University of the Aegean

Department of Product & Systems Design Engineering

Methodological framework, analysis, development and creation of design support environments of adaptive systems

PhD Thesis

Evangelos Vlachogiannis

Submitted: December 2007
Supported: 10 June 2008
Supervised by Prof. Ioannis Darzentas
Members of Support Committee: Dr. Thomas Spyrou, Dr. Carlos A Velasco
Members of Exam Committee: I. Darzentas, S. Katsikas, G. Vouros, N. Sapidis, S. Gritzalis, D. Lekkas, S. Vosinakis

to my parents, George and Katerina,

to my brother, George,

to my better half, Maria

Foreword and Acknowledgments

This thesis is the outcome of the research work taken place at the University of the Aegean, in the Department of Product & Systems Design Engineering located in Syros Island. I really feel lucky that I had the opportunity to work in the corpus of this academic environment that happened to be located at the place of my birth and residence. Thus, I feel the responsibility to thank the people who dreamed and realized this department both personally but also as a resident of Syros. It was an honor for me to be part of it and I really feel it is an honor for my place that it was given the opportunity to host such a cultural source. Here I will try to find some words to thank some people for both realizing such a dream and also providing me the opportunity to work with them.

I am proud to say that Professor Ioannis Darzentas, a pioneer of such a dream, has been my supervisor. It was a privilege, a unique opportunity and experience for me to work near him. It is amazing to think how such a personality has influenced me, not only in the scientific domain but also in my everyday life. He has been my guide towards to the philosophy of both my science and my life.

The second person that I would like to thank is the Associate Professor Thomas Spyrou who was also one of the pioneers to the above mentioned dream. I can safely say that he was the one that convinced me to follow a research career and opened my mind by introducing me to human centered design principles. During all these years he was always available for helping me with any kind of problem and always had the way to make me find an appropriate solution.

Dr. Carlos A. Velasco has been the third member of my advisor committee. I would firstly like to thank him for his interest to be in such committee. I have firstly met him under the umbrella of a European project and since then it was always an honor for me to work with him. He has been my guide mainly in technological issues.

Even if they had not been in my advisory committee, I was always feeling some people from our research team as my atypical mentors. Jenny S. Darzentas had always the patience to hear me for any kind of problem and always encourage me. I would also like to thank her for both her scientific advices and her tireless editing of my poor English texts. Then, I would like to give specials thanks to my friend and colleague Argyris Arnellos. Argyris was the one that initiated me to the extraordinary world of systems thinking though every days discussions. I have spent countless moments with him having fun and working. Another person that has really helped me is Panayiotis Koutsabasis. Panayiotis was the one who introduced me to the research work of projects and was always a person I could rely on.

Further, I would really like to thank my friend and colleague Nikos Viorres with whom I have spent days and nights working on projects. I really feel very lucky I had such cooperative and problem solving colleague and humorist friend and I cannot wait the time I will be working with him again. I would also like to thank my friends and colleagues Modestos Stavrakis and Sofia Kiratzi for their support, cooperation and friendly advices.

There are even more persons that supported me either directly or directly through every day discussions. I would like to say thanks you to Dimitris Lekkas and Spiros Vosinakis and generally to all people from my university that supported me during these years. I would also like to thank people outside my university that through research projects has influenced my research. I would like to give special thanks to Professor Gerhard Weber for spending his valuable time discussing with me and giving me early feedback. I would also like to thank the European Community for the financial support through Heraclitus scholarship. Finally, I want to thank Manolis and Flavianos for their heartily support and the adorable little Manos who offered me important relaxing moments.

Nothing would have been possible for me without the support of my family. I would like to thank my parents Georgios and Katerina for their love, their care, their psychological and even for their financial support for offering me the potentials for accomplishing my studies and this research. I would like to thank them and my lovely brother Georgios for their patience and permissiveness all these years.

Finally, I would like to address my very special thanksgiving and my deep most appreciation to my adorable fellow in life, Maria, for her support and patience. I want to heartily thank her for offering me so ungrudgingly her interminable support during a critical period of her life and career.

Table of Contents

Chapter 1. Introduction

1.1 Description of the problem

Our century has been a century of information society. The Information and Communication Technologies (ICT) have become part of our lives aiming at improving our everyday situations. More and more, the sometimes visible but most of the times invisible computer has settled in our everyday activities necessitating its existence. This is even reflected in culture and, generally, in every human activity of such technological developed countries, and has been further boosted by the fact that such technological advances become less and less expensive and consequently obtainable. Most member countries of European Union, including Greece, are at a transition phase moving their focus from the development of infrastructures to the development and exploitation of services, which would make human’s life better.

Human, as a sole entity, would have never made such achievements, if she had not been organized into societies with aims, principles and communication protocols. In the same manner, computers would not have offered to the former so much, if the infrastructures, for allowing their organization to “digital societies”, had not been realized. The networking infrastructures, with all accompanied well established protocols, have been the fundamental step towards to the realization of computers' societies. The, so far named, network-capable computer has now been abstracted and “humanized” in the form of autonomous agent. An agent is often a software entity that acts autonomously for accomplishing a task, like human acts autonomously based on the restrictions posed by her substance and role in the society she belongs to. This task is often a sub task of a set of tasks, required for reaching a goal, and in most of the cases one agent is not enough as more roles, and consequently more agents, need to effectively cooperate. Such agents' societies have been called in the corpus of Artificial Intelligence (AI) multi agent societies. Some of the researchers have been trying even to mitigate the nature towards to a more effective cooperation (Artificial Life). A very important class of agents has been the user agents, whose specialization is the accomplishment of the aims that human users have explicitly communicated to them. Consequently, it seems that nowadays we are living in a society of human and artificial agents that need an effective cooperation in order to preserve their existence. Note that, even scientific fields, such as virtual agents, have been emerged aiming at improving the cooperation between artificial and human agents by providing interfaces with humanoid characteristics (e.g. avatars).

The requirement of cooperating with such artificial beings would have not been appeared, if the complexity of nowadays digital world had not been huge. The technological advancement, that caused such a boom, was not other but the World Wide Web (WWW). The WWW has proven to be much more than a service. It could be at least considered as a base of a huge number of human centered services, which inherit the talented characteristics which made the former to expand so fast and in such an extent.

Through the Web, huge amount of information are now widely accessible. From such a viewpoint, the Web can be seen as a large corpus accessed through Web Information Systems (WIS), which can rely from a classical client/server to more modern service-oriented architectures. Moreover, due to recent advances in wireless and ubiquitous computing technologies the WIS can now have a wide range of use through a wide range of devices (laptop, mobile phone etc.). On the other hand, all these advances have turned out to prove the consumption of services, offered by the web, as an unusable or at least a non-user-friendly process due to the cognitive and information overload. Consequently, the response in front of a failure has been to prevent the users from experiencing such a cognitive and informational overload. Meeting this goal guarantees to some extent the usability and the durability of a WIS.

As a response to the cognitive overload, researchers and WIS designers have started to turn over a system-centered to a user-centered design approach and thus focusing on the user interface. For the first time, the end user participates in the early stages of the design process, and

Human Computer Interaction aspects are taken into account seriously, complementing them with appropriate quality assurance processes through automatic and user evaluation. Note that an interface in the domain of WIS often can consists of access devices, operating systems, web browsers (user agent: W3C glossary) and of course, the actual WIS user interface. It is obvious that, a WIS designer is only empowered of interfering with the WIS interface portion. For the rest of the interacting interface (supersystem), a WIS designer can only settle the system capability of sensing its environment, and recruit ways to improve the overall interface for the end user through the interaction with the WIS. Tangible, attentive and more generally pleasurable user interfaces can be met in related literature, as focused researchers' movements towards to more human-friendly interfaces. All these and even more can be classified under the umbrella of usability engineering that has appeared as an applausive interdisciplinary research field.

Thus, it is an apparent need for introducing ways of modeling the environment of a user often called in literature as “context of use”. The context of use can include parameters coming from the user's personality, experience, knowledge, location, device and many more that need to be sensed and conclude to several interaction characteristics. In that sense it is said that a service needs to be personalized. Furthermore, neither the so called “context of use” nor the requirements of a system are static. In contrary, these are continuously changing both during its “development time” and during its runtime challenging for endless design action iterations. Ideally, an adaptive system can be considered as adaptive if it is capable of self-designing during runtime.

The second major response to the revealed unfriendly WIS developments was to prevent the user from the informational overload. The huge number of web pages that daily is being increased offer valuable information but at the same time makes that unmanageable and unreachable. Keyword based search engines or even human maintained taxonomies of the web sites have given partial solutions. But as the complexity of the web increase former solutions proves to be inadequate. In narrow information domains, the paradigm of web portal as a hub of domain information has proved useful and this has been crucially straightened by their fundamental characteristic named personalization. A user can now in a sense personalize both the information she is interested in and the way this is communicated to her. However, portals cannot offer much in the case of wider domains such global web. Currently attempts like Google’s search engine provide the only chance for a user to locate the information she needs in an objective way, shotted of marketing parameters. But even such a powerful search engines have limited capabilities to deal with semantically connected concepts. Semantic web and its technologies come up to respond to such requirements. In all aforementioned solutions, whatever the mechanism is, it is obvious that the solution is called adaptation; adapt the interaction of a WIS to the changing conditions.

Furthermore, the services provided by the WWW have been consecrated by the governments as cardinal goods and protected by laws. Information society is seen as the successor to industrial society and can stimulate significant social change, such as the so called digital divide. “Digital Divide” refers to the gap between those able to benefit from digital technology and those who are not[1]. As more and more services become available through such technologies, people who cannot access them are of a serious disadvantage. This refers to concepts such as design for all, personalization, and accessibility. This means that there is more concern on user preferences and capabilities and the interaction might now be characterized the most critical point. All these, advocate to the fact that systems' design has to escape from the limited designer's perspective. Systems will be rather designed to be adaptive. Design and development of adaptive systems is currently a strong research theme which results from the dynamically changing requirements and preferences of the participants in interactive systems.

However, the notion of adaptivity is not at all new. Adaptivity is a feature that can characterize an interdisciplinary notion covering a very wide range of beings and artifacts in our world. In biology, adaptation can be considered either as the evolution of an anatomical structure, physiological process or behavioral trait by the process of natural selection that increases the likelihood of producing larger numbers of offspring or its reproductive success or even the transformation of bacteria in order to resist to antibiotics. The human also is said to have the ability to adapt to climate changes by even changing her internal organization. In psychology a behavior or trait is adaptive when it helps an individual to adjust and function well within their environment. Human organizations like families can also said to be adaptive containing adaptive entities; human beings. More and more adaptive man-made machines – artificial systems are

1 See www.itu.int/ITU-D/digitaldivide

designed to be adaptive and this often takes the form of control utilizing feedback loops in order to sense conditions in their environment and adapt accordingly. Robots and artificial agents incorporate many of these control systems often utilizing adaptive programming (e.g. Genetic algorithms) for optimizing an extrinsic fitness function imposed on their environment and even electronics like smart antennas can be characterized as adaptive.

Then the question raised is what mechanisms can be employed in making a WIS adaptive. It is very recently that the European Commission has proposed Challenge 2 (Cognitive Systems, Interaction, Robotics) as part of the ICT Work Programme 2007-08[2]. According to that "*a mix of innovative scientific theory and technology is needed, based on natural and artificial cognition, in conjunction with new systems design and engineering principles and implementations for machines, robots and other devices which are robust and versatile enough to deal with the real world and to behave in a user-friendly and intuitive way with people in everyday situations*". On the other side of the Atlantic Ocean, near the end of March 2007, the Ubiquitous Web Applications Working Group[3] was lunched under W3C Initiative. "*The UWA Working Group focuses on extending the Web to all kinds of devices including sensors and effectors. Application areas include home monitoring and control, home entertainment, office equipment, mobile and automotive.*"

Such movements seem to confirm the research approach of the current thesis. The thesis presents a view of the role of adaptivity in interactive systems design and in particular it discusses the necessity of acknowledging and containing the notion of adaptivity, conceptually as well as functionally, in any design support environment. This proposal adopts an abstract approach for the analysis of preferences and requirements in interactive systems. The main result of this is the general ability of the designed system to "grow" from a very simple and implementable structure, to a complex environment.

Research areas such as multi-agent systems, profiles, accessibility, usability, machine learning have been investigated towards a definition of a framework for creating design support

---

2 http://cordis.europa.eu/fp7/ict/

3 Ubiquitous Web Applications Working Group (http://www.w3.org/2007/uwa/)

environments. Such a framework thus follows a systems theory approach, so the adaptive systems are considered to consist of sub-systems, of their characteristics, of their relationships and of a holistic behavior. These parts compose systems that are characterized as dynamic, massively entangled, scale independent, transformative and emergent. Hence a framework as such should warrant that the resultant adaptive systems should have attributes such as self-maintenance, adaptability, information preservation and spontaneous increase in complexity.

1.2 Research area and Approach

This research aims to come up with a methodological framework, analysis, development and creation of design support environments for adaptive systems. The central concept that concerns this research is the notion / phenomenon of adaptivity. This phenomenon has either explicitly or implicitly been the subject of research in different scientific fields applying to a wide range of applications. However, the approaches and progress happening in such fields are often not communicated between these fields and this is often due to the publication of relevant scientific and technical articles to conferences and journals with narrow scientific subject.

This thesis aims at contributing to the research in the field of adaptive web information systems by bringing concepts, models and techniques from different scientific fields to the field of WIS. Such an approach has been introduced in the past in the form of General Systems Theory or its successor Systems theory and Systems Thinking. For this purpose it is apparent an introduction of the reader to the roots of this theory (see next section), which will be the epistemological basis of this research. More specifically this thesis aims to:

1. develop a framework in order to support the design and development process of adaptive web information systems and

2. provide a kind of methodological framework for designing Design Support Environments that would exploit knowledge of above mentioned framework.

The first one is actually the primary and main aim of this work. The second aim actually directs the first one so that the developed framework would have been designed in a way that can be exploited to the designer of adaptive web information systems effectively through a design support environment. In other words, exploitation requirements and roadmap though DSE would need to be inscribed. Now, focusing on the primary aim of this thesis, it is a fundamental requirement to clarify the basic entities employed in this research.

The central notion of this research is the “Web Information System”. From this thesis perspective a “System” is an entity that is characterized by a boundary that differentiates it from its environment and an internal organization. Such a system consists of several subsystems that cooperate in order to emerge an overall system behavior. In the case of an “information system” the subsystems are a combination of information, information technology, stakeholders that process the information. Moreover, when we speak about a “Web Information System” we are talking about an information system that relies on the web infrastructure (concepts, resources and technologies). Maybe the most important attribute of the web is its “openness”. Due to this attribute a WIS is a system that interacts with a wider range of entities comparing with non web information systems and this means that such a system has less knowledge for the interacting parties during its design time. In addition, its web nature adds to it more interaction layers (e.g. Browser, network etc). On the other hand the openness of the web can offer to the WIS more knowledge.

As this research aims to come up with a methodological framework we should clarify what such a framework would look like as the term sometimes ends up to be used as a buzzword. As a framework in general can be characterized a set of of ideas, principles, agreements and axioms that would determine the conceptual boundaries that a consecutive design will progress. More specifically to the domain of WIS it will provide to the WIS designer a conceptual model of a WIS accompanied with the underlying abstract design concepts and axioms and at the same time a kind of guidance regarding the crucial design issues and the design process itself. Such a framework will further provide to the designer the underlying methods, techniques, architectures and technologies so that the former will be able to start a design focusing only on the particularities of her design and not with abstract requirements of adaptiveness. Thus, such a definition of the proposed framework would be characterized as methodological as it aims at providing the set of principles for a method that need to be developed and followed by a designer for a particular design problem *[Checkland, 1999].*

Having set the epistemology, determined the aims of the thesis and clarified the important glossary, this section finally presents the followed research approach (Illustration 1).

As already mentioned, the actual research approach has its roots to systems thinking. As such, the thesis' problematic situation is considered as an ill-structured situation that needs to be investigated for building up a rich picture. This involves the an investigation of the notion of adaptivity in the General Adaptive Systems corpus. From this thesis perspective General Adaptive Systems (GAS) are called all (in any domain) “general” systems that exhibit any kind of adaptive behavior (a subset of General Systems in GST) and from now on this thesis will refer to them like this. Thus, representative GAS including both natural and artificial systems have been investigated and concepts, methods and techniques have been abstracted so they can inspire the design of adaptive WIS. Where it was possible an evaluation of the methods, the techniques and the relative technologies took place in several use case scenarios and its appropriateness of use to the domain of adaptive WIS has been examined. Having reached to several useful conclusions the next step was to identify axioms and agreements based on which the framework has to be developed. A conceptual model has been developed as an amalgamation of the GAS investigation outcome. Several components of the model has been applied to use cases and improvements have been made to the model after several iterations. Finally, the emerging framework has been stabilized through its empirical evaluation to a use case scenario.

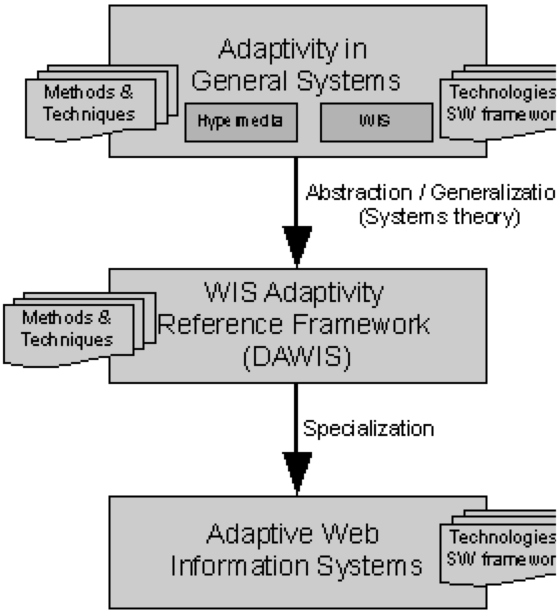


Illustration 1: Research Approach

Such a research approach has reached at a first version of the framework but several additional iterations could result to further improvements. Of course this was out of the scope of this thesis as the purpose of the thesis was to introduce an alternative approach and prove its validity in a fundamental extend. The following section is a very brief introduction of the reader to the very basic ideas of systems thinking as this seams fundamental for the apprehension of the research approach.

1.3 Brief introduction to Systems' Thinking

The scientific revolution of the 17th century brought to the the Western civilization the introduction of “science” method. This was initiated by Greeks long time before (the art of rational thinking) and till our days, our world is the world created by the activity of science. Science provided us a way of perceiving our world based on experiment and analytical reduction. It can actually be defined in terms of three characteristics; reductionism, repeatability and refutation.

Following Descrates rule, human divided up the problems being examined into separate parts. This assumes that such a division will not distort the phenomenon being studied. In other words, it assumes that the components are the same when examined separately as when examined as a whole.

The result of such an approach was actually the division of our knowledge to different "subjects" or "disciplines". Such a division, that continually evolves, has been so unconsciously and indissolubly connected with our knowledge that it makes it difficult to remember that such divisions are not divisions of nature bad man-made and arbitrary. So, physics, biology, psychology are not divisions of nature but divisions made by people for separating their knowledge.

Undoubtedly, the method of science worked great and we owe it the scientific revolution that added quality to our life. On the other hand, science seems to have some limitations. These limitations actually concerns methodology as we move from the "restricted" science to "unrestricted". The key factor here seems to be "complexity". Science method seems not to be able to deal with complex problems and the reason is that its assumption does not apply. In case of complex systems "the whole is more than the sum of the parts". Systems exhibit characteristics which cannot be found in any of its parts. These emergent properties of the whole are the most important and such properties cannot be investigated by dividing and examining under lab conditions.

Systems movement may be seen as a reaction exactly to the principle of reductionist approach. Being a systems thinker means that she will also be aware of the the problems with which the reductionist method of science cannot cope. This means that systems thinking is complementary and does not aim to replace classical science method.

In 1956 [Boulding, 1956] described systems ideas as: "*a level of theoretical model-building which lies somewhere between the highly generalized constructions of pure mathematics and the specific theories of the specialized disciplines*".

1.4 Thesis Outline

This section aims at providing an outline of the thesis. A short description for every chapter is followed accompanied with Illustration 2 that visualize how the research approach previously described is applied in terms of thesis chapters.

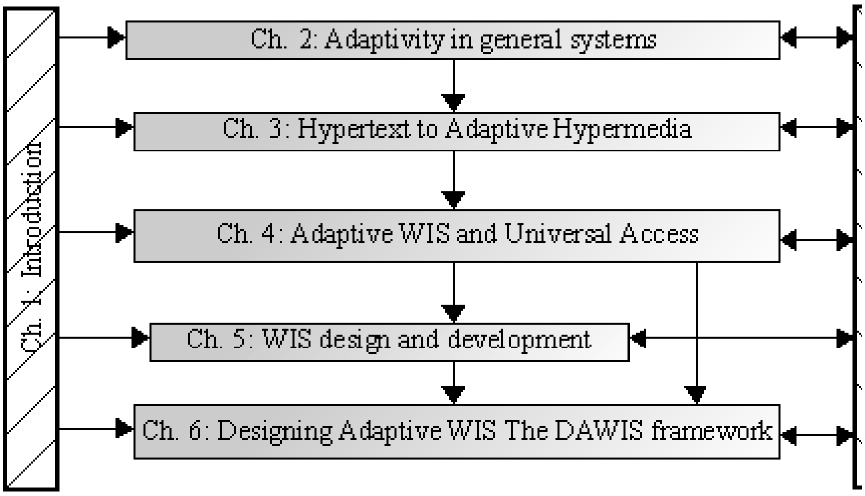


Illustration 2: Thesis outline

Chapter 1 : This is the current chapter containing introduction to the problem space, the research viewpoint and approach, this thesis' outline and a summary of contributions.

Chapter 2 : This chapter is responsible for investigating the attribute of adaptation in the field of the general systems. A classification of general systems is built as a combination of pre-existing classifications found in literature aiming at investigating systems with different characteristics and thus ensuring an interdisciplinary research. This investigation employs natural (biological and ecosystems), social and artificial systems.

Chapter 3 : After having investigated adaptivity attribute in General Systems, this chapter aims at focusing on the field of hypermedia. A throwback to the origins of hypertext and hypermedia takes place by also providing that period's relating technological context so that useful conclusions for the original requirements of such technological advances can be extracted. The most emphasis is put on the field of adaptive hypermedia. There are presented relating methods, techniques and applications. Adaptive hypermedia field appears to be strongly related to the field of user modeling which is also being discussed in the context of adaptive hypermedia.

Chapter 4 : This chapter employs the investigation of the role of adaptivity in the field of Web Information Systems with strong emphasis to the web accessibility requirements; seen as an extreme case of adaptive WIS. Thus, an introduction of web accessibility takes place following with a proposed classification of web accessibility approaches. Such a classification concludes to several adaptive techniques that have been applied to WIS. Then, emphasis is put on the WIS paradigm named portal. Portal paradigm and infrastructure is considered as a special – interesting case for the subject of this thesis thanks to its primary requirement; personalization (a case of adaptation). Thus a literature investigation of general portals but also e-commerce and m-commerce ones from an accessibility perspective takes place. Afterwards, the upcoming semantic web is being investigated including its design considerations, principles, technologies and applications. Having in mind the “disappearing computer” this thesis also investigates approaches and systems that have been used in ambient intelligence scenarios, most of them through the use of semantic web technologies. Finally, this chapter studies the use of multimedia technologies and abstract user interfaces for highly interactive systems.

Chapter 5 : This chapter aims at providing a methodological base for designing adaptive web information systems. This includes appropriate HCI design methods (SSM and User-Centered design), software design methods (waterfall, spiral and agile) and paradigms (objects, components, services and open source).

Chapter 6 : This chapter presents the proposed framework (DAWIS). First of all the design requirements for such a framework are identified. Then, according to them, axioms and definitions are stated aiming at clarifying, in the context of this work, the notion of adaptivity by also providing primitive models / metrics, required for further analysis and design (adaptivity measurement, effectiveness and capacity). Then, it is presented an attempt to support the designer on the decision whether and in what extent a problem situation requires to come up with an adaptive system. After that a two order (higher and individual order) conceptual model is being developed of an adaptive web information system. Afterwards, there is an attempt to identify the fundamental functional specifications of an adaptive system. Thus, a so named, interaction profile is proposed based on relating literature, experience and proposed classification. Then, the “adapt to what” question is discussed and the need for evaluating system's behavior is discussed. Moreover, this chapter proposes implementation architecture by identifying key aspect, providing indicative API and proposing useful technologies and software frameworks. Finally an approach for exploiting the framework through design support environments is proposed.

Chapter 7 : During the iterative design process of the aforementioned framework several methods, techniques, technologies and software frameworks have been evaluated. This chapter discusses four use cases by presenting either the developed prototype systems. An approach is presented based on IRIS research project introducing the blending of user and device profiles for adapting the presentation. IRIS demonstrates the blending of user and device profiles under an agent based architecture. Furthermore, some useful results from the BenToWeb research project are being discussed; how they can be used in the corpus of adaptive WIS. Especially, a remote testing framework named Amfortas, contribute in the field is being presented. Several portal accessibility guidelines have been reached through the proposed layered approach. Finally, the framework has been applied to the emerging application area of interactive television. Then, there is a short discussion how this framework could be exploited though a design support environment so that the designers of adaptive WIS would benefit. Overall conclusions and foreseen research are discussed the referenced bibliography is provided.

At the end of the thesis there can be found a few Annexes regarding the technologies, details of developments, list of referenced work, a glossary table and indexes of illustrations and tables.

1.5 Summary of contributions

This section consists of an outline of the main contributions took place during the research work to the relevant research fields.

- ✓ This thesis approaches the adaptivity attribute of web information systems and in general information systems from a systemic point of view resulting to an inspiration from the investigation of general systems. Classification of general systems is built as a combination of pre-existing classifications found in literature aiming at investigating systems with different characteristics and thus ensures an interdisciplinary research.

- ✓ Investigation of the attribute of adaptation in the field of the general systems including biological systems, ecosystems, social systems, artificial systems (AI), control systems engineering and autonomic computing.
- ✓ Throwback to the origins of hypertext and hypermedia takes place by also providing that period's relating technological context resulting to useful conclusions for the original requirements of such technological advances and techniques used.
- ✓ Investigation of adaptive hypermedia methods, techniques and applications and user modeling.
- ✓ Flashback to web accessibility approaches following with a proposed classification of web accessibility approaches that concludes to several adaptive techniques that have been applied to WIS.
- ✓ Portal paradigm and infrastructure is considered as a special – interesting case for the subject of this thesis thanks to its primary requirement; personalization (a case of adaptation). Investigation of its use in business cases (e-commerce and m-commerce).
- ✓ Semantic web investigation including its design considerations, principles, technologies and applications.
- ✓ Investigation of approaches and systems that have been used in ambient intelligence scenarios, most of them through the use of semantic web technologies.
- ✓ Multimedia technologies and abstract user interfaces investigation seen as modern highly interactive WIS interfaces.
- ✓ Identification and presentation of selected methods for designing and implementing AWIS and investigation of their use during runtime of the WIS.

- ✓ The development of the proposed framework (DAWIS). After identifying the design requirements axioms and definitions are stated aiming at clarifying, in the context of this work, the notion of adaptivity by also providing primitive models / metrics required for further analysis and design (adaptivity measurement, effectiveness and capacity). An attempt to support the designer on the decision whether and in what extent a problem situation requires to come up with an adaptive system. In addition, a two order (higher and individual order) conceptual model is being developed of an adaptive web information system. Afterwards, there is an attempt to identify the fundamental functional specifications of an adaptive system. Thus, a so named, interaction profile is proposed based on relating literature, experience and proposed classification. Then, the "adapt to what" question is discussed and the need for evaluating system's behavior is discussed. Moreover, there is a proposal for implementation architecture by identifying key aspect, providing indicative API and proposing useful technologies and software frameworks. Design requirements for design support environments that would exploit DAWIS framework have been finally identified.

- ✓ Four case studies are presented with prototype systems: Contribution to the BenToWeb IST research project whose results are being discussed in the context of adaptive WIS. Design, implementation and evaluation of Amfortas prototype under the BenToWeb IST research project; Contribution in the corpus of IRIS IST research project introducing the blending of user and device profiles for adapting the presentation; A layered approach for portals accessibility (Generic portals accessibility and indicative design guidelines, e-commerce portal accessibility, m-commerce portal accessibility). Portals seen from a systemic approach (whole/parts); Contribution to iTV accessibility as an overall preliminary evaluation of the proposed framework.

Chapter 2. Adaptivity in general systems

2.1 Classifying general systems

Having said that adaptivity is an interdisciplinary attribute, the investigation of its definition and application on heterogeneous fields seems of special interest. Of course, an exhaustive review of the interdisciplinary nature of adaptivity is out of the scope of this thesis and might also not be requisite. Actually, the aim of this section is to investigate the main characteristics of general adaptive systems (GAS). Such an investigation will probably allow the adoption of models, methods and techniques that can be transferred to web information systems and improve adaptivity. That kind of transfer is inspired by General Systems Theory whose first aim is "*to investigate the isomorphy of concepts, laws and models in various fields, and to help in useful transfers from one field to the other*" [Checkland, 1990]. Aiming to investigate adaptivity in general systems, it is important to investigate systems belonging to diverse domains for accomplishing the above mentioned aim of General Systems Theory. For ensuring such a diverging, investigation a classification of general systems is prerequisite.

In the past, several systems classifications have been developed. For instance, Jordan [Jordan, 1968] develops his classification based on three organizing principles that might enable human to perceive a group of entities as a system. Each principle defines a pair of systems properties that are polar opposites (Table 1).

