finally, an evaluation of the smarTag System concludes the thesis. The initial results of the evaluation have proven that the proposed framework do not degrade the

efficiency (in terms of speed and accuracy) during the Web content adaptation process as well as increases users' satisfaction and efficiency of information processing (both in terms of accuracy and task completion time), while users navigating in the personalized condition rather than the original one.

THE SMARTAG FRAMEWORK FOR THE DYNAMIC RECONSTRUCTION OF ADAPTIVE WEB CONTENT

Mario Belk

A Thesis

Submitted in Partial Fulfilment of the

Requirements for the Degree of

Master of Science

at the

University of Cyprus

Recommended for Acceptance

By the Department of Computer Science

June 2009

APPROVAL PAGE

Master of Science Thesis

THE SMARTAG FRAMEWORK FOR THE DYNAMIC RECONSTRUCTION OF ADAPTIVE WEB CONTENT

Presented by

Mario Belk

Research Supervisor Name: Prof. George Samaras

Committee Member: Prof. Constantinos Pattiches

Committee Member: Dr. Andreas Andreou

University of Cyprus

June 2009

ACKNOWLEDGEMENTS

Firstly, I would like to thank my supervisor, Prof. George Samaras, for giving me the opportunity to participate in this project, which has given me a comprehensive exposure and a rich experience to the core areas of research. The experience I gained is indescribable. I have therefore deepened my knowledge and skills and increased my horizons.

My appreciations go to my Co-Supervisor and friend, Dr. Panagiotis Germanakos, for his unrelenting help and support throughout the duration of this project. Panagiotis has been a true source of inspiration, always more than willing to take time out in order to assist me during crucial times in my project.

I furthermore would like to thank my AdaptiveWeb team fellows; Nikos Tsianos, Zacharias Lekkas, Prof. Costas Mourlas, Prof. Andreas Demetriou and Prof. George Spanoudes for the provision of the theoretical framework that the smarTag system development has been based upon. At this point it has to be mentioned that all the theoretical framework and analysis has been conducted by the AdaptiveWeb Team.

I also thank my lab fellows Anna Sialarou, Panagiotis Andreou, Dimosthenis Gerogiades and Chrys Panagiotou for their endless support and great ideas.

Finally, I would like to acknowledge my family; their support and patience has been endless. Their experience, wisdom and guidance have helped guide me through life. I thank them for their sacrifices, as without them I would not be half the man I am today.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Motivation

Peoples' lives today are more turbulent and diversified. The "one size fits all" [1, 2, 3] model could be considered out-of-date. People now want to be seen and treated as individuals and many are prepared to pay for this. They are better educated and informed; able and willing to make their own decisions [10].

Mass customization moves towards this direction and it aims to replace mass production, which is no longer suitable for today's chaotic markets, growing product variety, and opportunities for eCommerce and eServices (also referred to as eServices or On-line services) in general.

Mass customization is a broad term. It could be easily perceived as a working and profitable business model with a whole spectrum of ways and methodologies that can companies benefit from. At the most visible end of the spectrum, companies can mass customize products for individual customers.

However, with the rapid development of Internet technologies and the imminent change of business processes and services provision, there is always the question whether mass customization and internet can co-exist, or better is it actually happening [10]?

Nevertheless, we could perceive mass customization, together with personalization, as a combination that together tend to change the business information systems offering personalized service relationships as a way of connecting with customers over a number of platforms and of differentiating their services from those of competitors.

1.2 Mass Customization

Mass customization should be more than just configuring a specific component (hardware or software), but should be seen as the co-design of an entire system, including services, experiences and human satisfaction at the individual as well as at the community level. It is widely acceptable that individuals differ in the way they think, feel, perceive and learn. Factors that could affect individuals' behaviour range from cognitive and mental processes to visual and emotional characteristics liable to determine their degree of information assimilation and learning capacity at a given moment.

1.3 Web Personalization

Web Personalization can be defined as any set of actions and definitions of specific rules that can tailor the Web experience to a particular user or set of users. Defining Web experience can be simply Web browsing, trading stocks or purchasing an item from an eStore. To achieve effective personalization, a system must shape the content in such an intelligent way, in order to anticipate the users' needs and provide them with personalized and adaptive information. Henceforth, Web personalization systems should use technologies and techniques for enabling machines to make more sense of the Web, with the result of making the Web more useful for humans [4].

Web Personalization is a growing research area and many systems have been developed providing personalization and adaptation of content to the user according to his / her characteristics. Nowadays, most Web personalization systems implement various techniques to extract the user profiling, which serves as the main component of such systems, and based on given user preferences and / or navigation behaviours, returns the requested personalized content.

1.4 The Personalization Problem

In recent years, there has been a rapid growth in research and experiments that work on personalizing Web content, according to user needs and indeed, the challenges ranging in this area are not few. Hereafter, there are a great number of issues that need to be taken into consideration. Subjectively, the size and heterogeneous nature of the data as well as the user environmental issues, such as current location and time [5], are amongst the most critical ones in the further Web personalization and adaptation systems design and development.

Indisputably, the user population is not homogeneous. To be able to deliver quality knowledge, systems should be tailored to the needs of individual users providing them personalized and adapted information. Although one-to-one service provision may be a functionality of the distant future, user segmentation is a very valuable step towards that direction. User segmentation means that the user population is subdivided, into more or less homogeneous, mutually exclusive subsets of users who share common user profile characteristics. The subdivisions could be based on:

- *Demographic characteristics* (i.e. age, gender, urban or rural based, region)
- *Socio-economic characteristics* (i.e. income, class, sector, number of employees, volume of business, channel access)
- *Psychographic characteristics* (i.e. life style, values, sensitivity to new trends) *individual physical and psychological characteristics* (i.e. disabilities, attitude, loyalty).

User characteristics and needs, determining user segmentation and thus provision of the adjustable information delivery, differ according to the circumstances and they change over time [5, 6, 7, 8].

The issue of personalization is a complex one with many aspects that need to be analyzed. Some of these issues become even more complicated once viewed from a mobile user's perspective, when wireless communication media and mobile device constraints are involved. Such issues include, but are not limited to:

- *What content to present to the user*. How to decide what to show, using user profiles, using the user history to predict future needs etc. When using user profiles the need for (i) storing the interests of the user in a format that is easy to be used, be updated or moved, and (ii) relating interests and items based on a semantics level (e.g., the theme interest of “flowers” is related to “florists” or even fertile producers) must be addressed.
- *How to show the content to the user*. Many users want to see the same things presented in a different format. In the wireless environment this also relates to the specific characteristics of the mobile device.
- *How to ensure the user’s privacy*. Every personalizing system acquires information about the habits of each user. This leads to privacy concerns as well as legal issues [9]. It could also lead to lack of user trust and could result in the failure of the system due to avoidance of its use.
- *How to create a global personalization scheme*. The user does not mind if a set of sites can be personalized but could very well be annoyed when at each one of them they have to repeat the personalization process. This is especially annoying and cumbersome for the user on the move carrying a resource poor mobile device.

These major issues of personalization could be summarized in the following phrase: “What, how and for everything” [5]. There are many approaches to address these issues of personalization but usually, each one is focused upon a specific area, i.e. whether this is profile creation, machine learning and pattern matching, data and Web mining or personalized navigation.

1.5 The smarTag System overview

The research that is described in this thesis focuses on incorporating theories of individual differences in information processing within the context of eServices and the dynamic reconstruction and adaptation of any hypermedia content to the benefit of the unique user.

Previous research of the AdaptiveWeb team [11, 12], in the field of adaptive eLearning, focused upon the enhancement of the quality of information presentation and users' interactions in the Web by matching their specific needs and preferences with the information space. It has been demonstrated that the incorporation of human information processing factors in eLearning environments leads to better comprehension on behalf of the users and increase of their academic performance [11, 12]. The comprehensive three-dimensional perceptual preferences model used comprises of the following human factors: Cognitive Style, Cognitive Processing Efficiency and Emotional Processing. The first dimension is unitary, whereas Cognitive Processing Efficiency is comprised of (a) Visual Working Memory Span (VWMS) [13] and (b) speed and control of information processing and visual attention [14]. The emotional aspect of the model focuses on different aspects of anxiety [15, 16, 17] and self-regulation.

Furthermore, since the WWW is by definition a huge resource of information, it would make much sense that individuals' information processing characteristics should be taken into consideration into this more generic context where constraints and challenges are radically differentiated. To that direction, our extended research efforts are focused on improving the effectiveness of Web services, and more broadly generic hypertext / hypermedia structures, by employing methods of personalization.

Chapter 2

Background Theory

Mass Customization and Web Personalization are widely appreciated as viable and promising strategies, which aim to provide product and services that best serve individuals' personal needs with near mass production efficiency. Personalization is adapting or sequencing solutions to fit individual differences, expectations, and needs. In contrast, mass customization is adapting to fit common characteristics identified for groups of users.

2.1 Web Personalization

Web personalization is the process of customizing the content and structure of a Web site to the specific needs of each user by taking advantage of the user’s navigational behaviour. Being a multi-dimensional and complicated area a universal definition has not been agreed to date. Nevertheless, most of the definitions given to personalization [18, 19, 20. 21] agree that the steps of the Web personalization process include: (1) the collection of Web data, (2) the modelling and categorization of these data (pre-processing phase), (3) the analysis of the collected data, and the determination of the actions that should be performed. Moreover, many argue that emotional or mental needs, caused by external influences, should also be taken into account.

Personalization could be realized in one of two ways: (a) Web sites that require users to register and provide information about their interests, and (b) Web sites that only require the registration of users so that they can be identified [22]. The main motivation points for personalization can be divided into those that are primarily to facilitate the work and those that are primarily to accommodate social requirements. The former motivational subcategory contains the categories of enabling access to information content, accommodating work goals, and accommodating individual differences, while the latter eliciting an emotional response and expressing identity [21].

All these paradigms have already been discussed in [90, 91]

2.1.1 Web Personalization Categories

In order to have a more comprehensive insight for its context, it is necessary to classify personalization in categories. These include: Link Personalization, Content Personalization, Context Personalization, Authorized Personalization and Humanized Personalization.

Link Personalization. This strategy involves selecting the links that are more relevant to the user, changing the original navigation space by reducing or improving the relationships between nodes. E-commerce applications use link personalization to recommend items based on the clients' buying history or some categorization of clients based on ratings and opinions. Link personalization is widely used in Amazon.com to link the home page with recommendations, new releases, shopping groups, etc. [23, 24]

Content Personalization. When content becomes personalized, user interface can present different information for different users providing substantive information in a node, other than link anchors. Most of the content personalization research is relative to text and hypertext personalization and can be further classified into two types:

(a) *Node structure customization (personalization)*, usually appears in those sites that filter the information that is relevant for the user, showing only sections and details in which the user may be interested. The user may explicitly indicate their preferences, or these may be inferred (semi-) automatically either from the user profile or navigation activity. For example, google.com offers Google Personalized Search, turning it on for all users when they create a new Google account. Personalized Search learns from a searcher's previous queries and clicks, attempting to improve results by guessing at ambiguities. In my.yahoo.com or in www.mycnn.com users choose a set of “modules” and further personalize those modules by choosing a set of attributes of the module to be perceived. Some “automatic” customization may occur based on location information (e.g by using the zip code of the user to select local to the user sport events). The outcome of these applications is that the user should be able to “build” their own page.

(b) *Node content customization (personalization)*, occurs when different users perceive different values for the same node attribute; this kind of content personalization is finer grained than structure personalization. A good example can be found in online stores that give customers special discounts according to their buying history (in this case the attribute price of item is personalized) [23, 24].

Context Personalization. Personalizing navigational contexts is critical when the same information (node) can be reached in different situations [23]. A navigational context is a set of nodes that usually share some property. For example in a Conference Paper Review Application, it is possible to access papers etc. Notice that one paper may appear in different sets and that different users may have different access restrictions according to their role in the Review application. Context personalization can also be adapted to the preferences of the learner and semantics of the learner’s current environment. One subcategory of context personalization is *terminal adaptivity*. That is adapting information to the characteristics of a device. It is applied on the mobile devices to satisfy learner’s demand for “learning as you go”. **Terminal Personalization** occurs on a per session basis. Personalization can be achieved by applying many axes of adaptation effecting both the navigational structure and appearance of the learning experience. It involves the tailoring of a resource to the current environment of the learner [25, 26].

Authorized Personalization. In the personalized user interface, different users have different roles and therefore they might have different access authorizations. For example, in an academic application, instructors and students have different tasks to perform.

Instructors want to access their class materials, such as upload, edit their class syllabus and give students' grades etc. On the other hand, students want to access the interface to find out their current GPA, their enrolment status, and their course work status etc.

Humanized Personalization. Bonnie Kaplan and Ramesh Farzanfar presented and studied an intelligent interactive telephone system (Telephone-Linked Care (TLC)) that provided information whether they were talking to a machine or to a person during TKC relationships with the TLC system [27]. If the dimension of the "emotional user interface" could be involved, it will be a huge step towards a concrete and universal definition of Web personalization. Unquestionably, this category of personalization still needs to be explored, with an extensive use of Artificial Intelligence technologies, since there are a lot of ambiguity and technical obstacles at present [28].

2.1.2 Web Personalization technologies

Web personalization can be defined as the process of customizing the content and structure of a Web site to the specific individual needs of each user taking advantage of the user's navigational behaviour. The steps of the Web personalization process include: (1) The collection of Web data, (2) the modelling and categorization of these data (pre-processing phase), (3) the analysis of the collected data, and (4) the determination of the actions that should be performed. The technologies that are employed in order to implement these processing phases are distinguished to:

Content-based filtering. Systems that are implementing these kinds of techniques are solely based on individual users' preferences. The system tracks each user's behaviour and recommends items that are similar to items the user liked in the past. It is based on description analysis of the items rated by the user and correlations between the content of these items and user's preferences. It is an alternative paradigm that has been used mainly in the context of recommending items such as books, Web pages, news, etc. for which informative content descriptors exist [29, 30, 31]. This technique is primarily characterized by two weaknesses, *content Limitations* and *over-Specialization*. There are content limitations like IR methods that can only be applied to a few kinds of content, such as text and image, and the extent aspects can only capture certain aspects of the content. On the other hand content-based recommendation systems provide recommendations merely based on user profiles, therefore, users have no chance of exploring new items that are not similar to those items included in their profiles and thus leading to over-specialization. Consequently, some more drawbacks that have been identified in time are [31, 32, 36]:

(a) Search-based models build keyword, category, and author indexes offline, but fail to provide recommendations with interesting, targeted titles. They also scale poorly for customers with numerous purchases and ratings.

(b) User input may be subjective and prone to bias.

(c) Explicit (and non-binary) user ratings may not be available.

(d) Profiles may be static and can become outdated quickly.

(e) May miss other semantic relationships among objects.

At this point it would be noteworthy to mention a complementary technique of Content-based filtering, namely **Social Information filtering**, essentially automates the process "word-of-mouth" recommendations; items are recommended to a user based upon values assigned by other people with similar taste. The system determines which users have similar taste via standard formulas for computing statistical correlations. Social Information filtering overcomes some of the limitations of content-based filtering. Items being filtered need not be amenable to parsing by a computer. Furthermore, the system may recommend items to the user which are very different (content-wise) from what the user has indicated liking before. Finally, recommendations are based on the quality of items, rather than more objective properties of the items themselves [31, 32]. Some of the most popular systems using content-based filtering are WebWatcher [24], and client-side agent Letizia [33].

Rule-based filtering. The users are asked to answer a set of questions. These questions are derived from a decision tree, so as the user proceeds to answer them. What he finally receives is a result (e.g. list of products) tailored to his needs. Content-based, Rule-based, and

Collaborative filtering may also be used in combination, for deducing more accurate conclusions. Some of the rule-based filtering drawbacks are: User input may be subjective and prone to bias, explicit (and non-binary) user ratings may not be available, profiles may be static and can become outdated quickly, and for large systems it becomes burdensome to manage. Related interesting systems include Dell, Apple Computer, Amazon.com, CDNOW, and Broadvision [36, 31, 34, 35].

Collaborative filtering. Systems invite users to rate the objects or divulge their preferences and interests and then return information that is predicted to be of interest to them. This is based on the assumption that users with similar behaviour (e.g. users that are rate similar objects) have analogous interests. There are two general classes of collaborative filtering algorithms, memory-based methods and model-based methods [34, 35, 29, 30]. Moreover, the goals in a collaborative filtering system are basically focused upon the reduction of computation time, the increase of the extent in which predictions can be computed in parallel, and the increase of prediction accuracy. Collaborative filtering can further refine the process of giving each individual personal recommendation compared to rule-based filtering. It overcomes the drawbacks of the content-based filtering because it typically does not use the actual content of the items for recommendation. It usually works based on assumptions. With this algorithm the similarity between the users is evaluated based on their ratings of products, and the recommendation is generated considering the items visited by nearest neighbours of the user. In its original form, the nearest-neighbour algorithm uses a two-dimensional user-item matrix to represent the user profiles. This original form suffers from three problems, *scalability*, *sparsity*, and *synonymy* [32, 42]. Some more highlighted drawbacks of collaborative filtering are focused upon: (a) Collaborative-filtering techniques are often based in matching in real-time the current user's profile against similar records obtained by the systems over time from other users.

However, as noted in recent studies, it becomes hard to scale collaborative filtering techniques to a large number of items, while maintaining reasonable prediction performance and accuracy. Part of this is due to the increasing scarcity in the data as the number of items increase. One potential solution to this problem is to first cluster user records with similar characteristics, and focus the search for nearest neighbours only in the matching clusters. In the context of Web personalization this task involves clustering user transactions identified in the pre-processing stage; (b) traditional collaborative filtering does little or no offline computation, and its online computation scales with the number of customers and catalogue items. The algorithm is impractical on large data sets, unless it uses dimensionality reduction, sampling, or partitioning – all of which reduce recommendation quality; (c) user input may be subjective and prone to bias; (d) explicit (and non-binary) user ratings may not be available; (e) profiles may be static and can become outdated quickly; (f) they are not able to recommend new items that have not already been rated by other users. An object will become available for recommendation only when many users have seen it and rated it, making it part of their profiles first ("latency problem"); (g) they are not satisfactory when dealing with a user that is not similar enough with any of the existing users [36, 43, 44, 45]. Some systems applied with this technique are Yahoo, Excite, Microsoft Network, Net Perceptions [35, 36].

Web-usage Mining. The typical sub-categorization of the Web mining research field falls into the following three categories: Web-content mining, Web-structure mining, and Web usage mining. The prerequisite step to all of the techniques for providing users with recommendations is the identification of a set of user sessions from the raw usage data provided by the Web server. Web usage mining is the only category related to Web Personalization. This process relies on the application of statistical and data mining methods to the Web log data, resulting in a set of useful patterns that indicate users' navigational behaviour. The data mining methods that are employed are: Association rule mining, sequential pattern discovery, clustering, and classification. Given the site map structure and usage logs, a Web usage miner provides results regarding usage patterns, user behaviour, session and user clusters, click stream information, and so on. Additional information about the individual users can be obtained by the user profiles [46, 47, 35]. The overall process can be divided into two components. (a) The *offline component* is comprised of the pre-processing and data preparation tasks, including data cleaning, filtering, and transaction identification, resulting in a user transaction file, and (b) the *data mining stage* in which usage patterns are discovered via specific usage mining techniques such as association-rule mining, association-rule discovery and usage clustering [44]. The increasing focus on Web-usage mining as the time passes derives from some key characteristics which are summarized as follows: (a) the profiles are dynamically obtained, from user patterns, and thus the system performance does not degrade over time as the profiles age; (b) using content similarly alone as a way to obtain aggregate profiles may result in missing important relationships among Web objects based on their usage. Thus, Web usage mining will reduce the need for obtaining subjective user ratings or registration based personal preferences; (c) profiles are based on objective information (how users actually use the site); (d) there is no explicit user ratings or interaction with users (saves time and other complications); (e) it helps preserve user privacy, by making effective use of anonymous data; (f) the usage data captures relationships missed by content-based approaches; (g) it can help enhance the effectiveness of collaborative or content-based filtering techniques. Nevertheless, usage-based personalization can be problematic when little usage data is available pertaining to some objects or when the site content attributes of a site must be integrated into a Web mining framework and used by the recommendation engine in a uniform manner [36, 48]. Noteworthy applications are Alta-Vista, Lycos, WebSift, and SpeedTracer [37, 35].

Demographic-based filtering. This specific technique could be roughly described as an approach that uses demographic information to identify the types of users that prefers a certain object and to identify one of the several pre-existing clusters to which a user belongs and to tailor recommendations based on information about others in this cluster [51, 52].

Agent technologies. Agents are processes with the aim of performing tasks for their users, usually with autonomy, playing the role of personal assistants [60, 61]. Agents usually solve common problems users experience on the Web such as personal history, shortcuts, page watching and traffic lights [62]. Some of the agents' main characteristics could be distinguished according to their abilities used and according to the tasks they execute. The former include characteristics such as *intelligence*, *autonomy*, *social capacity* (inter-agent communication), and *mobility*; while the latter classify the agents into *information filtering agents, information retrieval agents, recommendation agents*, *agents for electronic market,* and *agents for network management* [60].

Since the mobility dimension is also incorporated in this paper and therefore addresses vital needs as to locate the required information, on time, under any circumstances the use of intelligent mobile agents for the a given wireless environment could be proved ideal for implementing various Web personalization processes. Intelligent mobile agents are identified by some specific capabilities focused upon: (a) *Reduction of the network load*, instead of relying on numerous communication protocols to achieve network interaction, which would increase the network traffic, mobile agents can carry with them the data that is required for an interaction and process it locally; (b) o*vercoming network latency*, mobile agents can help in critical real-time systems where a response to environment changes is required in real time and latencies will not be tolerated. Mobile agents can be dispatched from a central controller to act locally and directly execute the controller's directions; (c) *asynchronous and autonomous execution*, after a task is assigned to a mobile agent, the agent will be dispatched into the network and become independent of the creating process. It can operate asynchronously and autonomously, relieving its owner from having continuously an eye on its activities. The agent's owner will be able to collect it at some later time, if needed; and (d) *dynamic adaptation*, mobile agents are capable of monitoring the environment in which they operate and react to the changes accordingly. Last but not least, (e) mobile agents are naturally heterogeneous, robust and fault-tolerant, and able to encapsulate protocols considered vital for the universal development of open, modular, ubiquitous and personalized mobile learning adaptive hypermedia applications [23, 25, 61, 27]. Pioneer personalization systems implemented with intelligent agents are: ARCHIMIDES, Proteus, WBI, BASAR, 1:1 Pro, Haystack, eRACE, mPersona, Fenix system, and SmartClient [59, 60, 61].

Cluster Models. These types of techniques are found mostly in the area of eCommerce and could be characterized as eCommerce recommendation algorithms. To find customers who are similar to the user, cluster models divide the customer base into many segments and treat the task as a classification problem. The algorithm's goal is to assign the user to the segment containing the most similar customers. It then uses the purchases and ratings of the customers in the segment to generate recommendations. The segments typically are created using a clustering or other unsupervised learning algorithm, although some applications use manually determined segments. Using a similarity metric, a clustering algorithm groups the most similar customers together to form clusters or segments. Because optimal clustering over large data sets is impractical, most applications use various forms of greedy cluster generation. These algorithms typically start with an initial set of segments, which often contain one randomly selected customer each. They then repeatedly match customers to the existing segments, usually with some provision for creating new or merging existing segments. For very large data sets – especially those with high dimensionality – sampling or dimensionality reduction is also necessary. Once the algorithm generates the segments, it computes the user's similarity to vectors that summarize each segment, chooses the segment with the strongest similarity and classifies the user accordingly. Some algorithms classify users into multiple segments and describe the strength of each relationship [24]. Cluster models have better online scalability and performance than collaborative filtering because they compare the user to a controlled number of segments rather than the entire customer base. The complex and expensive clustering computation is run offline. However, recommendation quality is relatively poor. To improve it, it is possible to increase the number of segments, but this makes the online user segment classification expensive. Typical examples of eCommerce systems are Amazon.com [23], Dell [35], and IBM.com [41].

2.2 Mass Customization

Traditionally customization and low cost have been mutually exclusive. Mass production provided low cost but at the expense of uniformity. Customization was the product of designers and craftsman. Its expense generally made it the preserve of the rich. To-day, new interactive technologies, like the Internet, allow customers to interact with a company and specify their unique requirements which are then manufactured by automated systems. Whilst this may at first seem complicated and beyond the average consumer, there are various ways to hide the technical details. In some cases the process will be handled by an organization's staff, a third party, or intermediary.

Mass customization is the customization and personalization of products and services for individual customers at a mass production price [10]. It is actually a further step of enhancing an individual customers' relationship. It may not always be practical to support one user at a time or to build in total personalization capabilities specific to one user. It may be preferable to start with a mass customized solution that identifies a few common critical success attributes that are key for improved performance. Based on recent technological advances it is possible to implement On-line services and communication environments accessed via Internet or Web technologies which may be personalized on the basis of individuals' preferences or even the intrinsic characteristics of the specific user like cognitive and emotional parameters, often referred to as human factors. Both content and its way of presentation (modality, visual layouts, ways of interaction, structure) as well as functional elements of such communication environments may automatically adapt their behavior according to the user needs and preferences enhancing the quality of service delivery and user satisfaction.

The greatest benefit of mass customization done well is technology's ability to make complex instruction easier by alternatively presenting content for a particular learner/user – what the user wants to see in the appropriate manner and at the appropriate time. A well-tested framework, based on sound scientific and design foundations, can help identify the capabilities, resources, issues, and content that is relevant, useful, and attractive to the targeted group of users. It also helps designers tailor products and services to satisfy the wide variety of requirements and capabilities (business, learning, educational, and personal goals).