Principles	Systems Properties
rate of change	Structural (static) / Functional (non-static)
purpose	Purposive / Non-purposive
connectivity	Mechanistic (or “mechanical”) / Organismic

Table 1: Jordan systems classification principles and properties [Jordan, 1968]

The combination of principles and properties defines the classes of systems Jordan proposes. Jordan's contribution focuses on the need to start from an observer and also draws attention to the distinction between natural and designed systems.

Another classification has been introduced by Boulding [Boulding, 1985] who develops a classification of systems based on the characteristics of the domain being observed. He classifies systems into nine categories: Structures / Frameworks, Clock-works, Control mechanisms, Open systems, Lower organisms, Animals, Man, Socio-cultural systems and Transcendental systems. He presents his classification in an order based on complexity. Checkland [Checkland, 1990] indicates that “*the fact that everyone agrees that the hierarchy is convincing entails that still there is no definition for the scale of 'system complexity'*”. Mingers [Mingers, 1997] realizes certain inconsistencies but he also focuses on the same problem: “*there is no definition of the underlying scale of systems complexity*”. He also suggests that the underlying dimension can be seen as the types of relations upon which the different levels depend and he further claims that it clarifies the inconsistencies.

Checkland [Checkland, 1990] based on the aforementioned works, builds up a new classification of systems. He classifies the systems into Natural, Designed and Human activity Systems. He defines natural systems as the ones that has their origin in the origin of universe and whose characteristics are a result of the universe's forces and processes. According to this classification Designed systems are the systems which are a result of conscious design and are further classified to designed physical (having a physical substance) and designed abstract systems (represent the product of human mind). The last category is the Human Activity systems that is the actual human act of design. Furthermore, Checkland mentions the particularity of Social systems that could fit both in design and human activity systems.

A classification has been also proposed by Scott [Scott, 2002] who distinguishes organizations into natural, rational and open systems. It is interesting to note that this work adopts the posture of the observer.

Recently, Kovacic [Kovacic, 2005] proposed a three dimensional categorization. The first dimension relates to the solution form and it actually distinguishes whether the whole is being referenced in terms of components or elements. The second dimension is focused on the complexity of the whole and components and the third one is focused on the manner in which we treat reality. Each categorization can be applied individually or simultaneously. As categorizations are increased, an increasing level of information can be developed of the ensuing groupings.

Finally even if it is not about a formal taxonomy, it is essential to mention the categorization that Holland [Holland, 1975] uses for investigating adaptive systems. He actually classifies adaptive systems to natural and artificial ones.

For the aims of this work the simplified categorization of Holland is adopted as the first level of classification and this is further expanded based on Checkland's classification. Thus this thesis distinguishes the systems to Natural, Artificial – Designed and Social Systems. Artificial systems are further expanded to Abstract and Physical.

Thus, this section has provided to this research attempt a basic classification of the General Adaptive Systems. Thanks to this, a holistic investigation will be ensured by including into the investigation scientific fields that represent the set of the identified categories. Such an investigation takes place into the following sections including biological systems and ecosystems (Natural systems), social systems, artificial intelligence (Abstract Designed Systems), control systems engineering and autonomic computing (Physical Designed Systems). Then the thesis focuses on the fields (adaptive) hypermedia and web information systems (Physical Designed Systems).

2.2 Natural systems

This section aims at investigating the attribute of adaptivity into representative natural systems. Natural systems are not static systems. We have all been witness the blossom of a rose, the growing of a child, the expanding of organizations in society. Natural systems' structure is changing over the time and even their behavior in their environment is changing. Such a change is called evolution and is a common property of natural systems. In cases where the above mentioned restructuring happens combined with organization preservation, then the system can be said to be experiencing adaptive evolution. Investigations of this phenomenon are taking place in the scientific fields of biology, with emphasis to the work done regarding the origin and preservation of life, and ecosystems.

2.2.1 Biological Systems – Origin of spaces

The theory of evolution is one of the great revolutions of human history (even though this is now in crisis in light of the tremendous advances made in molecular biology, biochemistry and genetics over the past fifty years[4]). It has drastically changed our perception of the world and our place on it. Before Charles Darwin [Darwin, 1859] scientists believed that each organism and each adaptation was due to its creator/designer.

Darwin's theory of evolution has four main parts [Allan et al, 2006] identifying that the organisms are always changing over the time, that all are derived from common ancestors by a process of branching, that the change is gradual and takes a long time and finally and more interestingly that the mechanism of evolutionary change was natural selection.

Natural selection was the most important and revolutionary part of Darwin's theory. Its process consists of four main steps: reproductive ability, struggle for existence, natural selection and finally evolution. Natural selection was the first attempt to explain the mechanism used by natural systems in order to adapt to their environments. It considered evolution as an irreversible fact, in spite of classical Newtonian reversibility. The argument of natural selection shows that

4 *http://www.allaboutscience.org/darwins-theory-of-evolution.htm*

subtle but purely physical environmental factors lead to inheritable characteristics in populations of organisms [Bausch, 2001]. He mentions that "*the relaxation of selection favors an increase of variability and thus the origin of new species*", which results to an increase in the system's capability for adaptive behavior.

Natural selection was not an adequate mechanism to explain all the non-physics features of life and this was strengthened by the introduction of non-linear dynamics and self-dissipative structures [Nikolis & Prigogine, 1997]. According to Norfolk's dictionary of cybernetics [Norfolk, 1986], a dissipative structure is defined as *"A system that exists far from thermodynamic equilibrium, hence efficiently dissipates the heat generated to sustain it, and has the capacity of changing to higher levels of orderliness".* According to Prigogine, systems contain subsystems that continuously fluctuate. It can happen that a single fluctuation or a combination of them might be able to untune the preexisting organization. At such a moment it is impossible to determine in advance whether the system will be driven into "chaos" or leap to a new, higher level of "order". The latter case defines dissipative structures because they need more energy to sustain them than the simpler structures they replace and are limited in growth by the amount of heat they are able to disperse.

Based on dissipative structures, it comes out that matter presents a spontaneous activity, or else, an emergent adaptive behavior. From the perspective of far-from-equilibrium conditions, there is a strong need for adaptive behavior as the natural system acts in a dynamic irreversible framework. In the theory of dissipative structures, a natural system is an open dynamic system, which, when pushed far-from-equilibrium, results in emergent novel structures. In this instance, adaptive evolution is based on the system's dynamics.

This capability for autonomous, self-adaptation to a changing environment is called self-organization. The term "self-organizing" seems to have been first introduced by the psychiatrist and engineer W. Ross Ashby [Ashby, 1947]. Much later, Heylighen [Heylighen, 1997] defines self-organization *as "a process where the organization (constraint, redundancy) of a system spontaneously increases, i.e. without this increase being controlled by the environment or an encompassing or otherwise external system."* He further states that in dissipative structures the self-organization depends upon the non-equilibrium boundary conditions, i.e. the coupling of the system with the environment. If these conditions change, the structure of the system will generally be destroyed. In this sense dissipative structures are not very stable and this is because the forces that create and maintain the internal distinctions are external themselves. For such a structure to become stable an internal “control” is necessary for eliminating all fluctuations and perturbations which might destroy it. DNA can be seen as an example of such an internal control which selects chains of amino acids to form the structure proteins and enzymes needed for the proper functioning of the all. Two main principles of self-organizing and adaptive systems indicate the need of a systemic and interaction-oriented framework for adaptivity: the principle of self-organization and the “order from noise” principle.

The principle of self-organization, ([Ashby, 1952];[Ashby, 1956]) argues that a self-adaptive (a system that adapts in an autonomous manner, i.e. without external control) evolutionary system, always tends to evolve towards a state of temporary equilibrium (attractor). Consequently, the system reduces its uncertainty, expands its variety and its knowledge of available options and adapts to the environment.

Ashby developed a model of a system which named **“**ultrastable system” ([Ashby, 1952];[Ashby, 1956]). This can be visualized in Illustration 3 where the two subsystems, the environment and the reacting part are presented. R represents a subsystem of the organism that is responsible for sensing through the sensor channels environmental changes and for responding to them through its motor channels. Ashby models the environment using continuous *essential variables* which in their turn affect some mechanisms (motors) “*when and only when these variables pass outside given limits”.* He defines a form of behavior as adaptive “*if it maintains the essential variables within physiological limits that define the viability zone”*.

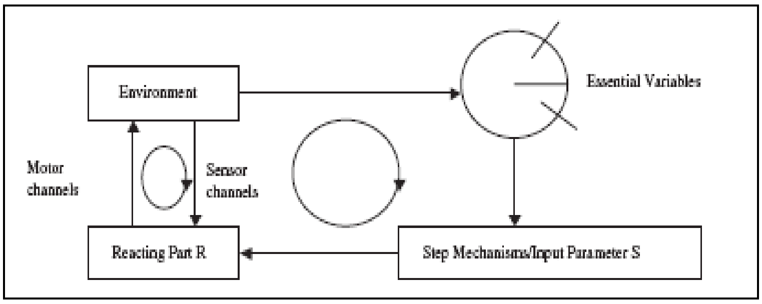


Illustration 3: The ultra-stable system architecture [Ashby, 1952]

Ashby, has further developed a very useful law for future research, the so called "the law of requisite variety". According to that, "*in active regulation only variety can destroy variety*". Heylighen & Joslyn [Heylighen & Joslyn, 2001b] comment that this leads to the observation that the regulator must have a sufficiently large variety of actions in order to ensure a sufficiently small variety of outcomes in the essential variables. This principle has important implications for practical situations: since the variety of perturbations a system can potentially be confronted with is unlimited, its internal variety should be always maximized, so as to be optimally prepared for any foreseeable or unforeseeable event.

The second principle that indicates the need of a systemic and interaction-oriented framework for adaptivity is the "order from noise" principle [Foerster, 1960]. This states that the larger the random perturbations ("noise") that affect a system, the more quickly it will self-organize (inner emergence, production of "order"). Having in mind that non-linear systems have in general several attractors, it is the interaction between system and environment that makes adaptation both necessary and possible. Moreover, system's adaptation is characterized by its fitness to the environment. Holland [Holland, 1975] suggests that the structure of a given self-organizing system has the ability to fit in different environments if it has the ability to perform "well" under them. In that case, adaptations to the environment are persistent properties of the sequence of structures generated by the adaptive system. There are two theories of evolution which expand these principles and embed them in a functional framework: The component-systems theory [Csanyi & Kampis, 1991] and the theory of autopoiesis [Maturana & Varela, 1980].

The first is based on the idea that the law of energy conservation and non-reversible dynamics express the necessary but not sufficient conditions for explaining the functionality of evolutionary and complex adaptive systems. Some components have abilities that influence and modify other components of the system with which they are related. These functions are the means of a system's organization [Bausch, 2001].

In the DNA example above, as Heylighen [Heylighen, 1989] remarks, the DNA itself is not invariant: it is itself produced and reproduced by the proteins it produces. In this sense the process is circular. This phenomenon may be called "autopoiesis", i.e. Self-production [Maturana & Varela, 1980]. An autopoietic system is a system whose internal processes produce the components needed for the internal process to occur. Thus the system maintains an invariant organization, even though the components, or substances, which carry this organization, change continuously. Thus, the theory of autopoiesis gives a phenomenological description of life from the viewpoint of a living organism. It actually describes the way living systems address and engage with the domains in which they operate. Autopoiesis states that in order a system to be adaptive it must be structure-determined.

At this point we should clarify the notions of "organization" and "structure". According to Maturana and Varela [Mingers, 1997] organization refers to the relations between components that define and specify a system as a composite unity of a particular class, and determine its properties as such a unity by specifying a domain in which it can interact as an unanalyzable whole endowed with constitutive properties. Again, according to Maturana and Varela, structure refers to the actual components and the actual relations that these must satisfy in their participation in the constitution of a given composite unit and determines the space in which it exists as a composite unity that can be perturbed through the interactions of its components, but the structure does not determine its properties as a unity.

This means that the actual changes a system undergoes depend on the structure itself. Additionally, autopoietic systems are organizationally closed as the product of their organization is that very organization itself. The fact that an autopoietic system interacts with its environment through its structure but simultaneously is organizationally closed leads to the conclusion that there is no "direct mapping" between the environment and the system itself. Thus, the environment is only able to trigger the system's structure and in no way determines or specifies the behavior of the system (structural coupling). Mingers [Mingers, 1997] notes that structural coupling is a reformulation of the idea of adaption but with the important condition, that the environment does not specify the adaptive changes that will occur. They either will occur - and thus maintain autopoiesis - or they will not - and the system will disintegrate. He supplements, that organisms become structurally coupled not only to their medium, but also to other organisms. Behaviors of one become triggers to other through the selections of their individual structures. The theory of autopoiesis offers an evolutionary framework, as it suggests that an organism undergoes a history of perturbations from its environment, which trigger its own state trajectories. If these triggers result in attractor changes that involve structural changes, then there is adaptation.

The systemic and phenomenological epistemology of autopoiesis has as its central idea that “knowing is doing and doing is knowing”. According to this and having in mind that a system is “intelligent” if it has a large internal variety of behaviors, its adaptation ability should be proportional to its variety and its ability/intelligence to manage its variety in its interaction with other systems.

In the context of this chapter, another important notion that need to be discussed is the notion of “cognition” as this is probably the main difference between natural and artificial systems. Mingers [Mingers, 1997] gives the general usage of cognition as a term that refers to the process of acquiring and using knowledge, and, as such, it is assumed to be limited to organisms with (fairly advanced) nervous system. The nervous system itself is viewed as a system that has developed to collect knowledge about the environment, enabling an organism to better survive. It is an evolutionary biological development, which increases the range of behavior that can be displayed by an organism – its requisite variety.

According to Mingers, the characteristics of the nervous system are: Maintaining Internal Correlations (acts so as to maintain or restore internal correlation between sensory and effector surfaces), Organizational closure (the state of its components at one instant determine the its state at the next - “structure-dependent”, Plasticity (structure can change over time - learning) and interactions with relations (act in response to the relations between events rather than the simple events themselves. This leads to the possibility of abstract thought, description and eventually language and the observer).

Finally, this section approaches the nervous system from a strict biological point of view [Kimball, 2003]. Such architecture is shown in Illustration 4

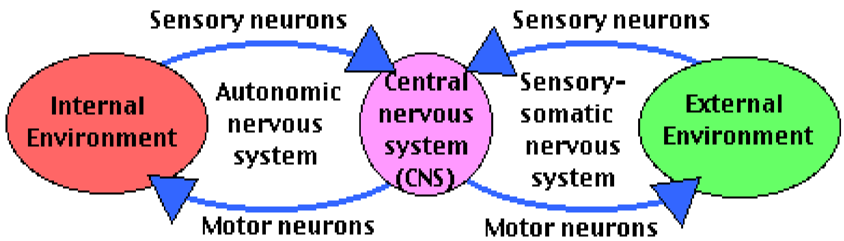


Illustration 4: Nervous System [Kimball2003]

According to this, the nervous system is divided into peripheral nervous system (PNS) and the central nervous system (CNS). CNS consists of the spinal cord and the brain while the peripheral nervous system is subdivided into the sensory-somatic nervous system and the autonomic nervous system.

2.2.2 Ecosystems

Aiming to investigate the role of adaptivity attribute, this section comes up with an indicative work in ecosystem studies. Smit et al [Smit et al, 1999] paper outlines what is meant by "adaptation" to climate change, and how it might be addressed in the IPCC[5] Assessments. More specifically, the authors investigate the attribute of adaptation in the climate change field by identifying two roles: adaptation as part of impact assessment (where the key question is: "what adaptations are likely?" and adaptation as part of the policy response, where the central question is: "what adaptations are recommended?". These two roles are placed into the objectives of the UNFCCC[6] (United Nations Framework Convention on Climate Change) (presented in Table 2).

Adaptation as part of:	**IMPACT ASSESSMENT**	**POLICY EVALUATION**
Analytical Function	Positive	Normative
Purpose	Predict, Estimate Likelihood	Evaluate, Prescribe

5 *Intergovernmental Panel on Climate Change (IPCC): http://www.ipcc.ch/*

6 *http://unfccc.int/resource/docs/convkp/conveng.pdf*

Central Question	What Adaptations are Likely?	What Adaptations are Recommended?
UNFCCC Article	Art. 2. :are the impacts likely to be dangerous for ecosystems, food production and sustainable economic development?	Art. 4: which measures should be formulated and implemented to facilitate adequate adaptation?

Table 2: Places for Adaptation Analyses in IPCC [Smit et al, 1999]

Illustration 5 presents the role of adaptation in the IPCC Assessment. The figure actually shows that the assessment of climate change impacts includes a consideration of adaptations that are expected to occur. Impact studies now use the term “net impacts” to denote the explicit consideration of adaptation in impact assessment. These adaptations have been called “autonomous” and “passive”, in that they would likely occur in the absence of specific policy initiatives to promote adaptive behavior.

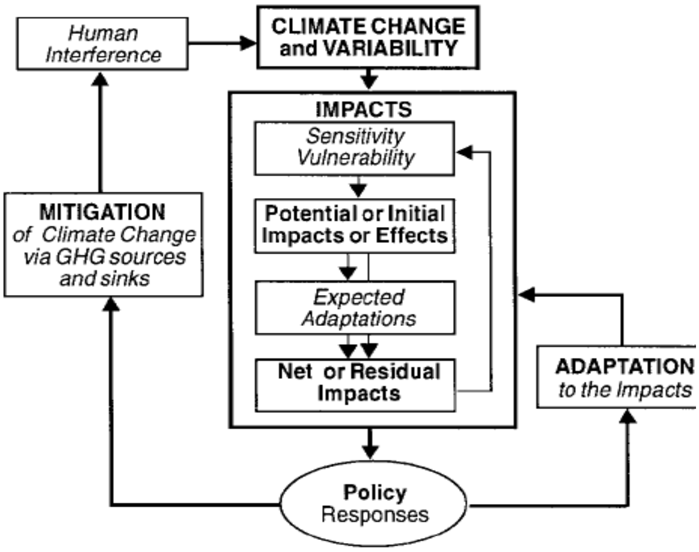


Illustration 5: Adaptation in IPCC Assessment. [Smit et al, 1999]

At this point the clarification of the terms evaluation and assessment is apparent as these are used interchangeably. Based on dictionary definitions, evaluation is to determine significance or worth or judging the effectiveness or worth of something. Assessment is to determine a rate or amount and is used as an activity to measure learning and other systems characteristics. From a time point of view an assessment is something continuously / progressively happening in contrast to the evaluation that is an overall, concluding and richer outcome.

The date the paper under discussion was written, the concept of adaptation was relatively new to the climate change research community, but as the authors state, this concept had a longer history of use in such related fields as ecology, natural hazards and risk management. Adaptations had been distinguished according to whether they are autonomous or planned, occur in natural or socio-economic systems, are anticipatory or reactive, and take technological, institutional or behavioral forms.

Before the authors proceed to the development of their framework they define key terms such as: “Adapt” means to make more suitable (or to fit some purpose) by altering (or modifying) and “Adaptation” refers to both the process of adapting and the condition of being adapted. In the climate-change literature, numerous definitions had been proposed (cited by [Smit et al, 1999]). Referring only to societal adaptation to climate, Burton (1992) defines it as “the process through which people reduce the adverse effects of climate on their health and well-being, and take advantage of the opportunities that their climatic environment provides”. Smith et al. (1996) state that “adaptation to climate change includes all adjustments in behavior or economic structure that reduce the vulnerability of society to changes in the climate system”. Smit (1993) describes it as involving “adjustments to enhance the viability of social and economic activities and reduce their vulnerability climate, including its current variability and extreme events as well as longer term climate change”. To Stakhiv (1993), the term adaptation means “any adjustment, whether passive, reactive or anticipatory, that is proposed as a means for ameliorating the anticipated adverse consequences associated with climate change”. Watson et al., (1996) define adaptability as “the degree to which adjustments are possible in practices, processes, or structures of systems to projected or actual changes of climate”, and note that “adaptation can be spontaneous or planned, and can be carried out in response to or in anticipation of change in conditions”.

Further, the authors distinguish between the terms of adaptation and mitigation as both represent responses to climate change and variability. However, “mitigation”, which means abate, moderate or alleviate, could be applied to impacts, as in mitigate vulnerabilities and effects. In this paper, mitigation is a response to the broad issue of climate change and involves reducing or stabilizing greenhouse gas emissions or levels, in order to mitigate changes in climate. “Adaptation” is applied to altering activities related to greenhouse gases. In that paper, adaptation refers to adjustments in ecological-social-economic systems in response to actual or expected climatic stimuli, their effects or impacts.

The outcome of this work is a framework which aims at providing a structure for the systematic analysis of adaptations. This is defined in terms of three questions: adapt to what, who or what adapts and how does adaptation occurs.

2.3 Social Systems

This section briefly presents fundamental issues of Social Systems research from the prism of adaptivity.

Starting from Luhmann [Luhmann, 1995] hypothesis that it is safe to say that psychic systems, and even social systems, are also living systems, social systems could be described as autopoietic systems. Luhmann, based on [Maturana, 1981] definition of autopoietic systems, notes that autopoietic systems are not only self organizing systems in a sense that they do not only produce and eventually change their own structures but their self-reference applies to the production of other components as well. He further states that social systems use communication as their mode of autopoietic reproduction and that their elements are communications which are recursively produced and reproduced by a network of communications and which cannot exist outside of such a network.

Thus, at this domain, such higher-order systems exhibit adaptive behavior in a communication oriented manner. Information is not something which the system takes in from the environment (pieces of information ready to consume) but it is actually something that is being produced by the system itself as selections in comparison with something else [Luhmann, 1995]. In other words, information is not the transportation of mere messages that will change the structure of a cognitive system, but perturbations which must be classified into the system's structure. An adaptive cognitive system will be able to structurally determine its behavior in order to perform "well" in terms of the respective message. In case such a system wants to proceed in a communication of its response (its emergent adaptive behavior), the interaction becomes a sociocommunicative one, resulting in a dynamic co-adaptation, or else, in a co-evolutionary fitness. This means that adaptation cannot be externally imposed on a system but “adaptation” must arise in a dynamic internal way as the system evolves.

2.4 Artificial systems

Based on the investigation of natural systems, researchers have tried to enrich artificial systems with intelligence. Like natural systems, artificial systems have to employ a mechanism for observing their changing requirements, tracking behavior of the interacting systems and interpreting this in a manner that would enable adaptation of their own behavior according to the new requirements, extracted from context of interaction.

2.4.1 Artificial Intelligence

Artificial intelligence is a field that was started aiming to replicate human level intelligence in a machine. Ziemke [Ziemke, 1998] states that AI's general goal is “*the modeling and synthesis of intelligent behavior and cognitive capacities in artifacts*”. AI definitions are organized into four categories [Russell & Norvig, 1995]:

Systems that think like humans (the cognitive model approach)	Systems that think rationally (the laws of though approach)
Systems that act like humans (Turing test approach)	Systems that act rationally (The rational agent approach)

Table 3: Organizing definitions of AI [Russell & Norvig, 1995]

2.4.1.1 Classical - Knowledge-based AI

The classical approach associates with the assumption that any intelligent agent needs a memory in which it stores information in form of abstract symbols about its environment. As its environment changes, the agent should be able to decode these perturbations and update its information (its perception of its environment). The agent aims to achieve its goals based on such perception and thus adapt to its plans. In other words, building an adaptive system has been based on the view that adaptive behavior is the result of assigning the system’s variety in abstract symbols, which should then, depending on the environmental perturbations, be manipulated based on rules that have been externally imposed [Newell, 1980]. Classical AI offered general purpose search mechanisms trying to string together elementary reasoning steps to find complete solutions. They were using weak information about the domain so that they have been called “weak methods”.

One of the first knowledge-based systems has been the DENDRAL program (cited by [Russell & Norvig, 1995]). This was arguably the first successful knowledge-intensive system. Classical AI traditionally emphasized a top-down approach by building systems that possess a certain amount of knowledge about a certain problem domain and then tried to model high-level cognitive capacities, such as planning, game playing, etc. This approach defines intelligence in terms of knowledge: “*A system is intelligent if it maximally applies the knowledge that it has*” (Newell's principle of rationality [Newell, 1982]).

It worth noting that according to Barry Smith [Smith, 2003] the concept of “ontology” has been associated with AI by John McCarthy in his 1980 paper on “circumscription”, where he argues that the proper treatment of common-sense reasoning requires that common-sense knowledge can be expressed in a form which allow human to express propositions like “a boat can be used to cross rivers unless there is something that prevents its use” and further introduces “ontology” concept in AI domain.

2.4.1.2 Behavior-oriented AI

Since mid-1980s, traditional AI and its underlying cognitivism paradigm in cognitive science have been questioned from a number of perspectives including robotics as such systems seemed to meet serious problems to dial with complex environments [Ziemke, 1998]. Nearly 1990s, researchers have started to stress embodied intelligence and made strong alliances with biology and research on artificial life [Steels, 1993]. Steels states that this had been characterized as a bottom-up AI and between others it has been called behavior-oriented AI. This has been defined in terms of observed behavior and self-preservation (or autonomy): “*The behavior of a system is intelligent to the extent that it maximizes the chances of self-preservation of that system in a particular environment*” [Steels, 1993]. Further, Steels [Steels, 1996] proposes a new class of systems called *evolving complex adaptive systems* and defines for them four characteristics including self-maintenance, adaptivity, information preservation and spontaneous increase in complexity.

Self-maintenance refers to the system's capability to rebuild itself by drawing materials from the environment and establish a boundary between itself and the rest of the environment. Maturana and Varela have called this process autopoiesis [Maturana & Varela, 1987]. Further, the characteristic of adaptivity emphasizes the requirement that such a system would be also capable of adapting to possible changes of its environment for preserving its existence. Additionally, an evolving complex adaptive system needs to preserve the information that defines itself even if some of its components happen to be destroyed. This means that the information regarding roles and relations of subsystems need to be preserved. Finally, such a system needs to be able to increase its own complexity. This often leads instances of the same system coming together to form a larger whole that operates as a single system evolving complex adaptive system at a higher level.

This generation of AI, is actually inspired by the notion of intelligence/cognition as a biological characteristic [Steels, 1993]. Here, the main idea is to start with the design of simple modules with multiple interaction capabilities, while expecting their interaction to emerge complex adaptive behavior. Systems of the complexity of animals have been called agents or multi-agent systems in case that several of them cooperate or compete. For a system to preserve itself under changing environment adaptivity and learning are required. Thus, a system is capable of adapting and learning if it changes its behavior so as to continue maximizing its intelligence, even if the environment changes.

A key issue relating to behavior-oriented AI is autonomy. Ziemke [Ziemke, 1998] distinguishes two kinds of autonomy: operational and behavioral. When he says that an agent is characterized by operational autonomy he means that its behavior is typically to higher degree preprogrammed, rather than self-determined. On the other hand Steels [Steels, 1993] argues on the meaning of behavioral autonomy (true autonomy): “But autonomy goes beyond automaticity, because it also supposes that the basis of self-steering originates (at least partly) in the agent's own capacity to form and adapt its principles of behavior”.

Further, Ziemke [Ziemke, 1998] distinguishes two kinds of adaptivity in terms of the factors the agent takes into account for adapting itself: long-term and short-term adaptation. The former one means that the agent decision is based on its whole or part lifetime experience and the later reacts only based only on current agent's state. Adaptive control of such an agent could be realized as in Araujo and Grupen's (1996) (cited by [Ziemke, 1998] control composition model shown in Illustration 6.

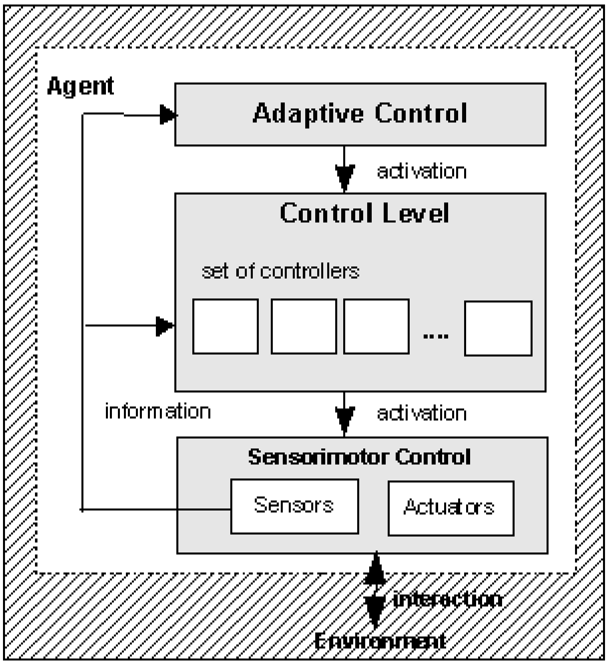


Illustration 6: Control composition of an agent. Redrawn from Araujo and Grupen (1996) as cited by [Ziemke, 1998].

Ziemke [Ziemke, 1998] distinguishes two kind of approaches for designing such behavior systems: the engineering and the learning approaches.

The engineering approaches are characterized by the fact that the autonomy is more or less predefined by the designer (operational autonomy). Two representative engineering approaches are the Subsumption architecture and the dynamic approaches.

The Subsumption architecture [Brooks, 1991] is based on the idea of “divide and conquer” and thus attempt to divide the problem to sub-problems aiming to reduce complexity and it is typically implemented by a finite state automaton (FSA). This, on one hand, means that such a system makes use of its internal state information (autonomy characteristic) but, on the other hand, each module behavior is preprogrammed and thus its variety is externally imposed. This means that interacting with a non-trivial environment will result to an increasing of the number of states; increase of the complexity.

In the case of Dynamic approaches, an adaptive artificial system consists of a number of processes running in parallel and being represented by means of differential equations establishing a continuous relationship between a set of quantities. This paradigm is closer to the descriptions used in biology, cognitive science and cybernetics. The self-organization component is emphasized, but the role of the environment is disguised.

Alternative approaches for designing such behavior systems are the so called learning approaches. This kind of approaches tends to reach “real” autonomy by attempting to design agents that could reach self-organization. Neural Networks and Evolutionary techniques are such approaches.

In the case of neural networks, the emphasis is given in the development of learning behaviors which are usually based on a direct coupling (mapping) between the system’s inputs and outputs. Although this functional mapping does not exist at the system’s initialization, it is being stabilized after a certain training phase. In most of the cases the learning is supervised or reinforced. Neural networks are being used in searching a whole static or dynamic landscape by tryinh to recognize and adapt to a static or dynamic environment.

Another approach in building adaptive artificial systems uses evolutionary techniques based on the Darwinian principle of reproduction and survival of the fittest and analogs of naturally occurring genetic operations such as crossover and mutation [Koza, 1997]. These techniques give the possibility to evolve a large number of individuals, each representing a possible adaptive behavior. It can be described as the mutation and crossover of genes to move the organism around a state space landscape, trying to find the most-fit point. Evolutionary algorithms can exhibit high adaptation where no reinforcement learning is available, and also, they can be used in combination with neural networks as the evolving mechanism of their connection weights [Mitchell, 1996]. John Holland [Holland, 1975] was one of the pioneers, by describing how an analog of the evolutionary process can be applied to solving mathematical problems and engineering optimization problems using what is now called the genetic algorithm.

Further it worth mentioning the viewpoint of Holland for an adaptive system. Holland denotes that adaptation, whatever its context, involves a progressive modification of some structure or structures. The mixture of operators acting on the structure at each stage actually builds up adaptation. According to Holland three major components are associated in the adaptation process (adaptive plan task): the environment of the system undergoing adaptation, the adaptive plan, whereby the systems' structures are modified to effect improvements and a measure, of performance, i.e. the fitness of the structures for the environment. The adaptive plan is responsible for controlling the internal changes in response to the environment. Since a given structure performs differently in different environments, the adaptive plan is responsible to produce structures which perform well (fit) in the environment.

2.4.2 Control Systems

The attribute of adaptation appeared in engineering field mainly in form of Control Systems Engineering. Control theory and Control Systems Engineering is rooted to cybernetics.

2.4.2.1 Cybernetics

Cybernetics, deriving from the Greek word for steersman (kybernetes), has been first introduced by the mathematician Wiener (1940s), as the science of communication and control in the animal and the machine [Heylighen et al, 1999]. Although first-order cybernetics (as have been called) had an interdisciplinary orientation, it might be called engineering approach, and focused on studying feedback loops and control systems, and on constructing intelligent machines [Geyer, 1995]. It grew out of Shannon's information theory, which was designed to optimize the transmission of information through communication channels, and the feedback concept used in engineering control systems.

Maybe the most fundamental contribution of cybernetics is its explanation of goal-directed behavior in terms of control and information [Heylighen & Joslyn, 2001]. A cybernetic system tries to achieve and maintain goal states through negative feedback control loops which were seen as basic models for the autonomy characteristic of organisms. Their behavior is not strictly determined by either environmental influences or internal dynamical processes.

First-order cybernetics was interested primarily in homeostasis (Newtonian order) or equilibrium-maintenance, or at least in restoring a system's equilibrium whenever it was disturbed by external influences impinging on that system. Thus, they were particularly interested in negative feedback loops, rather than positive ones (as second order cybernetics – living systems already discussed). When a negative feedback loop occurs, the performance or output of a system is compared with a preset goal, and corrective action is taken whenever there is a deviation from that goal. The thermostat of a central heating system may serve as an example: there is a feedback loop from the thermostat to the heater, whenever room temperature rises above a certain maximum, or falls below a certain minimum. It is noteworthy that even in this simple example, although it clearly is a control system, there is no specific controlling agent; "*control is dispersed through the system, and any part of it could be said to control the rest of the system*"

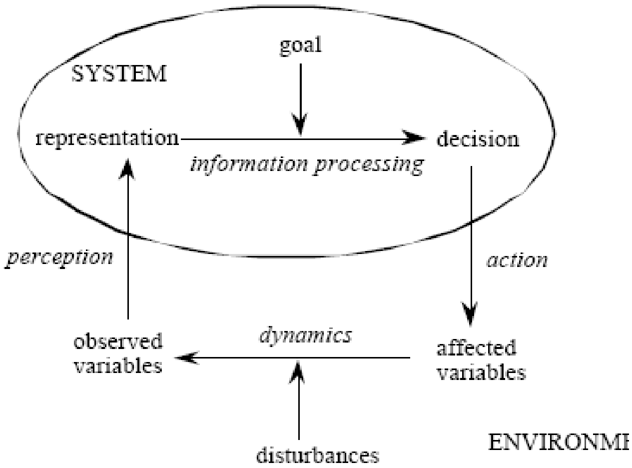


Illustration 7: Basic components of a control system [Heylighen & Joslyn, 2001]

[Geyer, 1995]. Van der Zouwen put the usage difference between positive and negative feedback succinctly :"*without negative feedback loops the organism cannot maintain itself in its environment, and without positive feedback loops it has no chance to survive as a species in view of environmental changes to which it has to adapt by setting new goals*". (Van der Zouwen comments on [Geyer, 1995])

On the other hand, second order cybernetics recognize that all our knowledge of systems is mediated by our simplified representations (models) of them, which ignore aspects of the system that is irrelevant to the purposes for which the model is constructed [Heylighen & Joslyn, 2001]. Thus, the properties of the actual system must be distinguished from the properties of the model. Illustration 7 shows the basic components of control systems from such a viewpoint.

2.4.2.2 The control engineering perspective

Now, looking from a pure control engineering perspective, at its simplest, a control system is a device in which a sensed quantity is used to modify the behavior of a system through computation and actuation. A modern view of control sees feedback as a tool for uncertainty management [Murray et al, 2003]. By measuring the operation of a system, comparing it to a reference, and adjusting available control variables, we can cause the system to respond properly even if its dynamic behavior is not exactly known or if external disturbances tend to cause it to respond incorrectly.

The evolution of control systems engineering can be briefly presented in Table 4.

Period	**Developments**
Classical Control (1940s)	designing feedback loops to avoid positive reinforcement of disturbances around a closed loop system, one can ensure that the system is stable and disturbances are attenuated.
Modern Control (1960s)	methods for multi-variable systems where many strongly coupled loops must be designed simultaneously.
Robust multi-variable control (1980s)	added powerful formal methods to guarantee desired closed loop properties in the face of uncertainties
Past 2 decades	adaptive, nonlinear, geometric, hybrid, fuzzy, and neural control frameworks

Today	a rich methodology and a supporting set of mathematical principles and tools for analysis and design of feedback systems linking four important concepts that are central to both engineered and natural systems: • **dynamics**: systems that are unstable can be stabilized, • **modeling**: input/output representations of systems, • **interconnection:** the dynamics of the resulting system is determined not only by the dynamics of the components, but by the interconnection structure between these components, • **uncertainty:** the control community has developed one of the most powerful collection of tools for dealing with uncertainty

Table 4: Control Systems Engineering evolution (summarizing [Murray et al, 2003])

Controllers are designed to eliminate the need for continuous operator attention. Cruise control in a car and a house thermostat are common examples of how controllers are used to automatically adjust some variable to hold the measurement (or process variable) at the set-point. A modern controller senses the operation of a system, compares that against the desired behavior, computes corrective actions based on a model of the system's response to external inputs, and actuates the system to effect the desired change. This basic feedback loop of sensing, computation, and actuation is the central concept in control. The key issues in designing control logic are ensuring that the dynamics of the closed loop system are stable and that dynamics have the desired behavior. The components of the model of such a modern control system can be seen in Illustration 8.

In practice, a control loop consists of: measurement by a sensor connected to the process (or the "plant"), decision in a controller element and action through an output device ("actuator") such as a control valve. As the controller reads a sensor, it subtracts this measurement from the "setpoint" to determine the "error". It then uses the error to calculate a correction to the process's input variable (the "action") so that this correction will remove the error from the process's output measurement.

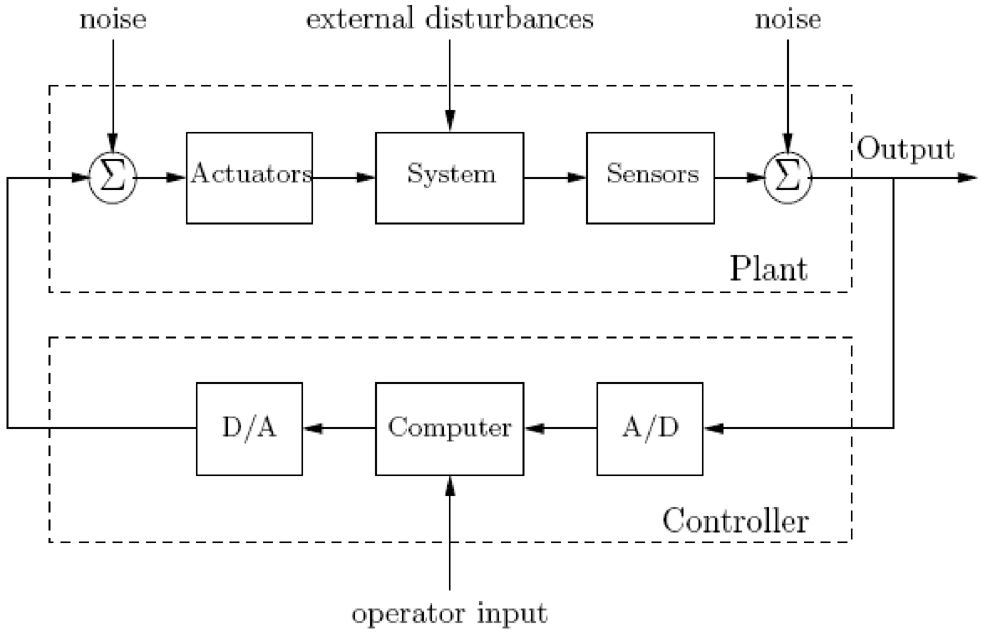


Illustration 8: Components of a modern control system [Murray et al, 2003]

A proportional-integral-derivative controller (PID controller) is a common feedback loop component in industrial control systems (First appeared in 1922 paper by Minorsky: “Directional stability of automatically steered bodies” under the name “three term control”). In a PID loop, correction is calculated from the error in three ways: cancel out the current error directly (Proportional), the amount of time the error has continued uncorrected (Integral), and anticipate the future error from the rate of change of the error over time (Derivative). The traditional PID controller is shown in Illustration 9.

For example, suppose a water tank is used to supply water for use in several parts of a plant, and it is necessary to keep the water level constant. A sensor would measure the height of water in the tank, producing the "measurement", and continuously feed this data to the controller. The controller would have a "setpoint" of (for example) 75% full. The controller would have its output (the "action") connected to a proportionally-controlled characterized control valve controlling the make-up water feed. Opening the valve would increase the rate of water entering the tank, closing the valve would decrease it. The controller would use the measurement of how the level is changing over time to calculate how to manipulate the control valve to maintain a constant level at the "setpoint" [WIKIPEDIA].

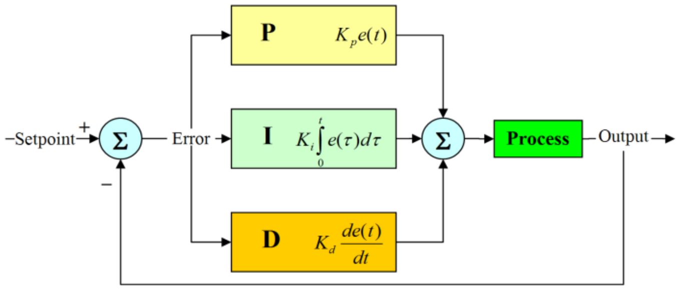


Illustration 9: A traditional PID controller

We could also mention here models for linear, time-invariant systems (LTI), adaptive PCM (pulse code modulation), adaptive digital filters and more which make use of “control” concept adapted accordingly to their domain.

Finally, the challenges currently facing the field (control) as described in the report of the Panel on Future Directions in Control, Dynamics, and Systems [Murray et al, 2003] identify several important aspects. According to the report future systems need to have their control distributed across multiple computational interconnected units. In addition these should be characterized by a higher level of coordination and autonomy. Finally, the report puts emphasis to the requirement for automatic synthesis of control algorithms, with integrated verification and validation. In other words, this requires designs that allow the system to automatically reconfigure itself so that its performance degrades gradually rather than abruptly.

2.4.3 Autonomic Computing

Nowadays, networked applications and information services have become complex, dynamic and heterogeneous. Current programming paradigms seem not to be adequate to dial with such characteristics so that such systems result to be unmanageable [Hariri et al, 2006]. In mid-October 2001, IBM released a manifesto observing that the main obstacle to further progress in the IT industry is a looming software complexity crisis. This is driven by the need to integrate several heterogeneous environments into corporate-wide computing systems, and to extend that beyond company boundaries into the Internet which actually introduces new levels of complexity [Kephar & Chess, 2003]. The situation becomes even worse if the trillions of wireless/mobile interconnected devices are considered as well.

IBM, aiming to face such problems, introduced the so called autonomic computing. Autonomic computing is an approach to self-manage computing systems with a minimum of human interference. The term derives from the body's autonomic nervous system which controls key functions without conscious awareness or involvement. Autonomic computing is the evolution of a long tradition of understanding and creating self-regulating systems. It's risen to the top of the ICT agenda because of the immediate need to solve the skills shortage and the rapidly increasing size and complexity of the world's computing infrastructure [IBM Autonomic].

IBM mentions that autonomic computing (AC)'s aim is not to produce intelligent machines; machines which embodies human cognitive power, but, systems that can adapt, learn and take over certain functions previously performed by human. In that sense, AC does not aim to replace AI, but the last one is a critical discipline that will help bring about autonomic computing.

The principles that govern all such systems have been summarized as eight defining characteristics ([Hariri et al, 2006];[IBM Autonomic]): Self-Awareness, Self-Configuring, Self-Optimizing, Self-Healing, Self-Protecting, Contextually Aware, Open and Anticipatory. According to these characteristics an autonomic system needs to know itself and its components can possess its identity. The system should have detailed knowledge of its components, current status, ultimate capacity, and all connections to other systems to govern itself. In addition, it should be able to automatically configure itself to best handle changing environments, and even, continuously looking for ways to optimize its workings. In cases of routine and extraordinary events that might cause some of its parts to malfunction the system should be able to recover by discovering problems or potential problems and find alternative ways to keep functioning smoothly. Additionally, It must detect, identify and protect itself against various types of attacks to maintain overall system security and integrity and always keep the implementation complexity hidden from the user.

In the context of this thesis, maybe the most interesting characteristics of autonomic computing systems are its Contextually Awareness and Openness. According to the first one, a system must know its environment and the context surrounding its activity, and act accordingly. "*It should be able to find and generate rules for how best to interact with neighboring systems. It will tap available resources, even negotiate the use by other systems of its underutilized elements,* *changing both itself and its environment in the process -- in a word, adapting"*. According to the latter one, such a system even independent it should follow open standards or in other words, "*an autonomic computing system cannot, by definition, be a proprietary solution*".

2.5 Summary and Discussion

The aim of the current chapter has been to investigate the attribute of adaptation in the field of general systems. A classification of general systems has been built as a combination of pre-existing classifications met in literature aiming at investigating systems with different characteristics and thus ensuring an interdisciplinary research. This investigation included natural (biological and ecosystems), social and artificial systems. This section will summarize what have been seen in this chapter and extract conclusions related to this research; adaptive web information systems.

Starting from the investigation of natural systems, the phenomenon of evolution that appears as a fundamental principle of life in biological systems is probably also the most fundamental characteristic of a WIS that need to be adaptive. The mechanism of evolution has been expressed in the form of natural selection that contradicts with preceding theories according to which adaptation was due to its creator / designer. Per contra, natural selection shows that subtle but purely physical environmental factors leads to the inheritable characteristics in populations of organisms and further concluded that this results in the system's capability for adaptive behavior. Transferring that to the domain of adaptive WIS, it can be said that the systems need not to be adaptive due to their designer but due to their capability of taking advantage of their interaction and interaction history with their environment.

This statement is further elaborated by the introduction of self-organization principle that has been based on the theory of non-linear dynamics and self dissipative structures. From the viewpoint of such theoretical advances, an adaptive WIS can be modeled as an open dynamic system that during its life cycle is being pushed far from its equilibrium due to environmental changes. According to dissipative structures theory, these conditions need to result to emergent novel structures. This evolution is based on the system's dynamics which is expressed through system's variety. According to the “law of requisite variety” only system's variety can “destroy” environmental variety / environmental changes. In other words, in order for a WIS to be adaptive this needs to be self-organized and thus depends on the coupling of the system with the environment and not externally imposed factors, which means that it is the interaction between system and environment that makes adaptation both necessary and possible. Such a conclusion is further elaborated from the theory of autopoiesis that states that in order such a system to be adaptive, this must be structure-depended. As such, this should always tend to evolve towards an attractor (a state that fits better to environmental changes). Based on the “order of noise principle” it seems that the more the interaction, the more quickly the system will self-organize. According to this and having in mind that a system is “intelligent” if it has a large internal variety of behaviors, a WIS adaptation ability should be proportional to its variety and its ability/intelligence to manage its variety in its interaction with other systems.

In addition, this chapter has presented the Ashby's ultra-stable system which seems that can be used as a basis for a model of a general adaptive system. This also seems to fit with the structures of human's nervous system as have been investigated through the prism of both cognitive science and biology.

Moreover, even the investigation of ecosystems showed off that aforementioned conclusions fully applies to systems from completely different domains by comparing them with a framework developed aiming to face climate change and variability through adaptation as part of impact assessment and policy response. Over and above, such an investigation emphasized the requirement for continues evaluation of the adaptation process and introduced a point of view for a generic framework based on three questions: adapt to what, who or what adapts and how does adaptation occurs. Such conclusion could be even incorporated as they are to a framework for adaptive WIS.

Moving to the brief investigation of social systems, a WIS can be seen as a social network of self-referenced subsystems that are characterized by both their individual and their emerging - holistic attributes. The concept of information as defined in the field of social systems (internal process of selection) is foreseen as a vehicle towards the “self-evaluation” of both subsystems and system. Again in the field of social systems it is emphasized that adaptation must arise in a dynamic internal way as the system evolves and thus cannot be externally imposed.

The idea of a social network of subsystems has been formulated in the past in the corpus of Artificial Intelligence and more specifically in the subfield of intelligent agents as multi-agent systems. The investigation in the field has shown off similar results and models. In other words intelligent agents research seems that can be parallelized with that of adaptive WIS and thus offer interesting feedback to the last one. For instance, engineering approaches like neural nets, evolutionary approaches and more even from classic AI that have been proved successful in the past can be applied to WIS and solve specific problems. For example, while speaking earlier about attractors, this, in implementation terms, introduces the requirement for setting thresholds and a foreseen good solution might be a well trained neural network. A second example could be the application of evolutionary techniques for coming up with “novel structures”. Finally, it seems interesting to consider the principles and attributes of robot design like [Ziemke, 1998] through a systems' thinking prism in a sense that a physical agent like robot is much closer to an interactive system like WIS than expert like systems.

Two other fields that seem to advocate to the interdisciplinary approach of adaptivity is Cybernetics and its succeeding Control Theory and Engineering. Cybernetics explain the goal-directed behavior in terms of control and information and such a system tries to achieve and maintain goal states through negative feedback control loops which were seen as basic models for the autonomy characteristic of organisms. Again, these systems' behavior is not strictly determined by either environmental influences or internal dynamical processes. Furthermore, cybernetics also support the distributiveness that such a system should have by stating that control is dispersed through the system, and any part of it could be said to control the rest of the system. Cybernetics also contribute with a kind of model that is in line with above mentioned. Even from the pure engineering perspective, control engineering offers such models that have been applied successfully to narrower domains like electronics. Finally, the foreseen research in the field of control, in a way, sets requirements for what need to be also satisfied towards adaptive WIS.

At the end of this chapter, autonomic computing and its principles has been presented. The complexity of the nowadays information systems lead to the requirement for more autonomic systems. Even though this research aim is to develop systems that would be able to “maintain” themselves, such a research is seen as much in line with this thesis research with the major difference that in this thesis' research the focus is on the interaction with the user. However the basic principles are similar.

Chapter 3. Hypertext to Adaptive Hypermedia

3.1 Hypertext to Hypermedia

This section realizes a flashback to the origins of Hypertext, a fundamental component of the latter web. Through such a flashback, the initial requirements of hypertext, and latter called hypermedia, are investigated and thus hypertext is being evaluated in its historical context. Investigating the incursion of hypertext it would allow us to inspire the research of adaptive WIS and clarify the potential capabilities of hypertext and its descendants in terms of adaptivity.

3.1.1 The Hypertext incursion

Even if "hypertext" term has been coined some decades before, there is still doubt about what hypertext really is and how this does relates with other terms that has been connected in literature such as hypermedia. Recently, in 2004, Noah Wardrip-Fruin [Wardrip-Fruin, 2004] discusses the different interpretations of the term and tries to clarify it. Anyway, the information revolution that we experience owes a lot to this artifact and still we are using this in our every day electronic life. We can really learn a lot out of its success story by doing a quick survey and find any connections with other important artifacts that led the information revolution.

Nearly 1940, in the middle of a world war, scientists of every side had to join their power towards victory against their enemies. That meant that scientists were working one close to the other and the knowledge was spreading by publishing. They did not have to “reinvent the wheel”. By the end of the war scientists were isolated and their work was only known to their close colleges. This caused a slackening of progression. For that reasons many scientists were pushing their research on introducing an effective way of exchanging and sharing in a knowledge network. Vannevar Bush, in 1945 [Bush, 1945], proposes something very innovative for that age. One could say that his proposal was prescribing the evolution of information technology and the raising of human computer issues. In his article, Bush [Bush, 1945] states: "*Consider a future device for individual use, which is a sort of mechanized private file and library. It needs a name, and to coin one at random, memex will do. A memex is a device in which an individual stores all his books, records, and communications and which is mechanized so that it may be consulted with extending speed and flexibility. It is an enlarged intimate supplement to his memory*" [Bush, 1945]. Trying to analyze Bush's vision a posteriori we can distinguish three key phrases that constitutes "memex":

1. "a future device for individual use", "private file and library"
2. "stores all his books, records, and communications"
3. "consulted with extending speed and flexibility"

Bush emphasized the requirement that memex would need to be for individual use and serve as a private file and library. He does not speak for a supercomputer for public use. Such a requirement could not be satisfied with no other way but the "incursion" of computer technology into our homes in the form of personal computer in 1970s.

For coming to that point, to have a personal computer, it was necessary that scientists get over a lot of issues (see Table 5). Such a device required input and output peripherals and would allow users to interact with. At Bush's time there was already a way for outputting by using CRT screens. QWERTY keyboards were working fine for computers with command line operating systems, but such operating systems were not user friendly at all to attract people and make them welcome these into their homes. The invention of the pointing device, commonly called "mouse", pushed the introduction of Graphical User Interface based operating systems that was really a friendly environment accompanied with appropriate "killer applications" so that it could visit our homes successfully.

The second fundamental extracted requirement for memex was the ability to store. Storing books, records and communications, required something more that simply storing binary files to a medium. This involves document structures and metadata, databases and communication protocols and applications. Having those, the third memex requirement can be satisfied by efficient retrieval algorithms, semantic extraction and inferencing.

Date	Event
1895	Cathode Ray Tube (CRT)
1940	First electronic computers in US, UK and Germany
1945	Vannevar Bush (Science Advisor to president Roosevelt during WW2) proposes Memex. He gaves the starting blocks to build the concepts of human computer interaction.
early 60s	The first document systems support the three core tasks: text editing, formatting (simple and low - printer level commands) and rendering. Soon, formatters become of handling non textual elements such as tables, mathematical expressions and diagrams.
1962	Technology allows Memex's concept to develop in the form of GUI. Sutherland's sketchpad program: A program that used a light pen, and allowed Sutherland to create and edit engineering drawings on a 9-inch CRT screen in WYSSYG interface. This was the 1st GUI long before the term was coined.
1962	Englebart's device: the first pointing device: the mouse. He believed that his device could be used to to drive what he called a “graphical windowed interface”.
late 60s	Generic coding using descriptive tags (no codding)
late 60s	The use of pointing device and the concept of direct manipulation set the foundation for a user friendly computing environment, but these concepts remained in academics and research till 70s.
1965	Ted Nelson coins the word "Hypertext"
1967	Andy van Dam et al build the Hypertext Editing System and FRESS
1968	NLS System: A “windowed GUI” that featured a windowed interface that was manipulated by the mouse, hyperlinked media, and even teleconferencing.
1968	Doug Engelbart demos NLS system at FJCC
1969	ARPANET: Ritchie & Tomson's UNIX operating system
1970s	The personal computer brought computer technology and the user friendly environment into the homes.
1975	ZOG (now KMS) at CMU.
1977	Apple personal computer
1978	Aspen Movie Map, first hypermedia videodisc, MIT.
1980	B. Reid divides documents into hierarchically-nested logical units, such as chapters, sections, and paragraphs, and links formatting instructions to these logical

Date	Event
	elements via style sheets.
1981	The Xerox Star introduced the idea of using metaphors as a language for a graphical interface. The Star called the working space on the screen the “computer desktop” and introduced the WIPM (Windows, Icons, Menus and Pointers) concepts that became the standard by which nearly every GUI is based.
1981	Nelson's Xanadu, centralized hypertext archive
1983	Apple released Lisa the 1st (with applications) GUI operating system. (too expensive – failed)
1983	TCP/IP replaces NCP and defined Internet
1984	1st version of MacOS GUI. Microsoft (a little company) was asked to build the “killer application” so that MacOS would survive. Microsoft Word and Microsoft Excel developed for the Macintosh. Both companies benefited.
1984	Filevision from Telos: hypermedia database for Macintosh
1984	L. Lamport applies B. Reid's technique to TEX.
1984	DNS: Domain Name Service
	Apple Macintosh bundled WYSWYG word processor
1985	Microsoft 1st attempt for GUI OS – Windows 1.0: no more than a shell over DOS (failure)
1985	Symbolics Document Examiner, Janet Walker.
1985	InterMedia, Brown University, N. Meyrowitz
1986	OWL introduces Guide, first widely available hypertext
1986	SGML as ISO [ISO86]
1987	Apple introduces Hypercard, B. Atkinson.
1987	Microsoft released Windows 2.0: lack of programs (failure)
1987	Hypertext'87 Workshop
1988	ODA as ISO. [ISO-88]
1980-1990	The first consumer GUI came into market – these interfaces would introduce a new way of computing that is still in use today.
1989	Tim Berners-Lee. WWW
1990	HTML
1990	ECHT (European Conference on HyperText)
1990	Microsoft Windows 3.0: with several applications
1993	Microsoft WindowsNT: truly GUI-based OS
1993	NCSA released 1st ver of Mosaic: friendly window based interaction
1994	Netscape 1st release

Date	Event
1995	W3C founded
1995	Lycos and Yahoo search engines
1995	Java makes applets net-portable
1998	XML 1st Recommendation
1998	Google Inc. opened its door in Menlo Park, California
1998	Introduction of handwriting recognition
2005	OpenOffice 2.0 – Next Microsoft office open standard, W3C Doc common standards activity.

Table 5: History around Hypertext (based on [Conklin,1988])

Our today's memexs are interconnected PCs and hand-held devices under the framework of WWW and its protocols so that the information is being published and shared, even if this does not happen trustworthy (see XANADU[7]). An individual can store all his books and publish them on the WWW. There are also mechanisms that they can be consulted in a means of search engines but under hard work there is a meaningful consultancy and this is expressed by semantic user agents that will be able to extend their flexibility and effectiveness by inferencing. Nevertheless, there are also a lot of issues opened towards Bush's vision.

The medium that this evolution uses is not other but hypertext. It was 1965, that Ted Nelson coined the word “hypertext” and defined it as: “***a body of written or pictorial material interconnected in a complex way that it could not be conveniently represented on paper. It may contain annotations, additions and footnotes from scholars who have examined it.”*** but it took about two decades till it was put to use in any sort of broad application that would impact an average computer user (1987, Apple's hyperCard).

As already mentioned in the beginning of this section, even after so many years there are still conflicts about Nelson's “hypertext”. In 2004, Noah Wardrip-Fruin [Wardrip-Fruin, 2004] tries to answer the question, by examining two of Nelson's early publications, “What is hypertext”. He remarks that: “*It is worthwhile to note the following: (1) “hypertext” and “hyperfilm” are coined within the same sentence; (2) both hypertext and hyperfilm are characterized as “new media”; (3)*

7 *http://xanadu.com/*

the larger category in which at least the hyperfilm is included is "hypermedia"; (4) while hypertext includes written and pictorial material, material that functions cinematically has its own term (hyperfilm); and (5) while what Nelson offers in this brief section does not explicitly contradict definitions of hypertext that focus on the link, links are not mentioned". And finally concludes with the following definition**: "***We can now, based on our examination of Nelson's texts, provide the first two sentences of a historically-based definition of hypertext appropriate for a world familiar with the Web "Hypertext is a term coined by Ted Nelson for forms of hypermedia (human-authored media that branch or perform on request) that operate textually. Examples include the link-based 'discrete hypertext' (of which the Web is one example) and the level-of-detail-based 'stretchtext'"***.** He gets off the link based definitions that most of literature uses and tries to ascribe the original idea, and not the most implemented and understandable one.

But how hypermedia arrived? We should go back again into history and observe the transition from the printed documents to electronic ones. The first, so called, document systems appeared in early 60s and supported the three core tasks: text editing, formatting (simple and low – printer level commands) and rendering. Soon, formatters became of handling non textual elements such as tables, mathematical expressions and diagrams.