Furthermore, mass customization raises the profits and lowers the costs. Whilst it is possible to manufacture at a mass produced price, there is the option to charge a premium whilst still retailing below the price of a custom product. This, in turn, will open a given product to a wider market. The uniqueness and profitability of customized products and services with the economies of scale and mass market penetration stemming from the mass production techniques that have to be adjusted and aligned with the current trends ruled by the dynamic contexts and environments, as is nowadays the Internet. World Wide Web introduces a new model of communication that differs from traditional media, since information is distributed in a variety of ways that enhances the proliferation of human networks [67], regardless of their social, educational, economic or political orientation.

2.3 User Profiling in Personalization Systems

User Profiling is considered the most vital component of Web Personalization and Adaptation Systems and therefore we dedicate the current section in presenting background theory regarding techniques and methodologies of User Profiling Construction.

One of the key technical issues in developing personalization applications is the problem of how to construct accurate and comprehensive profiles of individual users. How these can be used to identify a user and describe the user behaviour, especially if they are moving [68]. Term profile means according to Merriam-Webster dictionary “a representation of something in outline”. The user profile is a source of user requirements. The user profile specifies information regarding the general population (types) of individuals who will make daily use of the system functions. It can be thought of being “a set of data representing the significant features of the user”. A user profile consists of a set of keywords that describe the user preferred interest areas compared against information items. Its main objective is the creation of an information base that contains the preferences, characteristics and activities of the user.

User profiling is coming more and more important in the future with the introduction of the heterogeneous devices used to access all kinds of information and services. It is important for both the user and service provider to have more customized ways of accessing services. Within ubiquitous computing user profiling and their management is a valid research subject, when looking at use of profiles in services that really can have value for user profiles. That leaves out applications like intelligent saunas or toasters, which are often associated in ubiquitous environments. Clearly all kinds of devices, especially mobile device users and service providers can have benefits from profiling. Popular research subjects have been user profile replication in mobile services.

User profiling can either be static, when it contains information that rarely or never changes (eg. demographic information), or dynamic, when the data change frequently. Such information is obtained either *explicitly*, using online registration forms and questionnaires resulting in static user profiles, or *implicitly*, by recording the navigational behaviour and / or the preferences of each user. In the case of implicit acquisition of user data, each user can either be regarded as a member of group and take up an aggregate user profile or be addressed individually and take up an individual user profile. The data used for constructing a user profile could be distinguished into: (a) the Data Model which could be classified into the demographic model (which describes who the user is), and the transactional model (which describes what the user does); and (b) the Profile Model which could be further classified into the factual profile (containing specific facts about the user derived from transactional data, including the demographic data, such as "the favourite football team of user X is Football team A"), and the behavioural profile (modelling the behaviour of the user using conjunctive rules, such as association or classification rules. The use of rules in profiles provides an intuitive, declarative and modular way to describe user behaviour [68]).

Still, could current user profiling techniques be considered complete incorporating only these dimensions? Do designers and developers of Web-based applications take into consideration the real users' preferences in order to provide them a really personalized Web-based content? Many times this is not the case. How can a user profiling be considered complete, and the preferences derived optimized, if it does not contain parameters related to the user perceptual preference characteristics? We could define *User Perceptual Preference Characteristics* as all the critical factors that influence the visual, mental and emotional processes liable of manipulating the newly information received and building upon prior knowledge, that is different for each user or user group.

These characteristics determine the visual attention, cognitive and emotional processing taking place throughout the whole process of accepting an object of perception (stimulus) until the comprehensive response to it [74]. In further support of the aforementioned concepts, one cannot disregard the fact that, besides the parameters that constitute the "traditional" user profile (composed of parameters like knowledge, goals, background, experience, preferences, activities, demographic information, socio-economic characteristics, device-channel characteristics etc.), each user carries his / her own perceptual and cognitive characteristics that have a significant effect on how information is perceived and processed. Information is encoded in the human brain by triggering electrical connections between neurons, and it is known that the number of synapses that any person activates each time is unique and dependant on many factors, including physiological differences [69]. Since early work on the psychological field has shown that research on actual intelligence and learning ability is hampered by too many limitations, there have been a "number of efforts to identify several styles or abilities and dimensions of cognitive and perceptual processing" [70], which have resulted in what is known as learning and cognitive styles. *Learning and cognitive styles* can be defined as relatively stable strategies, preferences and attitudes that determine an individual's typical modes of perceiving, remembering and solving problems, as well as the consistent ways in which an individual memorizes and retrieves information [71]. Each learning and cognitive style typology defines patterns of common characteristics and implications in order to overcome difficulties that usually occur throughout the procedure of information processing. Therefore, in any Web-based informational environment, the significance of the fore mentioned users' differences, both physiological and preferential, is distinct and should be taken into consideration when designing such adaptive environments.

It is true that nowadays, there are not researches that move towards the consideration of user profiling incorporating optimized parameters taken from the research areas of visual attention processing and cognitive psychology in combination. Some serious attempts have been made though on approaching eLearning systems providing adapted content to the students but most of them are lying to the analysis and design of methodologies that consider only the particular dimension of cognitive learning styles, including Field Independence vs. Field Dependence, Holistic-Analytic, Sensory Preference, Hemispheric Preferences, and Kolb's Learning Style Model [72], applied to identified mental models, such as concept maps, semantic networks, frames, and schemata [73]. In order to deal with the diversified students' preferences such systems are matching the instructional materials and teaching styles with the cognitive styles and consequently they are satisfying the whole spectrum of the students' cognitive learning styles by offering a personalized Web-based educational content.

Chapter 3

Existing work in Dynamic Reconstruction of Web content

3.1 The Web today - A High-level Analysis of Major Web Services Sites based on a given Cognitive Framework

A mass production technique is to devise a cognitive framework that is, could assist providers to develop Web-sites that will embrace intrinsic values of customers, tailoring their On-line services accordingly.

Our previous findings [11, 12] show that cognitive factors have an important role in user satisfaction and identification of the products that are interested in. However, the way cognitive factors used today in order to design and develop Web services is considered unwisengly and principally based on provider's perception, without following particular rules that could achieve the appropriate mapping with selected content meta-characteristics; thus reconstructing any content to the benefit of the users.

In further support of the aforementioned concepts, one cannot disregard the fact that, besides the parameters that constitute the "traditional" user profile (composed of parameters like knowledge, goals, background, experience, preferences, activities, demographic information, socio-economic characteristics, device-channel characteristics etc., [75], each user carries his own perceptual and cognitive characteristics that have a significant effect on how information is perceived and processed. Information is encoded in the human brain by triggering electrical connections between neurons, and it is known that the number of synapses that any person activates each time is unique and dependant on many factors, including physiological differences [76]. Since early work on the psychological field has shown that research on actual intelligence and learning ability is hampered by too many limitations, there have been a “number of efforts to identify several styles or abilities and dimensions of cognitive and perceptual processing” [77], which have resulted in what is known as learning and cognitive styles. *Learning and cognitive styles* can be defined as relatively stable strategies, preferences and attitudes that determine an individual’s typical modes of perceiving, remembering and solving problems, as well as the consistent ways in which an individual memorizes and retrieves information [78]. Each learning and cognitive style typology defines patterns of common characteristics and implications in order to overcome difficulties that usually occur throughout the procedure of information processing. Therefore, in any Web-based informational environment, the significance of the aforementioned users’ differences, both physiological and preferential, is distinct and should be taken into consideration when designing such adaptive environments.

It is true that nowadays, there are not main researches, to our knowledge, that move towards the consideration of user profiles that incorporate optimized parameters taken from the research areas of visual attention processing and cognitive psychology in combination and used effectively in generic hypermedia structures and On-line services. Some serious attempts have been made though on approaching eLearning systems providing adapted content to the students but most of them are lying to the analysis and design of methodologies that consider only the particular dimension of cognitive learning styles, including Field Independence vs. Field Dependence, Holistic-Analytic, Sensory Preference, Hemispheric Preferences, and Kolb’s Learning Style Model [79], applied to identified mental models, such as concept maps, semantic networks, frames, and schemata [80, 81]. In order to deal with the diversified students’ preferences such systems are matching the instructional materials and teaching styles with the cognitive styles and consequently they are satisfying the whole spectrum of the students’ cognitive learning styles by offering a personalized Web-based educational content.

In our research, a selection of the most appropriate and technologically feasible learning styles (those that can be projected on the processes of selection and presentation of Web-content and the tailoring of navigational tools) has been studied, such as Riding’s Cognitive

Style Analysis (Verbal-Imager and Wholistic-Analytical) [82], Felder / Silverman Index of Learning Styles (4 scales: Active vs Reflective, Sensing vs Intuitive, visual vs. Verbal and Global vs. Sequential) [83], Witkin's Field-Dependent and Field-Independent [84], and Kolb's Learning Styles (Converger, Diverger, Accommodator, and Assimilator) [85] in order to identify how users transforms information into knowledge (constructing new cognitive frames).

We consider that Riding's CSA (as well as in many cases Felder / Silverman's ILS) implications can be mapped on the information space more precisely, since they are consisted of distinct scales that respond to different aspects of the Web-space (see Figure 1). Learning style theories that define specific types of learners, as Kolb's Experiential Learning Theory, have far more complex implications, since they relate strongly with personality theories, and therefore cannot be adequately quantified and correlated easily with Web objects and structures.

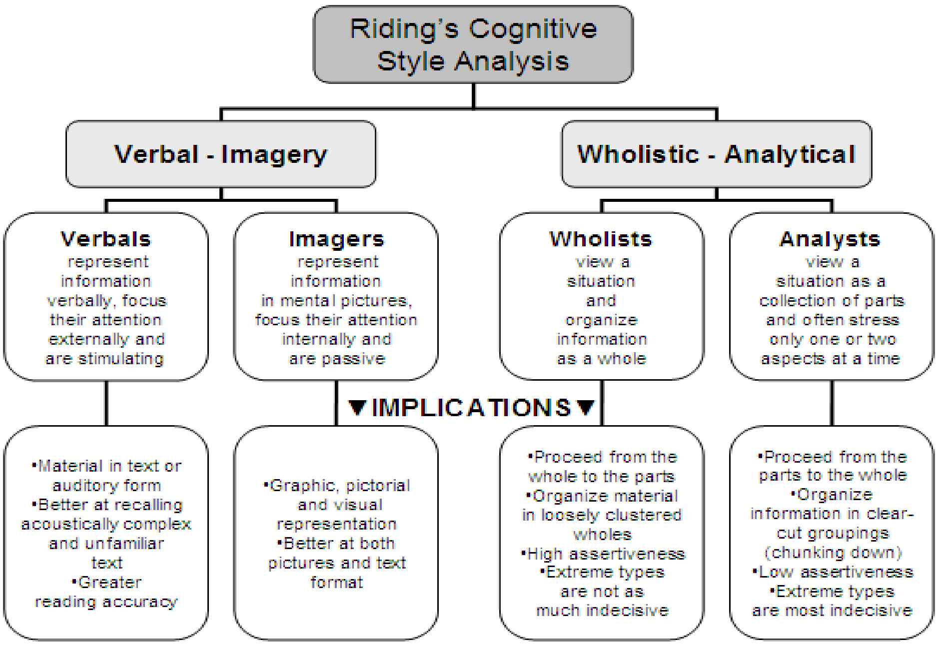

Figure 1. Riding's Learning Styles Characteristics and Implications

According to the theory behind CSA, individuals that are placed towards the edges of each axis have a strong preference for a specific method of information structure (Analyst / Wholist) or presentation (Imager / Verbalizer) - Table 1.

Table 1. Preferences of individuals according to cognitive style

Cognitive Style	**Preference**
Analyst	Internal (self-)guidance, non-linearity, index of interconnected concepts, view of situations in parts
Wholist	External guidance, linearity, defined framework, view of situations as a whole
Intermediate	No specific preference
Imager	Images, diagrams, schemes, better comprehension through visual representations.
Verbal	Predominance of text, better comprehension through verbal representations.

Consequently, when an individual is required to process information in the Web, it is most likely that the matching of his/hers preference to the structure and method of presentation of the Website would lead to better understanding, efficiency and satisfaction.

The first step to ground the need of personalization would be a preliminary inspection of the direction that major Web services sites are oriented to, with regards to cognitive style. For that reason, we selected five very deeply elaborated Web-sites of major commercial companies in the field of computers: www.dell.com, www.ibm.com, www.sony.com, www.apple.com, and www.hp.com. Due to the extreme breadth and depth of these sites, our analysis was limited to information related to the characteristics of computers that these companies offer, since this kind of information is factual and visitors are expected to understand and retain these data for further processing that could lead to commercial decisions.

In general, quite a few common patterns were observed: firstly, it is evident that in all five cases the lack of sequential organization and the extreme segmentation of the content require that the users should adopt an analytic path. External guidance is missing, and a general framework that would benefit Wholists is absent. Important information is available only through additional clicking and navigating.

Still, it is of high interest that when users successfully navigate to a specific product, the presentation is rather sequential, since information is provided without interconnections and links to concepts that would allow Analysts to form a deeper understanding; Wholists on the other hand would find this simplicity more to their liking.

It could as well be supported that this is not an intermediate approach, with all aspects of information processing being equally taken into account, but a mixed-mode that at instances may serve users' preferences in a random way. Of course, this is expected since Web-sites are not built around this kind of individual differences.

As it concerns the Imager / Verbalizer dimension, while all sites are aesthetically very pleasing with the inclusion of photos and banners, all significant information is mostly conveyed through text. The idea of schematically presenting important details is not actualized in any case; however, the Sony and Apple sites accompany many texts with relevant images that provide a somehow visual description of the information, as long as users are a little bit experienced with computers.

To this end, it could be supported that the 3 out of 5 sites are heavily suitable for Verbalizers; the remaining two adopt a rather intermediate approach which can be considered as balanced, even if this happens for aesthetical reasons.

At this point, the construct of working memory should be discussed. Working Memory (WM) has gained some popularity in terms of examining the interaction of WM span with different hypertext levels of complexity. DeStefano and LeFevre [17] reviewed 38 studies that address mainly the issue of cognitive load in hypertext reading, and working memory is often considered as an individual factor of significant importance, even at the level of explaining differences in performance. Lee and Tedder [86] examine the role of working memory in different computer texts, and their results show that low working memory span learners do not perform equally well in hypertext environments. The term Cognitive Load Theory is often used when referring to guidelines for designing hypermedia applications, and it is often correlated with working memory span [20].

In all five sites, the amount of links and information is rather exhaustive. Especially at the first levels of the navigational structure, there are so many links to information resources that could burden users with low WM span. The lack of a coherent pattern or even better an adaptive mechanism that would adjust the availability of information to users' capabilities could as well reduce the efficiency of navigation through the site.

The most demanding task is to keep a track of the paths that lead to different resources in order to avoid disorientation; it seems that, according to the abovementioned studies, this task requires a satisfactory level of WM span. The way we approach methodologically this issue is discussed in a next section of this thesis.

In our opinion and in relation to our work in the field of adaptive educational hypermedia [18], the sites that were inspected, though at a preliminary level, are not exactly biased towards specific preferences, neither well balanced. At each instance, a mode of information presentation predominates, but this is not stable; it may as well change, for example when an actual product is shown. Perhaps Analysts/Verbalizers would find these Web-sites more comprehensible than Wholists / Imagers, but not at all times.

Consequently, our research interest is whether we could dynamically alter a section of a Web-site (the computer section in this case) by personalizing the content and the structure to specific users' cognitive preferences. This could be achieved by enriching the existing Web structures with further design enhancements and specific content transformations based on the adaptation mapping rules derived from selected cognitive factors. In the event that this would be proven successful and meaningful, individuals would learn better the information that is important to them.

Therefore, based on our previous research, the way cognitive styles could be used effectively within the context of any content reconstruction is to identify the way we will reconstruct the content. The adaptation process involves the transformation and/or enhancement of a given raw Web-based content (provider's original content) based on the impact the specific human factors have on the information space [11, 12] (i.e., show a more diagrammatical representation of the content in case of an Imager user, as well as provide the user with extra navigation support tools).

Today's most popular Web-sites (http://www.alexa.com/site/ds/top_500) like Google, Microsoft Live, Yahoo, Amazon, eBay, BBC news etc. do not heavily use the abovementioned cognitive considerations but they rather mostly employ customization techniques where the user has direct control; the user explicitly selects between certain options. On the other hand, personalization is driven by the system which tries to serve up individualized pages to the user according his profile and needs. Although, personalization is used by many of these popular Web-sites (especially Google), the techniques they maintain are lying under the predetermined customization of services or products and not to the actual personalization and dynamic reconstruction of content based on user preferences.

3.2 Personalization and Mass Customization techniques used in today's most popular Web-sites

Indicatively, two live cases under this category are Google and Amazon personalization methods.

3.2.1 Google Personalization Methods

Google Inc. uses several methods and techniques that look at personalization, and provide a system for collecting information from a searcher that may make it easier for the search engine to deliver search results to them that more closely match what they may be looking for than from a non-personalized search. Some of them are:

- Systems and methods for analyzing a user's Web history
- Systems and methods for modifying search results based on a user's history
- Systems and methods for providing a graphical display of search activity

- Systems and methods for managing multiple user accounts
- Systems and methods for combining sets of favourites
- Systems and methods for providing subscription-based personalization

Profile building is one of the most popular techniques Google uses for providing personalization. A lot of information is collected in this process, including clicks on search results pages, which pages are viewed, how long someone stays on different pages, how long ago these activities happened, and more. Different algorithms might be used to identify other types of data, including pages that are similar to ones that users have interacted with.

3.2.2 Amazon Personalization Methods

Amazon.com has a much-vaunted personalization element that gives each customer individualized recommendations of books. Even though this feature is far from perfect, it usually succeeds in including some relevant titles.

The book recommendations succeed for two reasons: (a) Users do not need to do anything to set it up, and (b) the system learns their preferences by recording what books they buy.

By watching millions of buyers, the system learns which books are similar. If many people who buy some user's books also buy i.e. Don Norman's books, then it is a good idea to recommend Norman's new book to somebody who has bought the user's books in the past, even if they have never bought any of his books.

We have to note at this point that both steps happen without imposing any extra work on the users**.** Also, the fact that somebody buys a book is a pretty strong signal that they have an interest in the book (much more reliable data than most preference settings one can collect from users).

Amazon also uses the similarity data to include hypertext links between related books. Thus, when users are browsing the page for one book, they see links to three other books they are likely to want. This use of the data is much better than the personal recommendation list because the hypertext links are embedded in the context of the users' natural behaviour. When the users go to a book page, they will be shown recommendations that match their specific interest in that moment (as opposed to being derived from a generic model of the users' average interests).

3.3 Current Web-based Authoring Tools

Nowadays, most semantic Web authoring tools (neither HTML editors, nor CMS), provide the Web developer with techniques and easy-to-use tools to create and generate descriptive ontologies of eServices' content. These authoring tools, as well as any other kind of Web editing tools (CMS, HTML Editors etc.) do not take into consideration adaptation and personalization techniques. Ideally, a combination of a Web authoring process of Web-based content and the adaptation of this content based on a given framework would give a more comprehensive approach to the personalization of content production.

To our knowledge, there has not been a Web Development Editor that takes into consideration the above issues for mass customizing and personalizing Web products and services. A comprehensive review of current Web Development Editors will be presented in this section.

Web-based authoring tools are becoming standard issue in modern content management systems. They range from simple text editors to high powered graphical authoring tools and content management systems. This section contains a listing of some noteworthy research oriented and commercial Web authoring tools. Many of the editors listed below are "What You See Is What You Get" (WYSIWYG) HTML editors, some of which have the option to view the HTML source code. These are quite popular due to the low learning curve, yet it is important to get some understanding of HTML since WYSIWYG HTML editors can be limiting.

3.3.1 Research Oriented Web Authoring Tools

A selection of the most predominant non-commercial Web-based authoring tools is described below:

3.3.1.1 Protégé

Protégé [87] (http://protege.stanford.edu/) is a free, open-source platform that provides a growing user community with a suite of tools to construct domain models and knowledge-based applications with ontologies. At its core, Protégé implements a rich set of knowledge-modelling structures and actions that support the creation, visualization, and manipulation of ontologies in various representation formats. Protégé can be customized to provide domain-friendly support for creating knowledge models and entering data. Further, Protégé can be extended by way of a plug-in architecture and a Java-based Application Programming Interface (API) for building knowledge-based tools and applications.

An ontology describes the concepts and relationships that are important in a particular domain, providing a vocabulary for that domain as well as a computerized specification of the meaning of terms used in the vocabulary. Ontologies range from taxonomies and classifications, database schemas, to fully axiomatized theories. In recent years, ontologies have been adopted in many business and scientific communities as a way to share, reuse and process domain knowledge. Ontologies are now central to many applications such as scientific knowledge portals, information management and integration systems, electronic commerce, and semantic Web services.

3.3.1.2 Swoop: A Web Ontology Editing Browser

SWOOP ((http://code.google.com/p/swoop/) [88, 89]) is a tool for creating, editing, and debugging OWL ontologies. It was produced by the MIND lab at University of Maryland, College Park, but is now an open source project with contributors from all over.

Swoop is built primarily as a Web Ontology Browser and Editor, i.e., it is tailored specifically for OWL ontologies. Thus, it takes the standard Web browser as the UI paradigm, believing that URIs are central to the understanding and construction of OWL Ontologies. The familiar look and feel of a browser emphasized by the address bar and history buttons, navigation side bar, bookmarks, hyper-textual navigation etc are all supported for Web ontologies, corresponding with the mental model people have of URI-based Web tools based on their current Web browsers.

3.3.1.3 OntoStudio

OntoStudio (http://semanticWeb.org/wiki/OntoStudio) is an engineering environment for ontologies and for the development of semantic applications, with particular emphasis on rule-based modelling. It is the successor of OntoEdit which was distributed worldwide more than 5000 times. OntoStudio was originally developed for F-Logic but now also includes some support for OWL, RDF, and OXML. It also includes functions such as the OntoStudio Evaluator. The Evaluator is used for the implementation of rules during modelling; this procedure has been recently patented.

3.3.2 Commercial Web Authoring Tools

A selection of the most predominant commercial Web-based authoring tools is described below:

3.3.2.1 EditOnPro by Realobjects

RealObjects edit-on Pro (http://www.realobjects.com/) is a cross-platform WYSIWYG XHTML / XML editor as a Java applet, allowing individuals and teams to update, create, and publish Web content within Content Management, Knowledge Management, e-Learning or other Web-based systems.

The editor has an easy-to-use, intuitive user interface which provides word processor-like and XML editor-like features to Web based applications, empowering non-technical users to become content contributors without knowing HTML, XML or other cryptic mark-up languages.

It guarantees XHTML compliance of the contents created or pasted from other applications by validation. Thus corporate site standards for style, layout and code can uncompromisingly be enforced. The valid XHTML output assures portability, compatibility and interoperability. For example, content can easily be parsed and automatically be transformed using XSLT.

3.3.2.2 Cute Editor by Cute Soft

Cute Editor (http://cutesoft.net/) for ASP.NET is a WYSIWYG browser-based Online HTML Editor for ASP.NET. It is also available for PHP and ASP.

It enables ASP.NET Web developers to replace the text area in the existing content management system with a powerful, but easy to use WYSIWYG HTML editing component.

It empowers business users to make content updates easily and safely themselves while maintaining control over site design and content, all at an affordable price.

3.3.2.3 TinyMCE - Javascript WYSIWYG Editor

TinyMCE (http://tinymce.moxiecode.com/) is a platform independent Web based Javascript HTML WYSIWYG editor control released as Open Source under LGPL by

Moxiecode Systems AB. It has the ability to convert HTML text area fields or other HTML elements to editor instances. TinyMCE is very easy to integrate into other Content Management Systems.

3.3.2.4 JXHTMLEdit by Tecnick

JXHTMLEDIT (http://www.tecnick.com/) is a free Open Source browser-based HTML/XHTML content authoring tool based on the Java 2 Platform that allows WYSIWYG editing on multiple platforms (require the Sun JavaTM Plug-in 1.4 or higher installed on client).

It is a cross-platform WYSIWYG HTML/XHTML content authoring tool, a very small Java Applet based on the Java 2 Platform. JXHTMLEdit provides word processor-like user interface that allows users to edit the XHTML document directly in the final form (as will be rendered). This empowers non-technical users to become content contributors without any knowledge of HTML or XHTML.

JXHTMLEdit has been designed to offer great flexibility and could be used to easily integrate WYSIWYG authoring functionality into existing Websites, CMS, WMS or any other Web-based software. The Applet JAR archive is less than 150 KB and it's cacheable, so it loads very quickly.

Furthermore, we will give a comprehensive review of current well established Personalization and Adaptation Systems, describing the techniques and ideas for adapting and personalizing web content. Describing all of them though in detail will be beyond of the scope of this thesis.

3.4 Web Personalization Systems

3.4.1 Commercial Systems

3.4.1.1 BroadVision One-To-One

BroadVision's [59] emphasis is on what the company calls Enterprise Relationship Management (ERM). That is, it provides products that help to optimise the relationship between the organisation and its employees, business partners and customers. Specifically, it attempts to optimise the conditions under which applications such as these might work. BroadVision provides personalisation through One-To-One Command Centre. Command Centre supports 15 or more different forms of personalisation. There is no need to discuss each of these in detail but the most important capabilities include:

Rules - this is the facility to define business rules (such as, "if they do X, then show them Y") that apply to personalisation. It can be used to define who sees what information, where and when and is commonly used in a variety of situations, such as:

Cross-selling and up-selling - To do this, you need to know what the user has done so far in this session and what they have done when they have previously logged on to this site. The former is held in an in-memory cache throughout any single session, so that appropriate rules can be applied even where this is a new customer. For previous sessions BroadVision provides its One-To-One observation system, which records all pertinent data for use in conjunction with other personalisation techniques (such as feedback and learning – see below) in order to make the best use of selling opportunities.

Profile and community-based targeting - BroadVision provides profiling capabilities that allow you to record attributes against customers from a demographic or psychographic basis for segmentation and other purposes. These attributes may be multi-valued and the system is extensible so that you can define your own attributes.