In 1980 B. Reid divides documents into hierarchically-nested logical units, such as chapters, sections, and paragraphs, and links formatting instructions to these logical elements via style sheets. Four years later, this technique is applied by L. Lamport to TEX[8], two years prior to SGML establishment as an ISO standard. From the viewpoint of the author it was the editor the central component of a document processing system. That's why "What You See is What You Get" (WYCWYG) editors gave a big push.

Thus it is clear the parallel development of hypermedia and document systems that have a common pathway. The main differences between those are issues regarding reading and writing cognitive models and these are discussed in a following chapter because of their importance when approaching adaptive learning and creation systems.

8 See http://research.microsoft.com/users/lamport/pubs/pubs.html

3.1.2 Hypermedia Vs Document Systems

"Hypertext" systems introduce a new, non-sequential method for accessing information. They are coming closer to the structure of an encyclopaedia or a dictionary, in a sense that both are intended to be read non-sequentially and also their characteristic is cross-references. In other words they can be read in several different ways depending on the reader reading / processing / learning style or even his aim (Xanadu model principles).

At this point it is very important to consider how human reads and writes a document. For this purpose reading and writing models have been developed by psychologists. Regarding reading models, semiotics states that knowledge understanding take place at four levels: Lexical (the reader determines the definition of each word encountered), Syntactic (the reader determines the subject action and object of a sentence), Semantic (the reader determines the meaning of the sentence) and Pragmatic (the reader integrates the meaning of the sentence with her knowledge of self and of the world).

The aforementioned levels take place in that order, interact continuously and cannot be separated. Then, a mental representation of the meaning of the text is constructed which is in the form of propositions or relationships. This information that is stored in the short-memory is filtered by the related facts stored in long-memory. In memory, each concept is connected to other concepts, thus activating a concept, activates its adjacent concepts which activates their adjacent concepts e.t.c. This process determines what concepts need to be added or removed from the interpretation of the reading text. The process comes to an end when further activation of adjacent propositions does not change the propositions used to interpret the text.

Now, regarding writing models, these state that writing is constrained by goal and audience. In other words the writer who aims a certain goal has to take also into account what the audience is prepared to read. Writing involves three phases: Exploring (the author is brainstorming and keeping unstructured notes), Organizing (the author produces a hierarchy of the concepts and Encoding (the actual writing by composing sentences, paragraphs e.t.c).

Thus, reading employs writing processes in the reverse order. Furthermore both writing and reading processes merges the non-linear nature of thinking of human beings. Human cognition is essentially organized as a semantic network of concepts linked together with association. The last one, emphasizes the relation between the human cognition process and hypertext concepts.

Shapiro & Niederhauser [Shapiro & Niederhauser, 2004] say that the most basic feature of hypertext is its non-linearity which gives flexibility to the information access. Whereas traditional text allows the author to assume what information has already been encountered and present new information accordingly, information within a hypertext may be retrieved in a sequence specified by each user; greater degree of learner control. In other words, the reader is the one that needs to decide whether she has fill her learning gaps or what she needs to read for filling that.

A theoretical view of learning from hypertext is the Construction Integration Model (CIM) [Kintsch, 1988]. According to that model there are three stages for text comprehension: character or word processing, construction of text base and creation of situation model. According to the CIM, the integration of prior knowledge with new information is necessary to achieve a deep understanding of new material or in other words to achieve a meaningful understanding. Active learning is what this requires and this relates directly to the nature of hypertext.

Another important learning theory is Cognitive Flexibility Theory (CFT), a constructivist theory of learning from various media [Spiro et al, 1992]. According to Spiro et al, the implication of this model is that advances learning take place not only as a consequence of active learning and prior knowledge use, but also as a consequence of constructing knowledge anew for each novel problem. This perspective of learning is relevant to hypertext-based learning as hypertext offers the possibility of coming at a topic from various perspectives or in other words as the reader can access a resource page from multiple other sites and also having different learning goals. Further, Spiro et al note that a number of investigations have shown that increased metacognitive activity when reading hypertext can contribute positively to hypertext-assisted learning (HAL) outcomes.

Cognition related issues like the above mentioned / discussed need to be taken into account while designing adaptive web information systems. For instance in the case of accessibility, the metacognitive activity seems that needs to be reduced. But how can that affect the learning outcome? How can the adaptive WIS adapt to the user's cognitive model/style? Is it feasible for a system to extract such a model?

3.1.3 Designing Hypermedia

Several reference models have been developed for hypermedia systems aiming to describe the basic concepts such as the node/link structure [Koch, 2001]. Koch lists them by distinguishing them to formal and informal - semi-formal model. Dexter model [Halasz & Schwartz, 1990] seems to be the most important one since this has been the base model for other more complex models that were followed.

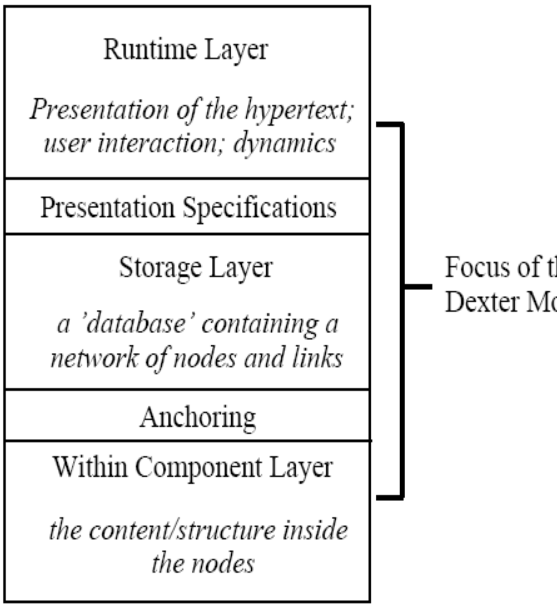


Illustration 10: Layers of Hypertext Dexter Reference Model [Halasz & Schwartz, 1990].

The Dexter reference model divides the hypermedia / hypertext system into three layers as viewed in Illustration 10. As Halasz & Schwartz [Halasz & Schwartz, 1990] discusses, the storage layer describes the network of nodes and links that is the essence of hypertext. The runtime layer describes mechanisms supporting the user's interaction with the hypertext. The within-component layer covers the content and structures within hypertext nodes. The focus of the model is on the storage layer as well as on the mechanisms of anchoring and presentation specification that form the interfaces between the storage layer and the within-component and runtime layers, respectively.

3.1.4 The Open Hypermedia movement

Work in hypermedia tended to be directed towards the traditional application domain of navigation (e.g. the authoring and browsing of structure over data). Aiming at reusing hypermedia content to diverse domains, it was necessary to consider opening the set of structural abstractions supported by an Open Hypermedia System (OHS).

Ossenbruggen et al [Ossenbruggen et al, 1998] compares between OH and traditional hypermedia and further argues that Web does not qualify as an OHS, because it requires other applications to adopt HTML as the main document format, which would require (at least) a major rewrite for most applications. They state that web browsers can not be easily extended with new document types but it seems that recent technologies like (XML/XSLT) enables such feature. An OH is a system does not have a single, fixed hypermedia document model and focus on the facilities supporting structural, domain-dependent markup, facilities to use common link structures across different document sets, and generic ways of defining how to present the encoded information, usually in the form of style sheets. In this way an open hypermedia system is able to offer generic hypermedia services to different applications. It has been seen as a middleware component offering link services and/or storage facilities to a wide variety of applications, each with their own data models and document formats. Open hypermedia system models focus on the design of the OHS architecture, the interfaces and (link) protocols which are defined by the various components in the OHS environment and the main component technology used (e.g. CORBA, DCOM etc). [Ossenbruggen et al, 1998]

At the time Ossenbruggen et al were writing their paper [Ossenbruggen et al, 1998] they were mentioning that one of the drawbacks of using XML [Bray et al, 2006] for building an open hypermedia environment was the fact that XML, and especially the standards related to XML, were in a very early state of development. Contrarily, nowadays XML is a mature standard with a wide range of supporting technologies and thus it is able to come about its role. For instance, recently, Halsey and Anderson [Halsey & Anderson, 2000] discussed the use of XLink [DeRose et al, 1999] and XPointer [Grosso et al, 2003], two emerging Internet standards designed to support the linking of XML documents, in the context of OH. A representative application example is Amaya[9] W3C editor which uses XPointer technology for associating document annotations that are placed outside the document (completely independent) in RDF(s) [Brickley & Guha, 2004] format. Furthermore, Goble et al [Goble et al, 2001] see the joining of Open Hypermedia and Ontology services as one particular implementation of the Semantic Web.

3.2 Reaching adaptive hypermedia

3.2.1 Defining Adaptive Hypermedia

Adaptive Hypermedia (AH) systems are a departure from the static, "one-size-fits-all" type of system and have emerged to serve the new demand for dynamic adaptation to the individual user. As the content and applications available over the Internet increase, the knowledge from research on AH becomes increasingly important in guiding the ways by which users can interact with content and applications in ways that are meaningful to them.

Stephanidis [Stephanidis, 2001] reports that "*In the broad domain of interactive software, adaptation has been identified as a characteristic of systems that can exhibit intelligent behavior and possess the ability to support and co-operate with their users in the attainment of interaction goals. Despite the high degree of attention that adaptation in interactive systems has drawn recently, the definition of what constitutes an adaptation-capable system remains broad and, in some cases, controversial.*" However the "original" definition of Adaptive Hypermedia was given by Brusilovsky in 1996 who defined: *"Adaptive hypermedia systems are hypermedia systems which reflect some features of the user in a user model and use this model by adapting various visible aspects of the system to the user." [Brusilovsky, 1996]*

AH systems can be useful in any application area where the system is expected to be used by people with different goals and knowledge and where the hyperspace is reasonably big. In other words, the system should satisfy three criteria: it should be a hypermedia system, it should have a user model and it should be able to adapt the hypermedia using this model [Brusilovsky, 1996].

9 *Amaya W3C editor: http://www.w3.org/Amaya/*

A clear distinction must be made between hypermedia systems that are customizable, known as adaptable systems, and adaptive hypermedia systems. In both cases the user plays a central role and the ultimate goal is to offer a personalized system. They differ in the way the adaptation is performed [Koch, 2001]. At the same year, Stephanidis [Stephanidis, 2001] also reports such a literature distinction between adaptation to user-invoked and to automatic one. Blending and summarizing such definitions we could define:

- An adaptable hypermedia, or in other words a hypermedia which experience user-invoked adaptation, is a system with predefined alternative interaction scenarios that can be triggered by the user during the runtime by explicitly changing some parameters. This presupposes considerable familiarization of the user with the system, which may limit the usefulness of the adaptation.

- An adaptive hypermedia system or in other words a hypermedia which experience automatic adaptation is a system that adapts autonomously. The system should be capable of identifying those circumstances that necessitate adaptation, and accordingly, select and affect an appropriate course of action. This means that such a system needs to be able to monitor user interaction using several explicit or implicit methods and by incorporating the data into its knowledge model inference about the modification it needs to realize to the interaction.

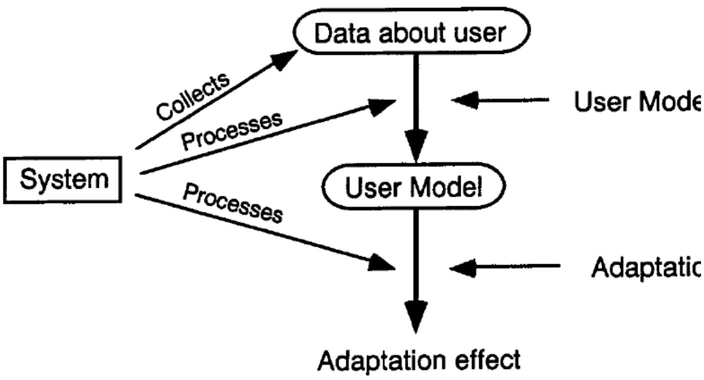


Illustration 11: Model of AHS [Brusilovsky, 1996]

Generally, in adaptive computer systems the adaptation process was described by Brusilovsky [Brusilovsky, 1996] as shown in Illustration 11.

Koch [Koch, 2001] shows (Illustration 12) a slightly different lifecycle model for adaptation graphically represented by UML state diagram which depicts the states of the lifecycle model and the possible transitions between these states. The life cycle consists of four main states: “Presentation”, where the system presents to the user presentation elements or a page appropriate to the properties the system knows about the user. The system remains in this state until the user becomes active or it receives a time-out signal. Then the system moves to the “Interaction” state, where it decides how to react to user action; either a non-adaptive or an adaptive reaction. Follows the state of “User observation” which aims to evaluate the information obtained from the user interaction with the system. The final state is the “Adjustments” which comprises of two sub-states that performed concurrently: The User model update, where the result of the acquisition is used by the system to update the user model and the “System adaptation” where the user model is utilized to adapt the presentation, content or links, e.g. to modify the user interface or generate a presentation that takes into account the user’s goals or characteristics.

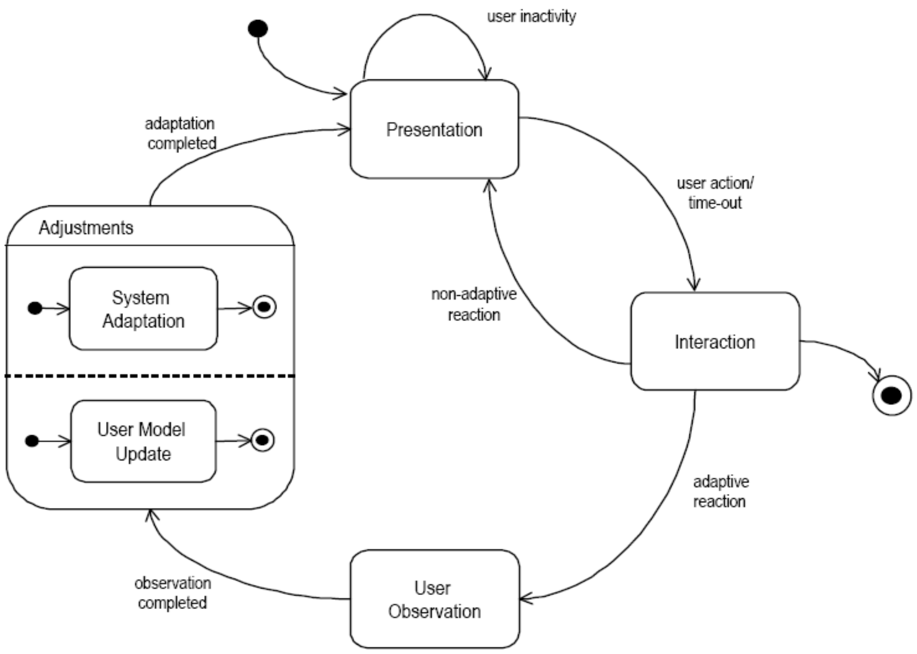


Illustration 12: Life cycle of Model of Hypermedia Adaptation [Koch, 2001]

Researching AH field requires a kind of classification of the relevant research issues. Brusilovsky [Brusilovsky, 1996] established the basis for the classification of adaptive hypermedia methods and techniques (Illustration 13). The identified dimensions are quite typical for the analysis of adaptive systems in general. The four identified dimensions are:

1. Where adaptive hypermedia systems can be helpful.
2. What features of the user are used as a source of the adaptation, i.e. to what features of the user the system can adapt its behavior.
3. What can be adapted by a particular technique? Which features of the system can be different for different users. From Brusilovsky's research two essentially different groups - content adaptation and link adaptation were identified
4. Which are the adaptation goals achieved by different methods and techniques: why these methods and techniques are applied, and which problems of the users they can solve. The adaptation goals are dependent on application areas.

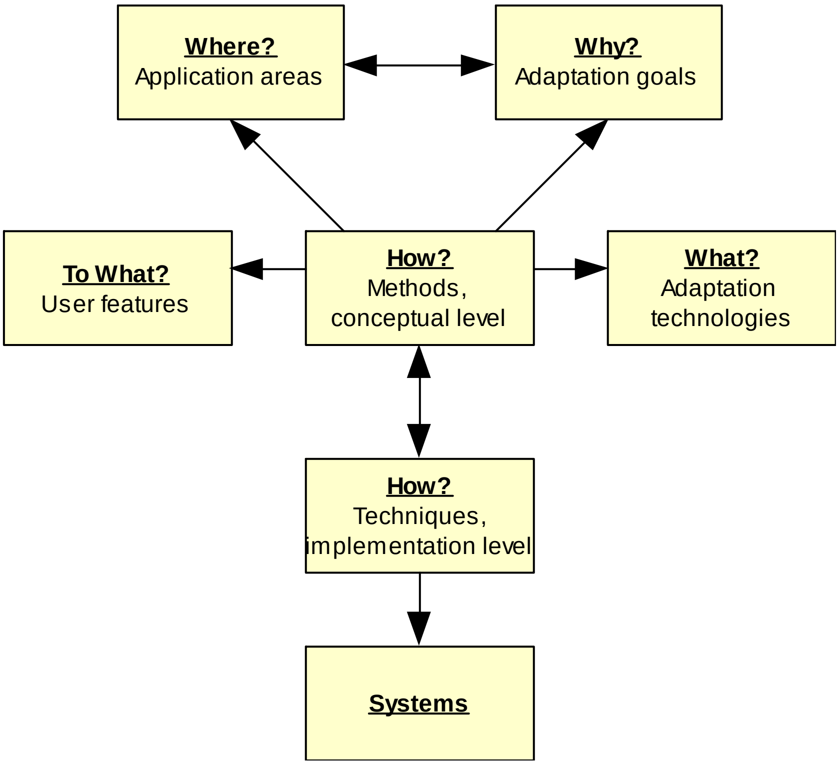


Illustration 13: Dimensions of adaptation in AH (redrawn from [Brusilovsky, 1996])

3.2.2 Applications of adaptive hypermedia systems

In terms of domains of applications, much work has been done in the area of e-learning applications [De Bra, 2002], following in the tradition of the work begun in the late eighties on Intelligent Tutoring Systems [Brusilovsky, 1996] but also in commercial settings, in the sense of allowing personalization, either in customizing content, or in the sense of adapting the display of content to the users preferences, either according to profiles, or by direct interaction with the user ([Stolze, 2002];[Fink & Kobsa, 2000]).

The wide use of AH in the area of educational hypermedia seems justified as in this area it is easier to build a detailed user model. For instance, consider a user studying a textbook: While this user studying the textbook her knowledge will change, thus her behavior will change as well. An adaptive hypermedia should be able to update her user model and adapt to that new model. It is important to mention the main difference between intelligent tutoring systems and adaptive hypermedia systems: While the formers' focus is on the tutor who decides what the learner should study (adaptive course sequencing), the latter's focus in on the learner.

Brusilovsky [Brusilovsky, 2001] reports several additional application areas including on-line information systems, on-line help systems, information retrieval hypermedia, institutional hypermedia, and systems for managing personalized views in information spaces.

Furthermore, AH research contributes to field of web accessibility (discussed in next chapter). For example, some accessibility problems have been faced by dealing with the presentation layer. Screen magnification, or providing the necessary code for assistive technologies such as text browsers. In the Avanti project [Fink et al, 1996] as well as Multireader project[10], the effort was on multimodal delivery of content and alternative content delivery. For instance in the Multireader

10 http://www.multireader.org

project: the same content could be text, have audio, have a video, with captioning, as well as some signing[11]. Users could choose the modality they preferred, or have a multimodal experience.

3.3 Designing Adaptive Hypermedia

3.3.1 Adapting to What- The role of user modeling

This section comprises an investigation of what features of the user have been used as a source of the adaptation. Traditionally (pre-1996), adaptation decision in adaptive systems was based on taking into account various characteristics of their users represented in the user model. As Brusilovsky [Brusilovsky, 2001] reports, currently the situation is different, a number of adaptive Web based systems are able to adapt to something else than user characteristics. Kobsa et al. [Kobsa et al, 1999] suggest to distinguish characteristics to user data (various characteristics of the users) , usage data (data about user interaction with the systems that can not be resolved to user characteristics and environment data (all aspects of the user environment that are not related to the users themselves). User and usage data seems to be strongly related to user modeling research and as [Brusilovsky, 2001] mentions, the development of adaptive hypermedia systems was a consecutive of both the maturity of user modeling and of hypertext.

Koch [Koch, 2001] attempts to clarify between model, user model and user modeling notions. According to Koch, a model is defined as an abstract representation of something of the real world and only some relevant properties for the application are included in the model. In the case of the user model, the real thing is the user, who is represented as a collection of data. It is the explicit representation of user aspects. Mainly the system’s belief about the user is portrayed. However, user modeling is a process covering the whole life cycle of a user model including acquisition of knowledge about the user, construction, update, maintenance and exploitation of the user model. Because of the importance of user modeling in adaptive user centered adaptive systems a brief presentation of its historical evolution follows.

11 *Signing refers to content in sign language usually in the form of a small animation file with an avatar, although sometimes video of a professional signer (person using sign language is used)*

3.3.1.1 User modeling history

Alfred Kobsa [Kobsa, 2002] states that user modeling has been studied since 1978 and has passed through various stages. The initial steps of user modeling were characterized by the fact the the user modeling process was part of the logic of the system and not as a separate subsystem. In later days, after ten years, of user modeling history, designers noticed that the user modeling components needed to be separated so that it can be reusable. According to Kobsa, in 1986, Tim Finin published his `General User Modeling System' GUMS. This software allowed programmers of user-adaptive applications the definition of simple stereotype hierarchies, and for each stereotype, of Prolog facts describing stereotype members and rules prescribing the system's reasoning about them. This system was never used together with an application system, it set the framework for the basic functionality of future `general' (i.e., application-independent) user modeling systems, namely the provision of selected user modeling services at runtime that can be configured during development time. Kobsa (1990) seems to be the first author who used the term “user modeling shell system” for such kinds of software tools. The term `shell system', or `shell' for short, was thereby borrowed from the field of Expert Systems. Five years later Kobsa built up a list of the frequently found services of shell systems based on which he identified several requirements for such systems: Generality, including domain independence; Expressiveness and Strong Inferential Capabilities.

In the late 90s web personalization has been proved as a hot theme in commercial settings aiming at attracting and keeping customers by predicting customer’s interests [Kobsa, 2001]. This information often comes from filled forms, navigations logs, purchase history, etcetera. Peppers and Rogers (1993, 1997), cited in [Kobsa, 2002], notes that personalization allows the relationship with customers on the Internet to migrate from anonymous mass marketing and sales to `one-to-one' marketing.

The central characteristic of user modeling shell commercial systems is their client-server architecture. Here, user modeling systems are not functionally integrated into the application but communicate with the application through inter-process communication and can serve more than one user/client applications at the same time. User modeling Client / Server architecture offers advantages including its central and reusable nature.

The main obstacle to furthering personalization here seems to be privacy issues that are tending to be faced by privacy frameworks (e.g. P3P [Cranor et al, 2002] and policies [Kobsa, 2002];[De Bra et al, 2004]. Nowadays, the aforementioned methods are expressed through the portal's “boom” that it is analytically discussed in next sections.

3.3.1.2 User modeling process

The process of user modeling consists of initializing, acquiring and adjusting the it. When a user reach an adaptive system for first time the system knows nothing about her so that the former could adapt to her needs. Or in other words, we could say that the user model is empty. Koch [Koch, 2001] reports two kind of user model initialization: explicit questioning and default assumption.

Explicit questioning involves that the user fills questionnaires for initializing her profile. Such an approach raises two main problems; the number of questions the use is ready to answer and how it could be ensured that these would be correct.

Default assumptions involves that the system makes default assumptions about the user and classify her for instance to a random stereotype. Such an approach would probably initially result to an undesirable behavior but progressively improves along with user interaction.

After having initialized the user model, an adaptive system has to keep it up-to-date. Such an approach has been called to Model Acquisition. The aforementioned model initialization can be considered as a special case of model acquisition when the system has absolutely no knowledge about the user. After initialization the system can recruit a number of techniques for updating its knowledge. Chin [Chin, 1993] characterize user model acquisition techniques along several orthogonal dimensions: *Active or passive*, based on the participation of the user in the acquisition, *Automatic or user initiated* based on who is the initiator of the acquisition, *Direct or indirect,* depending on the length of the inference chain, *Explicit or implicit*, based on the type of user feedback, *Logical or plausible,* according to the results produced and *Online or offline*, based on when the acquisition is performed. The most acknowledge user modeling techniques are discussed below:

Overlay model: The user's knowledge is modeled as an overlay of the structural model of the subject domain. Usually, the subject domain is modeled as a semantic network of concepts – elementary pieces of knowledge. For each domain model concept, an individual overlay model stores some value which are an estimation of the user knowledge level of this concept. This can be just a binary value, a qualitative measure, or a quantitative measure. The resulting overlay model of user knowledge then can be represented as a set of pairs "concept-value", one pair for each domain concept. Brusilovsky [Brusilovsky, 1996] mentions that overlay models are powerful and flexible, they can independently measure user knowledge of different topics.

Stereotypes: A stereotype user model distinguishes several user categories, referred as "stereotypes". For each dimension of user modeling the system can have a set of possible stereotypes (predefined values) or even by incorporating some probabilistic value (the probability that the user belongs to the stereotype). Brusilovsky [Brusilovsky, 1996] notes that stereotype model is simpler and less powerful than overlay model but it is also more general and much easier to initialize and to maintain.

Combination of overlay model and stereotypes: It seems that the combination of the two aforementioned techniques can prove the most optimal solution. Such a combination can take place by using stereotype modeling at the beginning of work to classify a new user and to set initial values for overlay model, then a regular overlay model is used.

Bayesian Networks: Bayesian networks are one of the most popular numerical techniques used to manage uncertainty in user modeling. A Bayesian network is a directed, acyclical graph in which the nodes correspond to variables (user properties) and links correspond to probabilistic influence relationships. In the context of user modeling such variables might be used to model domain knowledge, background knowledge, and cognitive model – learning styles.

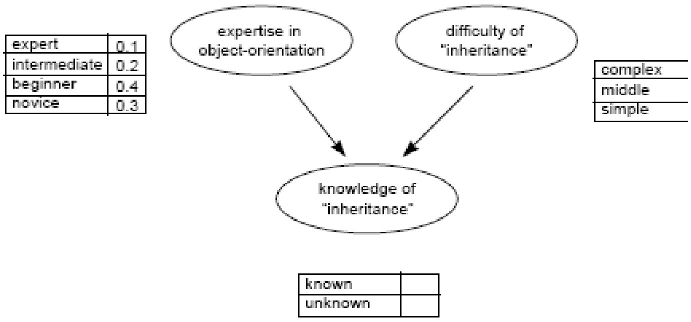


Illustration 14: Bayesian Network Example for User modeling [Koch, 2001]

Furthermore, Bayesian networks suite also to integration of distributed, fragmented user models [Tedesco et al, 2006]. Koch [Koch, 2001] presents a useful example that illustrates the use of Bayesian networks to the field of user modeling presented in Illustration 14. More computation techniques are referred in [Koch, 2001].

3.3.1.3 User modeling attributes used in AH

Brusilovsky's reviews ([Brusilovsky, 1996];[Brusilovsky, 2001]) showed up six kinds of user characteristics. A summarization of the superset of these is briefly presented below by classifying them to independent and to dependent of application context. The form category refers to features that are either stable or change rarely and more or less define the user as an individual. The latter category refers to frequently changing features that strongly depends on the context of the specific application.

Independent of application context

- **Knowledge:** User's knowledge is a particularity of the user which often changes and as such it needs to be monitored and updated. Modeling user's knowledge has been mainly achieved by overlay model, stereotype or the formers combination.
- **Interests:** This feature models long term interests and combined them with short-term ones aiming at improving information filtering and recommendations.
- **Individual traits:** These features define the user as an individual and include like personality factors (e.g. introvert/extravert), cognitive factors, and learning styles. These have been usually extracted by specially designed psychological tests. Brusilovsky cites Gilbert and Han (1999) who states that in order to progress in this area, we either

need to learn more about the relationships between user traits and possible interface settings, or treat user traits as a black box and attempt to model them and adapt to them using non-symbolic technologies.

- **Background:** This is defined as all the information related to the user's previous relevant experience outside the subject of the hypermedia system. This includes the user's profession, experience of work in related areas, as well as the user's point of view and perspective. This is often modeled using stereotypes and retrieved basically by interviewing users.

<u>**Dependent of application context**</u>

- **Goals/tasks:** This represents the actual goal a user has in mind at a specific time and this often changes. The goals are often modeled using overlay model but more advanced methods involve representations in tree hierarchies or even set of pairs "goal-value" where the value is usually the probability that the corresponding goal is the current goal of the user [Brusilovsky, 1996].
- **Experience:** This models the familiarity of the user with the structure of the hyperspace and how easy can the user navigate in it. This is often modeled using stereotypes.
- **Preferences:** This represents the preference of the user to some nodes and links over others and some parts of a page over others.

It is worth noting that these features are strongly related to the field of hypermedia and thus should be abstracted in order to be transferred to the field of WIS.

Furthermore, according to [Brusilovsky, 2001], web based systems introduced the requirement for considering the user's environment as an important factor to the adaptation process. With the arrival of web based systems, a single (server-side) system needs to serve several users with completed different working environments. Consequently, recent adaptive hypermedia systems have also involved to the adaptive process factors such as user location and user platform (hardware, software, network bandwidth). Such “context-aware” environment will be discussed in detail in the next chapter.

3.3.2 *What can be adapted - Methods and techniques for AH*

This section investigates what can be adapted in a hypermedia system in order to fit better to user's needs. In other words, what are the features of a hypermedia system that can differ for different users. Brusilovsky's review [Brusilovsky, 1996] comes up with taxonomy of such features. According to Brusilovsky, two types of adaptation are distinguished: Adaptive presentation (or content level adaptation) which refers to the content of a page and Adaptive navigation support (or link level adaptation) which refer to the way of presenting links. Adaptive presentation could be further classified to adaptive text presentation and adaptive multimedia presentation.

In his taxonomy, Brusilovsky illustrates methods for adaptive navigation such as link hiding, sorting, annotation, direct guidance and hypertext map adaptation and further discusses applications of them and their strengths and weaknesses. Table 6 summarizes and classifies them based on their level.

Technique	Description	Level
Direct guidance	What is the next "best" node for the user to visit according user's goal and other parameters represented in the user model.	Link
Sorting	Sort all the links of a particular page according to the user model and to some user-valuable criteria: the more close to the top, the more relevant the link is.	Link
Hiding	Restrict the navigation space by hiding links to "not relevant" pages. This category also includes link disabling, hiding, removal and dimming.	Link
Annotation	Augment the links with some form of comments which can tell the user more about the current state of the nodes behind the annotated links.	Link
Map adaptation	Adapting the form of global and local hypermedia maps presented to the user	Link
Insert / remove fragments	Inserting or removing fragments of text for prerequisite explanations (extra explanation for user who need it), additional explanations (more in-depth information) or comparative explanations (comparison between topics)	Text
Stretchtext	Items or paragraphs can be displayed or hidden and the system decides adaptively which items to open when the page is first displayed.	Text
Alter fragments	Select between a number of alternative explanations	Text

Technique	Description	Level
Sort fragments	Sorting for performing relevance ranking. Usually for adapting to different cognitive styles.	Text
Dim fragments	Makes text less visible (instead of hiding)	Text
Natural language generation	Many natural language generation systems make use of fragments (and even paragraphs) of canned text. A distinction here is made between those systems that use natural language technology as a foundation and those that do not.	Text
Alter media	Altering or even user more than one types of media simultaneously to present information more effectively.	Modality
Link generation	Link generation includes three cases: discovering new useful links between documents and adding them permanently to the set of existing links; generating links for similarity-based navigation between items; and dynamic recommendation of relevant links	Link

Table 6: Adaptive Hypermedia Techniques (summarized and classified from [Brusilovsky, 1996])

Five years later, [Brusilovsky, 2001], he further extends his taxonomy based on 1996's one. In this work the most notable additions (more in Illustration 15) are: Adaptation of modality under the adaptive presentation aiming at including the possibility of presenting the information in different types of media like video and speech and Link generation as a new category under adaptive navigation support aiming at including links generated mainly in recommender systems. Brusilovsky identified several different methods for such an adaptation on the basis of user preferences, abilities, learning style and context of work, in several kinds of adaptive hypermedia systems.

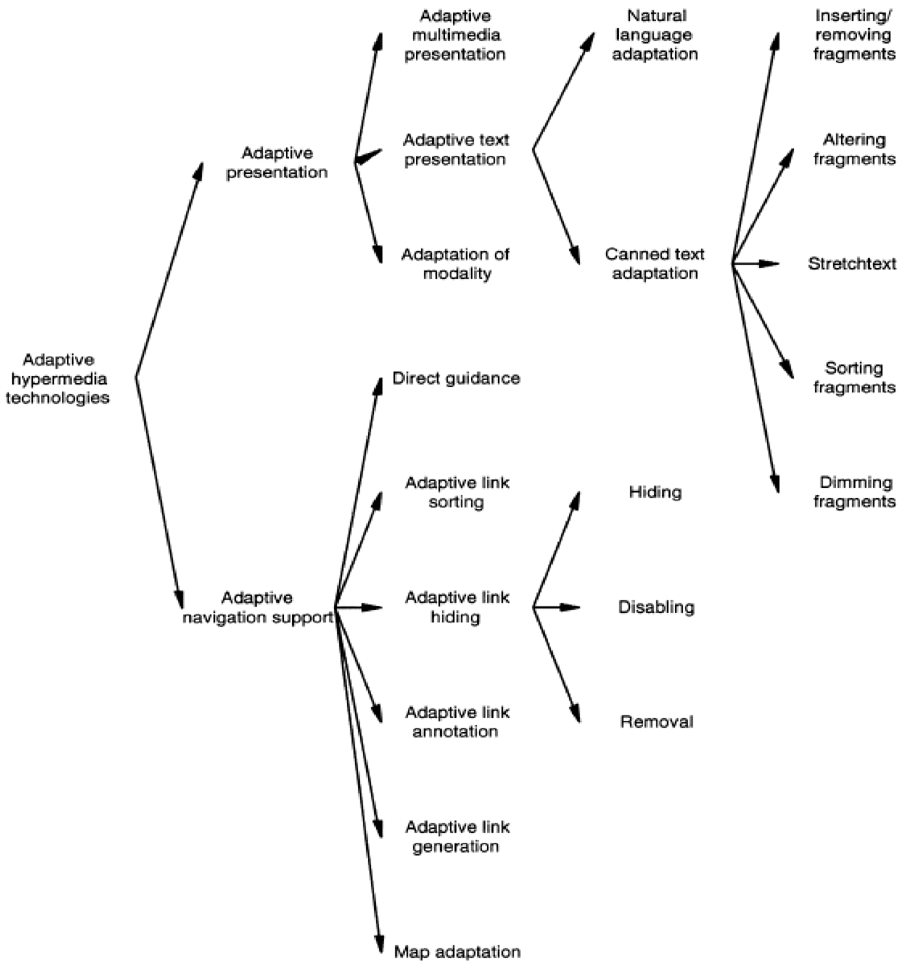


Illustration 15: Taxonomy of AH technologies [Brusilovsky, 2001]

Recently, Brusilovsky and Maybury, [Brusilovsky & Maybury, 2002] recognized the movement from adaptive hypermedia to adaptive web and even more to mobile web, they distinguish three generations: the “Pre-Web” generation which treat adaptive presentation and adaptive navigation support and concentration on modeling user knowledge and goals, the “Second-Web-generation” which further incorporates with adaptive content selection and adaptive recommendation based on modeling user interests and finally, the “Third-mobile-generation” that extends the basis of the adaptation by adding models of context to the classic user model aiming at adapting both to user and the user's context of use.

3.3.3 AH Reference models and methods

This section presents the most referenced reference models and relating methods in the field of adaptive hypermedia. Koch [Koch, 2001] perceives four objectives of hypertext or hypermedia reference models: “*to capture important abstractions found in current hypermedia applications, to describe basic concepts, such as node/link structure of these systems, to provide a basis to compare the systems and to develop a standard*”.

As already discussed in earlier chapter, Dexter reference model had provided a common language for the people involved in hypermedia development so that they could obtain common abstractions for the hypermedia systems existing at that time. Based on that model, several other models have been developed (Devise Hypermedia Model, Amsterdam Hypermedia model and more discussed thoroughly in [Koch, 2001]). As for the purposes of this thesis, reference models for adaptive hypermedia is of special interest as a possible basis to the foreseen framework for adaptive WIS these will be briefly presented below.

AHAM: Adaptive Hypermedia Application Model: The AHAM has been introduced by Houben and De Bra [Houben & De Bra, 1998] and as most AH work focus on educational hypermedia. AHAM extends Dexter reference model. According to Houben and De Bra [Houben & De Bra, 1998], Dexter's storage layer represents a domain model, i.e. the author's view on the application domain. In adaptive hypermedia applications the central role of the domain model is shared with user model as figured in Illustration 16.

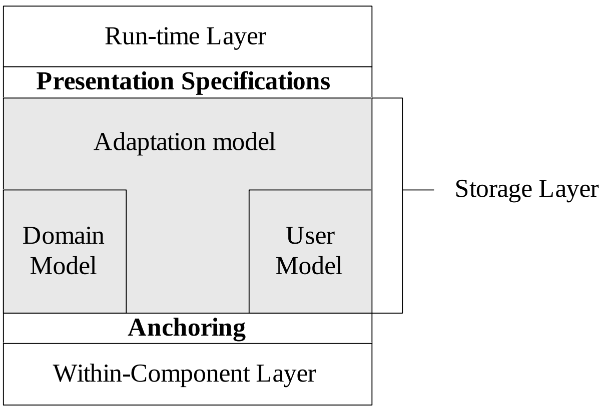


Illustration 16: AHAM reference model (adapted from [De Bra, 2002])

Thus, at the top there is the run-time layer which represents the user interface. AHAM does not describe what the user interface should do exactly. Instead it provides abstractions of what it should do (e.g content to be emphasized) by means of presentation specifications. The run-time layer is responsible to translate such specifications to presentation technologies (e.g. Style sheets). At the bottom, the within-component layer describes the internal, implementation-specific data objects that can be accessed by means of anchoring. The core of the AHAM model is the storage layer, like in the Dexter model. In AHAM this layer consists of three functionally different parts: the *domain model* which contains a conceptual representation of the application domain, the *user model* which contains a conceptual representation of all the aspects of the user that are relevant for the adaptive hypermedia application and the *adaptation model* that describes how an event results in a presentation, by combining elements from the domain model and the user model.

The Munich Reference Model: The Munich reference model is more or less a formalization of AHAM. Aiming at filling the gap of AH generalization, the main novelty of this approach is an object-oriented specification written in UML (Unified Modeling Language) which integrates both an intuitive visual representation and a formal unambiguous specification in OCL (Object Constraint Language) [Koch & Wirsing, 2002].

The Munich reference model preserves the three-layer structure of Dexter Model describing the network of nodes and links and the navigation mechanism but at the same time, also extends the functionality of each layer to include user modeling and adaptation aspects. Thus, in order to support adaptation the storage layer is divided into three sub-models [Koch & Wirsing, 2002]: the Domain Meta-Model that manages the basic network structure of the hypermedia system, the User Meta-Model that manages a set of users represented by their user attributes with the objective to personalize the application and the Adaptation Meta-Model that consists of a set of rules that implement the adaptive functionality, i.e. personalization of the application.

For the purpose of this thesis the most important part will be presented further is the Adaptation Meta-model. The adaptation is performed using a set of rules, such as in most adaptive hypermedia applications. These rules determine how pages are built and how they are presented to the user (presented as a UML class diagram in Illustration 17.

The core elements used to model the adaptation are the class Adaptation and the class Rule. The class Adaptation includes operations for resolving a component, triggering that for returning all connected rules and finding the rules. The class Rule (executor operation) allows the system to select the appropriate components, and to perform content-adaptation, presentation-adaptation and link-adaptation as well as to update the User Model. These operations play the role of the adaptive engine in AHAM. Rules are classified according to their objectives into: construction rules, acquisition rules and adaptation rules as shown in Illustration 17.

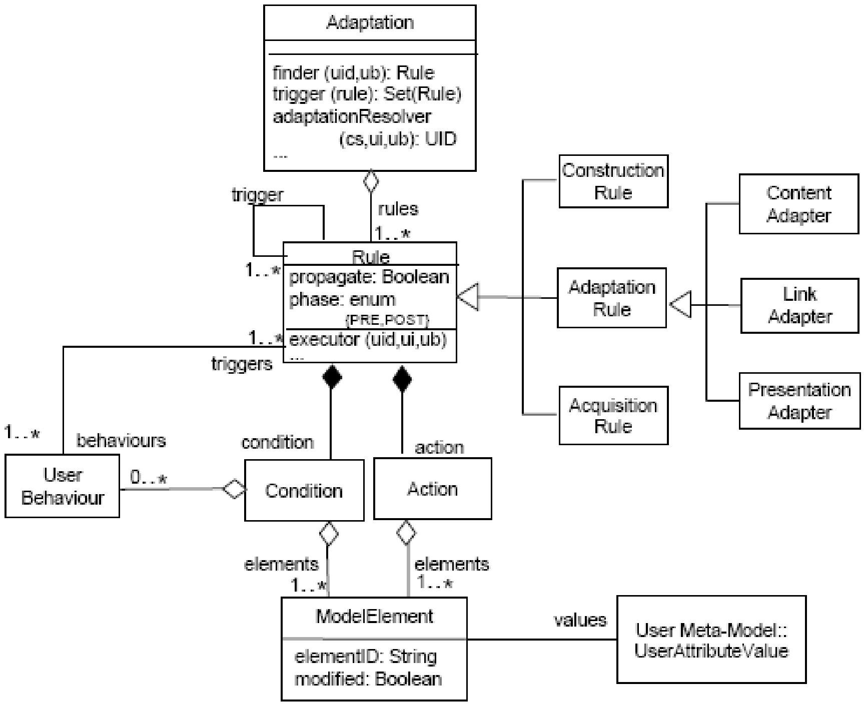


Illustration 17: View of the Adaptation Model of the Munich Reference Model

3.4 Summary and Discussion

The current chapter has attempted to transfer the focus of the thesis to the field of adaptive hypermedia, a “brother” scientific field of adaptive WIS.

Aiming at reconsidering the initial design requirements of such technological advances, a throwback to the origins of hypertext and hypermedia takes place by also providing that period's relating technological context so that a rich picture would be developed.

Thus, starting from the initial hypertext design requirements as expressed in the form of the memex, it is obvious that even from the very early stages the requirement for personalization was apparent; or in other words the requirement for personalized information retrieval / filtering that constitute a basic type of adaptation in our web days. However as came up from the flashback the need for developing the infrastructures had disoriented the research to more system-centric solutions and the initial design requirements had been neglected. It seems that we are exactly at the moment that the initial requirement is now given paramount importance by introducing more user-centric systems mainly expressed with the introduction of user modeling.

It is obvious that even at the initial stages of user modeling, the latter had not been considered as a separate subsystem but this was used as part of the system's logic. However, in more recent systems the user modeling is being separated and reused. In that sense, user modeling mechanism's nature seems to have similarities with the one's of adaptation. In a similar way, adaptation feature has been appeared into many systems as part of their logic. In the case of adaptation, the separation is seen possible only in the form of aspects (as used in aspect oriented programming discussed in later chapter). In other words, the adaptation as a mechanism need to be separated as "logic" but at the same time it must be distributed to the sub-systems.

In addition, an interesting point to discuss is the nature of hypertext. In this chapter it has been identified that the nature of hypertext fits to already developed human reading and writing models. This is a fact that needs to be considered as an advantage and utilized in favor of the user. It seems that there is a lot of interesting conclusions when comparing the cognitive model of human with the nature of hypertext that can be applied so that a future adaptive web information system would be capable of adapting to the cognitive style of the user. This is even more interesting in the case of cognitive disabled users.

Next, this chapter has presented several methods, models and techniques having been used in the field of adaptive hypermedia. Such an investigation concludes that many of these can be inherited to the field of adaptive WIS. However, as most of these are strongly related with the application of adaptive hypermedia, these need to be abstracted and adapted to fit the WIS requirements.

A last point that seems worth discussing is the emerging requirement, even from the early stages of adaptive hypermedia to model the environment of the user. However in such early stages [Brusilovsky, 2001] these have been introduced as sole attributes of, for instance, user and device. In contrary, as it will come up in later chapters there have been introduced blending approaches [Velasco et al, 2003].

Chapter 4. Adaptive WIS and Universal Access

4.1 Web accessibility

4.1.1 Web accessibility origins and benefits

Starting from the early days of computers, as briefly outlined in Table 5 ,the concept of "universal access" would have been meaningless. With the spreading of personal computers (PCs) and especially with the introduction of GUIs (Graphical User interfaces) human computer interaction issues have been increasingly raised. Early days software systems were being designed for a particular target group. The characteristics of the potential users were well known and the designer had to satisfy such predefined, narrow users' requirements. Even, in such cases the usability of systems' interfaces proved a difficult task and many of them had really failed to satisfy such requirements and have been rejected by the users. Software engineering methodologies such as object oriented programming, rational unified process, agile unified process and extreme programming (just to name some – see more in Chapter 5) have been developed for improving software design.

Accessibility is by definition a category of usability [Nielsen, 1993]: software that is not accessible to a particular user is not usable by that person. As with any usability measure, accessibility is necessarily defined relative to user task requirements and needs. Accessibility of user interfaces can be approached through usability. International Organization for standardization (ISO) 9241-11: Economic requirements for Office Work with Visual Display Terminals, Part 11: Guidance on Usability, defines usability as the "extend to which a product can be used by specified users to achieve specified goals effectively, efficiency and with satisfaction in a specified context of use". Accessibility focuses on including people with disabilities as the "specified users" and a wide range of situations, including assistive technologies, as the "specified context of use".

In 1989 while working at CERN, the European Particle Physics Laboratory in Geneva, Tim Berners-Lee invented the World Wide Web. WWW has surpassed its original design goals and, as the ways to access the Internet proliferates (along with the range of services offered and fast growing media) new design requirements emerged.

Since then, every human in the world has become a potential user of any system aiming to use that media. The potential user of such a system could be a child or elderly people, an illiterate or a scientist, a genius or a cognitive disabled human scattered all over the world. Tim Berners-Lee stated that "*The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect*" - Tim Berners-Lee

Universal design describes a process that everyone should keep in mind when designing a product. It could be seen as an attempt to merit all group - specific designs processes to a single – universal one. One of the definitions is given by Ron Mace who defines universal design as "*the design of products and environments to be usable by all people, to the greatest extent possible, without the need for adaptation or specialized design*" -Ron Mace [12]. The Center for Universal Design[13] (1997) have developed a list of principles for universal design including Equitable Use, Flexibility in Use, Simple and Intuitive Use, Perceptible Information, Tolerance for Error, Low Physical Effort and Size and Space for Approach and Use.

Nowhere is universal design approach more apparent than in the case of people with disabilities. This user-centered approach can be very useful for increasing the potential for people with disabilities (PwD) to access the contents of web-based applications. Until very recently, many of these new applications actually impeded, or worse, excluded PwDs from using them, because they were not accessible to them. Given that these people are some of those most in need of

12 *About UD: http://www.design.ncsu.edu/cud/newweb/about_ud/aboutud.htm*

13 *The Center for Universal Design (CUD) - http://www.design.ncsu.edu/cud/*

taking advantage of ‘e-life’ applications, such as e-learning, e-shopping etc. because their impairments make it difficult to engage in these activities in the traditional ways, one might have expected these users’ requirements to be central to these new developments, especially in applications that would enable enhancement of quality of life and independence. Accessible design is a narrower approach of universal design and its purpose is to make products accessible to people with disabilities.

The movement to the accessible or more general universal design is basically due to the increase of age of the population and the number of disabled people worldwide. More and more people use the web for accomplishing their everyday tasks. Among people having recently accessed the Internet, more than one third had made on-line purchases for personal purposes[14]. In Europe, about 17% of firms use ICT (Information and Communication Technology) solutions for supporting the marketing or sales processes[15]. On the other hand, lately the percentage of working-age people with a disability has increased in the US[16]. An estimated 10% of the world’s population experience some form of disability or impairment[17]. Only in UK, according to RNIB there are some 8.5 million people who have some form of disability. However, it is only lately that accessibility issues have begun to feature in these applications. This is in part due to public awareness of campaigns for the rights of all users (see video, “Websites that work” on http://www.idcnet.info/wai_video); to regulatory[18] measures in terms of safeguarding these rights (ADA[19], DDA[20]), and encouraging public procurement to stipulate accessibility (Section

14 *Statistics on the information society in Europe data 1996-2002 (http://ec.europa.eu/enterprise/ict/studies/is-stat-96-02.pdf)*

15 *The European e-Business Report 2005 (http://www.ebusiness-watch.org/resources/documents/eBusiness-Report-2005.pdf)*

16 *Rehabilitation Research and Training Center on Disability Demographics and Statistics. (2005). 2004 Disability Status Reports. Ithaca, NY: Cornell University. (http://www.ilr.cornell.edu/ped/disabilitystatistics/)*

17 *Disability and rehabilitation: WHO action plan 2006-2011 (http://www.who.int/disabilities/publications/en/index.html)*

18 *Policies Relating to Web Accessibility - http://www.w3.org/WAI/Policy/*

19 *http://www.usdoj.gov/crt/ada/adahom1.htm*

20 *http://www.disability.gov.uk/dda/*

508[21]), but perhaps most of all, to increased understanding on the part of designers of applications about what is needed.

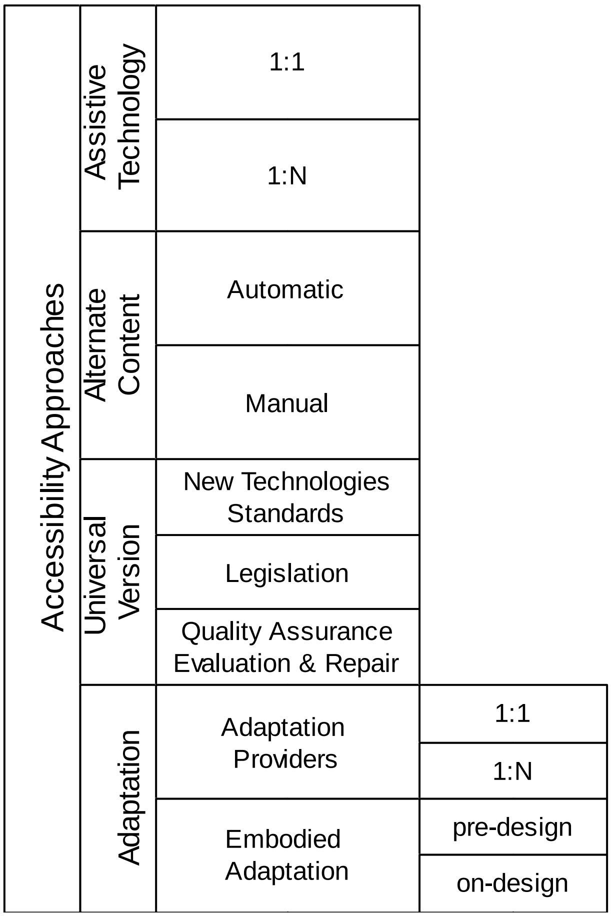
4.1.2 *Facing web accessibility – The mainstreams*

There have been several approaches developed aiming at reaching web accessibility. This section attempts to identify and overview the most important streams that, in this thesis, are considered as milestones in web accessibility history (briefly presented in Table 7)

Stream	Characteristics
Assistive Technology	Assistive Technology as a separate field (separated from application design – designed for a predefined target group)
Alternative Content	• Triggered by user – Customization • Automatically (ex. According to browser, NOSCRIPT, NOFRAMES) • Content Version selection
Universal Version and Standardization (Legislation and E&R)	• From design phase, consider both sighted and non-sighed users. • Emphasis on standardization and guidelines mainly by W3C WAI (also corporations like IBM, Microsoft, SUN). • Relating legislations introduce web accessibility as business requirement. Evaluation and Conformance requirements.
Adaptation	• Adaptation Providers: In this case, the adaptation is provided by separate systems that transforms the output of the application. • Embodied Adaptation: In this case, the adaptation mechanism is embodied into the application.

Table 7: Web Accessibility Streams

21 http://www.section508.gov/



Furthermore, a kind of taxonomy has been developed and presented in Illustration 18

4.1.2.1 Assistive Technology

According to Zajicek & Edwards [Zajicek & Edwards, 2004] first time that accessibility issue was raised originate to text-based interfaces days, which were inaccessible for blind users. Nevertheless, that issue was relatively easy to overcome with the introduction of speaking capabilities to special applications such as word processors (nearly 1983). Later it was realized that a better approach was the screen reader adaptation, through which a whole range of applications became accessible.

While the introduction of the GUI in mid-1980s had the effect of making computers more accessible to more people, this was only true as they were sighted. Even if in the beginning the GUI institution seemed doubtful Microsoft's Windows 3 came to vindicate a few researchers that have predicted such an intrusion. Now, non-slighted people were again in a backseat. Even if this time it was more difficult, researchers managed to overcome problems and develop screen readers that were able to handle GUI applications. Zajicek & Edwards [Zajicek & Edwards, 2004] discusses that screen readers have really succeeded if and when their users can perform all the same tasks as efficiently as their sighted persons. That was actually the beginning of a whole research field named assistive technology aiming to deal with disability problem in information society.

Disability Discrimination Act (DDA) in 1995 defined disability as “*a physical or mental impairment that has a long-term or substantial effect on a person’s ability to carry out day to day tasks*” - DDA, 1995. According to RNIB[22], the types of disability range from people with physical and sensory impairments to people with diabetes, disfigurements, heart disease and epilepsy. Not all of these conditions affect how an individual may use software. Such a classification includes Eyesight (people with no vision, or some functional vision), Hearing (people who are completely deaf or have partial hearing in one or both ears and require the use of a hearing aid), Mobility (a wide range of people with varying types of physical disabilities) and Cognitive (people with dyslexia and learning difficulties). It should be mentioned other similar classifications from organizations like WebAIM's one[23] (Visual, Auditory, Motor, Cognitive and Seizure) and GNOME Accessibility Project[24] (Visual impairments, Mobility impairments, Hearing impairments, Seizure disorders and Age-Related Impairments). W3C defines Assistive Technology as “*Software or hardware that has been specifically designed to assist people with disabilities in carrying out daily activities*”.

In the early period, according to Stephanidis [Stephanidis, 2001], accessibility problems were primarily considered as concerning only the field of Assistive Technology (AT), and consequently, access entailed meeting prescribed requirements for the use of a product by people with disabilities.

---

22 *RNIB (Royal National Institute of the Blind) - Types of disability - http://www.rnib.org.uk/xpedio/groups/public/documents/publicwebsite/public_sactypes.hcsp*

23 *WebAIM: http://www.webaim.org*

24 *GNOME Accessibility Project: http://developer.gnome.org/projects/gap/disability-types.html*

4.1.2.2 Alternative Content

With the global penetration of digital computer in business activities, the accessibility issue re-appeared, as disabled and elderly people faced serious problems in accessing computer-based services. Today, accessibility resurfaces as a critical quality target in the context of the emerging Information Society. Stephanidis [Stephanidis, 2001] mentions that currently Universal Access refers to the global requirement of coping with diversity in the target user population and their individual and cultural differences, the scope and nature of tasks and the technological platforms and the effects of their proliferation into business and social endeavors.

Recently, the problem of accessibility seems to be serious in the domain of the World Wide Web mainly due to its distributed nature (as discussed earlier) and relating emerging and mixed technologies like multimedia.

Early web accessibility engineering moves were characterized by the creation of different versions of pages ([Stephanidis, 2001];[Masuwa-Morgan & Burrell, 2004]). Text only versions that are more accessible to text only browsers and to screen reading technologies, print friendly version, versions in different languages could be triggered by the user explicitly (e.g. click) or semi-automatically (ex. NOSCRIPT, NOFRAMES HTML tags). Even from such an early stage a small number of simple adaptation techniques seem to appear (ex. Different version per browser product).

4.1.2.3 Universal Version and Standardization

Another accessibility direction was the “one size fits all” approach that looked away from multiple structure approach and focuses on a minimalist concept universal design (eg. single versions which are accessible to everyone, however they access the internet). “*In theory, the designer of information should not have to worry about producing several versions of specialized web pages or sites. Rather, the focus should be on designing the source page with a rich set of characteristics that can subsequently be rendered or viewed by a wider audience…*” [Paciello, 2000]. However, this has not precluded the incorporation of exciting technologies which enhance graphical presentation [Masuwa-Morgan & Burrell, 2004]. RNIB notes that:“*Accessible (web) pages* *should not be boring! They can be well designed, fun and attractive at the same time providing access to everyone*." - RNIB, 2001

Masuwa-Morgan & Burrell mention that, from such a viewpoint, what matters is that all users are able to manipulate the interfaces with the same ease, regardless of their ability/disability. To this direction the web was pushed by the introduction of new technologies. Various specialized extensible markups have emerged which are specially aimed at addressing particular accessibility problems including Cascading Style Sheets including CSS2 audio styles, W3C's Synchronised Multimedia Integration Language (SMIL), Xforms (the next generation of forms for the Web) and SVG (a language for describing two-dimensional graphics in XML).

For accomplishing web accessibility requirement the WWW Consortium introduced the Web Accessibility Initiative (WAI)[25]. Its aim is to develop strategies, guidelines and resources to make the Web accessible to people with disabilities. Activities of WAI include:

Development of guidelines and techniques: WAI has developed guidelines for accessible web content (WCAG), for authoring tools aiming to help authors developing content that conform WCAG and for user agents indicating how to make them accessible to people with disabilities. The tremendous evolution of web technologies made the initial version of WCAG (WCAG1.0) inadequate for covering web content accessibility issues (HTML centric). That led to the development of WCAG2.0 under a technology independent approach. General principles and technology independent success criteria have been developed aiming to catch upcoming technologies. Then, separate documents are developed specifying abstract web content accessibility requirements per technology. Currently WCAG2.0 is a Working Draft (27 April 2006, when a last call for review has been announced).

Managing accessibility: A very important task WAI has undertaken is to promote accessibility; convince owners and developers to seek for accessible web. Such activity involves investigation of

25 *Web Accessibility Initiative (WAI) – http://www.w3.org/WAI/*

policies and legislations supporting web accessibility such as Rehabilitation Act (Section 508[26]), introduction of business cases and offering training.

Evaluating accessibility: For retaining quality assurance from the accessibility point of view WAI investigates evaluation methods and tools. An important issue here is the “measurement” of web accessibility for conformance claiming. Recently WAI has also initiated the Evaluation and Repair Tools Working Group (ERT WG)[27] aiming to develop an infrastructure (EARL[28]) for communicating evaluation results in a standardized way promoting quality assurance of web content.