Context-based matching - This is used to construct dynamic pages where the content is related in some way.

Matching agents - this is where content is classified by the authors of that content, users specify the sort of content they are looking for, and the system puts the two together. For example, a particular investor's profile would be best suited by a particular portfolio of investment products. Putting the two together is the function of this method.

Feedback and learning - this is based on "do what I do, not what I say". That is, I may request certain information but the system may observe that the 80/20 rule applies. Of the information that I have requested, I spend 80% of my time looking at 20% of it. So perhaps my personalisation rules should be changed to reflect my actual behaviour rather than what I thought my behaviour was. The other aspect of this approach is that data mining and other analytical techniques can be applied to the information that is collected from the web site. BroadVision has partnership with BroadBase and NCR for click stream analysis and this can be provided by as a part of One-To-One Enterprise to provide analytical capabilities. Alternatively, you can analyse this data via a third-party product.

Searching - BroadVision has the Verity search engine built-in to it, and it supports both attribute search and full-text search. While the latter is well known the former is particularly useful when searching from particular products, services or content where you want to narrow down your field of search as much as possible. The only proviso is that in e-Procurement environments the range of attributes required may be extensive and a specialist solution based on a product database might be a better option.

Community rating and collaborative filtering - these are the sorts of facilities that Amazon has made popular: the ability to see what other people thought of what you are thinking of buying, and the facility to see what other people bought who also liked what you bought last time.

3.4.1.2 Microsoft's Firefly Passport (developed by the MIT Media Lab)

Microsoft Passport [60, 61] is a web-based authentication system which supports Single Sign On (SSO). Give Microsoft a user name and password, and you have a Passport. When you visit any Passport-aware Internet sites, you type in your same Passport name and password. You no longer have to remember different user names and passwords for every place that you visit or shop on the Internet. In fact, when you move from one Passport-enabled site to another, you don't even need to log on again. With the Passport, your personal data travels with you: name, address, and, if you purchased anything, your credit card number. Microsoft says its .NET Passport enables software, Internet services, and computer gadgetry to work together and share information, making the Internet easier for everyone to use.

The Passport service can be viewed as a web-based authentication system. It consists of three entities, Passport server, online Merchants, and customer clients.

The Passport server is hosted and supported by Microsoft, it is the single and central location where all the customer accounts information are stored and processed for authentication purposes. When the customer tries to log-on to a Passport-participating online merchant's website, the user authentication is transferred to Microsoft Passport server via browser redirection and secure cookie setting is stored at the customer's web browser as the authentication result.

The secure cookie stored in the customer's web browser is used as a proof for legitimate usage at the online merchant's web site. If the customer goes to another Passport participating web site, as long as the secure cookie is stored in the browser, the customer doesn't need to type in the username and password again. Instead, the secure cookie will be sent to the Microsoft Passport server for authentication purpose.

So, for the online merchants, they only need to trust the Microsoft Passport and let it handle the authentication process. After the authentication, the customers will be redirected back to the customers and they can read the customer account information from the cookies that have been set by the Passport server.

3.4.1.3 Macromedia LikeMinds Preference Server

Macromedia LikeMinds 5.0 [62] is a collaborative filtering personalization product that analyzes website visitors buying patterns in real time and makes recommendations based on click-through patterns, purchase history, and preference matching. LikeMinds 5.0 uses a technology called collaborative filtering to retain site visitor interest, Lynch said. With the goal of increasing average order size and customer retention, LikeMinds aims to engage the customer who is actively involved in the site and make recommendations. It matches current users with mentors who have expressed similar online behaviours to real-time recommendations.

3.4.1.4 IBM WebSphere

WebSphere [63] allows you to personalize the content of a Web site, intranet or extranet so that it matches the unique needs and interests of each visitor.

IBM WebSphere Personalization features are:

- **Personalization Workspace**: Define, control and preview a Web site's personalization behaviour using this browser-based graphical tool.
- **Campaign management**: Create and control audience-specific site content and e-mails for time-delimited campaigns.
- **Implicit profiling**: Update and personalize visitor profiles in real-time based on visitor activity.
- **Rule functions**: Exclude and limit content; personalize by browser; use sub attributes of data resources in your rules.

3.4.2 Research Oriented Systems

3.4.2.1 ARCHIMIDES

ARCHIMIDES [50] is an intelligent agent that aims to provide intelligent, adaptive and personalized navigation within a WEB server. Provided a subset of the set of keywords that characterize the server's contents, ARCHIMIDES undertakes the task to perform an intelligent information retrieval and afterwards to construct a personalized version of the server in the form of an index to pages that present some interest to the user. This index does not resemble what search engines return as a result of some query; it could be probably regarded as a much sorter version of the WEB server, with links that are dynamically inserted or deleted according to the user's interests, preferences and behaviour, providing ARCHIMIDES with the feature of adaptivity. As a result the user navigates in a WEB server that may completely present interest to him or her, thus relieving the user from undesired information overload.

3.4.2.2 Proteus

Proteus [51] is a system that constructs user models using artificial intelligence techniques and adapts the content of a Web site taking into consideration also wireless connections. The Proteus web site personalizer performs a search through the space of possible web sites. The initial state is the original web site of unadapted pages. The state is transformed by any of a number of adaptation functions, which can create pages, remove pages, add links between pages, etc. The value of the current state (*i.e.*, the value of the web site) is measured as the expected utility of the site for the current visitor. The search continues either until no better state can be found, or until computational resources (*e.g.*, time) expire.

3.4.2.3 WBI

Web Browser Intelligence (WBI, pronounced "WEB-ee") [52, 53] is an implemented system that provides a loosely confederated group of agents on a user's workstation capable of observing user actions, proactively offering assistance, modifying resulting web documents, and performing new functions. For example, WBI will annotate hyperlinks with net- work speed information, record pages viewed for later access, and provide shortcut links for common paths. WBI is an architecture in which small programs, or agents, connect to the information stream by registering their trigger conditions and then performing operations on the stream. This structure provides rich opportunities for personalizing the web experience by joining together personal and global information, as well as enabling collaboration among web users.

3.4.2.4 BASAR

BASAR [54] (Building Agents Supporting Adaptive Retrieval) provides users with assistance when managing their personal information spaces. This assistance is user-specific and done by software agents called web assistants and active views. Users delegate tasks to web assistants that perform actions on their views of the WWW, on the WWW itself, and on the history of all user actions.

3.4.2.5 mPERSONA

mPERSONA [39] is a flexible personalization system for the wireless user that takes into consideration user mobility, the local environment and the user and device profile. The system utilizes the various characteristics of mobile agents to support flexibility, scalability, modularity and user mobility.

3.4.2.6 INSPIRE

INSPIRE [55] is an Adaptive Educational Hypermedia system, which emphasizes the fact that learners perceive and process information in very different ways, and integrates ideas from theories of instructional design and learning styles. Its aim is to make a shift towards a more "learning-focused" paradigm of instruction by providing a sequence of authentic and meaningful tasks that matches learners' preferred way of studying. INSPIRE, throughout its interaction with the learner, dynamically generates learner-tailored lessons that gradually lead to the accomplishment of learner's learning goals. It supports several levels of adaptation: from full system-control to full learner-control, and offers learners the option to decide on the level of adaptation of the system by intervening in different stages of the lesson generation process and formulating the lesson contents and presentation. Both the adaptive and adaptable behaviour of INSPIRE are guided by the learner model which provides information about the learner, such as knowledge level on the domain concepts and learning style. The learner model is exploited in multiple ways: curriculum sequencing, adaptive navigation support, adaptive presentation, and supports system's adaptable behaviour.

3.4.2.7 ELM-ART II

ELM-ART II [56] is an intelligent interactive textbook to support learning programming in LISP. ELM-ART II demonstrates how interactivity and adaptivity can be implemented in WWW-based tutoring systems. The knowledge-based component of the system uses a combination of an overlay model and an episodic user model. It also supports adaptive navigation as individualized diagnosis and help on problem solving tasks. Adaptive navigation support is achieved by annotating links. Additionally, the system selects the next best step in the curriculum on demand. Results of an empirical study show different effects of these techniques on different types of users during the first lessons of the programming course.

3.4.2.8 AHA!

AHA [57] is an open Adaptive Hypermedia Architecture that is suitable for many different applications. This system maintains the user model and filters content pages and link structures accordingly. The engine offers adaptive content through conditional inclusion of fragments. Its adaptive linking can be configured to be either *link annotation* or *link hiding*. Even *link disabling* can be achieved through a combination of content and link adaptation.

3.4.2.9 InterBook

InterBook [58] is a tool for authoring and delivering adaptive electronic textbooks on the World Wide Web. InterBook provides a technology for developing electronic textbooks from a plain text to a specially annotated HTML. InterBook also provides an HTTP server for adaptive delivery of these electronic textbooks over WWW. For each registered user, an InterBook server maintains an individual model of user's knowledge and applies this model to provide adaptive guidance, adaptive navigation support, and adaptive help.

3.4.2.10 TANGOW

TANGOW [64] is a tool for developing Internet-based courses, accessible through any standard WWW browser. Courses are structured by means of Teaching Tasks and Rules which are stored in a database and are the basis of TANGOW guidance ability. In TANGOW a Student Process is launched for each student connected to the system. Each Student Process consists of two main modules: a Task Manager that guides the students in their learning process, and a Page Generator that generates the HTML pages presented to the student. The Student Process also maintains information about the actions performed by the student when interacting with the course in the Dynamic Workspace. This information is used by TANGOW to adapt the course contents to the student's learning progress. TANGOW has also information about student profiles, which is used to select, at run-time, the contents of each HTML page presented.

3.4.2.11 SQL-Tutor

SQL-Tutor [65] is a knowledge-based teaching system which supports students learning SQL. The intention was to provide an easy-to-use system that will adapt to the needs and learning abilities of individual students. The tailoring of instruction is done in two ways: by adapting the level of complexity of problems and by generating informative feedback messages.

SQL-Tutor is based on Constraint-Based Modeling (CBM), a student modeling approach proposed by Stellan Ohlsson. It is a very efficient approach which concentrates on the violations of the basic principles in the domain of instruction. Currently there are 406 constraints in the system, compiled into a structure resembling RETE networks, in order to speed up the matching process. The system also knows about various databases, problems and the ideal solutions to them, and uses this knowledge to diagnose students' answers. Currently the system only deals with the SELECT statement, but we believe that is not a serious limitation; the same approach can be used for other SQL statements, queries do cause most problems for students anyway, and many concepts covered by SELECT are directly applicable to other statements and other relational languages.

In order to adapt to individual students, SQL-Tutor maintains a model for each student. A student model contains information about the history of previous sessions (like a list of problems solved correctly etc) and also contains a model of the student's knowledge, expressed in terms of constraints.

When a student logs in for the first time, the system creates a new model, and lets the student select a database and a problem to work on. Every student solutions is analyzed (i.e., propagated to the constraint network), and the system reports on its findings. There are several levels of feedback messages:

- Positive / negative feedback simply informs the student whether his/her solution is correct or not;
- Error flag informs the student about the clause of the SELECT statement an error appeared in;
- Hint provides a general description of the error;
- Partial solution gives the correct version of the clause where an error appeared;
- Complete solutions present the ideal solution to the problem.

3.4.2.12 SKILL

SKILL [66] is a scalable Internet-based teaching and learning system. The primary objective of SKILL is to cope with the different knowledge levels and learning preferences of the students, providing them with a collaborative and adaptive learning environment utilizing new World Wide Web technologies. Basic components of SKILL are course material based on concepts organized in an ordinal rating derived from pre-requirements, an annotation facility suited for collaboration work, and a configuration environment for tailoring the system. Topics discussed include: (1) SKILL functionality, including adaptivity/progress control and collaborativity through annotations and course extensions; (2) components, including security, document management, and tutoring components; (3) implementation issues; and (4) related work.

Chapter 4

A proposed framework for the dynamic reconstruction of Web content

The smarTag framework (Figure 2) is an extension of the AdaptiveWeb [11, 75] framework aiming to improve the creation process of adaptive Web-pages based on given user's characteristics (cognitive factors). A visitor that wants to get personalized information of a Web-site that has been enhanced under the smarTag framework needs to have installed the smarTag Firefox browser plug-in in order to get personalized and adapted content. The browser plug-in is responsible for the mapping process of the user's comprehensive profile and the smart Web objects created under the smarTag framework.

A user initially navigates through any Web-site available on the net. So far the user views the raw content of a Web-site without any personalization of content taking place. If the particular Web-site consists of smart web objects and the user has not installed the browser plug-in, a message will appear on the screen prompting him / her to install the browser plug-in to see personalized and adapted content. In case the user wants to get personalized content (after installing the plug-in) (s)he authenticates into his / her profile (initially created in the Automatic Profile Extraction Component) and the Web content is parsed by the browser before shown to the user and all the smart objects in the Web-page will be reconstructed based on the user's profile.

The smarTag framework is composed of a number of interrelated components[1], each one representing a stand-alone Web-based system. The idea behind the framework is to enhance

[1] The technology used to build each Web system's component is ASP .Net (http://asp.net).

any Web services page (technology and language independent) with adaptive Web objects that will adapt according to a given user's profile (user's cognitive characteristics).

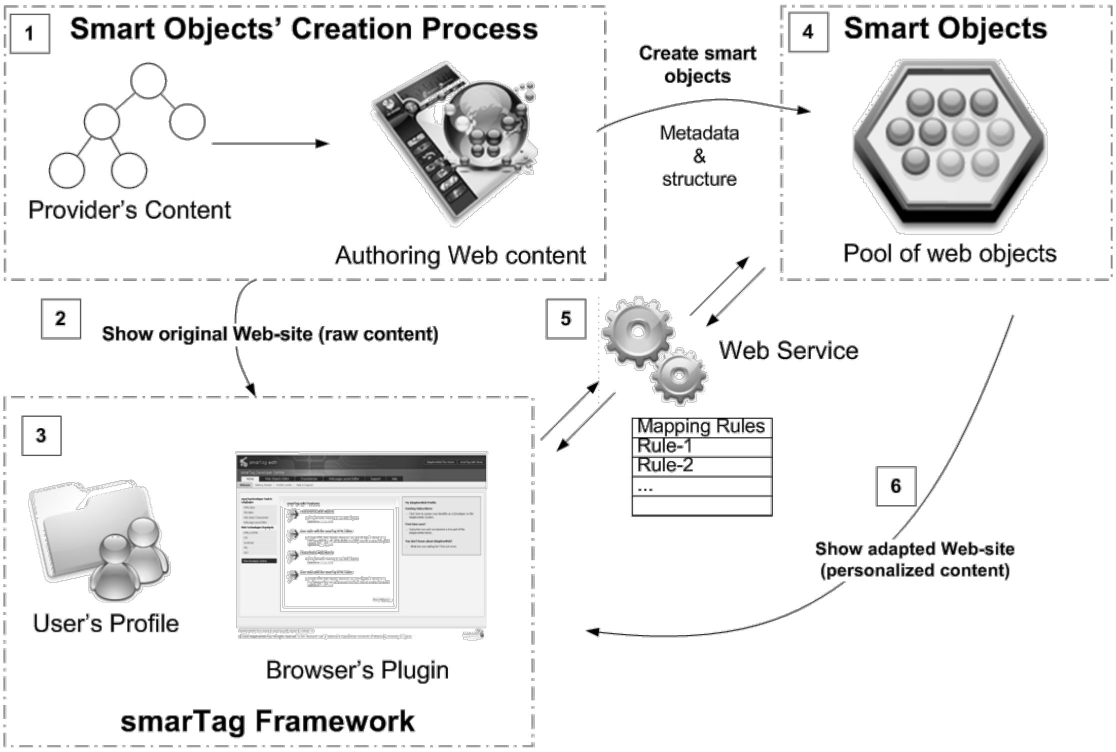

Figure 2. The smarTag Framework

A more detailed description and analysis of how the components in the smarTag framework interact and how the adaptation process works is presented in the following sections.

4.1 Authoring smart objects

A smart object under the smarTag Framework is conceptually similar to the traditional XML objects: they too consist of attributes and content. The content can either be in textual or diagrammatical form in case of a Verbalizer and Imager user respectively. The smarTag attributes are special meta-characteristics describing the possible behaviour the object can perform in its environment [11, 75]. All the objects are embedded in any Web-page which are used in the mapping process of a user's profile [74].

Since all the smart objects will be embedded as enhancements in an external Web-site, our main concern is to ensure openness and interoperability between the system's components and any external Web-site, as well as to ensure the Web security policies. In order to achieve this, the smart objects must be easily extendible and easy to handle. Using XML for implementing the smart objects' structure seems to be the best way to achieve this. Indeed XML[2] enables the extendibility we need and enhances interoperability and integration among systems' components. A more comprehensive description on this matter will take place in the following section.

4.1.1 Enhancing any Web-site with the smarTag framework

Our main concern was to create an easy to use framework that enables any Web developer / designer to enhance divisions of his / her Web-site with mass customization and personalization techniques. More specifically, the traditional methods of Web Development will take place in the process; based on the main requirements of the end-users of the Web-site and mainly on the "design taste" of the Web Designer / Developer.

Figure 3 depicts the Traditional Web-site development process. Initially, based on the Web-site requirements and specifications, all the needed information (text, data, graphics and pictures) of the Web-site is collected. The Web-site's layout and Navigation is then designed by the Web Designer and all the collected information is implemented in the Web-site.

[2] http://www.w3.org/XML/

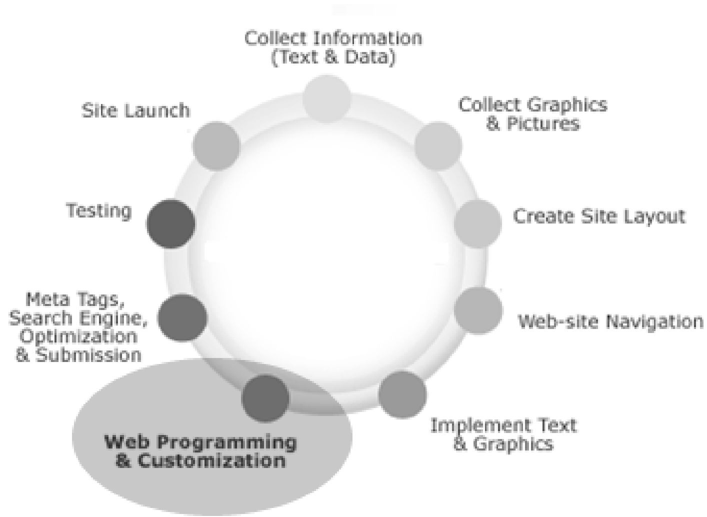

Figure 3. Traditional Web-site development process

So far, the Web-site's author will follow the traditional steps of the Web development process. The smarTag concept is used in the Web Programming and Customization phase. The Web Developer will define specific divisions in the Web-site that will adapt according to individual characteristics (cognitive styles).

4.2 Automatic Profile Extraction

This component (Figure 4) will be responsible for the user's automatic profile extraction. Its main objective is to replace it with the Profile Construction Component used in AdaptiveWeb [11, 12] and extract a comprehensive user profile in a more efficient way.

It is the initial step the user makes for the smarTag framework's personalization process. It is a vital part of the system. At this point the user creates his / her comprehensive profile, which is going to be mapped at a later stage with the personalized content.

The main idea behind this component is to let users navigate through a predefined Web-site for about 20 minutes and tracking their navigation pattern with the use of their click streams. While navigating the system will use intelligent algorithms for analyzing their navigation and then extract their profile.

The main parameters that will comprise the user's profile are the Imager / Verbalizer and Wholist / Analyst. More specifically, for the extraction of the Imager / Verbalizer parameter the system will track the navigation of the user to Web-pages containing more diagrams and images, thus the user is an Imager or to Web-page containing more textual content, thus the user is a Verbalizer. As for the Wholist / Analyst, weights are used in specific sections of the Web-site. The navigation pattern for each type of user is assumed to be different. The Wholist user has a more sequential navigation pattern, therefore the less the weight difference between the sections is, the more a Wholist user (s)he is. The Analyst user on the other hand has a diffused navigation pattern, therefore the more the weight difference between the sections is, the user tends to be an Analyst. There are also the cases where the user can be an Intermediate (parameter that is in between an Imager / Verbalizer and a Wholist / Analyst).

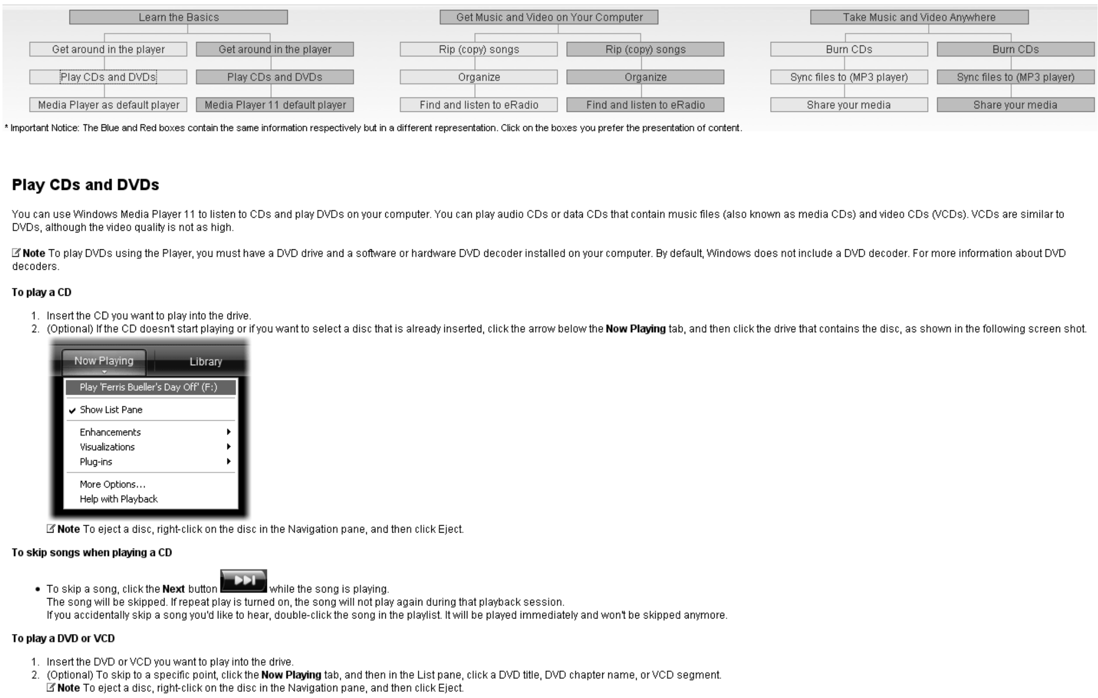

Figure 4. Automatic Profile Extraction Component

With the completion of the automatic profile extraction process, the system saves the results (user's parameters) in an XML file that will be used by the Firefox browser extension for mapping the profile with the personalized content.

4.3 Web content transformation

The proposed methodological approach for adapting Web content is related to the mapping process mechanism (that is, rules responsible for the content's transformation based on the correlation of the cognitive implications and the actual raw (provider's) data) and the imminent adaptation of any content based on the specific human factors, that is in the particular case the cognitive styles and working memory.

4.3.1 Web Browser Extension

A suggested precondition for the mapping process to work properly at this stage is to extend the well known html model with a new set of tags; <csl> (cognitive style list) and <csli> (cognitive style list item). A Web Browser (Mozilla Firefox) Extension (Figure 5) has been therefore developed in order for the browser to recognize and implement the set of tags. Fig. 6 shows a sample code that is extended with the new set of tags.

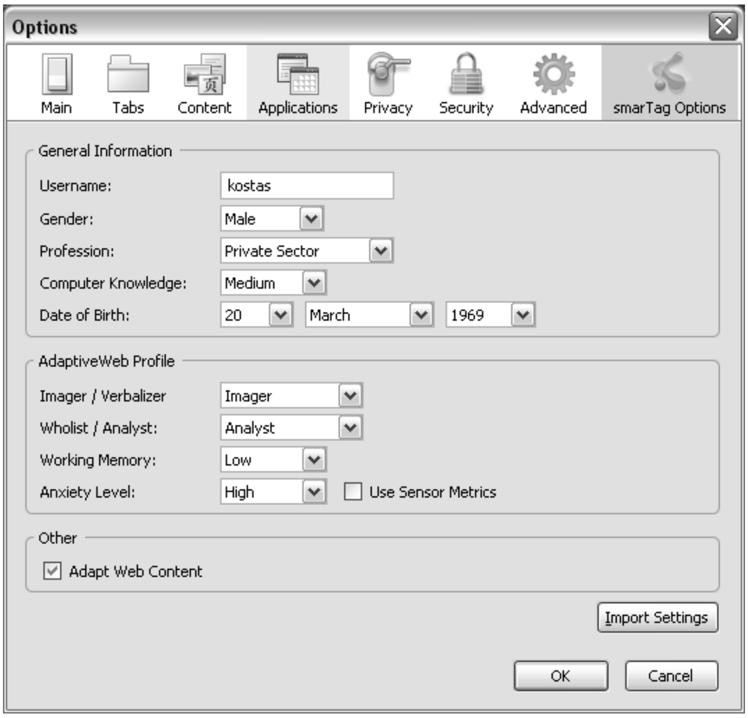

Figure 5. Firefox Browser Extension Options' Panel

This set of custom xml tags is interpreted by the Web browser extension for reconstructing a given Web content when mapped with a user's cognitive factors.

```
<csl>
      <csli name="General Information">
            <h2 style="...">Sony Vaio SZ791</h2>
            <p>Notebook</p>
            <p>Ultra Portable</p>
      </csli>
      <csli name="CPU Information">
            <p>Intel Core 2 Duo</p>
            <p style="...">2.5 GHz</p>
      </csli>
      <csli name="Memory Information">
            <p>4GB Ram Memory</p>
            <p>250 GB Hard Disk (7200 rpm)</p>
      </csli>
</csl>
```

Figure 6. Sample code extension with the new <csl> tag

Based on the sample code depicted in Figure 6, the corresponding browser's extension interpretation and content presentation is shown in Figure 7 and Figure 8.