Important work in this field takes place under the EU Web Accessibility Benchmarking Cluster (WAB)[29], a cluster of European projects to develop a harmonized European methodology for evaluation and benchmarking of websites. WAB has recently released the Unified Web Evaluation Methodology for websites (UWEM1.0[30]), a Web evaluation methodology that provides an evaluation procedure consisting of a system of principles and practices for manual and automatic evaluation of Web accessibility for humans and machine interfaces. WAB projects are also developing an accessibility quality mark (Support-EAM[31]) and benchmarking tools (BenToWeb) for accessibility, including test suites for WCAG2.0 (further details in chapter 7.2).

WAI summarizes the different components of Web development and interaction work together in order for the Web to be accessible to people with disabilities. These components include:

- content - the information in a Web page or Web application, including:
    - natural information such as text, images, and sounds

---

26 *Section 508: http://www.section508.gov/*

27 *WAI Evaluation and Repair Tools Working Group (ERT WG): http://www.w3.org/WAI/ER/*

28 *Evaluation and Report Language (EARL) 1.0 Schema Working draft - http://www.w3.org/TR/2006/WD-EARL10-Schema-20060927/*

29 *WAB Cluster - http://www.wabcluster.org/*

30 *Unified Web Evaluation Methodology 1.0: http://www.wabcluster.org/uwem/*

31 *Support-EAM: Supporting the creation of a e-Accessibility Quality Mark- http://www.support-eam.org*

    - code or markup that defines structure, presentation, etc.

- Web browsers, media players, and other "user agents"
- assistive technology, in some cases - screen readers, alternative keyboards, switches, scanning software, etc.
- users' knowledge, experiences, and in some cases, adaptive strategies using the Web
- developers - designers, coders, authors, etc., including developers with disabilities and users who contribute content
- authoring tools - software that creates Web sites
- evaluation tools - Web accessibility evaluation tools, HTML validators, CSS validators, etc.

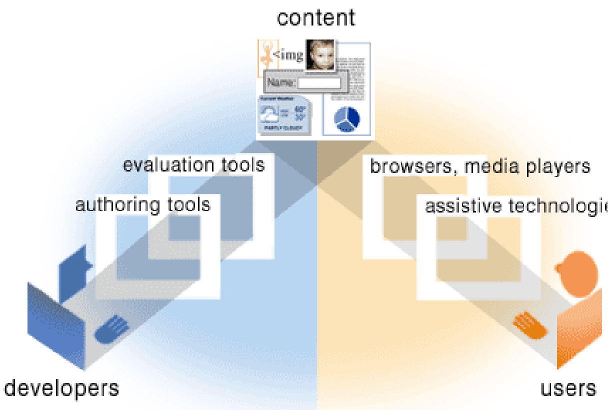


Illustration 19: How the Accessibility Components Relate (WAI web Site)

Finally, it should be mentioned that except for W3C initiative, industry initiatives have also been founded, as a result of active legislation, with major information providers industry leaders such as Microsoft (http://www.microsoft.com/enable/), IBM (http://www.research.ibm.com/access/), Sun Microsystems (http://www.sun.com/access/) and Apple (http://www.apple.com/accessibility/). Sun, for instance, developed the Java Accessibility API that is provided as part of Java Foundation classes and is intended to expose information on user interface objects so that the interface can be better translated into the varying modalities.

4.1.2.4 Adaptation

An important stream of universal access approaches has been adaptation. Adaptation approaches used so far could be distinguished to two categories:

- Adaptation Providers: the adaptation is provided by separate systems that transforms the output of the application.
- Embodied Adaptation: the adaptation mechanism is embodied into the application.

This is a similar classification to Stephanidis' one [Stephanidis, 2001], who distinguishes between Product-level adaptation and Environment-level adaptation. However, the proposed classification's point of view actually responds to the query whether the adaptation mechanism is a subsystem of the application (embodied) or a separate system (provider). Contrary to Stephanidis' classification, such a classification does not consider whether the adaptation functionality is involved during or after the design phase of the application. It seems that product-level adaptation, as defined by Stephanidis, could be a sub-category of embodied adaptation. Similarly, Adaptation Providers category seems to be a broader category comparing with environment-level adaptation. It should be noted that one could also classify here assistive technology as a “separate system” but for the purposes of this work, assistive technology is seen as a non-adaptive technology in the sense that it targets to special – predefined target group.

4.1.2.5 Adaptation Providers

From this thesis' point of view, an adaptation provider is a software that is capable of identifying user requirements aiming at transforming the output of the system of interest in a form that would improve user interaction (i.e. remove access barriers). Such movement could be probably seen in relation to automatic evaluation and repair discussed later. Follows indicative attempts that can be classified under Adaptation Providers accessibility approach:

BETSIE (BBC Education Text to Speech Internet Enhancer) http://www.bbc.co.uk/education/betsie/index.html). BETSIE is a text filtering software tool (written in Perlscript) created in 1998 by the BBC in collaboration with RNIB. BETSIE is quite effective in reading the contents of web pages. It however, has limitations when it comes to the interpreting of interaction support components (buttons, icons…). When a user makes a request for a web page, BETSIE removes all the images and the unnecessary formatting, so that what is displayed is the text content of the page, with headings at the top in their original form.

Lynx Viewer text browser[32] service allows web authors to see what their pages will look like (sort of) when viewed with Lynx, a text-mode web browser.

Tablin (http://www.w3.org/ WAI/Resources/TABLIN/) is a filter program developed by the WAI Evaluation & Repair (ER) group that can linearize HTML tables and render them accordingly to preferences set by the presentation layer (e.g. the screen reader end-user).

W3 Access for Blind Peoples (W3ABP) server [Perrochon & Kennel, 1995] uses the web's proxy server technology to automatically change format of a web page adding contextual information for digestion by screen readers.

NCAM[33] takes the view that the key to accessible media is to ensure that the media contains accessible code that can be read by access technologies.

Aurora [Huang, 2000] is an extensible transcoding system that targets and adapts content in existing Web pages to help the broadest population of users, particularly in the disabled community to obtain various Web-based services. The system adapts web content based on semantic rather than syntactic constructs – facilitating navigation by streamlining the web interface according to abstract user goals.

Web Accessibility Service: Fairweather et al [Fairweather et al, 2002] discusses the development of an intermediary-based strategy to enhance access to web that relieves some of the problems that beset conventional client-based assistive technology by moving the control point for accessibility away from the client and enable this service to compose adaptations tailored to a user's particular preferences and capabilities. The page is retrieved, parsed, and converted into its corresponding Document Object Model (DOM) form. Some transforms, such as those applied to text, can be executed with style sheets. Others, such as image transforms, require programmatic manipulation. Still others, such as auditory rendering or dynamic keyboard adaptation, are handled by injecting JavaScript and Java applets into the page so that the transformations can be handled locally on the client device.

32 http://www.delorie.com/web/lynxview.html

33 CPB/WGBH National Center for Accessible Media (NCAM) @ http://ncam.wgbh.org/

AcceSS: Parmanto et al [Parmanto et al, 2005] attempted to improve users with visual impairments web experience by simplification and summarization. Simplification is achieved by retaining sections of the web page that are considered important while removing the clutter. The purpose of summarization is to provide the users with a preview of the web page. Simplification and summarization are implemented as a “guide dog” that helps users navigate the entire web site.

Accessibility Agent [Kottapally et al, 2003]: This attempt aims at developing software agents in order to assist visually impaired users in navigating complex Web pages. The system relies on an explicit encoding of the navigational structure of the document, and on the use of planning technology to assist users in simple and complex query-answering tasks employing machine learning techniques. A planner allows users to give directive and simple goals as well as the first type of complex goals. This is integrated in an infrastructure that automatically captures user requests, analyzes incoming documents, and interacts with the user via keyboard input and aural output. According to the authors, the planner has been implemented but not yet integrated into the whole system. Further, the authors will be working on the development of a natural language processing interface to the software agent for visually-impaired individuals.

eAccessibilityEngine [Alexandraki et al, 2004] aims at transforming non-accessible web pages into accessible forms; the actual “output” of the tool can vary in accordance with specific user needs, and the assistive software and hardware available to the user for accessing the web. Disabilities are modeled using stereotypes, and each stereotype is associated with a set of accessibility transformations that adapt web pages according to the needs of the respective group of users. Transformation Processing can perform modifications both at the physical level of interaction (e.g., fonts and colors), and at the syntactic level of interaction (i.e. re-structuring) of web documents. Processing functionality is based, to a large extent, on the ability to “recognize” and address patterns of inaccessibility in web pages (e.g., HTML tables used for page layout). The eAcccessibility engine is implemented as a web-based service using technologies Java and XML technologies.

Squid-based adaptation [Canali et al, 2005] provides on-the-fly image transcoding services on the basis of the client device capabilities and operating on various image parameters, such as spatial geometry, color depth, quality factor and MIME subtype. The implementation was based on a standard Squid proxy server by extending it to support adaptation and multi-version caching functionalities on the intermediary node.

Triage Tool: Harper & Bechhofe [Harper & Bechhofer, 2005] propose that the inclusion of semantic information directly in XHTML is the only effective way to assist users who are visually impaired to access Web pages and, at the same time, avoid decreasing or compromising the creative activity of authors and designers. They name their research as Low-Cost Lightweight Instance Store (LLIS) research approach as it investigates how can semantic information can be built into general purpose Web pages, without compromising the page's design vision, such that the information is as accessible to visually impaired users as it is to sighted users. They have developed LLIS Triage Tool, as a Mozilla extension. This is unique in that it uses an ontology created from a preexisting CSS to allow triage of XHTML components. The CSS (component 1) is used to create concepts in the ontology (component 3). The LLIS application then retrieves the ontology, much as Mozilla retrieves the CSS document. The ontology is passed to the Ontology Service, and LLIS can now ask questions based on the actions required.

Semantic Web Accessibility Platform (SWAP) [Seeman, 2004] is a semantic web; knowledge based approach to accessibility that creates alternative renderings of sites aiming at enabling people with diverse special needs to smoothly and easily access the content. SWAP uses annotation, which reflects extra accessibility related information about each page. The annotations are invisible to the average user but are critical to enabling the disabled to access the web.

accessibilityWorks project [Hanson et al, 2005] provides software enhancements to the Mozilla Web browser and allows users to control their browsing environment. This is a continuation of **Web Adaptation Technology** (WebAdapt2Me[34], WebAdapt[35]) project which was implemented on Internet Explorer. Functionality include enlarging page content, Enhancing text, Reducing visual clutter, Enlarging browser controls, Adapting keyboard and mouse settings.

---

34 http://www-03.ibm.com/able/solution_offerings/WebAdapt2Me.html

35 http://www.webadapt.org/

As a conclusion to this section, follows a classification and summary of extracted features the systems under investigation offered:

- Page Content
  - Syntactic
    - remove all the images
    - linearize content and HTML tables
    - ensure accessible code that can be read by access technologies
    - re-structuring of the document
    - image transforms
  - Semantics
    - simplification and summarization
    - inclusion of semantic information directly in XHTML
    - uses annotation - extra accessibility related information about each page
- Formatting
  - Enlarging page content—magnifying pages and enlarging specific text or images
  - Enhancing text— changing colors, letter and line spacing, and text style
  - Enlarging browser controls
  - Remove unnecessary formatting
- Navigation
  - adapts web content based on semantic rather than syntactic constructs – facilitating navigation by streamlining the web interface according to abstract user goals
  - software agents & planning technology to assist users in simple and complex query-answering tasks employing machine learning techniques
  - natural language processing

4.1.2.6 *Embodied Adaptation*

From this thesis' point of view, a system is classified under embodied adaptation approach if the actual adaptation mechanism is a subsystem of the whole system.

People working on web accessibility issues have been inspired from adaptive hypermedia research and user modeling. In addition to aforementioned approaches that are either involved in design phase or developed as separate systems, accessible web has borrowed methods and techniques from adaptive hypermedia and adaptive web. According to [Stephanidis, 2001], the first attempts to bring the two communities closer were initiated in the 1990s (e.g. the ACCESS Project). The basis of adaptivity in AVANTI framework is its user modeling component, the User Modeling Server (see more about UMS in [Kobsa, 2001]).

Stephanidis [Stephanidis, 2001] reports that the user characteristics that trigger appropriate adaptation types at the content level mainly concern the type of disability, the expertise and the interests of the user. The resulting adaptations mostly concern alternative presentation using different media, additional functionality, conditional presentation of technical details, conditional presentation of details that are of interest to users with specific disabilities only and "role-taking" facilities allowing users to identify themselves as having a particular disability, active interest, etc.

Furthermore, the knowledge about the user and the interaction session is mostly based on information acquired dynamically during run-time (e.g. navigation monitoring, user selections, explicit user invocation), with the exception of the initial profile of the user, which is either retrieved from the UMS, acquired through a questionnaire during the initiation of the interaction, or retrieved from a smart card. One of the important conclusions of such work is "*The first important design lesson is that adaptation needs to be designed into the system rather than decided upon and implemented a posteriori.*" [Stephanidis, 2001]. Chapter 7 will discuss the contribution to the field of two IST projects; IRIS and BenToWeb and further present relating software prototype systems.

4.2 Web Portals' Revolution

Starting from 1983 (first assistive technologies) we count more that two decades of research and development in the field of web accessibility. Through all those years the technological advances offered solutions to accessibility issues but only after having raised them. GUI was designed to make use of ICT wider but at the same time excluded PwD. Nowadays Rich Client technologies like AJAX have appeared for making web user interface experience more effective but once again, PwD seem to be left out of the design process. And now researchers are working on offering accessibility to such technologies (Accessibility for Rich Internet Applications / WAI-ARIA Suite). Have these technologies been designed having accessibility in mind? Have the PwD been involved in their design phases? Should we spend time to make accessible applications using inaccessible technologies? Even if technologies are accessible does this mean that developed applications will be accessible? It seems that the answer on these questions is that the design of upcoming technologies needs to follow a universal design approach and further promote / enforce the use of accessibility features.

Such a requirement becomes now more than ever apparent with the introduction of web technologies to ubiquitous applications like handheld devices and interactive television. Last, but not least, the portal paradigm seems to be an example of upcoming domain reusable paradigm that meets special requirements mainly due to each complexity.

4.2.1 Reaching Web Portals

In its short life, the web has proven to be much more than a "single service"; instead it has become a successful media for delivering services in many areas, such as communication, education, entertainment, etc. From its origins based on the requirement of sharing scientific work that included simple text and images, its content representation vehicle was hypertext (later called hypermedia) coined by [Nelson, 1965]. However Nelson criticized, and still criticizes [Nelson, 2003] the web as an unsuccessful design mainly because of its unmanageable structure. In spite of this, the web simplicity concept and technology has made it popular, the web counts for some billions of pages. The one side of this coin was the availability of huge free information while flipping it around, a management and retrieving of this information was faced. Aiming to solve afore mentioned problems, search engines appeared offering a full text index of hypertexts. That was an important step coping with web chaotic structure. So, most of web surfers used those search engines as their entry point to the huge web. That was the birth of the “portal” term.

Portals such as Yahoo and Excite introduced the next portal phase that extend the pure search engines by offering content organized hierarchies organized by people in a more semantic way (more than keyword based).

Since then, portals have been evolved offering functionalities [Winkler, 2006] such as: 1) search and navigation, 2) information integration, 3) personalization, 4) notification, 5) task management and workflow, 6) collaboration and groupware, 7) integration of applications and business intelligence and 8) infrastructure functionality. However, the indisputable breakthrough in Web search technology and functionality pioneered by Google triggered a change of the web portal’s functional model; from one-track “information accessing”, to combined information accessing and service offering, with the emphasis put on the latter. That change took place in the late 90s with portal sites adding various features and services (e-mail, community tools, stock quotes, news, etc.) on their previously main function, “search” [Monohan, 1999], [Ledbetter, 1999].

At the same time, the introduction of personalization took place (as a fruit of user-modeling) and constituted a significant development for portals. Web has moved from a content-centric pattern to a user-centered one and that was pushed with the developments of user modeling [Kobsa, 2001]. Furthermore, new technologies has evolved web as a multimodal interaction service. Currently, we are experiencing a regeneration of the web with the introduction of semantic web, which might offer the manageability that Nelson is seeking for to an already wide-accepted service.

One of the distinctions of portals is made between “horizontal” and “vertical”: the formers refer to those that contain a huge variety of thematic content (also known as megaportals) like Excite and Yahoo; whereas vertical portals are those that serve a more specific and narrower scope and provide knowledge in greater depth. For instance, in EQUAL – SYMPOLITIA[36] a portal infrastructure is being used as a virtual incubator for communication and learning between educators, consultants and interested parties, while in EQUAL-EUNETYARD[37] it is used as the information and communication tool between the project's partners.

From a content management point of view, as the functionality and management of web applications become more and more complex, companies have been motivated to invest in portal systems as a mechanism by which they can manage information in a cohesive and structured fashion [Bellas et al, 2004]. Portals are used in enterprises both as their communication link with their customers, or maybe resellers, and often also as their intranet online management information tool. This also applies to education U-Portal[38], PORTAL, where there is a tendency from institutes to develop integrated and shared portal frameworks for moving all their services, back office operations and even teaching content online.

The wide use of portal term has introduced confusion regarding the term's use. There are many definitions for "portal" and very many web pages contain in their heading the term "portal". Smith [Smith, 2004] combined many of these and define portal as "*an infrastructure providing secure, customizable, personalizable, integrated access to dynamic content from a variety of sources, in a variety of source formats, whether it is needed*". [Smith, 2004]. Another widely accepted definition, which comes from the e-business perspective is:"*a portal is a single integrated point of comprehensive, ubiquitous, and useful access to information (data), applications, and people"* [Saha, 1999].

While "portal" has been consecrated as a paradigm for more and more applications such as e-commerce, collaborative environment and entertainment implementation infrastructures[39] and protocols have been developed. Recently, standardizations efforts have been taken place aiming at allowing for interoperability. Web Services for Remote Portlets (WSRP) [Kropp et al, 2003] and

---

36 *http://simpolitia.syros.aegean.gr*

37 *http://www.eunetyard.net*

38 *http://www.uportal.org/*

39 *Apache Portals: http://portals.apache.org*

JSR-168 [Abdelnur & Hepper, 2003] specifications are two well established specification documents that assure for interoperability. JSR-168 is a standard Java interface for portlets that builds on the J2EE programming model of servlets. This is an interface between a particular Java type of UI component and its hosting container. On the other hand, WSRP is a platform independent standard messaging interface for interacting with remote compliant UI components. In both specification these UI component are named "portlets". In the context of WSRP "*A portlet is a user-facing, interactive application component that renders markup fragments that can be aggregated and displayed by a portal.* "[40]. However, the formal definition of portal was given by the Java Community Process in the Java Specification Requests JSR-168 Portlet Specification: "*A portal is a web based application that –commonly- provides personalization, single sign on, content aggregation from different sources and hosts the presentation layer of Information Systems...*" [Abdelnur & Hepper, 2003]

While reviewing the relevant literature, a central notion to the meaning of "portal" is emphasized, that of an "*integrated, single point of access to a variety of (information) sources*". It is thus clear that integration is inherent to Portals as is the concept of "*a gateway to information*". The main portal characteristics are:

- **Search:** Search was the main function of the early portal implementations. Today's portal functional models position the search function in a differentiated, less central context and a less integral part of the overall portal strategy [Clarke & Flaherty, 2003]. Search now is moving from classic pure keyword based search to more personalized and semantic search.

- **Content aggregation**: The "single point of access" characteristic is given to portals through content aggregation from a variety of diverse information sources. The information providers may be either structured or unstructured, including databases, file systems, news feeds, the WWW, email servers etc, and either be internal or external. The process of aggregating content and presenting the information is transparent and seamless to the end user. Often, portals constitute the front-end (user interface) to Web Services (or/and

40 *http://www.oasis-open.org/committees/download.php/11774/wsrp-faq-draft-0.30.html#portlet*

WSRP). In addition, portals use content syndication standardized mechanisms like RSS to provide news headlines from various web capable news agencies.

- **Personalization - Customization**: A fundamental portal feature is the user ability to personalize portal's look and feel, layout, information and applications sources and services, according to her needs. This is done by providing customization facilities to certain aspects of the portal, which may include portal pages, layout-navigation, application preferences, services etc., the extent of which depends on the technology used and the design goals. Techniques coming from user modeling, adaptive hypermedia and context-aware systems are employed for more appropriate content delivery.

Furthermore, it is possible to draw a comparison between portals and Graphical User Interfaces (GUIs), in that portals provide a uniform way of accessing applications and content on the web, just as GUIs provided to operating systems ([Strauss, 2001];[Fred & Lindesmith, 2003]). Or, alternatively, portals could be seen as operating systems that provide a basic functionality and on which portlet applications can be based (according to certain specifications). So, portals could be seen as the future personal roaming desktop (working environment) by providing all the tools a user would need in a personalized manner. That brings memories from the terminal machines using the power of centralized servers, thus away from the user. Such an evolution would have a great impact both in software and hardware enterprise as both would need to move away from the end user.

Furthermore, WSRP (Web services for remote portlets) specifications, show off the new emergent web applications paradigm. These are blending the power of web services with the standardization of portlets and provide specifications for remote portlets both in a functional and user interface level (pushing the service oriented architecture), which means that a portlet designer would need to develop a portlet user interface without being aware of the rest of the portal.

Thus, WSRP, show off the emergent web applications paradigm; developing a web application as a portlet it makes it pluggable to any portal that conforms aforementioned specifications. It is blending the power of web services with the standardization of portlets and provide specifications for remote portlets both in a functional and user interface level (pushing the service oriented architecture), which means that a portlet designer would need to develop a portlet user interface without being aware of the rest of the portal. Furthermore, scenarios of m-commerce and location based services make such systems more distributed; introducing new requirements (see for instance Mobile Web Best Practices [Rabin & McCathieNevile, 2006]).

Portal, both as a paradigm and as web infrastructure has been used both in e-commerce and m-commerce fields showing off the importance of personalization and its accompanied attributes in the businesses domain. In the next sections a presentation of the former role with emphasis to HCI and more specifically to web accessibility issues takes place.

4.2.2 Portals in e-Commerce

Many e-commerce firms have adopted the “Portal” approach as the vehicle to perform their electronic commerce/business transactions. This is an integrative, user-centric and user-intensive approach, which offers various advantages and benefits to all stakeholders involved. Focusing on the end-customer, Van Riel et. al. [van Riel et al, 2001] state that “*portals offer many functional advantages over traditional media, such as easy access to several related services, access to almost unlimited content and excellent retrieval facilities*”. Furthermore, Grewal et al [Grewal et al, 2004], argued over portals’ potential to improve significantly the ability of customers to find specialized products via the plethora of search tools that they offer.

Regarding the corporate perspective, Meister et al [Meister et al, 2000] concluded that portals are valuable tools in building relationships with customers irrespectively of the e-business model and market space (B2B, B2C, C2C). They engender e-loyalty through the rich variety of content and the consequent ability to offer greater choice, emerging as “*top-of-mind destinations for one-stop shopping*” [Srinivasan et al, 2002], which is of paramount importance [Reichheld & Schefter, 2000]. However, the added aids to commercial success from a customer-centric point of view are not the only factors for the popularity of portals. Portals also function as infrastructures for information management, “*providing firms with a shared information work space that facilitates access to information content, organizational communications, and group collaboration*” [Detlor, 2002]. These Enterprise Information Portals constitute "*the vehicle to integrate and extend an entire business, delivering solutions and services over the Web"* [Hazra, 2002].

The existing literature with respect to portal and service quality assessment by consumers demonstrates the high importance of the user interface. However, the important issues of accessibility and usability are not properly addressed [Godwin & Haenel, 2002], although there are approaches that attempt to tackle these problems [Vlachogiannis et al, 2005].

4.2.3 Portals in m-Commerce

On 26th June 2006 Yankee Group announced the results of its 2006 Transatlantic Wireless Business Survey. According to that, "*the percentage of mobile workers in European small businesses continues to rise as mobile investments become a business priority. More than 50% of small business employees are classified as mobile workers, spending more than 20% of their time away from their primary workspace. This figure grew from 48% in 2005*".

Current generations of mobile phones and infrastructures named 2,5G and 3G are connected to digital communications infrastructures constituting a global network. Such a media can host a wide number of services including electronic commerce, known as m-commerce.

According to Okazaki [Okazaki, 2005], the first major m-commerce journal was published in Electronic Markets in 2002, followed by International Journal of Electronic Commerce, Decision Support Systems, and Journal of Business Research, amongst others. There are cited several definitions including: "*the emerging set of applications and services people can access from their Internet-enabled mobile devices.*" - *[Sadeh, 2002]* and "*the use of mobile, wireless (handheld) devices to communicate and conduct transactions through public and private networks*" - *[Balasubramanian et al,2002].*

To the contrary of the new services opportunities that m-commerce brings, it seems that its development has been comparatively slow [Mylonakis, 2004] and research suggests that it is seen as expensive, with poor service and usability [Jarvenpaa et al, 2003].

On the other hand, Dholakia et al [Dholakia & Rask, 2002] states that "*several characteristics of mobile networks make them more attractive than fixed networks for less developed countries and* *for those countries that want to "leapfrog" the leading IT nations*". This contradiction, between the high penetration rate of mobile devices and the low adoption rate of m-commerce, was recently Khalifa & Shen's study's motivation. For their study [Khalifa & Shen, 2006] they have investigated the theory of planned behaviour and technology acceptance model (TAM). Their findings include identification of five expected benefits that perceive to be important by potential m-commerce adopters, e.g., cost, convenience, privacy, efficiency and security. Further, Okazaki [Okazaki, 2005] states that "*one major problem in m-commerce research is the lack of standards in terms, concepts, and theories*". In addition, Harris at al [Harris et al, 2005] research findings provide some support for the view that culture plays a major role in shaping usage of and attitudes to m-commerce services.

In the research agenda, after having developed mobile internet infrastructures, security issues etcetera for enabling services delivery to the user, human-mobile interaction issues are being raised which prove to be the ultimate factor for the adoption and successfulness of mobile commerce. According to [Chan et al, 2002], "*early days research has focused on very narrow tasks for m-commerce and only on the impact of single form factors*". Chan et al article discusses usability findings due to constraints imposed by form factors. It is stated that "*the greatest challenge for various m-commerce application is their usability*" and this work concludes with additional recommendations for mobile interaction design. Also, [Venkatesh et al, 2003] cites studies that have shown that "*user interface features, such as page and content design are key determinants of sales in online stores*" and also that surveys of mobile internet users indicate that "*usability is the biggest source of frustration*".

Summarizing, the emphasis on mobile interface design is revealed around 2003-2004 by [Tarasewich, 2003], [Venkatesh et al, 2003], [Lee & Benbasat, 2003] and [Lee & Benbasat, 2004]. Also, in 2006 there seem to be a lot of interest both in enterprise ([Frank, 2006], (Adobe XD Team Weblogs[41])) and in academia [Jones & Marsden, 2006]. As appearing in [Jones & Marsden, 2006], mobile interaction design has been introduced as a subtopic of HCI. But how that differs from common interaction design?

41 *http://weblogs.macromedia.com/xd/*

Tarasewich [Tarasewich, 2003] describes the benefits and limitations of various wireless device interfaces by distinguishing between input and output interaction devices. Such a survey seems really useful for designers new to the mobile-world. This actually illustrates the different interaction patterns introduced in wireless devices comparing with common desktops ones. He cites a summarization of such differences [Holland & Morse, 2001]:

- Limited user attention given to the device and application (interactions with the real world being more important)
- User's hands being used to manipulate physical objects other than the device
- High mobility during the task, with the adoption of a variety of positions and postures
- Context dependent interactions with the environment
- High speed interactions with the device, driven by the external environment

Further, he investigates ways that such devices' usability could affect m-commerce success. He concludes with the additional challenges a designer faces designing for wireless devices including:

- Whether or not graphical or windows-based interfaces are appropriate for mobile devices
- Contexts such as location, time of the day, temperature and weather conditions can be taken into account
- Minimal Attention User Interfaces (MAUI)
- Sentient computing – Devices might also receive input from the surroundings rather than from the user.

Lee and Benbasat's [Lee & Benbasat, 2004] interpretation of previous studies is that "*e-commerce interfaces should not be directly applied to m-commerce interfaces, given the substantially different computing environments and device constraints*" but at the same time that "*current perspective on e-commerce is also applicable to m-commerce interfaces*". Based on that, they have investigated [Rayport & Jaworski, 2002] framework of seven design elements for customer's interfaces (7cs framework) and adapt it to m-commerce case by complementing it with two more that apply horizontally to 7 formers. Consequently we have context, content, community, customization, communication, connection and commerce, plus mobile setting and mobile device constraints.

In the article of Brian Frank [Frank, 2006] more practical issues of mobile interface design are illustrated from a commercial point of view. Some of merging issues include:

- Mobile devices have limited capabilities in comparison with desktops – the design needs to enable the interaction (input devices, visual display)
- Limit primary pieces for interaction with user to three to five. Moreover, AI techniques have been developed to eliminate the species personalized manner [Smyth & Cotter, 2003].
- The relationship user-mobile is more personal than user-desktop.
- The mobile user most of the times does short-term tasks and need to be done quickly
- Situation design: no more design for capabilities but for environment.
- Time (NOW) is important
- The diversity of the devices themselves presents a challenge.

Venkatesh et al [Venkatesh et al, 2003] results strongly suggest that “*relevance, structure, and personalization are essential to creating a positive wireless interface experience”*. So, personalization and more value added services need to accompany m-commerce services, for driving them to success. Dholakia & Rask [Dholakia & Rask, 2002] suggests that m-portal need to “*focus on personalization, permission and specification of content in order to offer extended mobility and locability for the user*”.

In the area of implementation of m-portals Chen et al [Chen et al, 2005] introduces the term “m-service” that extends the concept of web service to the wireless domain. They further propose a service oriented architecture of an “m-service portal” giving emphasis to “intelligent m-services”, context-aware / semantic-enabled agent-like architectures to improve adaptability and flexibility of m-service portal. Liao et al [Liao et al, 2005] also propose the introduction of semantic web technologies for improving the matchmaking between user requirements and product specifications and make m-commerce systems more intelligent in taking different actions according to different user context environments.