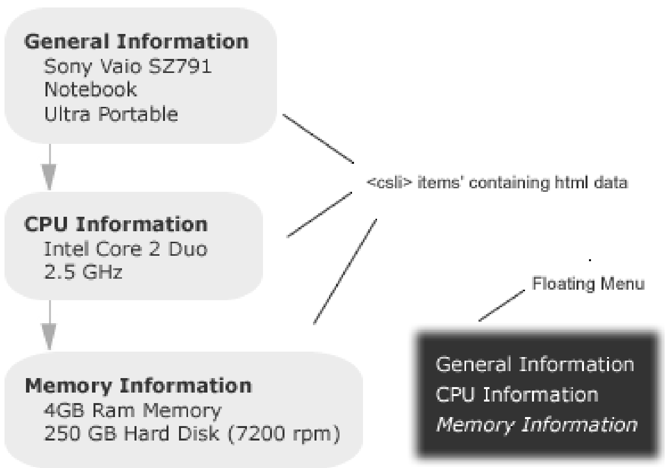

Figure 7. Browser content interpretation based on user's profile (Wholist / Imager)

In case a user is a Wholist / Imager (with regards to his cognitive style – Figure 7), the browser will enhance each <csli> item, along with its containing data, with a diagram box

(Imager) and will also create a floating menu (Wholist) that contains each <csli> item's name so to help the user navigate through the items by clicking on the corresponding link.

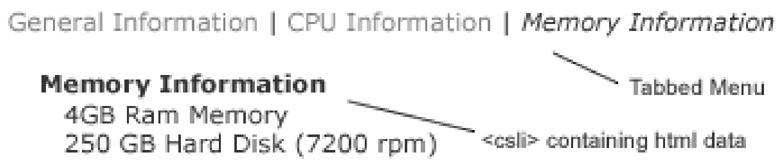

Figure 8. Browser content interpretation based on user's profile (Analyst / Verbalizer)

In Figure 8, the user happens to be an Analyst / Verbalizer. In this case, the browser shows the <csli> item's containing data in a textual form (Verbalizer) and will also enhance the Web-page with a tabbed menu for each <csli> item (Analyst). Each time the user clicks on a link of the tabbed menu, the corresponding data of the <csli> item is shown on the Web-page.

4.3.2 Mapping the <csl> tag with the User's Cognitive Characteristics

Our main goal in this section is to show in a more detail how the web browser extension interprets the <csl> tag and adapts the containing information based on the user's profile and consequently the abovementioned cognitive factors. The adaptation process involves the transformation and / or enhancement of a given raw Web-based content (provider's original content) based on the impact the specific human factors have on the information space [12, 74] (i.e., show a more diagrammatical representation of the content in case of an Imager user, as well as provide the user with extra navigation support tools). Figure 9 shows the possible Web content transformations / enhancements based on the mapping process that take place during adaptation process, the influence of the human factors and the theory of individual differences.

According to Fig. 6, the meta-characteristics of a user profile are deterministic (at most 3); Imager or Verbalizer, Analyst or Wholist and Working Memory level (considered only when low).

For a better understanding, a user that happens to be an Imager gets as mentioned above a diagrammatical representation of the containing information of the <csl> tag. The <csli> tag is used by the web browser extension to distinguish the logical meaning of a sentence when creating the diagrammatical representation. In other words, the <csli> tag is used for a new paragraph sentence in the <csl> division. As we will see later, the <csli> tag is interpreted differently by the browser when the user types change. On the other hand, when a user is a Verbalizer (prefers text instead of diagrammatical representations), no changes are made to the containing custom xml tags of <csl>.

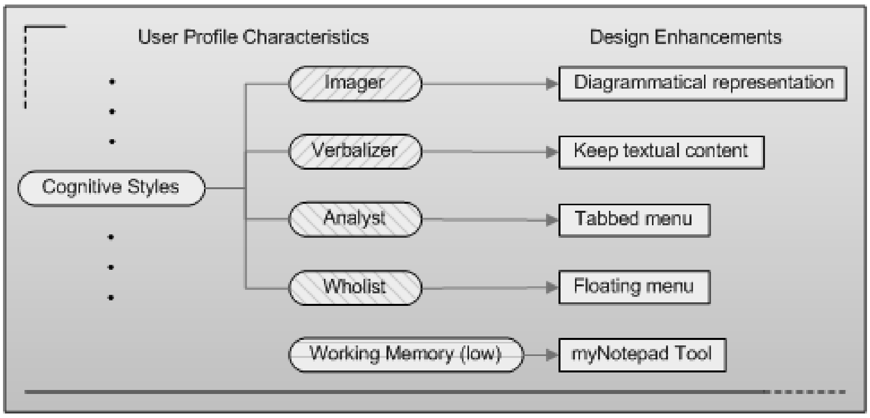

Figure 9. Web design enhancements / transformations

Furthermore, if a user is an Analyst, the information will be enriched with a tabbed menu to be easier accessible. The menu will consist of the <csl> element's containing <csli> tags. The <csli> tags along with the "name" attribute (Figure 6) are used in this case to create the tabbed menu with the name of each <csli> element comprising an item of the menu. Each <csli> element is added to the tabbed menu and is used as a dynamic link to the containing information of the particular tag. The same logic of transformation is used when mapping the <csl> with a Wholist user. In this case, a dynamic floating menu with anchors is created so to guide the users on specific parts into the content while interacting. Again, the <csli> elements comprise the menu's items.

Finally, when users happen to have a low working memory level, the browser will provide them with the “myNotepad” tool (temporary memory buffer) for storing a section (<csli> element content) of the page and keep active information that is interested in until the completion of a cognitive task at hand.

4.4 An eCommerce Adaptation Paradigm

We have designed and authored an experimental environment in the application field of eCommerce using the <csl> tags. The eCommerce (Web) environment that has been developed used the design and information content of an existing commercial Web-site of Sony[3]. This Web-site provides products’ specifications of the Sony Company. We have developed an exact replica of the Sony Notebooks’ section in sonystyle.com using the <csl> tags. Figure 10a depicts the Sony Web-site without any personalization made, while Figure 10b and Figure 10c shows the same Web-site after the personalization and adaptation process has been initiated, with the content to be adapted according to the user’s comprehensive profile and consequently the UPPC ontology.

As we can easily observe, the original environment has been altered based on rules that define the typologies of the users in terms of content reconstruction and supportive tools. For example, a user might be identified as an “Analyst-Imager” with low working memory and therefore the Web environment during interaction time would be as in Figure 10b. The information will be presented in a diagrammatic form (imager), will be enriched with menu tabs (analyst) to be easier accessible and with the “myNotepad” tool (temporary memory buffer) for storing sections' summaries (low working memory). In case that a user is identified as “Wholist-Verbalizer” the content will be automatically reconstructed as in Figure 10c, where a floating menu with anchors Wholist) have been added so to guide the users on specific parts into the content while interacting. In this case no diagrammatical presentation will be used because the user is a Verbalizer.

[3] See http://www.sonystyle.com (date extracted: September 19, 2007)

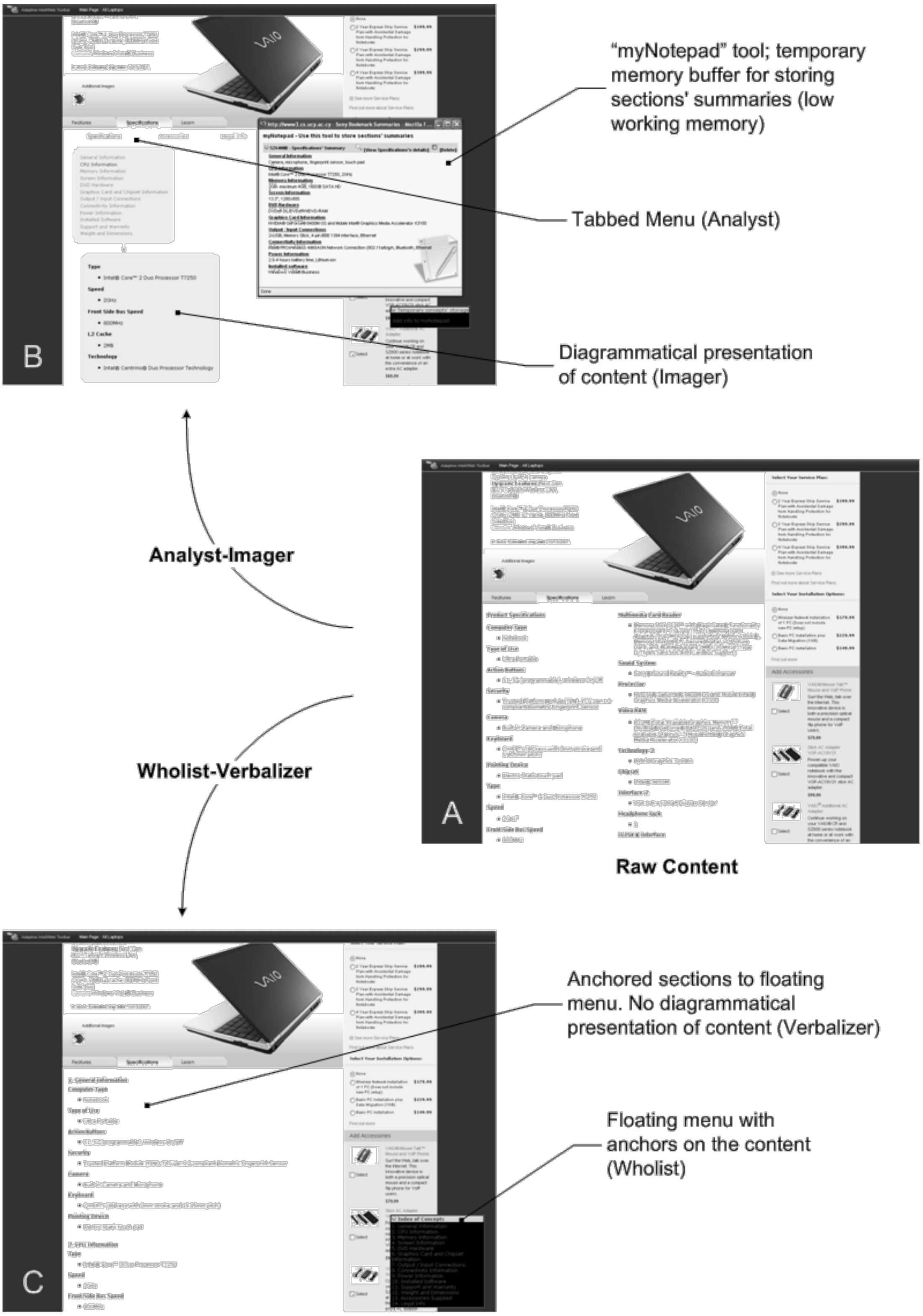

Figure 10. Content adaptation according to user's comprehensive profile

Chapter 5

The smarTag system and architecture

This section will explain in detail the smarTag system architecture, its components, their development and functionality. Throughout the system's description, we will also present you how data flows inside the system with the use of scenarios. Database tables' relation, data collation and retrieval, functionality, and usage of views and stored procedures as well as code snippets can be found in Appendix A and B at the end of this thesis.

This section is broken into two phases: i. Requirements Elicitation and Specifications Phase and, ii. Design and Implementation Phase. Each phase refers to each one of the System's Components and encounters them individually as if they were a standalone system.

At this point it has to be mentioned once more that all the theoretical framework and analysis of the system has been developed by the AdaptiveWeb Team. Furthermore, all the system components further discussed have been implemented from scratch.

5.1 Requirements Elicitation and Specifications

This section contains the specification for the implementation of an intelligent adaptation and personalization web system that is based on a Comprehensive User Profiling. The specification formally states the requirements for the smarTag System, and deliberates their implementation

The smarTag System (and more specifically the AdaptiveWeb System as a general framework) aims at establishing a more concrete definition of the Web Personalization notion introducing, and technologically evaluating methodologies and models derived from the research areas of visual processing, attention economy, social-cognitive psychology, affective computing. As highlighted in Section 3, many systems have been developed that tailor Web-based content to the needs of individual users providing them personalized and adapted information, with main premise to deliver quality knowledge.

This section explains the steps taken to pin down a set of requirements for a usable web personalization and adaptation framework that improves upon existing products by moving closer to the core goals of improvements in personalization and adaptation techniques; adapting content based on a comprehensive user profiling in order to deliver quality knowledge.

5.1.1 Automatic Profile Extraction Component Requirements

This component will be responsible for the user's automatic profile extraction. Its main objective is to create a comprehensive user profile that consists of the following parameters:

5.1.1.1 "Traditional" Profile Construction

User's "Traditional" Characteristics: A unique username, password, date of birth, gender, profession (Private Employee, 1^{st}, 2^{nd}, 3^{rd}, 4^{th} +Year Student, M.Sc. Student, Ph.D. Student, Other), knowledge on computers (High, Medium or Low) and User's Department name for university students.

5.1.1.2 Cognitive Styles Profile Construction

User's Cognitive Styles: The main parameters that will comprise the user's profile are the Imager / Verbalizer and Wholist / Analyst. These parameters will be extracted with the navigation of a user in a predefined environment with the use of click streams. More specifically, for the extraction of the Imager / Verbalizer parameter the system will track the navigation of the user to Web-pages containing more diagrams and images, thus the user tends to be an Imager or to Web-pages containing more textual content, thus the user tends to be a Verbalizer. As for the Wholist / Analyst, weights are used in specific sections of the Web-site. The navigation pattern for each type of user is assumed to be different. The Wholist user has a more sequential navigation pattern, therefore the less the weight difference between the sections is, the more a Wholist user (s)he is. The Analyst user on the other hand has a diffused navigation pattern, therefore the more the weight difference between the section is, the user tends to be an Analyst. There are also the cases where the user can be an Intermediate (something in between an Imager / Verbalizer and a Wholist / Analyst).

User's Working Memory: For the extraction of this parameter, no function has been developed in the smarTag System although it is used for the adaptation of content through the Firefox browser extension. A user can extract this parameter using the AdaptiveWeb System by participating in a specific psychometric test for Working Memory span and then can update his / her profile in the Firefox browser extension. More specifically, this test consists of 21 questions (7 levels of 3 questions). Initially, a small image with objects (text, numbers or schemas) is shown on the right upper corner for 2 seconds. After that, a bigger image with more objects is shown. The user has to answer true if all the objects of the small image are contained in the bigger image, otherwise false. Each time the user answers 3 questions his working memory level raises 1 level. When the user makes a mistake, the test is ended and his working memory level is the current level when he answered wrong. A user has low memory if he has a level of 1 or 2, medium memory if he has a level of 3, 4 or 5 and a high memory if he has a level of 6 or 7. A user has low memory if he has a level of 1 or 2, medium memory if he has a level of 3, 4 or 5 and a high memory if he has a level of 6 or 7.

A user will be able to update / complete his / her profile whenever he wishes; (s)he can navigate through the predefined environment as often as he likes as well as update his / her "Traditional" Profile.

Table 2 shows the decision making tables for the user's comprehensive profile with his / her actual characteristics.

Table 2. Decision making tables for constructing the user’s comprehensive profile

	Formula	**Cognitive Style**
Wholist / Analyst	if (counter <= 10)	Wholist
	if (counter > 10 && counter < 15)	Intermediate
	if (counter >= 15)	Analyst
Imager / Verbalizer	**Formula**	**Cognitive Style**
	if (imagerClickCount - verbalizerClickCount > 5)	Imager
	if (imagerClickCount - verbalizerClickCount == 0)	Intermediate
	if (verbalizerClickCount - imagerClickCount > 5)	Verbalizer
Working Memory Level	**Level (out of 7)**	**Actual Level**
	1, 2	Low
	3, 4 ,5	Medium
	6, 7	High

5.1.2 Mapping Rules Component (implemented in the Firefox browser extension)

The system’s “Mapping Rules” are functions that are implemented and run in the Firefox browser extension and comprise the main body of adapting and personalizing the provider’s content, according to the user’s cumulative profile.

According to the user’s Comprehensive Profile, specific adaptation of content will take place. Furthermore, we will present you different scenarios of a user profile to show the mapping between the user’s profile characteristics and the content adaptation process.

Table 3. Content adaptation according to the user's learning style (Imager / Verbalizer)

	Cognitive Style	Content Adaptation
Mapping Rule 1	Imager	Show content in a diagrammatical representation
	Intermediate	Keep textual content and provide some diagrams
	Verbalizer	Keep textual content

Table 4. Content adaptation according to the user's learning style (Wholist / Analyst)

	Cognitive Style	Content Adaptation
Mapping Rule 2	Wholist	Use a tabbed menu
	Intermediate	Use floating menu and tabbed menu
	Analyst	Use floating menu

Table 5. Content adaptation according to the user's working memory level

	Working Memory Level	Content Adaptation
Mapping Rule 3	Low	Use "myNotepad" tool
	Medium	No use of tools
	High	No use of tools

Based on the mapping rules' tables (Table 3, Table 4, Table 5), a user that happens to be an Imager gets a diagrammatical representation of content. On the other hand, when a user is a Verbalizer (prefers text instead of diagrammatical representations), no changes are made to the content; thus keep textual content. In the case of an Intermediate, most content is kept in textual form and some diagrams are presented.

Furthermore, if a user is an Analyst, the information will be enriched with a tabbed menu to be easier accessible. The menu will consist of the content's main sections. Each section element is added to the tabbed menu and is used as a dynamic link to the containing information. The same logic of transformation is used with a Wholist user. In this case, a dynamic floating menu with anchors is created so to guide the users on specific parts into the content while interacting. Again, the section elements comprise the menu's items.

Finally, when users happen to have a low working memory level, the browser will provide them with the "myNotepad" tool (temporary memory buffer) for storing a section of the page and keep active information that is interested in until the completion of a cognitive task at hand.

5.2 System Design and Implementation

The developed system, smarTag, is a Web application that can be ported to desktop computer. It is composed into a number of interrelated components, each one representing a stand alone Web system as described in section 4:

- Automatic Profile Extraction Component
- Firefox browser extension Component
- Adaptation and Personalization Process (Mapping Rules) Component

5.2.1 System Overview

A good design is the basis of a good final application, so before defining exactly how we will be implementing it we will describe an abstract overview of our application. Furthermore, a brief overview of the main components of our system and how they will interact is presented.

<u>*Component 1*</u> - <u>*Automatic Profile Extraction*</u> - This is the initial step the user makes for the smarTag System's personalization process. It is a vital part of the system. At this point the user creates his / her comprehensive profile, which is going to be mapped at a later stage with the personalized content.

Component 2 - *Firefox browser extension* - The second component, is the main component of the system that is responsible for the adaptation process. It is an extension of the Firefox browser and is used for displaying the raw or personalized and adapted content on the browser. The main concept of this component is to provide a framework where all personalized Web-sites can be navigated. Using this interface the user will navigate through the provider's content. Based on his / her profiling a further support will be provided to him / her with the use of tools adjusted accordingly.

Component 3 - *Adaptation and Personalization Process (Mapping Rules)* - In this section, all the system's components interact with each other in order to create and give personalized and adapted content to the end user. The author of a page creates the content based on the theoretical model behind the <csl> tags, which will be mapped after with the system's "Mapping Rules". The system's "Mapping Rules" are functions that run on the smarTag browser extension and comprise the main body of the adaptation and personalization procedure of the provider's content, according to the user's comprehensive profile.

5.2.2 Designing each component individually

A more detailed and a low-level description of each component will take place; describing individually each component's data flow inside the system, data models and schemas.

5.2.2.1 Automatic Profile Extraction Component

To get personalized and adapted content, a user has to create his / her comprehensive profile. The "Automatic Profile Extraction" component is responsible for the creation of this content (Figure 11).

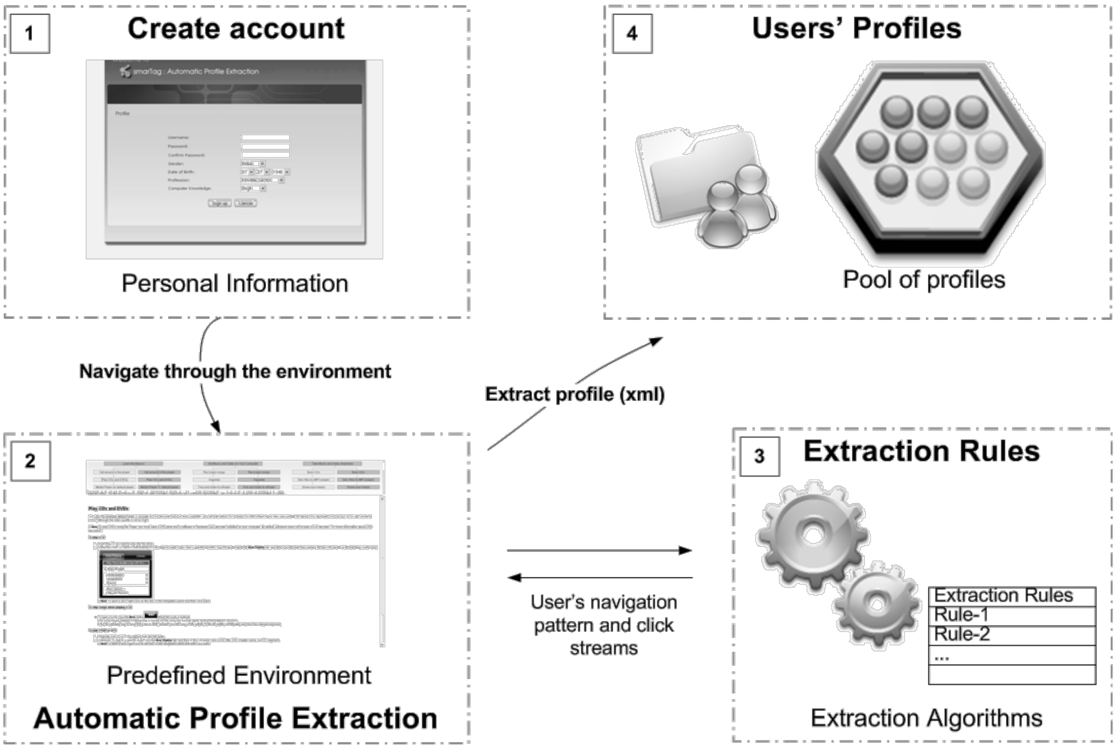

Figure 11. Automatic Profile Extraction Architecture

At this point the user has to provide his / her "Traditional" Characteristics and further navigate through a predefined environment (cognitive processing efficiency grabbing tool). If a user has not completed all of the available tests, the system will not be able to provide an adapted Web-page.

We will step lower and present you how data flows inside the Automatic Profile Extraction Component.

There are two types of users; a new user that wants to create a new user profile, or an existing user that wants to create or update an existing profile with his / her cumulative characteristics. The visitor has to choose if he wants to create a new user's record in order to become a member of the system and to enable him / her building his / her profile. On the other hand (s)he could be an existing user that wants to finish creating his / her profile or update specific characteristics of his / her profile. Figure 12, Figure 13 and Figure 14 shows the component's data flow with the use of scenarios breaking into a deeper level each time.

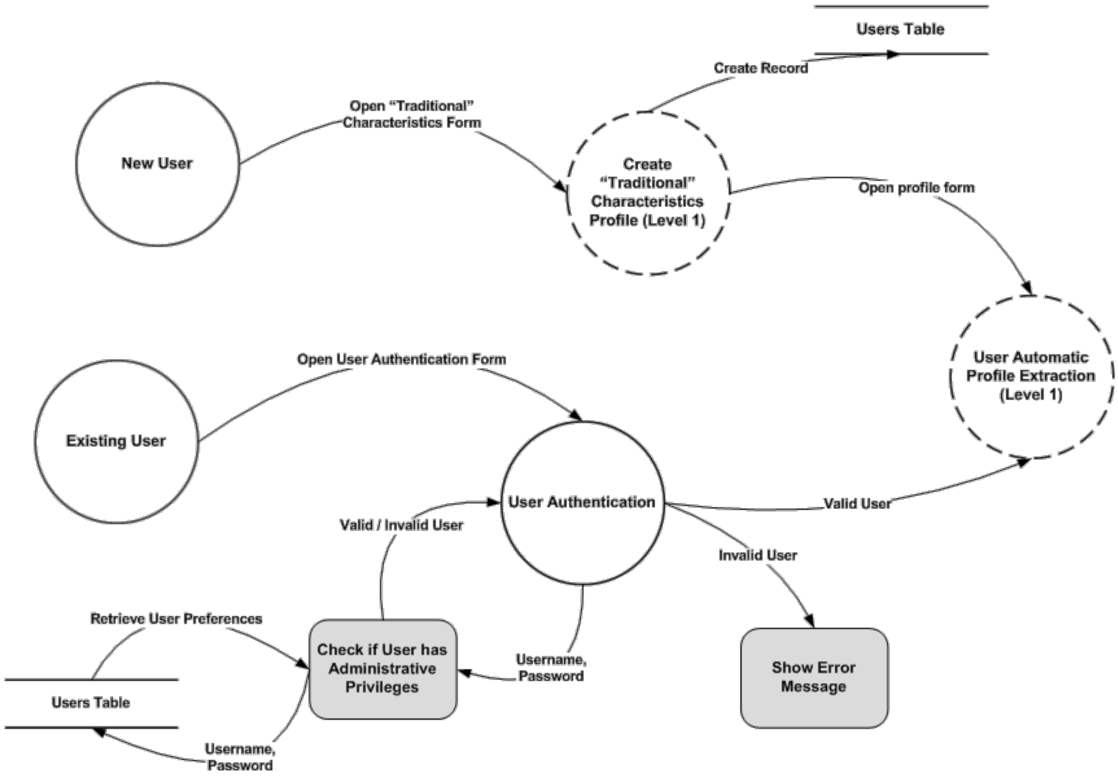

Figure 12. Profile Construction Scenario (Level 0)

Someone who chooses being a "New User" has to complete the first step of the process, which is filling in his / her "Traditional" Characteristics. More specifically, (s)he gives his / her username and password, which is a unique combination across users. This combination will be used by the user afterwards for logging in the system in order to create or update his / her profile. (S)He also gives his / her age, gender, educational knowledge on computers, and profession. The user has the ability to go back to the previous forms and change any of his / her characteristics. By confirming that all data given are correct, a new record is created in the system's database. The unique key being used to distinguish each user is his / her username which is unique across users. Now being a valid user of the system, the user can create / update his / her profile whenever (s)he likes.

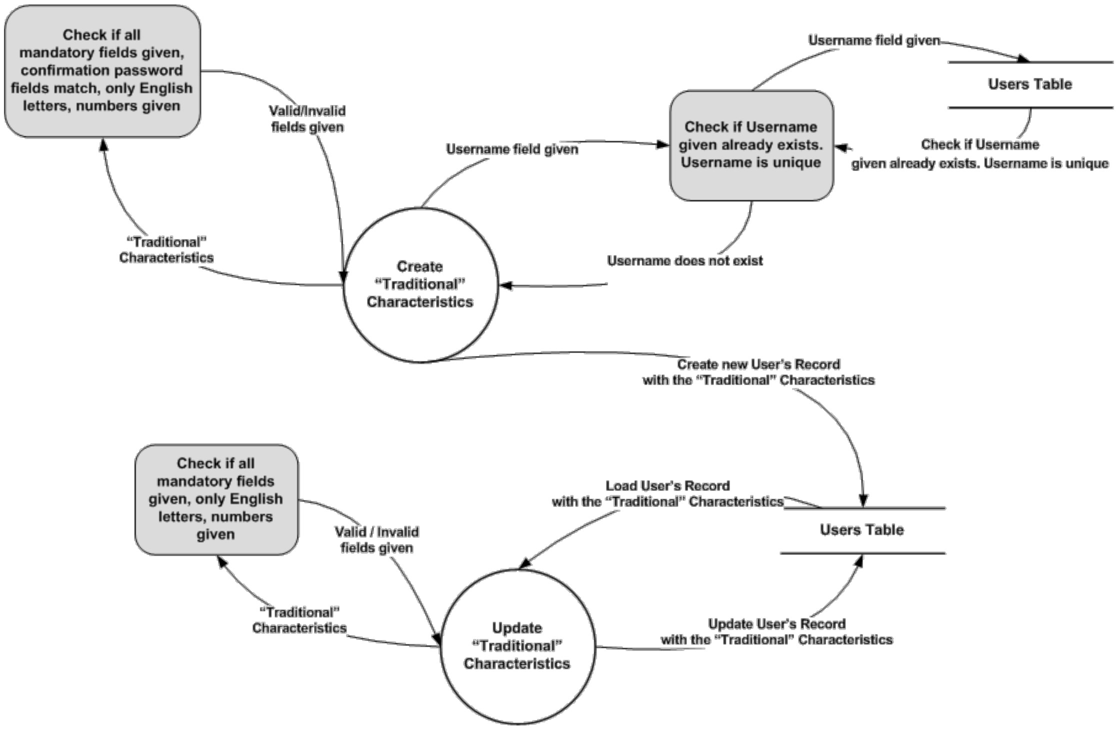

Figure 13. "Traditional" Profile Construction Scenario (Level 1)

A user that wants to create / update his / her profile has to pass the authentication page where (s)he will be asked to give his / her username and password that (s)he gave at the initial steps, creating his record. When a user passes the authentication successfully, a list with all available actions and functions will be shown. When a user hasn't completed all the profile extraction process, the system will not be able to give him / her personalized and adapted content of a page. The user has the ability to click and launch the navigation process whenever (s)he likes in order to create / update his / her profile. When the navigation process is launched the environment is being loaded on his / her computer (along with all the algorithms and extraction rules). After 20 minutes of navigation, all extraction results are sent to the system's database. A new profile record is created in the database which is unique across all profiles by using a unique key.

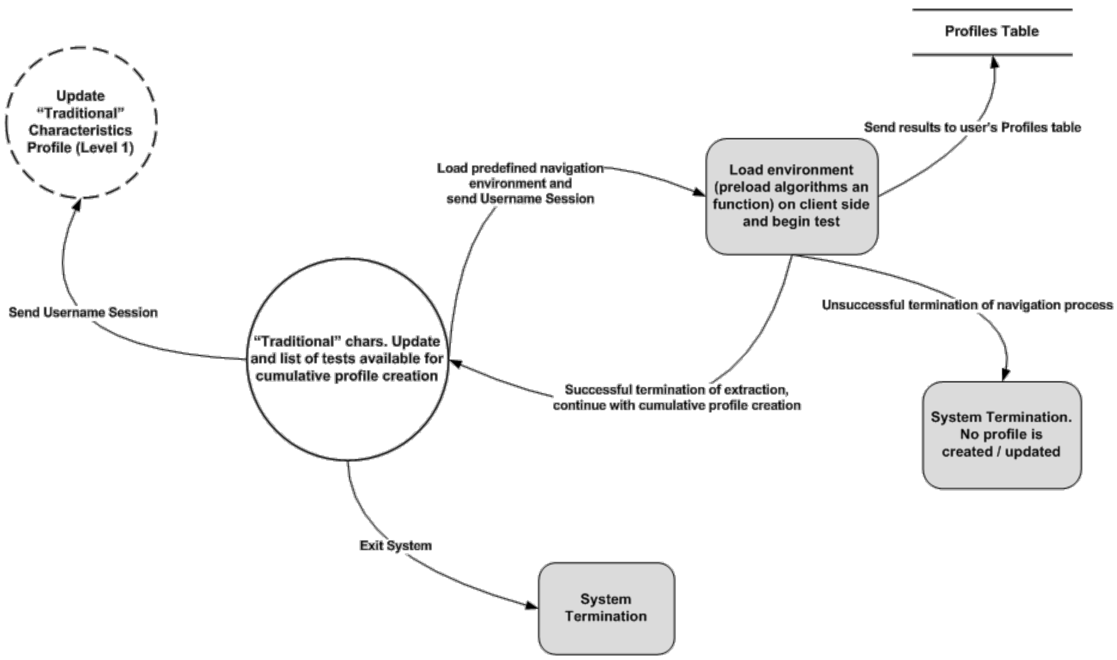

Figure 14. Profile Construction Scenario (Level 1.1)

A user can update his / her “traditional” characteristics, as well as navigate through the predefined environment for the automatic profile extraction as often as (s)he likes. The system will only consider the most recent characteristics of a user for creating his / her cumulative profile. All other characteristics that are out dated (history) will be only used for research purposes from the administrators of the system.

Furthermore, our main concern is to ensure openness and interoperability within and between system’s components. In case an external component wants to access the user’s profile, either for adaptation, for historic or statistic calculations, the system must be able to support extraction of the user’s profile. In order to achieve this, the user’s profile must be easily extendible and easy to handle. Using XML for implementing the user’s profile seems to be the best way to achieve this. Indeed XML enables the extendibility we need and enhances interoperability and integration among systems’ components.