4.3 Towards the semantic web

4.3.1 SW Introduction

Maybe the most recent response to the unmanageable structure of the web, mainly due to the latter prodigious evolution, is named semantic web. According to [Shadbolt et al, 2006], the original Scientific American article on the Semantic web appeared in 2001. In that article Berners-Lee et al [Berners-Lee et al, 2001] state that *"The Semantic Web will bring structure to the meaningful content of Web pages, creating an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users ... The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation."* [Berners-Lee et al, 2001]. In other words, the Semantic Web is an evolving – parallel - extension of the current one, in which information is given well-defined meaning. The vision of the Semantic Web is as an extension of Web principles from documents to data. Currently, data on the web is hidden away in HTML files. It is designed for humans to read, not for computer programs to manipulate "meaningfully". Computers can adeptly parse Web pages for layout and routine processing but in general, computers have no reliable way to process the semantics. According to W3C Semantic Web Activity Web Site[42], the Semantic Web will allow two things;It will allow data to be surfaced in the form of real data, so that a program doesn't have to strip the formatting and pictures and ads off a Web page and guess where the data on it is and It will allow people to write (or generate) files which explain—to a machine—the relationship between different sets of data. Thus, the Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries.

Often there is a misconception about the relation of the Semantic Web and Artificial Intelligence. As *"for the semantic web to function, computers must have access to structured collections of information and sets of inference rules that they can use to conduct automated reasoning"* [Berners-Lee et al, 2001], eventually some parts of the Semantic Web technologies are

42 *W3C Semantic Web Activity Web Site: http://www.w3.org/2001/sw/*

based on results of Artificial Intelligence research, like knowledge representation (e.g., for ontologies), model theory (e.g., for the precise semantics of RDF and RDF Schemas), or various types of logic (e.g., for rules). On the other hand, for instance, even if knowledge representation has done some progress, this has not changed the world; *"It contains the seeds of important applications, but to realize its full potential it must be linked into a single global system*" [Berners-Lee et al, 2001]. WWW can be seen as such a system but at the same time decentralized; a system that could offer the required variety for a rich and evolving knowledge system. The worries that semantic web cannot be realized as in a similar way that AI did not fulfilled its too high expectations is unjustified. The realization of the SW does not rely on human-level intelligence; "*if the ultimate goal of AI is to build an intelligent agent exhibiting human-level intelligence (and higher), the goal of the Semantic Web is to assist human users in their day-to-day online activities*"[Antoniou & Harmelen, 2004].

4.3.2 SW Components

The development of the SW follows a layered approach. A layer is being built on top of another step by step. Having several research groups moving in different directions, it is difficult for the industry to adopt the results and the SW vision becomes unrealizable. This leads to the requirement for standardization, the role of W3C. Under W3C a common road map has been designed and layer by layer technologies are being developed for realizing the vision. Illustration 20 presents the layers of SW with the relevant technologies.

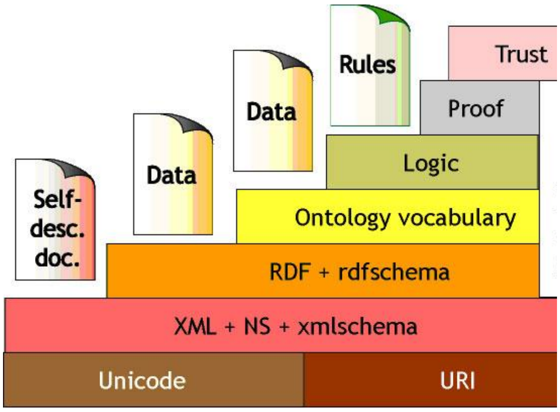


Illustration 20: The Semantic Web Stack (presentation by Tim Berners-Lee)

Starting from the bottom, we can identify six layers discussed in the following sections.

4.3.2.1 The Syntax layer

As already mentioned, current web can be characterized as a syntax-based web. At this "version" of the web, exchanging documents protocols and technologies have been matured. HTML for presenting documents and URI for allocating them have worked pretty well along several years. The introduction of Unicode has also offered a unique number for every character, no matter what platform, what the program, or what the language is. These are essential components / technologies for the interchange of documents.

Besides that, the interchange of data represented in the SW must be facilitated through concrete serialization syntax. For this purpose, W3C has introduced XML, a recently well adopted language. Comparing with HTML, XML was proposed as an extensible language allowing the user to define data structures (his own tags). A data object is said to be an XML document if it follows the guidelines for well-formed documents provided by W3C. However, it should be mentioned that SW is not tied to a particular syntax.

Now, considering XML as the default syntax of SW further technologies extend its capabilities for making it more effective. As XML is a metalanguage for markup that does not have a fixed grammar (set of tags) – but allows users to define tags of their own – it appears that for the communication between two or more parties a way for defining specialized vocabularies - common language of understanding – is required. This role had been initially undertaken by DTD but now this is smoothly being replaced with XML Schema. Such technologies can impose grammatical constraints and ensure successful data interchange. Finally, more technologies are being recommended by W3C for formatting, querying (XPATH, XPOINTER), processing (XSLT), linking (XLINK) and including (XINCLUDE) XML documents.

4.3.2.2 The Data Model layer

Even though XML provides a uniform framework for interchanging data between applications, XML does not provide a way for expressing the semantics of data. For instance, there is no intended meaning associated with the nesting of tags; it is up to each application to interpret this. Resource Description Framework (RDF) has been developed in the corpus of W3C Semantic Web Activity to respond to such a requirement.

RDF is a data model (often called language) which is based on an object-attribute-value triple, called a statement. An abstract data model, like RDF, needs a concrete syntax in order to be represented and transmitted. The most popular syntax of RDF has been XML. However, RDF can also be represented both as triples and graphically (consisting semantic nets). XML syntax has been mainly adopted as this is the most “machine processable” one.

However, RDF offers only binary predicates in a domain-independent manner, in a sense that no assumptions about a particular domain of use are made. Thus, under different domain, terms can be interpreted differently. RDF Schema (RDFS) is introduced by W3C to offer to the designers the ability to define their own vocabularies for their data models by introducing classes and properties hierarchies.

Finally, working with RDF documents XML query technologies like XPATH is not adequate as XML is located at a lower level of abstraction than RDF. RDF query technologies need to also understand the semantic of RDF(s) vocabularies. Currently, RDF Data Access Working Group (part of the Semantic Web Activity) is working on SPARQL[43] a query language for getting information from RDF graphs.

43 http://www.w3.org/TR/rdf-sparql-query/

4.3.2.3 The Ontology layer

From a philosophical point of view, ontology usually is used by researchers as a synonym for metaphysics, as it was used to refer to Aristotle work. The actual "ontology" term was coined in 1613, independently by two philosophers: Rudolf Gockel and Jacob Lorhard and was initially defined as "an Account of being in Abstract" [Smith, 2003]. Philosophical ontology actually seeks not explanation but rather a description of reality in terms of a classification of an exhausted list of entities. In other word it aims to create an exhaustive replication of the world containing all entities and all their relations.

Recently, ontologies have moved from a topic in philosophy to a topic in applied artificial intelligence with great interest on their application to the WWW and more specific to the semantic web. According to [Noy & McGuinness, 2001], an ontology is *“a formal explicit description of a domain, consisting of classes, which are the concepts found in the domain (also called entities)”*. Each class may have one or more parent classes (is-a or inheritance links), formulating thus a specialization/generalization hierarchy; a class has properties or slots (also called roles or attributes) describing various features of the modeled class, and restrictions on the slots (also referred to as facets or role descriptions). Each slot, in turn, has a type and could have a restricted number of allowed values, which may be of simple types (strings, numbers, booleans or enumerations) or instances of other classes. Classes may have instances, which correspond to individual objects in the domain of discourse; each instance has a concrete value for each slot of the class it belongs to. An ontology together with a set of individual instances of classes constitutes a knowledge base.

But what can ontologies offer to the semantic web? The expressibility of RDF and RDF Schema already presented is deliberately very limited [Antoniou & Harmelen, 2004]. The reason is twofold: RDF is limited to binary ground predicates and RDF Schema is limited to subclass hierarchy and property hierarchy, with domain and range definitions of these properties. With RDF Schema, one can define classes that may have multiple subclasses and super classes, and can define properties, which may have sub properties, domains and ranges. In this sense RDF-S is a simple ontology language. However, in order to achieve interoperation between numerous, autonomously developed and managed schema, richer semantics are needed. Antoniou & Harmelen [Antoniou &

Harmelen, 2004] identify the limitations of the RDF(S) to local scope of properties, disjointness of classes, boolean combination of classes, cardinality restrictions and special characteristics of properties. On the other hand, formal semantics that ontologies bring, offer reasoning about class membership, equivalence of classes, consistency, and classification.

The ontology language that have been recently recommended by W3C aiming at satisfying such requirements was named Ontology Web Language (OWL). Its starting points originate to DAML+OIL developed by a joint initiative consisting research groups from both US and Europe. The W3C working group aiming to encounter the requirements for efficient reasoning support and convenience of expression, defined OWL as three different -increasingly-expressive sub-languages, each geared toward fulfilling aspects of this full set of requirements. OWL Lite provides support for classification hierarchy and simple constraints. OWL DL offers maximum expressiveness while retaining computational completeness and decidability (all computation will finish in finite time). The third one, OWL Full, provide the maximum expressiveness and the syntactic freedom of RDF but with no computational guarantees. Thus, OWL Full can be seen as an extension of RDF, while OWL Lite and OWL DL can be viewed as extensions of a restricted view of RDF. An extensive review of ontology languages can be found in [Pulido et al, 2006]. Probably, having only such an infrastructure is not enough for using it effectively. For supporting ontologies' design and development several methodologies and tools have been emerged.

According to De Nicola et al [De Nicola et al, 2005], the first contributions to ontology building methods originate to 1993 when Gruber discussed some basic ontology design criteria; clarity, coherence, extendibility, minimal encoding bias and ontological commitment. Maybe the first complete ontology development process, METHONTOLOGY, is proposed by Fernández et al. [Fernández-López, et al, 1997]. The process is composed by the following phases: specification, conceptualization, formalization, integration, implementation, maintenance. Its life cycle is based on evolving prototypes and specific techniques peculiar to each activity. Other activities, like control, quality assurance, knowledge acquisition, integration, evaluation and documentation are carried out simultaneously with the ontology development activities. Later, Sure et al. [Sure et al, 2004] propose On-To-Knowledge, an ontology development process consisting of five main phases: feasibility study, kick-off, refinement, evaluation, application and evolution. Each phase consists of a number of sub-steps. Other approaches, often tied to industry or research projects, include the methods used for building CyC, SENSUS, and KAKTUS [Fernández-López et al, 2002]. A complete overview of ontology building methods is provided by [Corcho et al, 2003].

For adopting semantic web it is important that tools would be developed and support the whole process described by above mentioned methodologies. Pérez et al [Pérez et al, 2002] in their survey group such tools in the following clusters: Ontology development tools, used for building a new ontology from scratch or reusing existing ontologies; Ontology merge and integration tools, used for merging or integrating different ontologies on the same domain; Ontology evaluation tools, used for the quality assurance of both ontologies and their related technologies; Ontology-based annotation tools, that allow users inserting and maintaining (semi)automatically ontology-based markups in Web pages; Ontology storage and querying tools, for using and querying ontologies easily and Ontology learning tools, for (semi)automatically deriving ontologies from natural language texts. In Pérez et al [Pérez et al, 2002] deliverable there is an extensive review and evaluation of tools that can support in above mentioned tasks.

4.3.2.4 The upper layers

Current research is mainly focused on the first three layers. Below the upper layers are outlined. The logic layer consists of rules that enable inferences, e.g. to choose courses of action and answer questions. The Proof layer executes the rules and evaluates together with the Trust layer mechanism for applications whether to trust the given proof or not. Finally Digital Signature layer aiming at detecting alterations to documents. These are the layers that pare currently being standardized in W3C working groups.

4.3.3 SW Developments

This section aims to outline some of the most important developments in the field of the semantic web.

SHOE: The Simple HTML Ontology Extensions (SHOE) [Heflin et al, 1999] was an early approach for ontology management associated with web pages. SHOE was developed as an extension to HTML in order to enrich web pages with semantic information. It can be used for the representation of concepts, their taxonomies, n-ary relations, instances and deduction rules.

SESAME: Sesame [Broekstra et al, 2002] is a generic architecture for the storage and querying of RDF and RDFS ontologies. Sesame can be coupled with a variety of repositories for the storage of ontologies, including relational databases, RDF triple stores, or remote storage services on the web. RQL, a declarative query language, is used for querying RDF data at a semantic level.

KAON: The Karlsruhe Ontology and Semantic Web Tool Suite (KAON) is an infrastructure called by its developers as an Ontology Software Environment. The Ontology Software Environment facilitates re-use of existing ontology stores, editors, and inference engines. It provides the basic technical infrastructure to coordinate the information flow between such modules, to define dependencies, to broadcast events between different modules and to transform between ontology-based data formats [Volz et al, 2003]. Both internal and external components of KAON can be seen in Illustration 21.

The main components of KAON illustrated are OI-Modeler (ontology editor), KAON API (programming interface for developers), KAON Engineering Server (server for distributed ontology engineering, to be used in combination with OI-Modeler as front-end) and TextToOnto (workbench for ontology learning from texts, feeds learned ontologies into the OI-Modeler)

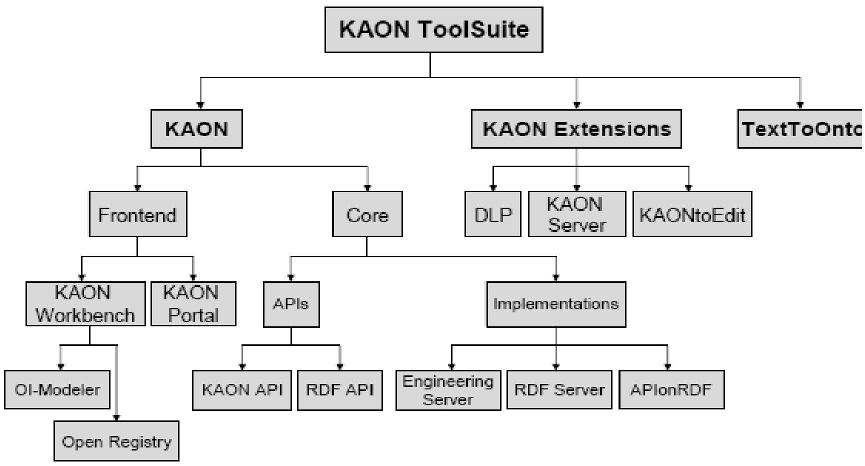


Illustration 21: KAON Tool Suite (http://kaon.semanticweb.org/)

Ontobroker: Ontobroker [Fensel et al, 1998] is a complete system with a broker architecture to manage ontologies, developed under the umbrella of european project. It consists of a query interface for formulating queries and an inference engine to derive new facts. The inference engine is a software program which interacts with users and which processes the results from the rules and data in the knowledge base. The inputs of the inference engine consists of an ontology, collected facts and queries formulated in a logic language (e.g, F-Logic). Ontobroker reads the rules and stores them in an internal database. It uses its inference engine to evaluate concepts, entities and relationships in its database and to compute an answer to the query submitted. It provides a graphic interface for indexing and querying a service. The strength of this approach is the tight coupling of informal, semiformal and formal information and knowledge.

SEAL: Another important application field of semantic web is portals, known as semantic portals. As already mentioned in previous chapters, portals often need to integrate many different information sources and they also require an adequate web site management system. SEAL ([Maedche et al, 2002];[Maedche et al, 2001]) aims to use ontologies as key elements for managing community Web sites and Web portals. In their paper Maedche et al [Maedche et al, 2002] distinguish between ontology and knowledge base and they further introduce a methodology for ontology engineering which can be characterized as an application-driven approach (Ontology kickoff, refinement, evaluation and maintenance phases). For supporting the ontology engineering process they also developed an ontology environment named OntoEdit.

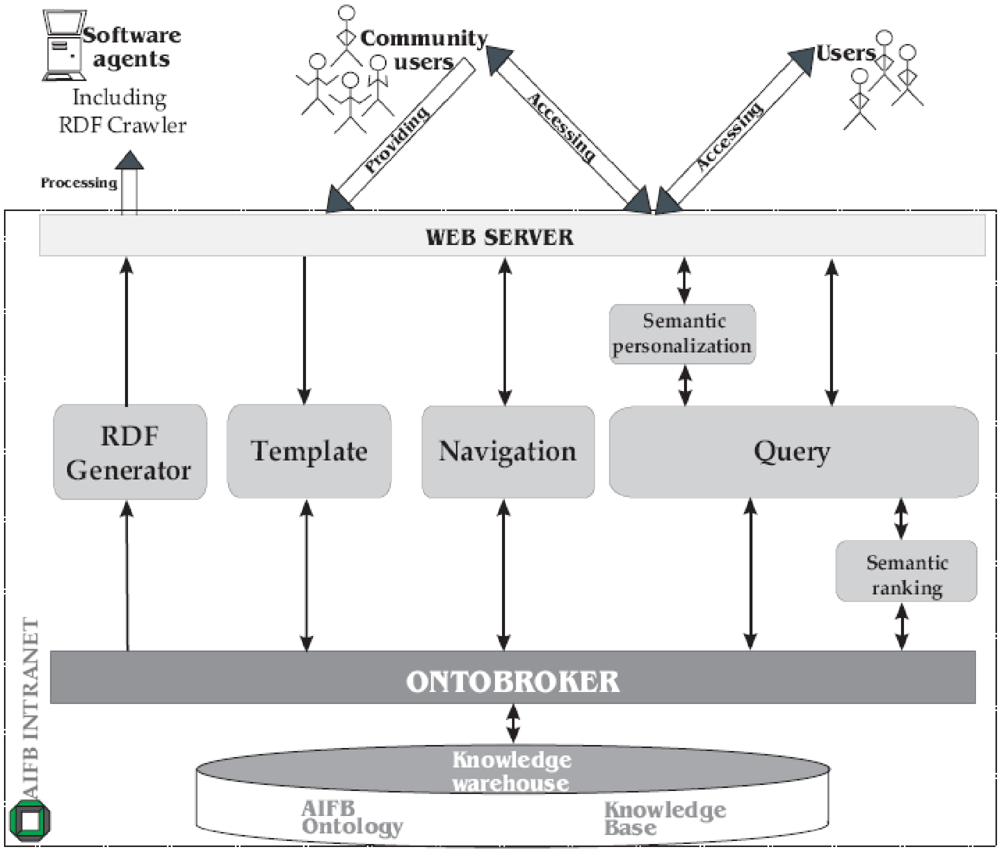


Illustration 22: SEAL Architecture [Maedche et al, 2002]

Now, regarding the actual SEAL system, its architecture is being shown in Illustration 22. The core modules of the systems are: Ontobroker, used as an inference engine which reads input files containing the knowledge base and the ontology, evaluates incoming queries, and returns the results derived from the combination of ontology, knowledge base and query. The Knowledge warehouse serves as a repository for data represented in the form of F-Logic statements. The ontology instances are organized around a relational database, where facts and concepts are stores in a reified format. The Navigation module is responsible of presenting the instances in the knowledge base by automatically generating links to all related instances. The Query module is an easy-to-use interface on the query capabilities of the F-logic query interface of Ontobroker. The Template module is responsible for generating an HTML form for each concept that a user may instantiate. Finally, there is also a module for handling the lexicon of the ontology.

ONTOLOGER: This is a system for usage-driven management of querying for information in ontology-based information portals. The focus of this work [Stojanovic et al, 2003] is on improving querying mechanism based on the usage data extracted from the analysis of so the called semantic log files. It is shown that a portal can benefit for such a system by mainly adjusting its vocabularies to users' needs.

Swoogle: Searching has been one of the most challenging applications for the semantic web. Recently, Ding et al [Ding et al, 2004] have designed a system, named Swoogle, that automatically discovers Semantic Web Documents (SWDs), indexes their metadata and answers queries about it. According to Ding et al, such an approach distinguishes it from other semantic web repositories and query systems in the literature; "Ontology based annotation systems, such as SHOE, Ontobroker, WebKB, QuizRDF and CREAM, focus on annotating online documents. However, their document indexes are based on the annotations rather than on the entire document, and they use their own ontologies which may not be suited for Semantic Web documents. It is notable that CREAM had indexed `proper reference' and `relational metadata'. Ontology repositories, such as the DAML Ontology Library, SemWebCentral and Schema Web, do not automatically discover semantic web documents but rather require people to submit URLs. They only collect ontologies which constitute a small portion of the Semantic Web. In addition, they simply store the entire RDF documents." [Ding et al, 2004]. Further, they present an architecture based on which a working prototype is available online[44].

4.4 Context-aware WIS – AmI

The aim of Ambient Intelligence (AmI) computing infrastructures is to provide intelligent services to the user targeting software towards generic context before delivery, and adapting it to a changing context after delivery [Preuveneers et al, 2004]. The "ambient intelligence" term comparing with the preceding "ubiquitous computing" one, currently mainly implemented by mobile technology, emphasizes that "*it does not solely rely on ubiquitous computing (i.e. useful, pleasant and unobtrusive presence of computing devices everywhere) but also on ubiquitous networking (i.e., access to network and computing facilities everywhere) and on intelligent aware interfaces (i.e., perception of the system as intelligent by people who naturally interact with the*

44 Swoogle: http://swoogle.umbc.edu/

system that automatically adapts to their preference)." [Issarny et al, 2005]. The term mobility is quite broad and may be subdivided into three categories [Roman et al, 2000]: personal mobility that deals with the mobility of people who may not necessarily carry a device, computer mobility that deals with the mobility of devices, and computational mobility that deals with the migration of code over physical nodes.

In such environments, the need for adaptivity is even more apparent comparing with classical web information systems mainly for two reasons: the change of user environment is more frequent and even more significant and the increased diversity of access devices' characteristics.

In such dynamic heterogeneous environments context-aware adaptation is a key concept to meet the varying requirements of different interacting systems. Held et al [Held et al, 2002] endorse that in order to enable context-aware adaptation, context information must be gathered and eventually presented to the application performing the adaptation. But what "context" actually is? One of the definitions for the notion of "context" defines it as "*any information that can be used to characterize the situation of entities (i.e whether a person, place or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves*" [Dey & Abowd, 2000]. Context can be categorized in human user context and surroundings context and may also be categorized according to persistence (permanent and temporary) and evolution (static and dynamic) [Golemati et al, 2007].

4.4.1 Context Modeling Approaches

A context model is needed in order to define and store context data in a machine processable form. To develop flexible and usable context ontologies that cover the wide range of possible contexts is a challenging task. Strang and Linnhoff Popien [Strang & Linnhoff–Popien, 2004] summarized the most relevant context modeling approaches which are based on the data structures used for representing and exchanging contextual information in the respective system including Key-Value Models, Markup Scheme Models, Graphical Models, Object Oriented Models, Logic Based Models and Ontology based Models: Strang and Linnhoff-Popien [Strang & Linnhoff–Popien, 2004] adopted such classification and further evaluated them based on six requirements including: Distributed composition, Partial validation, Richness & quality of information, Incompleteness and ambiguity, Level of formality and Applicability to existing environments. Bases on the results of this evaluation it seems safe to conclude that ontologies are the most expressive models and fulfill most of such requirements.

4.4.2 Ontology Context Models

Gu et al [Gu et al, 2004] propose a formal context model based on ontology using OWL to address issues including semantic context representation, context reasoning and knowledge sharing, context classification, context dependency and quality of context. It supports semantic context representation by defining the common upper ontology for context information in general and providing a set of low-level ontologies which apply to different sub-domains (see Illustration 23). It models the basic concepts of person, location, computational entity and activity and further describes the properties and relationships between these concepts.

In addition, they classify a wide range of contexts into two main categories - direct context and indirect context based on the means by which context is obtained. Direct context can be further classified into sensed context and defined context. Furthermore, as context information is inconsistent due to highly dynamic nature of pervasive computing systems and imperfect sensing technologies, Gu et al introduce quality constraints attributes aiming at more successful context reasoning.

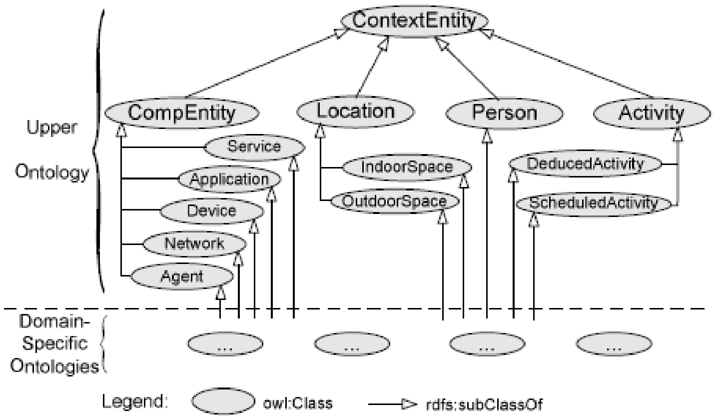


Illustration 23: Class hierarchy of context ontologies [Gu et al, 2004]

A narrower application domain's context aware ontology is COBRA-ONT [Chen et al, 2004]. COBRA-ONT expressed in the Web Ontology Language OWL, is a collection of ontologies for describing places, agents and events and their associated properties in an intelligent meeting-room domain. The ontologies are utilized by CoBra system, a broker-centric agent architecture for supporting context-aware systems in smart spaces.

Preuveneers et al [Preuveneers et al, 2004] conclude to six requirements for a basic context model: Application adaptivity (up-to-date information about entangled entities), Resource awareness (sufficient information about maximum and currently available resources), Mobile services (services migration on the move), Semantic service discovery (Semantic discovery based on context information enhances, Code generation (generate a dedicated implementation of a high-level service specification to broaden the range of devices on which services can be deployed), Context-aware user interfaces (user interfaces that are adapted to their context of use)

Based on such requirements and related research Preuveneers et al determine four main entities around which they build their ontology[45]:User, Environment, Platform, Service in relationship presented in by Illustration 24.

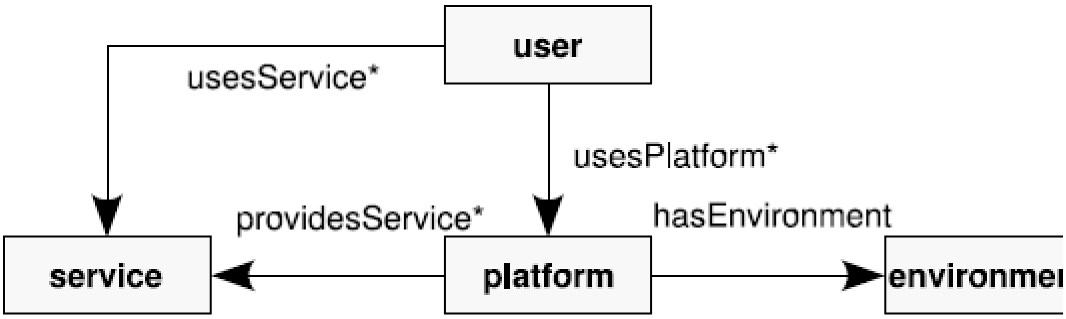


Illustration 24: Main entities of [Preuveneers et al, 2004]'s Context Ontology

To the aforementioned frameworks on context-aware models, also non context-aware specific work could contribute towards a more generic and complete ontology that could be used on the design and implementation of adaptive systems.

45 *Full ontology: www.cs.kuleuven.ac.be/cwis/research/distrinet/projects/CoDAMoS/ontology/*

Regarding user modeling, there have been identified two important research outcomes: Heckmann et al [Heckmann et al, 2005] aiming mainly at simplifying exchanging of user model between different user-adaptive systems introduce the GUMO (General User Model Ontology) based on OWL. Finally, Golemati et al [Golemati et al, 2007] present a user profile ontology based on OWL and focusing to the static profile of the user, his/her more or less permanent characteristics and not the dynamic ones, like his/her current position.

4.5 Recent User Interfaces

The web, as a non a passive source of information provides interaction with the user. Currently, the web has been turned out to host highly interactive applications. The wide range of access devices and users' preferences and requirements has led to multimodal interaction. Thus, today, an information can be either communicated as text, audio or video and the same information can be accessed via a web application, a hand held device or even interactive television (just to name some). The two main ways that the web nowadays interacts with a user are multimedia and user interfaces, discussed in following chapters.

4.5.1 Multimedia for Adaptation and Accessibility

A big proportion of information spread in the web is multimedia. This section aims to briefly present some of the latest developments on the field.

4.5.1.1 Scalable Vector Graphics (SVG)

SVG[46] is a platform for two-dimensional graphics. It has two parts: an XML-based file format that may be rendered in a resolution independent manner, and a programming API for graphical applications. Key features include shapes, text and embedded raster graphics, with many different painting styles. It supports scripting through languages such as ECMAScript[47] and has comprehensive support for animation.

46 http://www.w3.org/Graphics/SVG/

47 http://www.ecma-international.org/publications/standards/Ecma-262.htm

SVG is used in many business areas including Web graphics, animation, user interfaces, graphics interchange, print and hardcopy output, mobile applications and high-quality design. SVG is a royalty-free vendor-neutral open standard developed under the W3C Process. It has strong industry support. SVG builds upon many other successful standards such as XML (SVG graphics are text-based and thus easy to create), JPEG and PNG for image formats, DOM for scripting and interactivity, SMIL for animation and CSS for styling.