We have designed a Web Service (a software system designed to support interoperable Machine to Machine interaction over a network) for retrieving the users’ comprehensive profile. Depending on the needs of a third party system that interacts with our system through this middleware; calculations are made and are finally exported in XML. For a better insight, an XML instance of the Comprehensive User Profiling, giving emphasis on the comprehensive user profile structure, is depicted in Figure 15.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<profile>
  <traditionalChars>
    <gender>0</gender>
    <dateOfBirth>21/2/1985</dateOfBirth>
    <profession>8</profession>
    <computerKnowledge>0</computerKnowledge>
  </traditionalChars>
  <uppcChars>
    <imagerVerbalizer>0</imagerVerbalizer>
    <wholistAnalyst>2</wholistAnalyst>
    <workingMemory>1</workingMemory>
  </uppcChars>
</profile>
```

Figure 15. The Tree Structure of the Comprehensive User Profiling XML document

5.2.2.2 Content Authoring and Adaptation Process

We have already discussed the content authoring methodological approach for the enhancement of an external Web-site with the use of smart objects in section 4. This section will emphasize on the implementation part of the Firefox browser extension and how the smart objects interact with the user's profile.

The basic idea behind the Firefox browser extension is to enhance the current browser to understand a set of xml tags (smart objects) while parsing any HTML document and adapt these objects based on the user's profile that is embedded in the browser. Any web developer can enhance the content of a Web-site with the use of these objects, which will be mapped after with the system's "Mapping Rules". The system's "Mapping Rules" are functions that run in the Firefox browser extension and comprise the main body of the adaptation and personalization procedure of the provider's content, according to the user's comprehensive profile.

To get a better insight how the smart objects interact with the user's profile we hereafter depict the Content Description Schema of a smart object and the user's profile Description Schema (Figure 16 and Figure 17 respectively), while Figure 18 shows the whole mapping process.

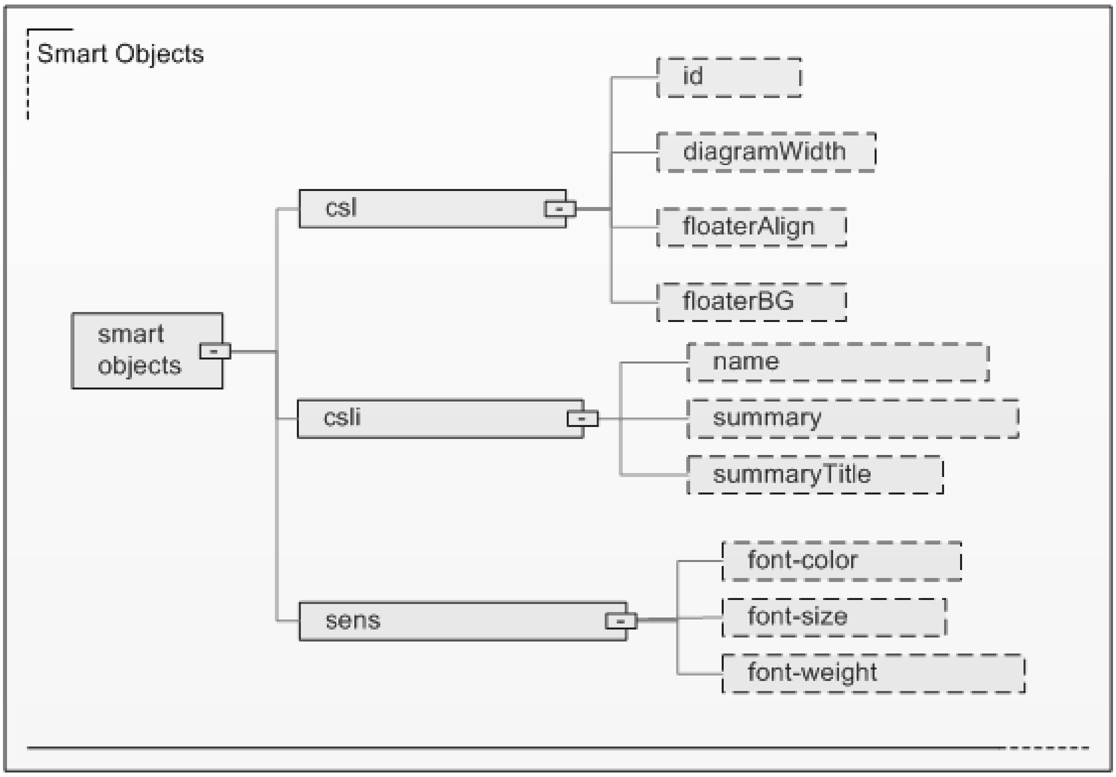

Figure 16. Content Description Schema

The current content description schema (Figure 16) of the smarTag framework consists of 3 smart objects and can be extended to any number of smart objects.

CSL object: It consists of 4 attributes and 1 sub-element (the CSLI object). Each CSL object has i) a unique id to distinguish it from other CSL objects in the same Web-page, ii) a "diagramWidth" attribute to define the width of the diagram in case the content has to be presented in a diagrammatical form, iii) a “floaterAlign” and “floaterBG” attribute to define the alignment and the background colour of the floating menu respectively in case one needs to be used as an enhancement to support the navigation of a Web-page. The CSL object also has a sub-element, the CSLI object.

CSLI object: The CSLI object is a sub-element of the CSL object and is used to distinguish between sub sections of an object and to create an assistive navigation menu (tabbed or floating) each menu link representing a CSLI object. It consists of i) a “name” attribute which is used for the menu link created, ii) a “summary” and a ”sumaryTitle” attribute that are used in case the “myNotepad” tool is used to store summaries of the particular object.

SENS object: This object is not part of the implementation of the particular thesis. We only present it to show the easy extendibility the smart objects description schema has. The SENS object can be used in a Web-site where users’ emotions play a vital role for adapting Web content (e.g. eLearning) and more specifically changing the aesthetics of an object in case users have high levels of anxiety. It consists of 3 attributes, i) “font-color” to define the colour of the object, ii) “font-size” to define the font size of an object and iii) “font-weight” to define the font weight (e.g. bold) of an object.

By using these smart objects in a Web-page, the Firefox browser extension will parse the HTML document and will interpret accordingly the smart objects based on the user’s profile that is embedded in the browser.

The user’s profile description schema (Figure 17) used in the Firefox browser extension consists of the following parameters; i) Cognitive Styles and more specifically, Imager / Verbalizer and Wholist / Analyst, ii) Working Memory Span, iii) Current Anxiety and the iv) “Tradition” Characteristics. For the scope of this thesis the Firefox browser extension uses only the first three parameters for the mapping of the parameters with the smart objects.

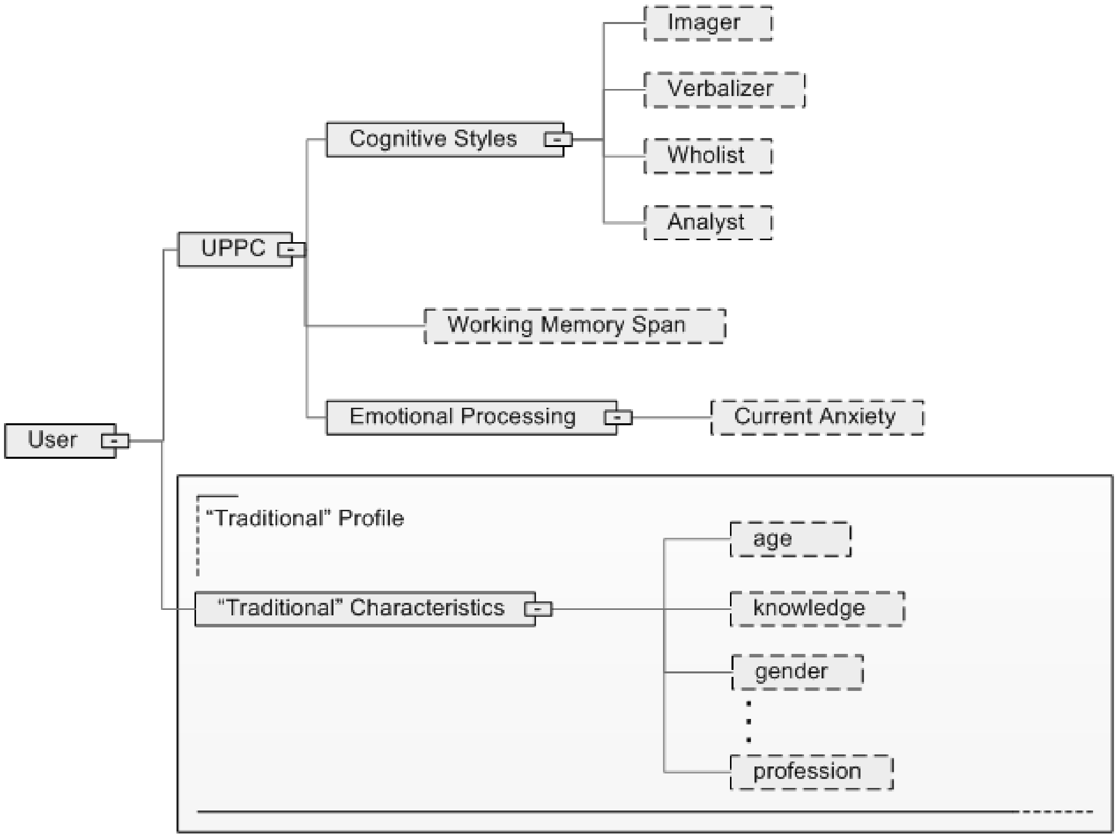

Figure 17. User's Profile Description Schema

In the following example (Figure 18), the user happens to be an Imager / Analyst with regards to the Cognitive Style and a low Working Memory Span (weighting 2/7). Using these preferences the data-implications correlation diagram is evaluated.

Every Web-page is detached into smart objects, each one having special characteristics. In our example, the user visits the "WebPage_Y" Web-page. First, the main HTML document of this Web-page is retrieved which contains all the needed information for building the Web-page; that is, (i) the HTML Web-page itself which is a predefined HTML document (designed by the provider) keeping information of specified divisions / frames in the page for positioning each object, (ii) all objects (text, image, audio, video etc.) and smart objects that comprise the content of the Web-page, and (iii) a generated JavaScript file from smarTag that is responsible for the proper integration of the smart objects into the divisions' Web-page.

At this point we have all the information we need for adapting the content; the data-implications correlation diagram based on the user's comprehensive profile and the content description of the particular Web-page. The next step is to map the implications with the Web-page's content, for assembling the final version of the provider's content.

The interpretation of the user's data-implications correlation diagram results in the following conclusions: (a) the user is an Imager, therefore the provision of visual information (diagrammatical representation) is predominant, (b) is provided with the "myNotepad" tool; temporary memory buffer for storing sections' summaries, as well as (c) extra navigation support tools are provided, devised to be more applicable while interacting with an eCommerce environment.

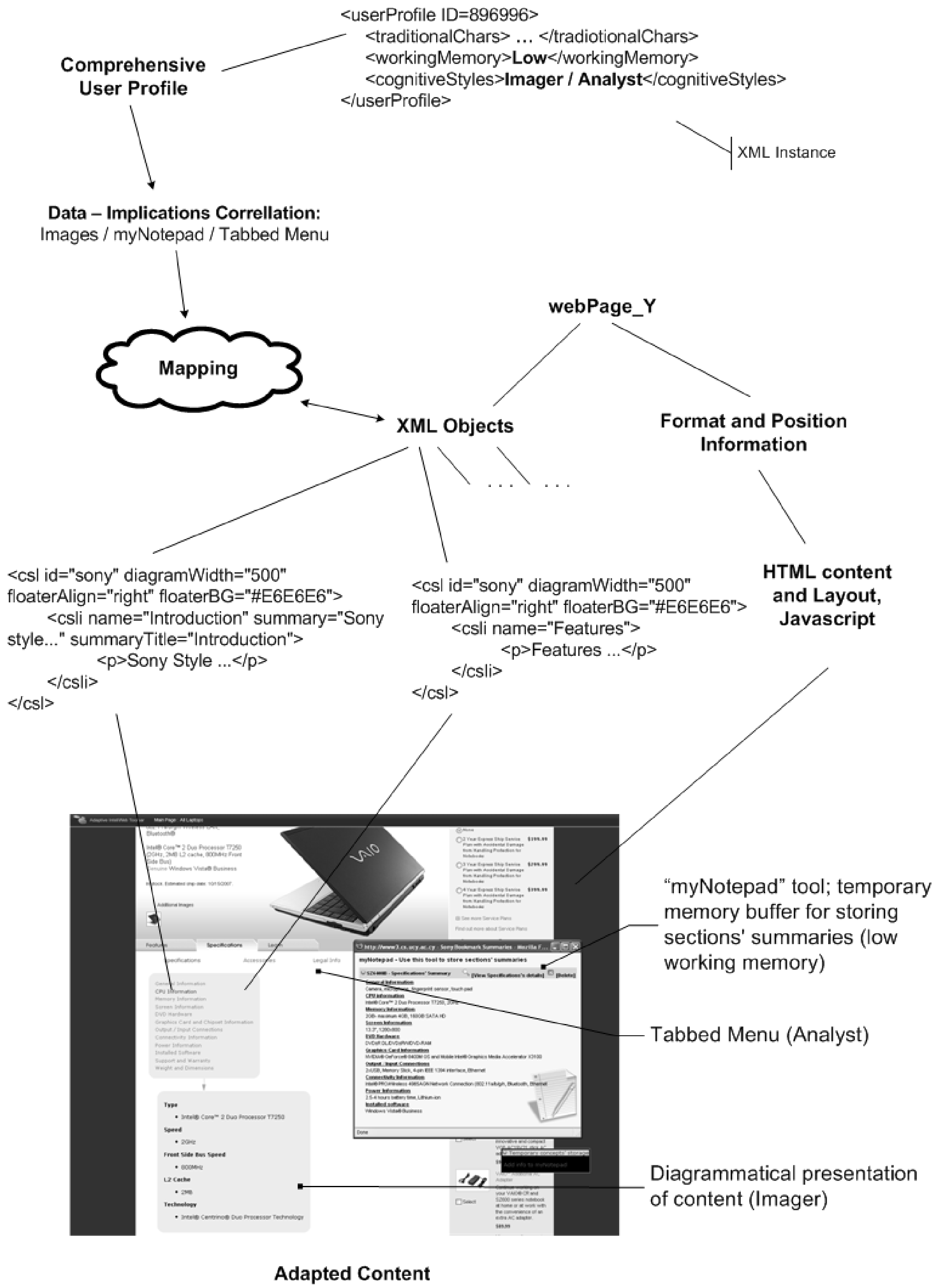

Figure 18. The Adaptation Process

Furthermore, Figure 19 shows the mapping process using our example; explained in pseudo code.

Algorithm : Mapping Process Phase

Input: User's profile (Learning Styles, Working Memory level), <csl> containing content

Output: Adapt the content inside the <csl> object and present the adapted and personalized html content

Execute these steps (top-down):

1) Parse the whole html document

    Filter out all <csl> objects and all their containing data;

2) Break each <csl> object into <csli> items

3) Change the <csli> items' presentation

    if (learningStyle == Imager)

      For each <csli> item in <csl> object, transform the content into a diagram box and create an arrow at the end of each diagram (<csli> item);

    elseif (learningStyle = Verbalizer)

      For each <csli> item in <csl> object, show only the containing data of the <csli> item (plain html content);

3) Create Dynamic Enhancement tools

    if (learningStyle == Wholist)

      For each <csli> item in <csl> object, get its "name" attribute and use it to create a floating menu. Assign a unique id to each menu item and link it with the <csli> item's containing data;

    elseif (learningStyle = Analyst)

      For each <csli> item in <csl> object, get its "name" attribute and

---

use it to create a tabbed menu. Assign a unique id to each menu item and link it with the <csli> item's containing data. When a menu item is clicked, the corresponding <csli> item will be shown;

4) Create "myNotepad" tool (temporary memory buffer)

Provide the user with this tool for storing a section (<csli> element content) of the page and keep active information that is interested in until the completion of a cognitive task at hand;

Create a link next to each <csli> item, along with a unique id for the corresponding item;

OnUserClick store the <csli> content into "myNotepad" tool;

Figure 19. Mapping Process Example (pseudo code)

The content will be adapted according to the user's preferences. The new, adapted content will then be loaded onto the user's device.

Chapter 6

Evaluation

The current environment and the dynamic transformation mechanism are currently at the evaluation stage. However, the whole procedure is driven by our previous findings [74, 75], whereby the alteration of presentation based on various cognitive factors has been proved efficient and effective. The predefined environment devised in this case concerns the Sony Company and the section of laptops' specifications.

6.1 Assessing System's Performance

To measure system's performance, functional behaviour and efficiency of our system we have run two different simulations with 100 threads (users) each: (a) users retrieving raw content without any personalization and adaptation taking place and (b) users interacting with adapted and personalized content. In the second scenario, there is a significant increase of functions and modules ran, compared to the first one (raw content scenario), like user profile processing, dynamic content adaptation, navigational support etc. Based on the simulations made (Figure 20) we assume the following: (i) Deviation for raw content is 67ms and for personalized content 98ms. This difference is expected since the system uses more functional components in the case of personalized content like profile loading, dynamic content, etc. Thus, this consumes more network resources, due to the enhanced/extended content, causing the deviation of our average to be greater than that of the raw content test.

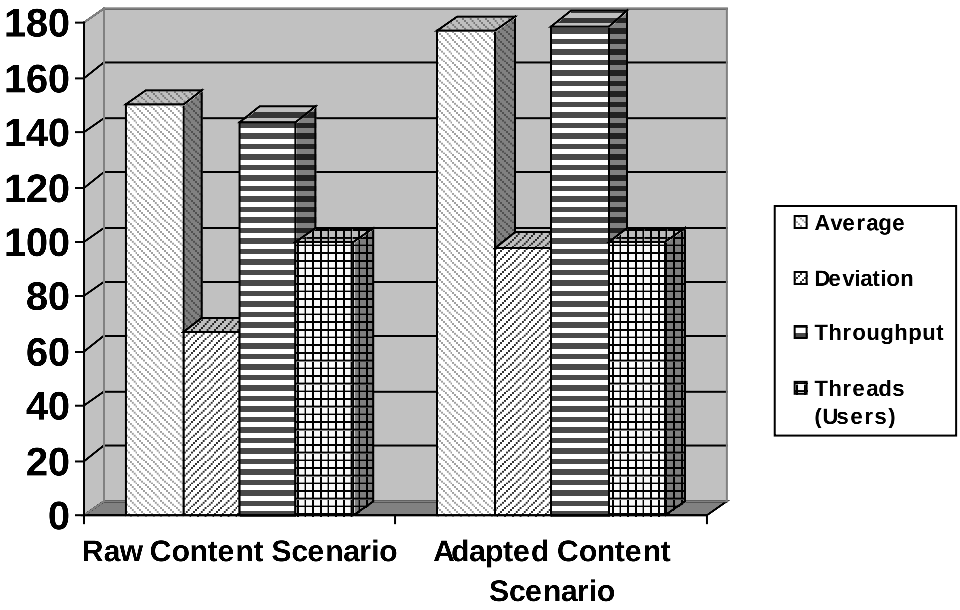

Figure 20. Raw Content and Adapted Content Scenarios

The deviation is not considered to be significantly greater and thus this metric result is proving the system to be stable and efficient; (ii) the throughput for the raw content scenario was 144Kb/sec while for the personalized content was 179Kb/sec. Based on the latter results, the system is again considered efficient mainly due to the fact that the difference in throughput between the two scenarios is minimal. Taking in consideration that major component functionality is used in the case of personalized content this small difference underlines the efficiency of the system; (iii) the same arguments are true in the case of the average response times. The average response time for the raw content scenario was 150ms while for the personalized content was 177ms. This difference is again marginal that proves the efficiency of the system.

6.1.1 smarTag System vs. AdaptiveWeb System

Furthermore, since the smarTag framework is an extension and optimization of the AdaptiveWeb System, we compared the two systems' performance, speed and efficiency. We have used an online Web Server and Web-site Monitoring Service, called Web Site Pulse (http://www.websitepulse.com/) to measure the systems' performance. The following table shows the significant difference and improvement of the personalization and adaptation process of the smarTag framework.

Table 6. Comparison table of smarTag and AdaptiveWeb frameworks

	smarTag framework	**AdaptiveWeb framework**
Average response time (in personalization mode)	0.313 sec	1.211 sec
Average size of files	50KB	300KB
Functions run	On Client side	On Server side
User's profile	On Client side	On Server side

As we can clearly observe from the above figure, the smarTag framework has made an important optimization in terms of system's performance, speed and memory usage. Both frameworks provide the same functionality of personalization. The response time in personalization mode in the smarTag framework is 0.313 sec, whereas in the AdaptiveWeb framework 1.211 sec; a decrease of 75% of response time. The average size of a page in smarTag is about 6 times less than the average size of AdaptiveWeb predefined Web-pages. As for the functionality, most of the functions and the user's profile in smarTag are stored and run on the client side and in AdaptiveWeb are stored and run on the server side which adds an overhead to the response time and complexity since connectivity has to be established for the retrieval of the user's profile, navigation tools etc.

6.2 Assessing the impact of Human Factors in Web services development process

As mentioned above, previous research [11, 12, 74] related to the use of human factors in the design and development of eServices/eCommerce (as well as eLearning) systems, it has been proven to have a positive effect to the end user customer (increase satisfaction, easier navigation, faster completion of tasks/goals). Therefore, an extended version would include the measurement of satisfaction of the content provider, in terms of efficiency and effectiveness of use for developing and designing their products for promotion using the particular framework.

6.2.1 Methodology and Design Implications

In order to evaluate such an approach a within participants experiment was conducted, seeking out to explore if the personalized condition based on the particular cognitive factors serves users better at finding information more accurately and fast.

The number of participants was 89; they all were students from the Universities of Cyprus and Athens and their age varied from 18 to 21, with a mean age of 19. They accessed the Web environments using personal computers located at the laboratories of both universities, divided in groups of approximately 12 participants. Each session lasted about 40 minutes; 20 minutes were required for the user-profiling process, while the remaining time was devoted to navigating in both environments, which were presented sequentially (as soon as they were done with the first environment, the second one was presented).

The content was about a series of Sony laptops: general description, technical specifications and additional information were available for each model. We considered that the original (raw) version of the environment was designed without any consideration towards cognitive style preferences, and the amount of information was so high and randomly allocated that could increase the possibility of cognitive overload. The personalized condition addressed these issues by introducing as personalization factors both cognitive style and working memory span. The psychometric materials that were used are the following: i) Cognitive Style: Riding's Cognitive Style Analysis, ii) Working Memory Span: Visuospatial working memory test [14, 15].

In each condition, users were asked to fulfil three tasks: they actually had to find the necessary information to answer three sequential multiple choice questions that were given to them while navigating. All six questions (three per condition) were about determining which laptop excelled with respect to the prerequisites that were set by each question. There was certainly only one correct answer that was possible to be found relatively easy, in the sense that users were not required to have hardware related knowledge or understanding.

As soon as users finished answering all questions in both conditions, they were presented with a comparative satisfaction questionnaire; users were asked to choose which environment was better (1-5 scale, where 1 means strong preference for environment A and 5 for environment B), regarding usability and user friendliness factors.

The dependent variables that were considered as indicators of differences between the two environments were:

a) Task accuracy (number of correct answers)

b) Task completion time

c) User satisfaction

The within participants design allowed the control of differences and confiding variables amongst users.

Regarding the design implications in this eServices/eCommerce setting, the content enhancements and transformation considerations discussed in previous sections regarding users' particular typologies were followed. More specifically, users with low working memory received a "myNotepad" tool that allowed them to make entries of goal-related information, while as it concerns cognitive style Table 7 shows the implications for each preference.

Table 7. Implications for cognitive style preferences in the eCommerce environment

Imager	**Verbalizer**	**Analyst**	**Wholist**
Presentation of information is visually enhanced as to resemble a diagrammatical form of representation	The usage of text is predominant, unaccompanied by any visual enhancements	The structure of the environment is chunked to clear cut links, as to match the analytical way of thinking	The structure of the environment is less segmented and follows a more holistic pattern. Users are shown where they are and what they have viewed, while a more sequential approach is encouraged

Intermediates in both axes received a condition that was balanced between the opposite preferences.

6.2.2 Results

The most robust and interesting finding was the fact that users in the personalized condition were more accurate in providing the correct answer for each task. The same user in the raw condition had a mean of 1 correct answer, while in the personalized condition the mean rose to 1.9.

Since the distribution was not normal and the paired samples t-test assumptions were not met, Wilcoxon Signed Ranks Test was performed, showing that this difference is statistically significant at zero level of confidence (Z= -4.755, p=0.000). This is probably a very encouraging finding, implying that personalization on the basis of these factors (cognitive style and working memory span) benefits users within an eServices/eCommerce environment, as long as there are some cognitive functions involved of course (such as information seeking).

Equally interesting is the fact that users in the personalized condition were significantly faster at task completion. The mean aggregated time of answering all three questions was 541 seconds in the raw condition, and 412 in the personalized. A paired samples t-test was performed (t(88)=4.668, p=0.000) demonstrating significance at zero level of confidence. Again, this second dependent variable (time) shows that the personalized environment is more efficient (Figure 21).

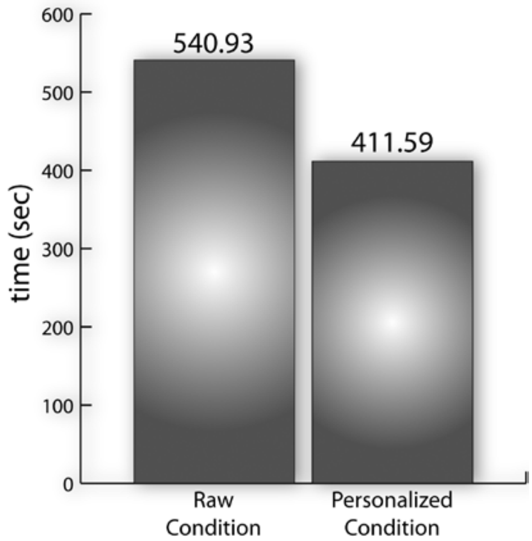

Figure 21. Difference in task completion time between the two conditions

As it concerns the satisfaction questionnaire, 31 users leaned towards the personalized environment, 38 had no preference while 20 preferred the raw. This descriptive statistic is merely indicative of whether participants would consciously observe any positive or negative effects of the personalized condition. A considerable percentage leaned towards that condition (or at least users did not seem somehow annoyed by such a restructuring), but overall it cannot be supported that they were fully aware of their increase in performance, as shown by the abovementioned findings.

In sum, the specific experiment shows in a rather clear way that users performed better within the personalized environment, and these findings are statistically very robust. It could be argued of course that there is no way to be fully aware if information processing was more efficient at a deeper level, or users simply found the personalized condition more of their (perhaps unconscious) liking, thus devoting more conscious cognitive effort.

Nevertheless, such an increase in performance, which is consistent to the findings of previously conducted experiments in the field of eLearning [92], provides support for the further development and application of the particular cognitive factors in different Web-based services environments and generic hypertext/hypermedia contents.

6.3 Assessing the performance of the Automatic Profile Extraction Component

The third phase of our research was to evaluate the performance and the effect of the Automatic Profile Extraction Component for extracting the user's profile. For the purposes of such an empirical validation, we created a predefined commercial environment, in order to extract the users' profiles. The parameters that were evaluated are the Imager / Verbalizer and Wholist / Analyst.

The number of participants was 32; they all were students from the Universities of Cyprus and their age varied from 20 to 22. They accessed the Web environments using personal computers located at the laboratories of the university. Each session lasted about 30 minutes; 10 minutes were required for the user-profiling construction process used in the AdaptiveWeb System, while the remaining time was devoted to navigating in the predefined environment for the automatic extraction of the users' profiles with the use of click streams and by tracking their navigation behaviour in the environment

The content was about the Microsoft Windows Media Player 11 (Figure 4): How it works, how to share music on the computer and through the network. The user was assisted with a site map tree of the whole Web-site. Each link was divided in two versions, the one having more textual content, while the other more images. By this we were tracking the amount of use of the two versions. We assumed that the more a user prefers the version with more images, the more Imager type (s)he is.

As for the Wholist / Analyst, weights are used in specific sections of the Web-site. The navigation pattern for each type of user is assumed to be different. The Wholist user has a more sequential navigation pattern, therefore the less the weight difference between the sections is, the more a Wholist user (s)he is. The Analyst user on the other hand has a scattered navigation pattern, therefore the more the weight difference between the section is, the user tends to be an Analyst. There are also the cases where the user can be an Intermediate (parameter that is in between an Imager / Verbalizer and a Wholist / Analyst).

The initial results of the Automatic Profile Extraction Component have been proved effective and efficient not only regarding the information flow within and between the various standalone system's components but also in respect to the actual output data gathered which reveals that the whole approach turned out to be initially successful.

Table 8. Automatic Profile Extraction Experiments' Results table (Wholist / Analyst)

AdaptiveWeb Profile Results	Average counter clicks
Wholists (11)	7.9
Intermediates (19)	10.9
Analyst (2)	13

Table 9. Automatic Profile Extraction Experiments' Results table (Imager / Verbalizer)

AdaptiveWeb Profile Results	Average counter clicks on images version	Average counter click on textual version
Imagers (6)	11.5	6.3
Intermediates (10)	10.7	8.2
Verbalizer (16)	9.9	6.6

More specifically, we compared the Cognitive Styles results (Imager / Verbalizer, Wholist / Analyst) derived by the real-time psychometric tests that each user had to execute under the

AdaptiveWeb framework with the Cognitive Styles results of each user identified by the Automatic Profile Extraction Component in smarTag. The experiments have shown that 18 out of 32 students have matched results regarding the Imager / Verbalizer parameter and 19 out of 32 students have matched results regarding the Wholist / Analyst parameter.

In this regards, the average counter clicks (Table 8) of the 11 Wholists was 7.9, for the 19 Intermediates was 10.9 and for the 2 Analysts was 13. This supports, the theory, that Wholists undertook less click stream due to their serial navigational nature as opposed to the other Cognitive Style extreme, Analyst, which undertook more click stream due to their scattered navigational behaviour.

As far as Imager / Verbalizer Cognitive dimension is concerned, the results were not that impressive. More specifically, the average counter clicks (Table 9) on the image version of the 6 Imagers were 11.5 and 6.3 on the textual version as opposed to the 10 Intermediates that scored 10.7 and 8.2 for image and textual version respectively and the 16 Verbalizers that had an average of 9.9 and 6.6 clicks for image and textual version respectively. However, even though the Imager and Intermediate parameters of this dimension support the given theory, these findings do not support the theory that Verbalizer users prefer textual content as it clearly showed using the AdaptiveWeb System. This might be due to the small size of sample which is not normally distributed and the general speculative tendency that the user population prefers images.

Table 10. Automatic Profile Extraction Component Accuracy

Profile Result	Accuracy
Imager / Verbalizer	56.25%
Wholist / Analyst	59.3%

To conclude with, the initial results (Table 10) of the Automatic Profile Extraction Component show that regarding the Imager / Verbalizer parameter there is 56.25% accuracy in comparison with the Profile Construction Component used in the AdaptiveWeb System and a 59.3% accuracy regarding the Wholist / Analyst parameter.

Chapter 7

Conclusions and future work

The explosive growth in the size and use of the World Wide Web as a communication medium has been enthusiastically adopted by the mass market to provide an electronic connection between progressive businesses and millions of customers bringing to light the eCommerce sector. eCommerce aims to deliver better quality of eServices increasing productivity with focused services to be provided by various channels, at a lower cost and time and in a personalized style.

Research implications and challenges of the Web Personalization and mass customization concepts could be considered as an enabler of eCommerce services sustainability. To succeed this, customers must not be spatially disoriented and be able to have continuous and adapted access on information and services requested.

In this regards, the basic objective of this thesis was to present a framework, namely smarTag, for the dynamic reconstruction of any Web content based on human factors for providing a comprehensive personalized result. According to these attributes the main content of a Web-page will be adjusted to the various typologies of users (mainly presentation, flow of content as well as quantity of content based on users' working memory). This approach is liable of enhancing efficiency and effectiveness of users' interaction with eServices in terms of information assimilation and accuracy of finding their cognitive targets (products or services).

Based on previous findings, it has been proven that user's cognitive factors have an important impact in the information space and on specific content meta-characteristics. Accordingly, the smarTag system provides an easy to use framework for enhancing any Web-site with smart objects that take into consideration human factors for the adaptation of the content. Towards this point we extended the well known html model with a new set of tags; <csl> (cognitive style list) and <csli> (cognitive style list item). A Web Browser (Mozilla Firefox) Extension has been therefore developed in order for the browser to recognize and implement the set of tags for the dynamic reconstruction and adaptation of Web content to the individual characteristics of the users.

The initial results of the system's evaluation have shown that the proposed framework do not degrade the efficiency (in terms of speed and accuracy) in the Web content adaptation process and could be efficiently used for targeting the mass market by encapsulating customers' distinct characteristics. Such a method could be considered nowadays fundamental for the provision of adapted and personalized eServices, via any medium, increasing this way one-to-one service delivery and integrity, enabling businesses to retain their customers and therefore to gain a substantial competitive advantage.

Future and emerging trends include: Further analysis and testing of the current cognitive factors framework with the use of the IBM experimental setting and the automatic content reconstruction approach; a more detailed analysis of the current model as well as the relationship between its different sub-dimensions; further investigation of constraints and challenges arise from the implementation of such issues on mobile devices and channels; study on the structure of the metadata coming from the providers' side, aiming to construct a Web-based personalization architecture that will be based on human factors and will serve as an dynamic personalization filter in different domains and contexts.

Bibliography

[1] Stonebraker M. (2007). One size fits all: A concept whose time has come and gone.

[2] Stonebraker M., Çetintemel U. (2008). "One Size Fits All": An Idea Whose Time Has Come and Gone.

[3] Brown D. (2004). Wave goodbye to the era of one-size-fits all.

[4] Edd Dumbill (2000). The Semantic Web: A Primer, [online], http://www.xml.com/pub/a/2000/11/01/semanticweb/index.html

[5] Panayiotou C., & Samaras G, (2004). mPersona: Personalized Portals for the Wireless User: An Agent Approach, *Journal of ACM/ Baltzer Mobile Networking and Applications (MONET), special issue on "Mobile and Pervasive Commerce"*.

[6] Anderson C., Domingos P., & Weld D. (2001). Personalizing Web sites for mobile users, [online], http://citeseer.ist.psu.edu/444187.html.

[7] Lankhorst M.M., Kranenburg, Salden A., & Peddemors A.J.H. (2002). Enabling Technology for Personalizing Mobile Services, *Proceedings of the 35th Annual Hawaii International Conference on System Sciences* (HICSS-35'02).

[8] Anderson C., Domingos P., & Weld D. (2001). Adaptive Web navigation for wireless devices, [online], http://www.cs.washington.edu/ai/proteus/ijcai01.pdf.

[9] Volokh E. (2000). Personalization and Privacy, *from the Communications of the Association for Computing Machinery*, vol. 43, issue 8, at 84.

[10] cyLEDGE Media (2008). Disentangling Web 2 and Mass Customization.

[11] Germanakos P., Tsianos N., Lekkas Z., Mourlas C., Belk M., Samaras G. (2008). Towards an Adaptive and Personalized Web Interaction using Human Factors. *A chapter to appear in: Advances in Semantic Media Adaptation and Personalization*, Vol.2. M. Angelides (Ed.), Taylor & Francis Group, LLC.

[12] Tsianos N., Lekkas Z., Germanakos P., Mourlas C., & Samaras G. (2008). User-centered Profiling on the basis of Cognitive and Emotional Characteristics: An Empirical Study. *Proceedigns of the 5th International Conference on Adaptive Hypermedia and Adaptive Web-based Systems (AH 2008)*, Hannover, Germany, July 28 - August 1, 2008, Springer-Verlag Berlin Heidelberg, LNCS 5149, pp. 214-223.

[13] Baddeley, A. (1992). Working Memory. *Science, Vol, 255*, 1992, pp: 556 - 559.

[14] Demetriou, A., Efklides, A., and Platsidou, M. (1993). The architecture and dynamics of developing mind: Experiential structuralism as a frame for unifying cognitive development theories. *Monographs of the Society for Research in Child Development, 58* (Serial No. 234), 1993, 5-6.

[15] Cassady J. C., Johnson, R. E. (2002). Cognitive Test Anxiety and Academic Performance. *Contemporary Educational Psychology*, 2002, Vol. 27 No 2, pp. 270-295.

[16] Cassady, J. C. (2004). The influence of cognitive test anxiety across the learning–testing cycle. *Learning and Instruction*, 2004, Vol. 14 No 6, pp. 569-592.

[17] Spielberger, C. D. (1983). Manual for the State-Trait Anxiety Inventory (STAI). *Palo Alto, CA: Consulting Psychologists Press*, 1983.

[18] Cingil I., Dogac A., & Azgin A. (2000). A broader approach to personalization, *Communications of the ACM*, Vol. 43, No. 8.

[19] Blom J. (2000). Personalization - A Taxonomy, *ACM 2000*. ISBN:1-58113-248-4., 2000

[20] Kim W. (2002). Personalization: Definition, Status, and Challenges Ahead. *Published by ETH Zurich, Chair of Software Engineering* JOT, 2002, Vol. 1, No. 1., 2002

[21] Wang J., & Lin J. (2002). Are personalization systems really personal? – Effects of conformity in reducing information overload, *Proceedings of the 36th Hawaii International Conference on Systems Sciences (HICSS'03)*, 0-7695- 1874-5/03, © 2002 IEEE., 2002

[22] De Bra P., Aroyo L., & Chepegin V. (2004). The next big thing: Adaptive Web-based systems, *Journal of Digital Information*, Vol. 5 Issue 1, Article no. 247.

[23] Rossi G., Schwade D., & Guimaraes M.R. (2001). Designing Personalized Web Applications, ACM 1-58113-348-0/01/0005.

[24] Perkowitz M, & Etzioni O. (2003). Adaptive Websites, Towards Adaptive Web sites: Conceptual Framework and case study, [online], http://www.cse.buffalo.edu/ ~sbraynov/ seninar2003/presenttations/adaptive.pdf.

[25] Dagger D., Wade V., & Conlan O. (2003). Towards "anytime, anywhere" learning: The role and realization of dynamic terminal personalization and adaptive eLearning, [online],

http://www.cs.tcd.ie/Owen.Conlan/publications/EdMedia2003v1_Conlan.pdf.

[26] Lankhorst M.M., Kranenburg, Salden A., & Peddemors A.J.H. (2002). Enabling Technology for Personalizing Mobile Services, *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, HICSS-35'02.

[27] Hjelsvold R., Vdaygiri S., & Leaute Y. (2001). Web–based personalization and management of interactive video, ACM 1-58113-348-0/01/0005.

[28] Kaplan B., Farzanfar R., & Freeman H.R. Research and ethical issues arising from ethnographic interviews of patients' reactions to an intelligent interactive telephone health behavior advisor, [online], http://is.lse.ac.uk/Support/ifip_wg82/ StLouis/kaplan.pdf.

[29] Pazzani J. M. (2005). A framework for collaborative, content-based and demographic filtering, [online], http://www.ics.uci.edu/~pazzani/Publications/ AIREVIEW.pdf.

[30] Basilico J., & Hofmann T. (2004). Unifying collaborative and content-based filtering, *Proceedings of the 21st International Conference on Machine Learning*, Banff, Canada.

[31] Sharbanand U., & Maes P. (1998). Social information filtering: Algorithms for automating "Word of Mouth", [online], http://citeseer.ist.psu.edu/ shardanand95social.html.

[32] Shahabi C., & Chen Y. (2003). Web information personalization: Challenges and approaches, [online], http://infolab.usc.edu/DocsDemos/DNIS2003.pdf.

[33] Chaffee J., & Gauch S. (2000). Personal Ontologies for Web Navigation, ACM 2000, 1-58113-320-0/00/11.

[34] Wang J., & Lin J. (2002). Are personalization systems really personal? – Effects of conformity in reducing information overload, *Proceedings of the 36th Hawaii International Conference on Systems Sciences (HICSS'03)*, 0-7695-1874-5/03.

[35] Eirinaki M., & Vazirgiannis M. (2003). Web Mining for Web Personalization, *ACM Transactions on Internet Technology*, Vol. 3, No. 1, Pages 1-27.

[36] Mobasher B., Dai H., Luo T., Nakagawa M., & Wiltshire J. (2000). Discovery of aggregate usage profiles for Web personalization, [online] http://maya.cs.depaul.edu/~mobasher/papers/webkdd2000. pdf.

[37] Pierakkos D., Paliouras G., Papatheodorou C., & Spyropoulos D.C. (2001). KOINOTITES: A web usage mining tool for personalization, [online], http://iit.demokritos.gr/~paliourg/papers/PCHCI2001.pdf.

[38] Delicato F., Pirmez L., & Carmo L. (2001). Fenix – personalized information filtering system for WWW pages, Internet *Research: Electronic Networking Applications and Policy*, Vol. 11, No. 1, pp. 42-48.

[39] Panayiotou C., & Samaras G, (2004). mPersona: Personalized Portals for the Wireless User: An Agent Approach, *Journal of ACM/ Baltzer Mobile Networking and Applications (MONET), special issue on "Mobile and Pervasive Commerce"*.

[40] Barett R., Maglio P., & Kellem D. How to personalize the Web, [online], http://www.raleigh.ibm.com/wbi/wbisoft.htm.

[41] Karat M.C., Brodie C., Karat J., Vergo J., & Alpert S.R. (2003). Personalizing the user experience on ibm.com, *IBM Systems Journal*, Vol. 42, No. 4.

[42] Papagelis M., Plexousakis D., Rousidis D., & Theoharopoulos E. (2004). Qualitative Analysis of User-based and Item-based Prediction Algorithms for Recommendation Systems, CIA 2004: 152-166.

[43] Mobasher B., Dai H., Luo T., Nakagawa M., & Wiltshire J. (2000). Discovery of aggregate usage profiles for Web personalization, [online], http://maya.cs.depaul.edu/ ~mobasher/papers/webkdd2000. pdf.

[44] Mobasher B., Colley R., & Strivastava J. (1997). Automatic Personalization Based on Web Usage Mining, [online], http://maya.cs.depaul.edu/~mobasher/personalization.

[45] Vozalis E., Nicolaou A., & Margaritis K. G. (2001). Intelligent Techniques for Web Applications: Review and Educational Application, *Presented at the Fifth Hellenic-European Conference on Computer Mathematics and its Applications* (HERCMA), Athens, Greece.

[46] Deshpande M., & Karypis G. (2004). Selective Markov models for predicting Web page accesses, *ACM Transactions on Internet Technologies*, Vol. 4, No. 2, Pages 163-184.

[47] Cingil I., Dogac A., & Azgin A. (2000). A broader approach to personalization, *Communications of the ACM*, Vol. 43, No. 8.

[48] Mobasher B., Dai H., Luo T., Sun Y., & Zhu J. (2000). Combining Web usage and Content mining for more effective personalization, [online], http://citeseer.ist.psu.edu/ mobasher00combining.html.

[49] Billsus D., Brunk C., Evans C., Galdish B., & Pazzani M. (2002). Adaptive interfaces for ubiquitous Web access, *Communications of the ACM*, Vol. 45, No. 5.

[50] Bogonicolos N. et al. (1999). ARCHIMIDES: an intelligent agent for adaptive-personilized navigation within a WEB server, *Proceedings of the 32nd Annual Hawaii Intl. Conf. On System Science*, HICSS-32. Vol 5.

[51] Anderson C. et.al. (2001). Personilizing Web Sites for Mobile Users, *Proceedings of the 10th Conference on the World Wide Web,* 2001.

[52] Maglio, P. & Barret, R. (2000). Intermediaries Personalize Information Streams, *Communications of the ACM,* Vol. 43(8), pp. 96-101.

[53] Barret, R., Maglio, P. & Kellem, D. (1997). How to Personalize the Web, Proc. CHI 97.

[54] Thomas, C. & Fischer, G., (1997). Using agents to personalize the Web, *Proceedings ACM* , IUI'97, pp. 53 60, Florida Usa.

[55] Papanikolaou K.A., Grigoriadou M., Kornilakis H., & Magoulas G.D. (2003). Personalizing the Interaction in a Web-based Educational Hypermedia System: the case of INSPIRE, *User-Modeling and User-Adapted Interaction*, 13(3), 213-267.

[56] Weber, G. & Specht, M. (1997). User Modeling and Adaptive Navigation Support in WWW-Based Tutoring Systems. *Proceedings of User Modeling '97,* pp. 289-300.

[57] De Bra, P. & Calvi, L. (1998). AHA! An open Adaptive Hypermedia Architecture. *The New Review of Hypermedia and Multimedia,* 4, Taylor Graham Publishers, pp. 115-139.

[58] Brusilovsky, P., Eklund, J. & Schwarz, E. (1998). Web-based education for all: A tool for developing adaptive courseware. Computer Networks and ISDN Systems. *Proceedings of the 7th International WWW Conference*, 14-18 April, 30(1-7), 291-300.

[59] Bloor Research (2001), BroadVision One-To-One Product Evaluation, [online], http://www.bloor research.com/research/product_evaluation/355/
broadvision_one_to_one.html

[60] Niral S., Ruopeng Y., Understanding Microsoft Passport, [online],
www.simson.net/ref/2004/csg357/projects/shah-ye.doc

[61] Windows XP for dummies, Understanding Microsoft .NET Passport, [online], http://www.dummies.com/WileyCDA/DummiesArticle/id-349,subcat-USING.html

[62] Glascock S., Macromedia Upgrades LikeMinds, Aria, [online],
http://www.techweb.com/wire/story/TWB20000619S002

[63] IBM, WebSphere Personalization, [online],
http://www-306.ibm.com/software*/genservers/personalization/*

[64] Carro R.M., Pulido E., Pilar R., TANGOW: Task-based Adaptive learNer Guidance on the WWW

[65] Mitrovic A., Brent M., Evaluating the Effects of Open Student Models on Learning

[66] Neumann G., Zirvas J., "SKILL: A Scalable Internet-Based Teaching and Learning System", *Proceedings of WebNet 98 World Conference of the WWW*, Internet & Intranet, pp. 688-693. Orlando, Florida, November 7-12, 1998.

[67] Mason, M.S., Hacker, L.K., (2003). Applying Communication Theory to Digital Divide Research. *IT & Society, Volume 1, Issue 5*, Summer 2003, pp 40-55.

[68] Adomavicious, G., and Tuzhilin, A. (1999). User Profiling in Personalization Applications through Rule Discovery and Validation. *Proceedings of the ACM Fifth International Conference on Data Mining and Knowledge Discovery (KDD'99)*, 377-381.

[69] Graber, D., A. (2000). Processing Politics. *Chapter 2*, The University of Chicago Press.

[70] McLoughlin, C. (1999). The implications of the research literature on learning styles for the design of instructional material. *Australian Journal of Educational Technology*, 15(3), p. 222-241.

[71] Pithers, R., T. (2002). Cognitive Learning Style: a review of the field dependent- field independent
approach, *Journal of Vocational Education and Training*, Volume 54, Number 1, p. 117-8.

[72] Yuliang, L. and Dean, G. (1999). Cognitive Styles and Distance Education, *Online Journal of Distance Learning Administration*, Volume II, Number III.

[73] Ayersman, D., J. and Reed, W., M. 1998. Relationships among hypermedia-based mental models and hypermedia knowledge. *Journal of Research on Computing in Education*, 30 (3), pp.222-238.

[74] Germanakos P., Tsianos N., Lekkas Z., Mourlas C., & Samaras G. (2007). Capturing Essential Intrinsic User Behaviour Values for the Design of Comprehensive Web-based Personalized Environments, *Computers in Human Behavior Journal, Special Issue on Integration of Human Factors in Networked Computing*, doi:10.1016/j.chb.2007.07.010.

[75] Germanakos P., Tsianos N., Lekkas Z., Mourlas C., Belk M., Samaras G. (2007). A Semantic Approach of an Adaptive and Personalized Web-based Learning Content - The case of AdaptiveWeb, *Proceedings of the 2nd International Workshop on Semantic Media Adaptation and Personalization (SMAP 2007)*, London, U.K., December 17-18, IEEE Computer Society, pp. 68-73.

[76] Graber, D., A. (2000). Processing Politics. *Chapter 2*, The University of Chicago Press.

[77] McLoughlin, C. (1999). The implications of the research literature on learning styles for the design of instructional material. *Australian Journal of Educational Technology*, 15(3), p. 222-241.

[78] Pithers, R., T. (2002). Cognitive Learning Style: a review of the field dependent- field independent approach, *Journal of Vocational Education and Training*, Volume 54, Number 1, p. 117-8.

[79] Yuliang, L. and Dean, G. (1999). Cognitive Styles and Distance Education, *Online Journal of Distance Learning Administration*, Volume II, Number III.

[80] Reed, W., M., Ayersman, D., J. and Liu, M. (1996). The effects of students' computer-based prior experiences and instructional exposures on the application of hypermedia-related mental models. *Journal of Educational Computing Research*, 14 (2), pp. 175-187.

[81] Ayersman, D., J. and Reed, W., M. (1998). Relationships among hypermedia-based mental models and hypermedia knowledge. *Journal of Research on Computing in Education*, 30 (3), pp.222-238.

[82] Riding R. (2001). Cognitive Style Analysis – Research Administration, *Published by Learning and Training Technology*.

[83] Felder, R.M., Silverman, L.K. (1988). Learning and Teaching Styles in Engineering Education, *Engineering Education 78*: 674-681.

[84] Witkin H., Moore C., Gooddenough D., Cox P. (1977). Field- dependent and field-independent cognitive styles and their educational implications, *Review of Educational Research 47*, pp 1-64

[85] Kolb, A.Y., and Kolb, D.A. (2005). The Kolb Learning Style Inventory. *Technical Specifications, Experience Based Learning Systems, Inc.*, Version 3.1 2005.

[86] De Bra P., Aroyo L., & Chepegin V. (2004). The next big thing: Adaptive Web-based systems, *Journal of Digital Information*, Vol. 5 Issue 1, Article no. 247.

[87] Noy N., Sintek M., Decker S., Crubézy M., Fergerson R., & Musen M. (2001). Creating Semantic Web Contents with Protégé-2000, Stanford University

[88] Kalyanpur A., Parsia B., Hendler J. (2005a). A Tool for Working with Web Ontologies. *In Proceedings of the International Journal on Semantic Web and Information Systems, Vol.1, No.1, Jan-Mar 2005*

[89] Kalyanpur A., Parsia B., Sirin E., Cuenca-Grau B., James Hendler (2005b). Swoop: A 'Web' Ontology Editing Browser. *Journal of Web Semantics* Vol 4(2), 2005

[90] Germanakos P., Mourlas C., Isaia C., & Samaras G. (2005). An Optimized Review of Adaptive Hypermedia and Web Personalization - Sharing the Same Objective, *Proceedings of*

the 1st International Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces (WPRSIUI 2005) of the 2nd International Conference on E-business and Telecommunications Networks (ICETE2005), Reading, October 3-7, 2005, pp. 43-48.

[91] Germanakos P., Mourlas C., Panayiotou C., & Samaras G. (2005). Personalization Systems and Processes Review based on a Predetermined User Interface Categorization, *Proceedings of the III International Conference on Communication and Reality, Digital Utopia in the media: from discourses to facts*. A balance, Barcelona, May 20-21, 2005, pp. 431-444.

[92] Lekkas, Z., Tsianos, N., Germanakos, P., Mourlas, C., and Samaras, G. (2008). The Role of Emotions in the Design of Personalized Educational Systems. *Proceedings of the 8th IEEE International Conference on Advanced Learning Technologies (ICALT 2008)*,

Appendix A

Appendix A consists of important code used for the implementation of the Firefox browser plug-in.

Chrome Manifest files

```
content     smarTag    chrome/content/ contentaccessible=yes

locale    smarTag   en-US   chrome/locale/en-US/smarTag/

overlay chrome://browser/content/browser.xul chrome://smarTag/content/browser.xul
overlay chrome://browser/content/preferences/preferences.xul
chrome://smarTag/content/preferences.xul
```

RDF file for installing the plugin to the Firefox Browser

```
<?xml version="1.0"?>

<RDF xmlns="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:em="http://www.mozilla.org/2004/em-rdf#">

  <Description about="urn:mozilla:install-manifest">
    <em:id>belk@cs.ucy.ac.cy</em:id>
    <em:version>0.1</em:version>
    <em:type>2</em:type>

    <!-- Target Application this extension can install into,
         with minimum and maximum supported versions. -->
    <em:targetApplication>
      <Description>
        <em:id>{ec8030f7-c20a-464f-9b0e-13a3a9e97384}</em:id>
        <em:minVersion>1.5</em:minVersion>
        <em:maxVersion>3.0.*</em:maxVersion>
      </Description>
    </em:targetApplication>

    <!-- Front End MetaData -->
    <em:name>smarTag</em:name>
    <em:description>A Firefox Extension to support the AdaptiveWeb
```

```
Framework</em:description>
    <em:creator>Mario R. Belk</em:creator>
    <em:homepageURL>http://www3.cs.ucy.ac.cy/adaptiveweb</em:homepageURL>
  </Description>
</RDF>
```

Localization file for multi-lingual support of the Plug-in

```
<!ENTITY  menuitem.label             "HP Scheduler Options...">
<!ENTITY  statusbar.tooltip.default  "Nothing Scheduled">
<!ENTITY  prefpane.label             "HP Scheduler">
<!ENTITY  locations.label            "Location(s):">
<!ENTITY  useCurrentPage.label       "Use Current Page">
<!ENTITY  useBookmark.label          "Use Bookmark...">
<!ENTITY  useBlankPage.label         "Use Blank Page">
<!ENTITY  groupbox.defaultHomepage   "Default Home Page">
<!ENTITY  groupbox.scheduledPages    "Scheduled Home Pages">
<!ENTITY  dateColumn                 "Date">
<!ENTITY  startColumn                "Start Time">
<!ENTITY  endColumn                  "End Time">
<!ENTITY  locationsColumn            "Location(s)">
<!ENTITY  newSchedule.button         "New Schedule">
<!ENTITY  editSchedule.button        "Edit Schedule">
<!ENTITY  deleteSchedule.button      "Delete Schedule">
<!ENTITY  up.label                   "Up">
<!ENTITY  down.label                 "Down">
<!ENTITY  groupbox.editSchedule      "Schedule Editor">
<!ENTITY  groupbox.timeRange         "Time Range">
<!ENTITY  radio.timeRange            "Between the following times:">
<!ENTITY  timeRange.to               " to ">
<!ENTITY  radio.allDay               "All day">
<!ENTITY  groupbox.dateRange         "Date Range">
<!ENTITY  radio.oneDay               "Only on the following date:">
<!ENTITY  radio.daily                "Daily">
<!ENTITY  saveSchedule.label         "Save Schedule">
<!ENTITY  cancelSchedule.label       "Cancel Schedule">
<!ENTITY  month.1                    "January">
<!ENTITY  month.2                    "February">
<!ENTITY  month.3                    "March">
<!ENTITY  month.4                    "April">
<!ENTITY  month.5                    "May">
<!ENTITY  month.6                    "June">
<!ENTITY  month.7                    "July">
<!ENTITY  month.8                    "August">
<!ENTITY  month.9                    "September">
<!ENTITY  month.10                   "October">
<!ENTITY  month.11                   "November">
<!ENTITY  month.12                   "December">
<!ENTITY  gender.male                "Male">
```

```
<!ENTITY  gender.female              "Female">
<!ENTITY  profession.privateSector   "Private Sector">
<!ENTITY  profession.publicSector    "Public Sector">
<!ENTITY  profession.academic            "Academic">
<!ENTITY  profession.1stYearStudent  "1st Year Student">
<!ENTITY  profession.2ndYearStudent  "2nd Year Student">
<!ENTITY  profession.3rdYearStudent  "3rd Year Student">
<!ENTITY  profession.4thYearStudent  "4th Year Student">
<!ENTITY  profession.4PlusYearStudent "4+ Year Student">
<!ENTITY  profession.mscStudent         "MSc Student">
<!ENTITY  profession.phdStudent         "PhD Student">
<!ENTITY  profession.other                 "Other">
<!ENTITY  level.low                         "Low">
<!ENTITY  level.medium                      "Medium">
<!ENTITY  level.high                        "High">
```

XUL file for extending the Firefox browser's interface

```
<?xml version="1.0"?>
<overlay id="smarTag_browser_overlay"
xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">

  <script type="application/x-javascript" src="chrome://browser/content/utilityOverlay.js" />
  <script type="application/x-javascript" src="chrome://smarTag/content/test.js" />

  <!-- Add the smarTag icon to the bottom right of screen -->
  <statusbar id="status-bar">
    <statusbarpanel class="statusbarpanel-iconic" id="smarTag_sbi"
                                    tooltiptext="smarTag Options"
onclick="openPreferences('smarTagOptionsPane');"
             src="chrome://smarTag/content/images/smarTag18.png" />
  </statusbar>

 <!-- Create a smarTag Options menu item -->
  <menupopup id="menu_ToolsPopup">
    <menuitem label="smarTag Options..."
onclick="openPreferences('smarTagOptionsPane');" />
  </menupopup>
</overlay>
```

XUL file for extending the Firefox preferences' interface

```
<?xml version="1.0"?>

<!-- Localization Information -->
```

```
<!DOCTYPE overlay SYSTEM "chrome://smarTag/locale/smarTag.dtd">

<overlay id="smarTag_preferences_overlay"
xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">

<!-- Merge with the BrowserPreferences Window -->
<prefwindow id="BrowserPreferences">
	<script type="application/x-javascript" src="chrome://smarTag/content/test.js" />
	<script type="application/x-javascript"
src="chrome://browser/content/utilityOverlay.js" />

	<!-- Create a new pane (tab) for HP Scheduler. -->
	<prefpane id="smarTagOptionsPane" label="smarTag Options"
onpaneload="loadDates();" image="chrome://smarTag/content/images/smarTag32.png">

		<preferences>
	<preference id="prefUsername" name="extensions.smarTag.username" type="string"
/>
			<preference id="prefGender" name="extensions.smarTag.gender"
type="int" />
			<preference id="prefProfession"
name="extensions.smarTag.profession" type="int" />
			<preference id="prefComputerKnowledge"
name="extensions.smarTag.computerKnowledge" type="int" />
			<preference id="prefBirthDay"
name="extensions.smarTag.dateOfBirth.day" type="int" />
			<preference id="prefBirthMonth"
name="extensions.smarTag.dateOfBirth.month" type="int" />
			<preference id="prefBirthYear"
name="extensions.smarTag.dateOfBirth.year" type="int" />
			<preference id="prefImagerVerbalizer"
name="extensions.smarTag.imagerVerbalizer" type="int" />
			<preference id="prefWholistAnalyst"
name="extensions.smarTag.wholistAnalyst" type="int" />
			<preference id="prefWorkingMemory"
name="extensions.smarTag.workingMemory" type="int" />
			<preference id="prefAnxiety" name="extensions.smarTag.anxiety"
type="int" />
			<preference id="prefEnableSensor"
name="extensions.smarTag.enableSensor" type="bool" />
			<preference id="prefDoAdapt" name="extensions.smarTag.doAdapt"
type="bool" />
    </preferences>

		<groupbox id="gpbGeneral">
			<caption label="General Information" />
			<vbox>
				<hbox align="center">
					<label control="lblUsername" style="width:120px;"
value="Username:"/>
					<textbox id="txtUsername"
preference="prefUsername" preference-editable="true" />
				</hbox>
```

```
						<hbox align="center">
							<label control="lblGender" style="width:120px;"
value="Gender:"/>
							<menulist id="mnulGender"
preference="prefGender" preference-editable="true">
								<menupopup>
									<menuitem label="&gender.male;"
value="0" />
									<menuitem
label="&gender.female;" value="1" />
								</menupopup>
							</menulist>
						</hbox>

						<hbox align="center">
							<label control="lblProfession" style="width:120px;"
value="Profession:"/>
							<menulist id="mnulProfession"
preference="prefProfession" preference-editable="true">
								<menupopup>
									<menuitem
label="&profession.privateSector;" value="0" />
									<menuitem
label="&profession.publicSector;" value="1" />
									<menuitem
label="&profession.academic;" value="2" />
									<menuitem
label="&profession.1stYearStudent;" value="3" />
									<menuitem
label="&profession.2ndYearStudent;" value="4" />
									<menuitem
label="&profession.3rdYearStudent;" value="5" />
									<menuitem
label="&profession.4thYearStudent;" value="6" />
									<menuitem
label="&profession.4PlusYearStudent;" value="7" />
									<menuitem
label="&profession.mscStudent;" value="8" />
									<menuitem
label="&profession.phdStudent;" value="9" />
									<menuitem
label="&profession.other;" value="10" />
								</menupopup>
							</menulist>
						</hbox>

						<hbox align="center">
							<label control="lblComputerKnowledge"
style="width:120px;" value="Computer Knowledge:"/>
							<menulist id="mnulComputerKnowledge"
preference="prefComputerKnowledge" preference-editable="true">
								<menupopup>
									<menuitem label="&level.low;"
value="0" />
```

```
                                        <menuitem label="&level.medium;"
value="1" />
                                        <menuitem label="&level.high;"
value="2" />
                                    </menupopup>
                                </menulist>
                            </hbox>

                            <hbox align="center">
                                <label control="lblDateOfBirth"
style="width:120px;" value="Date of Birth:"/>
                                    <menulist id="mnulBirthDay"
preference="prefBirthDay" preference-editable="true">
                                        <menupopup></menupopup>
                                    </menulist>

                                    <menulist id="mnulBirthMonth"
preference="prefBirthMonth" preference-editable="true">
                                        <menupopup>
                                            <menuitem
label="&month.1;" value="1" />
                                            <menuitem
label="&month.2;" value="2" />
                                            <menuitem
label="&month.3;" value="3" />
                                            <menuitem
label="&month.4;" value="4" />
                                            <menuitem
label="&month.5;" value="5" />
                                            <menuitem
label="&month.6;" value="6" />
                                            <menuitem
label="&month.7;" value="7" />
                                            <menuitem
label="&month.8;" value="8" />
                                            <menuitem
label="&month.9;" value="9" />
                                            <menuitem
label="&month.10;" value="10" />
                                            <menuitem
label="&month.11;" value="11" />
                                            <menuitem
label="&month.12;" value="12" />
                                        </menupopup>
                                    </menulist>

                                    <menulist id="mnulBirthYear"
preference="prefBirthYear" preference-editable="true">
                                        <menupopup></menupopup>
                                    </menulist>
                            </hbox>
                        </vbox>
                    </groupbox>
                    <groupbox id="gpbAdaptiveWebProfile">
```

```
                    <caption label="AdaptiveWeb Profile"/>
                    <vbox>
                        <hbox align="center">
                            <label control="lblImagerVerbalizer"
style="width:120px;" value="Imager / Verbalizer"/>
                            <menulist id="mnulImagerVerbalizer"
preference="prefImagerVerbalizer" preference-editable="true">
                                <menupopup>
                                    <menuitem label="Imager"
value="0" />
                                    <menuitem label="Intermediate"
value="1" />
                                    <menuitem label="Verbalizer"
value="2" />
                                </menupopup>
                            </menulist>
                        </hbox>
                        <hbox align="center">
                            <label control="lblWholistAnalyst"
style="width:120px;" value="Wholist / Analyst:"/>
                            <menulist id="mnulWholistAnalyst"
preference="prefWholistAnalyst" preference-editable="true">
                                <menupopup>
                                    <menuitem label="Analyst"
value="0" />
                                    <menuitem label="Intermediate"
value="1" />
                                    <menuitem label="Wholist"
value="2" />
                                </menupopup>
                            </menulist>
                        </hbox>
                        <hbox align="center">
                            <label control="lblWorkingMemory"
style="width:120px;" value="Working Memory:"/>
                            <menulist id="mnulWorkingMemory"
preference="prefWorkingMemory" preference-editable="true">
                                <menupopup>
                                    <menuitem label="&level.low;"
value="0" />
                                    <menuitem label="&level.medium;"
value="1" />
                                    <menuitem label="&level.high;"
value="2" />
                                </menupopup>
                            </menulist>
                        </hbox>
                        <hbox align="center">
                            <label control="lblAnxiety" style="width:120px;"
value="Anxiety Level:"/>
                            <menulist id="mnulAnxiety"
preference="prefAnxiety" preference-editable="true">
                                <menupopup>
                                    <menuitem label="&level.low;"
```

```
value="0" />
                                                        <menuitem label="&level.medium;"
value="1" />
                                                        <menuitem label="&level.high;"
value="2" />
                                                </menupopup>
                                        </menulist>
                                        <checkbox id="chkEnableSensor" label="Use
Sensor Metrics" preference="prefEnableSensor" preference-editable="true" />
                                </hbox>
                        </vbox>
                </groupbox>
                <groupbox id="gpbOtherOptions">
                        <caption label="Other"/>
                        <checkbox label="Adapt Web Content" checked="true"
preference="prefDoAdapt" preference-editable="true" />
                </groupbox>
                <vbox align="right">
                        <hbox align="center">
                                <button label="Import Settings" accesskey="I"
onclick="loadXMLProfile();" />
                        </hbox>
                </vbox>
        </prefpane>
</prefwindow>
</overlay>
```

Functions responsible for the parsing of HTML document and adaptation process

```
var doc;
var divFloatMenu;
var WorkingMemory;
var anxietySensorTimer;

function loadDates() {
        // Gain access to the Preferences service
        var prefManager = Components.classes["@mozilla.org/preferences-service;1"]
                  .getService(Components.interfaces.nsIPrefBranch);
        var mnulBirthDay = document.getElementById("mnulBirthDay");
        var mnulBirthYear = document.getElementById("mnulBirthYear");
        var date = new Date();
        var year = date.getFullYear();

        for (var i = 1; i <= 31; i++)
                mnulBirthDay.appendItem(i, i);

        for (var j = 1900; j <= year; j++)
                mnulBirthYear.appendItem(j, j);
```

```
		mnulBirthDay.selectedIndex = 
prefManager.getIntPref("extensions.smarTag.dateOfBirth.day") - 1;
		mnulBirthYear.selectedIndex = 
prefManager.getIntPref("extensions.smarTag.dateOfBirth.year") - 1900;
}

window.addEventListener("load", function() { myExtension.init(); }, false);

var myExtension = {
  init: function() {
    var appcontent = document.getElementById("appcontent");   // browser
    if(appcontent)
      appcontent.addEventListener("DOMContentLoaded", myExtension.onPageLoad, true);
    var messagepane = document.getElementById("messagepane"); // mail
    if(messagepane)
      messagepane.addEventListener("load", function () { myExtension.onPageLoad(); }, true);
  },

  onPageLoad: function(aEvent) {
		// Gain access to the Preferences service
		var prefManager = Components.classes["@mozilla.org/preferences-
service;1"].getService(Components.interfaces.nsIPrefBranch);
		var DoAdapt = prefManager.getBoolPref("extensions.smarTag.doAdapt");
		var AnxietyLevel = prefManager.getIntPref("extensions.smarTag.anxiety");
		var ImagerVerbalizer = 
prefManager.getIntPref("extensions.smarTag.imagerVerbalizer");
		var WholistAnalyst = prefManager.getIntPref("extensions.smarTag.wholistAnalyst");
		var IsImager = false;

		WorkingMemory = prefManager.getIntPref("extensions.smarTag.workingMemory");

		// Initialize Float Menu
		divFloatMenu = null;

		if (ImagerVerbalizer == 0 || ImagerVerbalizer == 1) // Either imager or intermediate
			IsImager = true;

    doc = aEvent.originalTarget; // doc is document that triggered "onload" event
    // do something with the loaded page.

		if(DoAdapt) {
			activateAnxietyTimer();
			if (AnxietyLevel == 0) {
				var sensNodes = doc.getElementsByTagName("sens");

				for (var j = 0; j < sensNodes.length; j++) {
					var hashSENSAttributes = new Object();
					hashSENSAttributes = retrieveSENSAttributes(sensNodes, j, 
hashSENSAttributes);
					var newSpan = doc.createElement("span");
					if (hashSENSAttributes["font-weight"] == "bold") 
newSpan.innerHTML = "<b>" + sensNodes[j].innerHTML + "</b>";
					if (hashSENSAttributes["font-size"] != null) 
newSpan.style.fontSize = hashSENSAttributes["font-size"];
```

```
					else newSpan.innerHTML = sensNodes[j].innerHTML;
					sensNodes[j].innerHTML = "";
					newSpan.style.color = hashSENSAttributes["font-color"];
					// Append new adapted element
					sensNodes[j].appendChild(newSpan);
				}
			}

			var cslNodes = doc.getElementsByTagName("csl");
			var cslID;

			for (var i = 0; i < cslNodes.length; i++) {
				var hashCSLAttributes = new Object();

				hashCSLAttributes = retrieveCSLAttributes(cslNodes, i,
hashCSLAttributes);

				var csliNodes = cslNodes[i].getElementsByTagName("csli");
				retrieveCSLChildElements(cslNodes[i], csliNodes, hashCSLAttributes,
WholistAnalyst, IsImager);
			}
		}
  }
}

// Retrieve all the attributes of the SENS tag
function retrieveSENSAttributes(sensNode, nodeIndex, hashSENSAttributes) {
		for (var j = 0; j < sensNode[nodeIndex].attributes.length; j++) {
			switch(sensNode[nodeIndex].attributes[j].nodeName.toLowerCase()) {
				case "font-color":
					hashSENSAttributes["font-color"] =
sensNode[nodeIndex].attributes[j].nodeValue;
					break;
				case "font-weight":
					hashSENSAttributes["font-weight"] =
sensNode[nodeIndex].attributes[j].nodeValue;
					break;
				case "font-size":
					hashSENSAttributes["font-size"] =
sensNode[nodeIndex].attributes[j].nodeValue;
					break;
			}
		}
		return hashSENSAttributes;
}

// Retrieve all the attributes of the CSL tag
function retrieveCSLAttributes(cslNode, nodeIndex, hashCSLAttributes) {
		for (var j = 0; j < cslNode[nodeIndex].attributes.length; j++) {
			switch(cslNode[nodeIndex].attributes[j].nodeName.toLowerCase()) {
				case "id":
					hashCSLAttributes["id"] =
cslNode[nodeIndex].attributes[j].nodeValue;
					break;
```

```
				case "diagramwidth":
					hashCSLAttributes["diagramWidth"] = 
cslNode[nodeIndex].attributes[j].nodeValue;
					break;
				case "floateralign":
					hashCSLAttributes["floaterAlign"] = 
cslNode[nodeIndex].attributes[j].nodeValue;
					break;
				case "floaterbg":
					hashCSLAttributes["floaterBG"] = 
cslNode[nodeIndex].attributes[j].nodeValue;
					break;
			}
		}
	return hashCSLAttributes;
}

// Retrieve all the child elements of the CSL tag
function retrieveCSLChildElements(cslNode, csliNodes, hashCSLAttributes, cognitiveStyle, 
IsImager) {
	var cslID =  hashCSLAttributes["id"];
	var hashCSLIAttributes = new Object();
	// Gain access to the Prefences service
	var prefManager = Components.classes["@mozilla.org/preferences-
service;1"].getService(Components.interfaces.nsIPrefBranch);

	if (cognitiveStyle == 0) { // Analyst
		var j = csliNodes.length;
		// Create new div for csl content menu
		var divID = "contentMenu" + cslID;
		var divContentMenu = doc.createElement("div");

		divContentMenu.setAttribute("id", divID);

		// Create diagram for content menu
		if (IsImager) {
			divContentMenu.setAttribute("class", "diagramContent");
			if(hashCSLAttributes["diagramWidth"] != null)
				divContentMenu.style.width = 
hashCSLAttributes["diagramWidth"] + "px";
		}
		else
			divContentMenu.setAttribute("class", "contentMenu");

		// Go through all csli nodes
		while (j > 0) {
			// Create content menu item for each csli element
			var anchorID = "anchor" + j;
			var anchorContentMenuItem = doc.createElement("a");
			// Assign attributes
			anchorContentMenuItem.setAttribute("id", anchorID);
			anchorContentMenuItem.setAttribute("content", 
csliNodes[0].innerHTML);
			anchorContentMenuItem.setAttribute("href", "javascript:");
```

```
				anchorContentMenuItem.setAttribute("onclick", "showDiagram('" +
anchorID + "', 'divContent" + cslID + "', '" + hashCSLAttributes["id"] + "', '" +
hashCSLAttributes["diagramWidth"] + "'," + IsImager + ");");
				hashCSLIAttributes = retrieveCSLIAttributes(csliNodes[0])
				var csliName = hashCSLIAttributes["name"];
				anchorContentMenuItem.innerHTML = csliName;
				divContentMenu.appendChild(anchorContentMenuItem);
				if (hashCSLIAttributes["summary"] != null &&
hashCSLIAttributes["summaryTitle"] != null && WorkingMemory == 0) {
					// Add to myNotepad link
					var anchorMyNotepad = doc.createElement("a");
					var ip =
prefManager.getCharPref("extensions.smarTag.username"); //new
java.net.InetAddress.getLocalHost();
					anchorMyNotepad.setAttribute("href", "javascript:");
					anchorMyNotepad.setAttribute("onclick",
"window.open('http://www3.cs.ucy.ac.cy/adaptivewebplus/smarTag/tools/myNotepad.aspx?sess
ion=" + ip + "&title=" + hashCSLIAttributes["summaryTitle"] + "&summary=" +
hashCSLIAttributes["summary"] + "', 'bookmarksPage', 'menubar=no,
resizable=yes,scrollbars=yes,status=yes,titlebar=no,toolbar=no,width=700px,height=460px,left
=230px,top=130px');");
					anchorMyNotepad.innerHTML = " <img
src='images/kedit.png' border='0' alt='' />";
					divContentMenu.appendChild(anchorMyNotepad);
					/////////////////////////
				}
				j--;

				if (IsImager)
					divContentMenu.appendChild(doc.createElement("br"));
				else if (!IsImager && j != 0) { // Not imager then create menu with
separators '|'
					var newSeparator = doc.createElement("span");
					newSeparator.innerHTML = " | ";
					divContentMenu.appendChild(newSeparator);
				}
				// Append new adapted element
				cslNode.appendChild(divContentMenu);
				// Remove previous csli element
				cslNode.removeChild(csliNodes[0]);
			}

			// Add an arrow to the diagrams
			if (IsImager) {
				var imgArrow = doc.createElement("img");
				imgArrow.setAttribute("src", "images/arrowBig.png");
				imgArrow.setAttribute("class", "arrow");
				cslNode.appendChild(imgArrow);
			}

			// Create div element for each menu item's content to be displayed
			var divContent = doc.createElement("div");
			divContent.setAttribute("id", "divContent" + cslID);
			cslNode.appendChild(divContent);
```

```
            }
            else if (cognitiveStyle == 1 || cognitiveStyle == 2) { // Intermediate OR Wholist
                        var j = csliNodes.length;
                        var titleCounter = 1;
                        var separator;
                        // Create new div for csl content menu
                        var divID = "contentMenu";
                        var divContentMenu = doc.createElement("div");
                        divContentMenu.setAttribute("class", "contentMenu");

                        if (cognitiveStyle == 1) { // Intermediate
                                    var anchorMenuTop = doc.createElement("a");
                                    anchorMenuTop.setAttribute("name", "top" + cslID);
                                    cslNode.appendChild(anchorMenuTop);
                                    separator = " | ";
                        }
                        else { // Wholist
                                    // If float menu already exists just append the new csli items to the
previous ones
                                    if (divFloatMenu != null)
                                                divContentMenu = divFloatMenu;
                                    divContentMenu.setAttribute("class", "floatDiv");
                                    divContentMenu.style.background = hashCSLAttributes["floaterBG"];
                                    if (hashCSLAttributes["floaterAlign"] == "right")
                                                divContentMenu.style.right = "10px";
                                    else if (hashCSLAttributes["floaterAlign"] == "left")
                                                divContentMenu.style.left = "10px";
                                    separator = "<br />"
                        }

                        cslNode.appendChild(divContentMenu);
                        divFloatMenu = divContentMenu;

                        // Go through all csli nodes
                        while (j > 0) {
                                    var divID = "contentMenu" + j + cslID;
                                    var divWholeContent = doc.createElement("div");
                                    divWholeContent.setAttribute("id", divID);
                                    if (IsImager) {
                                                divWholeContent.setAttribute("class", "diagramContent");
                                                if(hashCSLAttributes["diagramWidth"] != null)
                                                            divWholeContent.style.width =
hashCSLAttributes["diagramWidth"] + "px";
                                    }

                                    var anchorID = "anchor" + j + cslID;
                                    var anchorContentMenuItem = doc.createElement("a");
                                    anchorContentMenuItem.setAttribute("name", anchorID);
                                    if (!IsImager)
                                                anchorContentMenuItem.setAttribute("class", "contentMenu");
                                    hashCSLIAttributes = retrieveCSLIAttributes(csliNodes[0]);
                                    var csliName = hashCSLIAttributes["name"];
                                    anchorContentMenuItem.innerHTML = "<b>" + titleCounter + ". " +
csliName + "</b>";
```

```
                              // Add the content menu titles
                              if (j - 1 > 0) {
                                        if (cognitiveStyle == 2) // Wholist
                                                  divContentMenu.innerHTML += separator + "<b>" +
titleCounter + ". <a href='#" + anchorID + "'>" + csliName + "</a></b>";
                                        else // Intermediate
                                                  divContentMenu.innerHTML += "<b>" + titleCounter
+ ". <a href='#" + anchorID + "'>" + csliName + "</a></b>" + separator;
                              }
                              else {
                                        if (cognitiveStyle == 2) // Wholist
                                                  divContentMenu.innerHTML += separator + "<b>" +
titleCounter + ". <a href='#" + anchorID + "'>" + csliName + "</a></b>" + separator +
separator;
                                        else // Intermediate
                                                  divContentMenu.innerHTML += "<b>" + titleCounter
+ ". <a href='#" + anchorID + "'>" + csliName + "</a></b>";
                              }
                              divWholeContent.appendChild(anchorContentMenuItem);
                              if (hashCSLIAttributes["summary"] != null &&
hashCSLIAttributes["summaryTitle"] != null && WorkingMemory == 0) {
                                        // Add to myNotepad link
                                        var anchorMyNotepad = doc.createElement("a");
                                        var ip =
prefManager.getCharPref("extensions.smarTag.username");//new
java.net.InetAddress.getLocalHost();
                                        anchorMyNotepad.setAttribute("href", "javascript:");
                                        anchorMyNotepad.setAttribute("onclick",
"window.open('http://www3.cs.ucy.ac.cy/adaptivewebplus/smarTag/tools/myNotepad.aspx?sess
ion=" + ip + "&title=" + hashCSLIAttributes["summaryTitle"] + "&summary=" +
hashCSLIAttributes["summary"] + "', 'bookmarksPage', 'menubar=no,
resizable=yes,scrollbars=yes,status=yes,titlebar=no,toolbar=no,width=700px,height=460px,left
=230px,top=130px');");
                                        anchorMyNotepad.innerHTML = " <img
src='images/kedit.png' border='0' alt='' />";
                                        divWholeContent.appendChild(anchorMyNotepad);
                                        ////////////////////////
                              }
                              divWholeContent.appendChild(doc.createElement("br"));

                              // Append the actual content of the csli element
                              var divContent = doc.createElement("div");
                              divContent.setAttribute("class", "actualContent");
                              divContent.innerHTML = csliNodes[0].innerHTML;
                              divWholeContent.appendChild(divContent);

                              // Append a 'Back to Top' link for each content element
                              if (cognitiveStyle == 1) { // Intermediate
                                        var anchorBackToTop = doc.createElement("div");
                                        anchorBackToTop.setAttribute("align", "right");
                                        anchorBackToTop.innerHTML = "<a href='#top" + csIID +
"'>Back to Top</a>";
                                        divWholeContent.appendChild(anchorBackToTop);
                              }
```

```
                        // Append whole csli element
                        cslNode.appendChild(divWholeContent);
                        // Remove previous csli element
                        cslNode.removeChild(csliNodes[0]);

                        j--;
                        titleCounter++;

                        // Add an arrow to the diagrams, except the last one
                        if (IsImager && j != 0) {
                                var imgArrow = doc.createElement("img");
                                imgArrow.setAttribute("src", "images/arrowBig.png");
                                imgArrow.setAttribute("class", "arrow");
                                cslNode.appendChild(imgArrow);
                        }
                }
        }
}

// Retrieve all the attributes of the CSLI tag
function retrieveCSLIAttributes(csliNode) {
        var hashCSLIAttributes = new Object();
        for (var j = 0; j < csliNode.attributes.length; j++) {
                switch(csliNode.attributes[j].nodeName.toLowerCase()) {
                        case "name":
                                hashCSLIAttributes["name"] =
csliNode.attributes[j].nodeValue;
                                break;
                        case "summary":
                                hashCSLIAttributes["summary"] =
csliNode.attributes[j].nodeValue;
                                break;
                        case "summarytitle":
                                hashCSLIAttributes["summaryTitle"] =
csliNode.attributes[j].nodeValue;
                                break;
                }
        }
        return hashCSLIAttributes;
}

function pickaFile() {
        var nsIFilePicker = Components.interfaces.nsIFilePicker;
        var fp =
Components.classes["@mozilla.org/filepicker;1"].createInstance(nsIFilePicker);
        fp.init(window, "Select a File", nsIFilePicker.modeOpen);

        var res = fp.show();
        if (res == nsIFilePicker.returnOK){
          var thefile = fp.file;
          alert(fp.file.path);
          // --- do something with the file here ---
        }
```

```
}

function loadXMLProfile()
{
  // Gain access to the Preferences service
  var prefManager = Components.classes["@mozilla.org/preferences-service;1"]
                        .getService(Components.interfaces.nsIPrefBranch);

  var req = new XMLHttpRequest();
  req.open("GET", "http://www3.cs.ucy.ac.cy/adaptivewebplus/smarTag/ape/profiles/" +
document.getElementById("txtUsername").value + ".xml", false);
  req.send(null);

  var xmlDoc = req.responseXML;

  var ELEMENT_NODE = 1 // TEXT_NODE

  var traditionalChars = xmlDoc.getElementsByTagName('traditionalChars');

  for (i=0; i < traditionalChars.length; i++)
  {
        for (j=0; j < traditionalChars[i].childNodes.length; j++)
        {
          if (traditionalChars[i].childNodes[j].nodeType != ELEMENT_NODE) continue;
          switch(traditionalChars[i].childNodes[j].nodeName) {
                case "gender":
                  document.getElementById("mnulGender").value =
traditionalChars[i].childNodes[j].firstChild.nodeValue;
                  prefManager.setIntPref("extensions.smarTag.gender",
traditionalChars[i].childNodes[j].firstChild.nodeValue);
                  break;
                case "dateOfBirth":
                  var dateOfBirth = traditionalChars[i].childNodes[j].firstChild.nodeValue;
                  var dateOfBirth = traditionalChars[i].childNodes[j].firstChild.nodeValue;
                  var dayStringLength = 2;
                  var monthStringLength = 2;
                  if (dateOfBirth.substring(1,2) == "/")
                        dayStringLength = 1;
                  if (dateOfBirth.substring(dayStringLength + 2,dayStringLength + 3) == "/")
                        monthStringLength = 1;
                  document.getElementById("mnulBirthDay").value =
dateOfBirth.substring(0,dayStringLength);
                  document.getElementById("mnulBirthMonth").value =
dateOfBirth.substring(dayStringLength + 1,dayStringLength + monthStringLength + 1);
                  document.getElementById("mnulBirthYear").value =
dateOfBirth.substring(dayStringLength + monthStringLength + 2,dayStringLength + 1 +
monthStringLength + 5);
                  prefManager.setIntPref("extensions.smarTag.dateOfBirth.day",
dateOfBirth.substring(0,dayStringLength));
                  prefManager.setIntPref("extensions.smarTag.dateOfBirth.month",
dateOfBirth.substring(dayStringLength + 1,dayStringLength + monthStringLength + 1));
                  prefManager.setIntPref("extensions.smarTag.dateOfBirth.year",
dateOfBirth.substring(dayStringLength + monthStringLength + 2,dayStringLength + 1 +
monthStringLength + 5));
```

```
                        break;
                      case "profession":
                        document.getElementById("mnulProfession").value =
traditionalChars[i].childNodes[j].firstChild.nodeValue;
                        prefManager.setIntPref("extensions.smarTag.profession",
traditionalChars[i].childNodes[j].firstChild.nodeValue);
                        break;
                      case "computerKnowledge":
                        document.getElementById("mnulComputerKnowledge").value =
traditionalChars[i].childNodes[j].firstChild.nodeValue;
                        prefManager.setIntPref("extensions.smarTag.computerKnowledge",
traditionalChars[i].childNodes[j].firstChild.nodeValue);
                        break;
             }
           }
  }

  var UPPCChars = xmlDoc.getElementsByTagName('uppcChars');

  for (i=0; i < UPPCChars.length; i++)
  {
           for (j=0; j < UPPCChars[i].childNodes.length; j++)
           {
             if (UPPCChars[i].childNodes[j].nodeType != ELEMENT_NODE) continue;
             switch(UPPCChars[i].childNodes[j].nodeName) {
                      case "imagerVerbalizer":
                        document.getElementById("mnulImagerVerbalizer").value =
UPPCChars[i].childNodes[j].firstChild.nodeValue;
                        prefManager.setIntPref("extensions.smarTag.imagerVerbalizer",
UPPCChars[i].childNodes[j].firstChild.nodeValue);
                        break;
                      case "wholistAnalyst":
                        document.getElementById("mnulWholistAnalyst").value =
UPPCChars[i].childNodes[j].firstChild.nodeValue;
                        prefManager.setIntPref("extensions.smarTag.wholistAnalyst",
UPPCChars[i].childNodes[j].firstChild.nodeValue);
                        break;
                      case "workingMemory":
                        document.getElementById("mnulWorkingMemory").value =
UPPCChars[i].childNodes[j].firstChild.nodeValue;
                        prefManager.setIntPref("extensions.smarTag.workingMemory",
UPPCChars[i].childNodes[j].firstChild.nodeValue);
                        break;
             }
           }
  }
}

// Not in use
/////////////////////////////////////////////////////////////////////////////////////////////////////////
function enableSensor(){
           var checked = document.getElementById("chkEnableSensor").checked;
           if (!checked) { document.getElementById("mnulAnxiety").disabled = true; }
           else { document.getElementById("mnulAnxiety").disabled = false; }
```

```
}

function checkSensorStatus() {
	if (anxietySensorTimer == null)
document.getElementById("chkEnableSensor").checked = false;
	document.getElementById("mnulAnxiety").disabled =
document.getElementById("chkEnableSensor").checked;
}
//////////////////////////////////////////////////////////////////////////////////////////////////////

function activateAnxietyTimer() {
	var prefManager = Components.classes["@mozilla.org/preferences-
service;1"].getService(Components.interfaces.nsIPrefBranch);
	var sensorIsEnabled = prefManager.getBoolPref("extensions.smarTag.enableSensor");

	if (sensorIsEnabled) {
	  var req = new XMLHttpRequest();
	  req.open("GET",
"http://www3.cs.ucy.ac.cy/adaptivewebplus/smarTag/ape/profiles/anxiety/" +
document.getElementById("txtUsername").value + ".xml", false);
	  req.send(null);

	  var xmlDoc = req.responseXML;

	  var ELEMENT_NODE = 1 // TEXT_NODE

	  var emotionalChars = xmlDoc.getElementsByTagName('emotions');

	  for (i=0; i < emotionalChars.length; i++)
	  {
		for (j=0; j < emotionalChars[i].childNodes.length; j++)
		{
		  if (emotionalChars[i].childNodes[j].nodeType != ELEMENT_NODE)
continue;
		  switch(emotionalChars[i].childNodes[j].nodeName) {
			case "anxiety":
			  prefManager.setIntPref("extensions.smarTag.anxiety",
emotionalChars[i].childNodes[j].firstChild.nodeValue);
			  break;
		  }
		}
	  }
	  anxietySensorTimer = setTimeout("activateAnxietyTimer()", 10000);
	}
	else clearTimeout(anxietySensorTimer);
}
```

Chrome Manifest files

```
content    smarTag    chrome/content/ contentaccessible=yes

locale    smarTag   en-US   chrome/locale/en-US/smarTag/

overlay chrome://browser/content/browser.xul chrome://smarTag/content/browser.xul
overlay chrome://browser/content/preferences/preferences.xul
chrome://smarTag/content/preferences.xul
```

Chrome Manifest files

```
content    smarTag    chrome/content/ contentaccessible=yes

locale    smarTag   en-US   chrome/locale/en-US/smarTag/

overlay chrome://browser/content/browser.xul chrome://smarTag/content/browser.xul
overlay chrome://browser/content/preferences/preferences.xul
chrome://smarTag/content/preferences.xul
```

Appendix B

Appendix B consists of some important Stored Procedures and Views used in SQL Server 2000 for managing the system's data.

Retrieve User's Comprehensive Profile

```
CREATE PROCEDURE dbo.[AWEBSYS_UserProfile]
@Username nvarchar(20) = '-All Users-',
@Type int = -1

AS

SELECT ta.USERNAME,
	traditional.DATE_OF_BIRTH_VALUE,
	traditional.GENDER_VALUE,
	traditional.KNOWLEDGE_VALUE,
	traditional.PROFESSION_VALUE,
	isNull(traditional.DEPARTMENT_VALUE, '') as Department,
	device.SCREEN_RESOLUTION_VALUE,
	ta.TEST_TYPE,
	ta.Right_Answer_Count,
	ta.Wrong_Answer_Count,
	ta.Average_Response_Time,
	isNull(environment.MATCH_ENVIRONMENT, 1)
FROM AWEBSYS_TestAnalytics as ta
	LEFT OUTER JOIN INTELIWEB_USERS_ENVIRONMENT as environment ON
environment.USERNAME = ta.USERNAME
	RIGHT OUTER JOIN AWEBSYS_GetLatestTraditionalChars as traditional ON
traditional.USERNAME = ta.USERNAME
	RIGHT OUTER JOIN AWEBSYS_GetLatestDeviceChars as device ON
device.USERNAME = ta.USERNAME
where ta.TEST_OCCURENCE = (SELECT MAX(toc.TEST_OCCURENCE)
				FROM AWEBSYS_TEST_OCCURENCES as toc
				WHERE toc.USERNAME = ta.USERNAME AND
toc.TEST_TYPE = ta.TEST_TYPE)
	AND (@Username = '-All Users-' OR ta.USERNAME = @Username)
	AND (@Type = -1 OR ta.TEST_TYPE = @Type)
	AND ta.TEST_TYPE <> 0  AND ta.TEST_TYPE <> 1
ORDER BY ta.USERNAME, ta.TEST_TYPE, ta.TEST_SECTION

GO
```

Retrieve User's Actual Test Results

```
CREATE PROCEDURE dbo.[AWEBSYS_GetTestResults]
@Username nvarchar(20) = '',
@Type int = -1,
@TestOccurence int = -1

AS

SELECT COUNT(occurence.TEST_OCCURENCE)
FROM AWEBSYS_TEST_OCCURENCES AS occurence
WHERE occurence.USERNAME = @Username AND occurence.TEST_TYPE = @Type

SELECT  occurence.TEST_OCCURENCE,
        question.QUESTION_NUMBER,
        question.GIVEN_ANSWER,
        question.RESPONSE_TIME,
        question.IS_RIGHT,
        rightAnswer.RIGHT_ANSWER,
        occurence.TIME_ENTERED,
        DateOfBirth.DATE_OF_BIRTH_VALUE,
        Gender.GENDER_VALUE,
        Knowledge.KNOWLEDGE_VALUE,
        Profession.PROFESSION_VALUE,
        Department.DEPARTMENT_VALUE
FROM        AWEBSYS_TEST_OCCURENCES AS occurence
        JOIN AWEBSYS_TEST_QUESTIONS AS question ON
occurence.TEST_OCCURENCE = question.TEST_OCCURENCE
        LEFT JOIN AWEBSYS_TEST_RIGHT_ANSWERS AS rightAnswer ON
rightAnswer.QUESTION_NUMBER = question.QUESTION_NUMBER
                AND rightAnswer.TEST_TYPE = @Type
        JOIN dbo.AWEBSYS_USER_DATE_OF_BIRTH DateOfBirth  ON
DateOfBirth.USERNAME = occurence.USERNAME
        JOIN dbo.AWEBSYS_USER_PROFESSION Profession ON
occurence.USERNAME = Profession.USERNAME
        JOIN dbo.AWEBSYS_USER_GENDER Gender ON occurence.USERNAME =
Gender.USERNAME
        JOIN dbo.AWEBSYS_USER_KNOWLEDGE Knowledge ON
occurence.USERNAME = Knowledge.USERNAME
        LEFT OUTER JOIN dbo.AWEBSYS_USER_DEPARTMENT Department ON
occurence.USERNAME = Department.USERNAME
WHERE occurence.USERNAME = @Username
        AND occurence.TEST_TYPE = @Type
        AND (@TestOccurence = -1 OR occurence.TEST_OCCURENCE =
@TestOccurence)
        AND (Knowledge.TIME_ENTERED = (SELECT
MAX(LatestKnowledge.TIME_ENTERED)
                        FROM dbo.AWEBSYS_USER_KNOWLEDGE
LatestKnowledge
                         WHERE LatestKnowledge.USERNAME =
occurence.USERNAME
                                AND LatestKnowledge.TIME_ENTERED <
```

```
occurence.TIME_ENTERED))
            AND Profession.TIME_ENTERED = (SELECT
MAX(LatestProfession.TIME_ENTERED)
                                                FROM dbo.AWEBSYS_USER_PROFESSION
LatestProfession
                                                 WHERE LatestProfession.USERNAME =
occurence.USERNAME
                                                            AND LatestProfession.TIME_ENTERED <
occurence.TIME_ENTERED)
            AND Department.TIME_ENTERED = (SELECT
MAX(LatestDepartment.TIME_ENTERED)
                                                  FROM dbo.AWEBSYS_USER_DEPARTMENT
LatestDepartment
                                                WHERE LatestDepartment.USERNAME =
occurence.USERNAME
                                                            AND LatestDepartment.TIME_ENTERED <
occurence.TIME_ENTERED)
ORDER BY occurence.TEST_OCCURENCE desc, question.QUESTION_NUMBER asc

GO
```

Analytic Calculations of Tests

```
CREATE VIEW dbo.AWEBSYS_TestAnalytics
AS
SELECT toc.USERNAME,
            toc.TEST_OCCURENCE,
            toc.TEST_TYPE,
            AVG(validAvg.AverageResponseTime) AS Average_Response_Time,
            SUM(q.IS_RIGHT)  AS Right_Answer_Count,
            MAX(tt.NUMBER_OF_QUESTIONS) - SUM(q.IS_RIGHT) AS
Wrong_Answer_Count,
            q.TEST_SECTION
FROM        dbo.AWEBSYS_TEST_TYPES tt INNER JOIN
                    dbo.AWEBSYS_TEST_OCCURENCES toc ON tt.TEST_TYPE =
toc.TEST_TYPE INNER JOIN
                    dbo.AWEBSYS_TEST_QUESTIONS q ON q.TEST_OCCURENCE =
toc.TEST_OCCURENCE INNER JOIN
                    dbo.AWEBSYS_GetValidAverageResponseTimes validAvg ON
validAvg.TEST_OCCURENCE = q.TEST_OCCURENCE AND
                    validAvg.TEST_SECTION = q.TEST_SECTION
WHERE     (toc.TEST_TYPE <> 0) AND (toc.TEST_TYPE <> 1)
GROUP BY toc.USERNAME, toc.TEST_OCCURENCE, toc.TEST_TYPE,
q.TEST_SECTION
```