The specification defines the visual representation of the elements, which can be used in a stand-alone SVG file or included in another XML document within the SVG namespace. SVG 1.1 is a W3C Recommendation since 2003 and forms the core of the current SVG developments. SVG 1.2 is the specification currently being developed as is available in draft form. The SVG Mobile Profiles: SVG Basic and SVG Tiny are targeted to resource-limited devices and are part of the 3GPP[48] platform for third generation mobile phones. SVG Print is a set of guidelines to produce final-form documents in XML suitable for archiving and printing.

SVG allows for three types of graphic objects: vector graphic shapes (e.g., paths consisting of straight lines and curves), images and text. Graphical objects can be grouped, styled, transformed and composited into previously rendered objects. Text can be in any XML namespace suitable to the application, which enhances search ability and accessibility of the SVG graphics. The feature set includes nested transformations, clipping paths, alpha masks, filter effects, template objects and extensibility.

Scalable Vector Graphics (SVG) offers a number of features to make graphics on the Web more accessible than is currently possible, to a wider group of users (Accessibility Features of SVG[49]). SVG provides many accessibility benefits, some originating from the vector graphics model, some inherited because SVG is built on top of XML, and some in the design of SVG itself, for example, SVG-specific elements for alternative equivalents.

48 http://www.3gpp.org/

49 http://www.w3.org/TR/SVG-access/

4.5.1.2 Synchronized Multimedia Integration Language (SMIL)

SMIL[50] is also a W3C recommendation that enables simple authoring of interactive audiovisual presentations. It is a markup language (like HTML) and is designed to be easy to learn and deploy on Web sites. SMIL was created specifically to solve the problems of coordinating the display of a variety of media (multimedia) on Web sites. By using a single time line for all of the media on a page their display can be properly time coordinated and synchronized.

Similarly to SVG, SMIL offers opportunities for adapting content to user and system settings. Even, since version 1.0, SMIL includes "test attributes" that the author may use to suggest how a presentation should vary according to user preferences for subtitles, overdubs, captions, content language, connection speed, screen depth, and screen size. For each test attribute, players should allow users to set appropriate preferences.

For instance in the domain of accessibility, there are two continuous equivalents that promote accessibility.

- **Captions:** A caption is a text transcript of spoken words and non-spoken sound effects that provides the same information as a presentation's audio stream and is synchronized with the video track of the presentation. Captions benefit people who are deaf, hard of hearing, or who have auditory learning disabilities. They also benefit anyone in a setting where audio tracks would cause disturbance, where ambient noise in the audio track or listening environment prevents them from hearing the audio track, or when they have difficulties understanding spoken language.

- **Auditory descriptions:** An auditory description is a recorded or synthesized voice that describes key visual elements of the presentation including information about actions, body language, graphics, and scene changes. Like captions, auditory descriptions must be synchronized with the video stream they describe. Additionally, they must be synchronized with other audio streams. Auditory descriptions are generally timed to play during natural pauses in dialog. However, there may be cases where these natural pauses are not long

50 *http://www.w3.org/AudioVideo/*

enough to accommodate a sufficient auditory description. In such cases, it will be necessary to pause the video in order to provide enough time for an extended auditory description. At the end of the description, the video should resume play automatically. Auditory descriptions benefit people with blindness, low vision, or some kinds of visual perceptive learning disabilities. They also benefit anyone in an eyes-busy setting or whose devices cannot show the original video or graphical media object.

The test attributes that may be used with synchronization elements are:

- **system-captions**: Tests user preferences for captions
- **system-overdub-or-caption:** Tests user preferences for overdubs or subtitles
- **system-language:** Tests natural language preferences. Note. This test attribute does not specify the natural of language of an element's contents or an attribute's value; this is the role of the "xml:lang" attribute, defined in XML 1.0 ([XML10], section 2.12). However, "xml:lang" is not included in the SMIL 1.0 DTD. Therefore documents that include it will not validate to this DTD.
- **System-bitrate:** Tests preferences for the minimum approximate bandwidth (in bits per second) required to display the element. This attribute can be used to suggest changes in a presentation based on available bandwidth.
- **System-screen-depth:** Tests preferences for the minimum depth of the screen color palette (in bits) required to display the element. This attribute can be used to suggest changes in a presentation based on the ability of the screen to display images or video at a certain color depth.
- **System-screen-size:** Tests preferences for the minimum required screen size (in X and Y pixels) to display the element. This attribute can be used to suggest changes in a presentation based on screen size.

```
<switch>
   <video title="My Favorite Movie" longdesc="MyFavMovie"
          src="high-quality-movie.rm" system-bitrate="40000"/>
   <video title="My Favorite Movie" longdesc="MyFavMovie"
          src="medium-quality-movie.rm" system-bitrate="24000"/>
   <video title="My Favorite Movie" longdesc="MyFavMovie"
```

For example, Code 1 suggests that different video tracks need to be delivered according to the user's connection speed preferences. The player evaluates each of the choices in the switch element in order and chooses the first one whose system-bitrate value is equal to or greater than the user's preferred connection speed. No video will be played for connection speeds less than 10,000 bits per second.

4.5.1.3 MPEG-21 Multimedia Framework

Thanks to multimedia, communication is much widespread and therefore more powerful. However, there rise a serious problem of heterogeneity in users' terminals, networks, and in the people who ultimately consume and interact with the information presented to them [Vetro, 2004]. More and more different types of resources appear and can constitute complex entities. Such entities can involve the delivery of audio sound (music and spoken word), accompanying artwork (graphics), text (lyrics), video (visual) e.t.c.. The content providers' concerns include management of content, re-purposing of content based on consumer/device capabilities, protection of rights, protection from unauthorized access/modification, protection of privacy of providers and consumers, etc.

This motivates the MPEG-21 (ISO/IEC 21000) Multimedia Framework initiative that aims to enable transparent and augmented use of multimedia resources across a wide range of networks and devices, specifically taking into account Intellectual Property Management and Protection and the heterogeneity of the access and delivery infrastructure.

The MPEG-21 vision can be summarized as follows "*to define a multimedia framework to enable transparent and augmented use of multimedia resources across a wide range of networks and devices used by different communities*". - (ISO/IEC 21000-1). ISO/IEC 21000 consists of the following parts, under the general title Information technology — Multimedia framework (MPEG-21):

- Part 1: Vision, Technologies and Strategy [TR]
- Part 2: Digital Item Declaration
- Part 3: Digital Item Identification
- Part 5: Rights Expression Language
- Part 6: Rights Data Dictionary
- Part 7: Digital Item Adaptation
- Part 8: Reference Software
- Part 9: File Format
- Part 10: Digital Item Processing (under preparation)
- Part 11: Evaluation Tools for Persistent Association Technologies

For the purposes of this thesis, it is interesting to investigate how MPEG-21 could contribute on having adaptive content. The most relative part is part 7 that aims at fulfilling such requirements. This aims at defining the syntactics and the semantics that would allow for digital content adaptation. Thus, between others, there are explicitly defined characteristics such as:

1. User Characteristics

   (a) User information relating user preferences and usage history (UsagePreferences, UsageHistory)

   (b) User information relating to specific characteristics of digital content (AudioPresentationPreferences, DisplayPresentationPreferences, ColorPreference, StereoscopicVideoConversion, GraphicsPresentationPreferences, ConversionPreference, PresentationPriorityPreference)

(c) Specific issues regarding accessibility (FocusOfAttention, AuditoryImpairment, VisualImpairment, ColorVisionDeficiency)

(d) Services based on location (MobilityCharacteristics, Destination)

2. Terminal Capabilities (CodecCapabilities, DisplayCapabilities, AudioOutputCapabilities, UserInteractionInputs, DeviceClass)

3. Network Characteristics (NetworkCapability, NetworkCondition)

4. Characteristics of natural environments (Location, Time, AudioEnvironment, IlluminationCharacteristics)

4.5.2 Abstract User interfaces

The design of user interface is very important as this is actually the subsystem that interacts with the most changing environment; human and her context. This becomes even more important when considering that human now comes verge on more and more interfaces. This adds an overhead to user's cognitive process and has to be minimized (This issue is referred in literature as Attentive User Interfaces field [Vertegaal, 2003]).

For years Human Computer Interaction (HCI) is strangling for developing an abstract representation of user interface [Trewin et al, 2004] (currently based on XML syntax see for instance [Souchon & Vanderdonckt, 2003]). This would offer possibility of UI adaptation according to system's environment (or, including user and context of use). Such languages / frameworks include UIML, XIML, XFORMS, AIAP, XUL, JSF.

Further, the successful so far metaphor of desktop for human computer interaction is being disputed. Xerox's desktop and WIMP (Windows, Icons, Menus, Pointers) metaphor worked well with personal computers. The nowadays “memex” vehicle comes to be mobile, hand-held devices and the metaphors for user interfaces have to be adapted to the new requirements of new input forms such as handwriting recognition and new multimodal output forms such as voice synthesis. For such artefacts the desktop metaphor can prove not to be appropriate or enough for an effective interaction with the user. On the other hand personal computer does not seem to being disappeared, at least not straight out. This means that the requirement for user interface metaphor adaptation has been merged.

Such an adaptation could be realized through the abstraction of user interface. Having that, context sensitive pipelines would be introduced so that the interface can be adapted according to the system's environment (context of use). One of the promising technologies that even if it is not as abstract as it should be, moves research to such direction are XFORMS.

4.5.2.1 XFORMS

XForms[51] is an XML application that represents the next generation of forms for the Web. XForms is not a free-standing document type, but is intended to be integrated into other markup languages, such as XHTML or SVG. An XForms-based web form gathers and processes XML data using an architecture that separates presentation, purpose and content (model-view-controller). The underlying data of a form is organized into instances of data schema (though formal schema definitions are not required). An XForm allows processing of data to occur using three mechanisms:

- a declarative model composed of formulate for data calculations and constraints, data type and other property declarations, and data submission parameters
- a view layer composed of intent-based user interface controls
- an imperative controller for orchestrating data manipulations, interactions between the model and view layers, and data submissions.

Thus, XForms accommodates form component reuse, fosters strong data type validation, eliminates unnecessary round-trips to the server, offers device independence and reduces the need for scripting. The design goals of XForms meet the shortcomings of HTML forms point for point:

- Excellent XML integration (including XML Schema)
- Provide commonly-requested features in a declarative way, including calculation and validation

51 *http://www.w3.org/MarkUp/Forms/*

- Device independent, yet still useful on desktop browsers
- Strong separation of purpose from presentation
- Universal accessibility

As such, XFORMS can be seen as promising technology to enable multi-device / multi-context interaction. Even if a this moment there is not enough support for XFORMS its abstract form can be involved on the design and development of “server-side” components that would deliver adaptive forms. For instance, AJAXFORMS project transforms XFORMS to AJAX (a rich client technology based on Javascript) forms and thus making them operable on todays user agents.

4.6 Summary and Discussion

This chapter employed the investigation of the role of adaptivity in the field of Web Information Systems with strong emphasis to the web accessibility requirements; seen as an extreme case of adaptive WIS.

Thus, an introduction of web accessibility takes place by presenting a brief survey starting from the origins. The survey comes up with a proposed classification of web accessibility approaches with accompanied methods, techniques, architectures and technologies. All these can be applied as to the field of adaptive WIS and even more provide concrete approaches for classes of problem situations met in the past in the field of web accessibility.

Then, emphasis is put on the WIS paradigm named portal. Portal paradigm and infrastructure is considered as a special – interesting case for the subject of this thesis thanks to its primary requirement; personalization. An investigation of its role to the e-commerce and m-commerce shows off its importance and at the same time the requirement to make that as accessible as possible. In Chapter 7, such an approached will be presented from the point of view of this thesis.

Follows an investigation of context-aware WIS that aims at including into the adaptation process parameters that come mainly from the nature of mobile and hand held devices for making the former more effective. Then, interesting developments of context modeling approaches, mainly based on semantic web concepts and technologies, have been presented.

This is one of the reasons that an investigation of semantic web takes place. However, semantic web can also offer much more, specifically on the problems of inferencing and semantic content adaptation. The upcoming semantic web is being investigated including its design considerations, principles, technologies and applications. Having in mind the “disappearing computer” this thesis also investigates approaches and systems that have been used in ambient intelligence scenarios, most of them through the use of semantic web technologies.

Finally, this chapter has studied the use of multimedia technologies and abstract user interfaces for highly interactive systems. Multimedia technologies seem to be the upcoming hot subject for adaptation basically through the spreading of the use of interactive television. Technological advances like MPEG-21 have been studied providing useful information for the proposed framework. Last, but not least, the concept and developments of abstract user interfaces is seen as a stopover to the adaptive WIS as this would allow an abstracted designed interface to be capable of adapting its presentation according to the interaction context.

Chapter 5. WIS design and development

5.1 HCI design methods for WIS

Web information systems can borrow design methodologies from both fields of Human Computer Interaction and Software engineering aiming at including their concepts into the proposed framework.

Methodologies used for designing systems and especially adaptive ones are crucial to a framework. After a long period of classical methodologies that were predictive rather than adaptive, and process oriented rather than people-oriented, several methodologies such as SSM, RUP as well as new lighter weight methodologies emerged, such as: XP (Extreme Programming), Crystal Family, Adaptive Software Development, Feature Driven Development (FDD), Dynamic System Development Method (DSDM), USERfit (referenced in [Spyrou et al, 2003]).

This sections aims to give an overview of the most important design and development approaches that could be involved in the design and developments of adaptive WIS.

5.1.1 Soft Systems Methodology

A well established methodology designed and evaluated in the corpus of HCI is Soft Systems Methodology (SSM). SSM has been introduced by Peter Checkland [Checkland, 1976] primarily as a way of analyzing complex situations where there are divergent views about the definition of the problem — "soft problems". This distinguishes SSM from other methodologies which deal with “hard” problems that are often more technology-oriented.

However, SSM originated from the understanding that "hard" Systems Thinking, such as Operations Research techniques, is inadequate for enquiring into large, complex organizational issues. The methodology consists of seven stages (iterative and some times with different order) [Checkland, 1999] (see Illustration 25):

1. **The Problem Situation Unstructured:** Investigate not the problem but the situation in which there is perceived to be a problem; the unstructured problem.

2. **The problem situation expressed:** This stage is an expression phase during which an attempt is made to build up the richest possible picture, not of the problem but of the situation in which there is perceived to be a problem.

3. **Root Definition of relevant Systems:** This phase belongs into the systems thinking domain. This involves naming the systems that look as though they might be relevant to the putative problem and preparing concise definitions of what these systems are – as opposed to what they do. From what different perspectives can we look at this problem situation? Root definitions are written as sentences that elaborate a transformation. There are six elements that make a well formulated root definition. They are summed up in the acronym CATWOE:

    - Customer. Everyone who may gain benefits from a system is considered as a customer of the system. If the system involves sacrifices such as layoffs, then those victims must also be counted as customers.

    - Actor. The actors transform inputs into outputs and they perform the activities defined in the system.

    - Transformation process. This is shown as the conversion of inputs to outputs.

    - Weltanschauung. The German expression for world view. This world view makes the transformation process meaningful in context.

    - Owner. Every system has some proprietor, who has the power to start up and shut down the system (power of veto).

- Environmental constraints. These are external elements that must be considered. These constraints include organizational policies as well as legal and ethical matters.

4. **Conceptual models:** Stage 4 consists of making conceptual models of the human activity systems named and defined in the root definitions. A structured set of verbs is assembled which describes the minimum necessary activities required in a human activity system which is that described by the root definition.

   (a) **Formal system concept**: is the use of a general model of any human activity system which can be used to check that the models built are not fundamentally deficient.

   (b) **Other system thinking**: consists of modifying or transforming the model, if desired, into any other form which may be considered suitable in a particular problem.

5. **Comparison of 4 with 2**: At this stage the problem is considered back to the real world a set against the perceptions of what exists there.

6. **Feasible, desirable changes**: There will be defined possible changes which simultaneously meet two criteria: that they arguably desirable and at the same time feasible given prevailing attributes and power structures and having regard to the history of the situation under examination.

7. **Action to improve the problem situation:** Involves taking action based on stage 6 to improve the problem situation.

Even if SSM primarily fits to organizational - unstructured / ill-defined problems, it has been successfully applied in the field of information systems design ([Mathiassen, 1991];[Xu, 2000]). Moreover this seems to fit even more to the situation of adaptive information systems where the situation is more complex and the “problem” continuously changes.

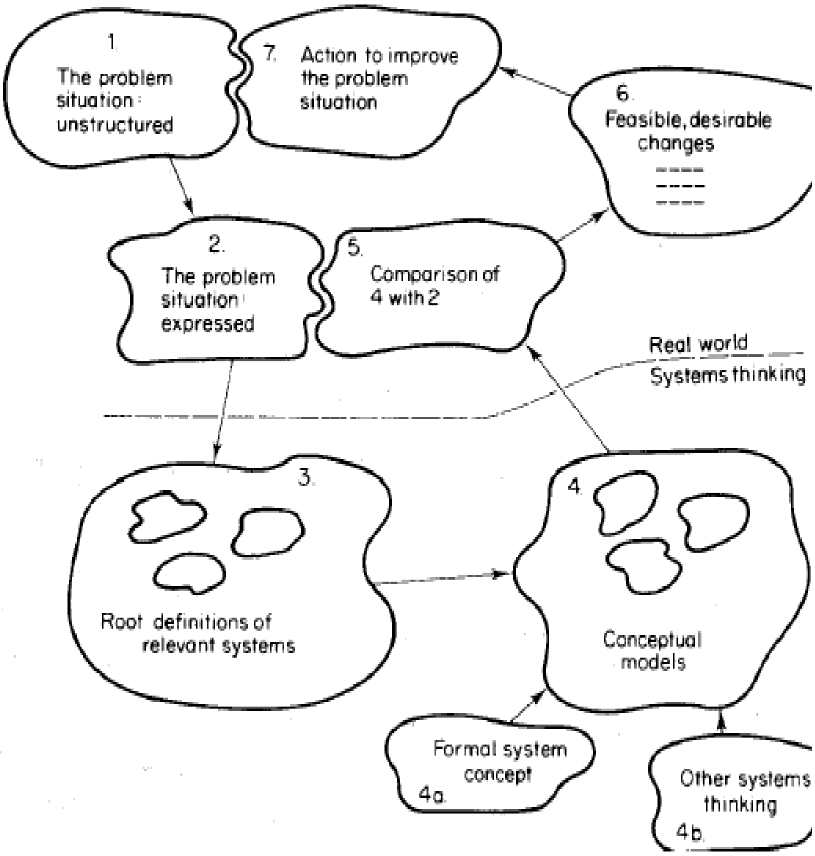


Illustration 25: SSM Outline [Checkland, 1999]

5.1.2 User-Centered design

User-centered design (UCD) ([Norman, 1988];[Nahl, 1996]) or Human-centered design (HCD) is a design philosophy that tries to optimize a system around how people can, want, or need to work, rather than forcing the users to change how they work to accommodate the system or function. UCD is a philosophy that places the person (as opposed to the 'thing') at the center and a process that focuses on cognitive factors (such as perception, memory, learning, problem-solving, etc.) as they come into play during peoples' interactions with things.

In his book, Norman [Norman, 1988] uses the term "user-centered design" to describe design based on the needs of the user, leaving aside what he considers to be secondary issues like aesthetics. User-centered design involves simplifying the structure of tasks, making things visible, getting the mapping right, exploiting the powers of constraint, and designing for error.

According to Maguire [Maguire, 2001] UCD approach is a complement to software development methods rather than a replacement for them. The key principles of UCD are as follows:

- The active involvement of users and clear understanding of user and task requirements.
- An appropriate allocation of function between user and system.
- Iteration of design solutions. Iterative software design entails receiving feedback from end-users following their use of early design solutions.

In literature can be found several converging approaches that keep based on same UCD philosophy, they "materialize" certain aspects. Such approaches includes participatory design (PD), focusing on the participation of users, contextual design, "customer centered design" focusing in the actual context, Universal Design - Design for all focusing on the inclusion of all users (disabled, elderly people e.t.c.) to the design.

5.2 Software engineering for WIS

5.2.1 *Software design methods*

In the corpus of software engineering the design and development process has been supported by several methodologies and accompanied tools. This sections aims to very briefly outline the most acknowledged ones.

5.2.1.1 Waterfall model

The waterfall model is an one-way sequential software development model consisting of requirements analysis, design, implementation, testing (validation), integration, and maintenance phases and has been initial introduced by Winston Royce [Royce, 1970].

During the process, the deliverables that are produced in each phase, are validated at the end of the phase, and then used as input for the next phase. The main characteristic of this process is that these deliverables are considered complete, almost frozen, and revisited only to fix a major issue, or in other words there is a minimal feedback from one phase to the other.

And this is actually the main drawback of the model as the requirements are changing while designing and developing and even more issues are emerged moving from one phase to the other. Royce himself advocated an iterative approach to software development and did not even use the term "waterfall." Starting from Royce himself, there appeared many iterative “versions” of the waterfall model. The most acknowledged is the spiral model.

5.2.1.2 Spiral model

The spiral model was introduced by Boehm in 1986 [Boehm, 1986]. In contradiction to waterfall model this is a purely iterative model. It might not be the first iterative one but it is the first one to explain why the iteration matters. This model of development combines the features of the prototyping model and the waterfall model. The spiral model is favored for large, expensive, and complicated projects. The steps in the spiral model are presented in Illustration 26.

An iterative lifecycle exploits the "soft" nature of software, and proceeds by developing in iterations that encompass the activities of requirements analysis, design, implementation, integration, and test. One of the best descriptions is in Professor Barry Boehm's paper on the "spiral" model, summarized as "Analyze a little, design a little, test a little, and loop back."

The spiral model is used most often in large projects. For smaller projects, the concept of agile software development is becoming a viable alternative.

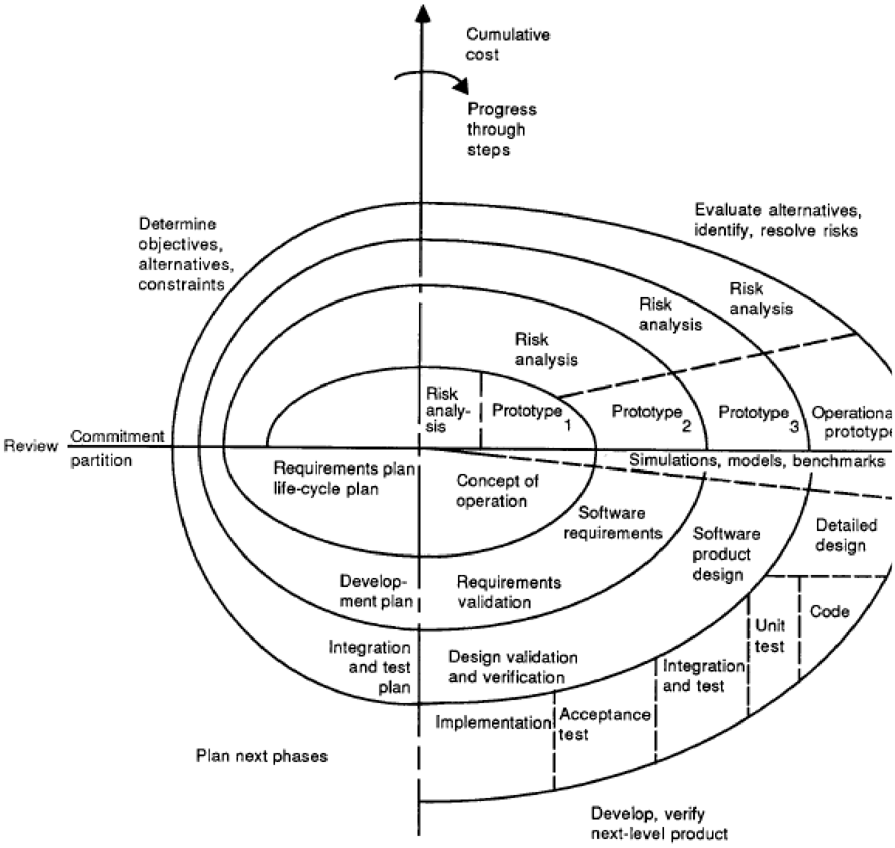


Illustration 26: Spiral Model [Boehm, 1986]

5.2.1.3 Agile Software Development

This is more a conceptual framework consisting of several methodologies. Most agile methods attempt to minimize risk by developing software in small iterations. Each iteration, which last one to four weeks, includes all the tasks necessary to release the increment of the new functionality: planning, requirements analysis, design, coding, testing, and documentation. Agile methods emphasize realtime communication, preferably face-to-face, over written documents.

While it may not have been the first agile method, Extreme Programming (usually abbreviated as "XP") established the popularity of agile methods. Extreme Programming was created by Kent Beck in 1996 [Beck, 2000]

Some of the principles behind the Agile Manifesto[52] are:

- Customer satisfaction by rapid, continuous delivery of useful software
- Working software is delivered frequently (weeks rather than months)
- Working software is the principal measure of progress
- Even late changes in requirements are welcomed
- Close, daily, cooperation between business people and developers
- Face-to-face conversation is the best form of communication
- Projects are built around motivated individuals, who should be trusted
- Continuous attention to technical excellence and good design
- Simplicity
- Self-organizing teams
- Regular adaptation to changing circumstances

These methods are characterized as adaptive comparing with predictive one.

5.2.2 Software Design paradigms

5.2.2.1 Objects (OOP)

Object-oriented programming (OOP) actually consists of a set of cooperating objects. This contradicts to a traditional procedural programming in which a program is a sequential list of instructions to the computer. In OOP, each object is capable of receiving messages, processing data, and sending messages to other objects. In other word each object is an entity having a distinct role or responsibility. Object oriented programming have been introduced in 1960s as a response to the crisis that had appeared due to the hardware an software increasing complexity. The modularity of OOP was seen as an solution to maintain the software quality.

52 *http://www.agilemanifesto.org/*

A survey by Deborah J. Armstrong [Armstrong, 2006], of nearly 40 years of computing literature, identified a number of "quarks," or fundamental concepts, identified in the strong majority of definitions of OOP. These include:

- **Abstraction**: Creating classes to simplify aspects of reality using distinctions inherent to the problem.
- **Class**: A description of the organization and actions shared by one or more similar objects.
- **Encapsulation**: Designing classes and objects to restrict access to the data and behavior by defining a limited set of messages that an object can receive.
- **Inheritance**: The data and behavior of one class is included in or used as the basis for another class.
- **Object**: An individual, identifiable item, either real or abstract, which contains data about itself and the descriptions of its manipulations of the data.
- **Message Passing**: An object sends data to another object or asks another object to invoke a method.
- **Method**: A way to access, set, or manipulate an object's information.
- **Polymorphism**: Different classes may respond to the same message and each implement it appropriately.

Maybe the most acknowledged object oriented methodology is the Rational Unified Process (RUP) [Kruchten, 2000]. RUP follows iterations of four phases (see Illustration 27):

1. **Inception phase**: Scope the project, define the business case.
2. **Elaboration phase**: Refine the requirements, establish an architecture, mitigate the most technical risk.
3. **Construction phase**: Complete the system up to a point where it can be deployed in limited context ("beta version").
4. **Transition phase**: Finish the product and reach product final release.

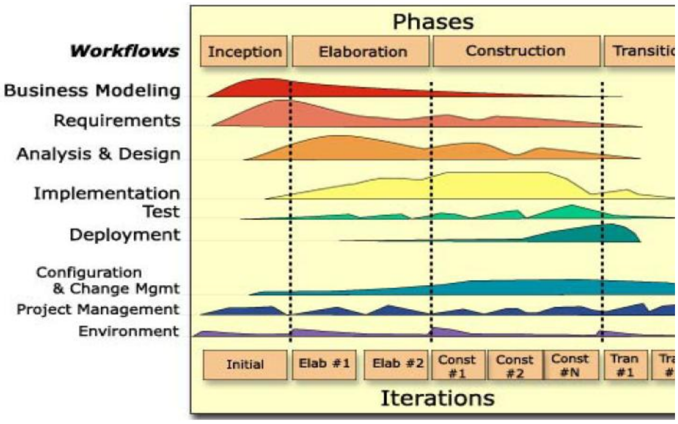


Illustration 27: The Rational Unified Process (RUP) [Kruchten, 2000]

The first (horizontal) dimension represents the dynamic aspect of the process expressed in terms of cycles, phases, iterations, and milestones. In the RUP, a software product is designed and built in a succession of incremental iterations. This allows testing and validation of design ideas, as well as risk mitigation, to occur earlier in the lifecycle. The second (vertical) dimension represents the static aspect of the process described in terms of process components: activities, disciplines, artifacts, and roles.

5.2.2.2 Components (COP)

The definition of COP was first formulated at the 1996 European Conference on Object-Oriented Programming (ECOOP) as one outcome of the Workshop on Component-Oriented Programming. According to that "*A software component is a unit of composition with contractually specified interfaces and explicit context dependencies only. A software component can be deployed independently and is subject to composition by third parties*" - (WCOP'96 Summary in ECOOP'96 Workshop Reader, dpunkt Verlag, 1997, ISBN 3-920993-67-5).

Component-oriented programming aims to replace traditional monolithic software systems with reusable software components and layered component frameworks [Szyperski, 1998]. COP is seen as the natural extension of object-oriented programming and has resulted to several important approaches over the recent years (e.g.CORBA and JavaBeans). A component, comparing with an object, provides the resources to instantiate objects and appropriate interfaces to several closely related classes. In that sense, COP is a kind of architecture and packaging.

In COP, each facility has a work interface and contracts surrounding that interface allowing easy replacement of Component instances without affecting code in other parts of the systems. Thus, the major distinction between Object Oriented Programming (OOP) and COP is the level of integration. COP offers easier management due to fewer interdependencies among classes, promoting the level of code reuse. Another, important, benefit of COP is the ability to have multiple implementations of the Component that can be selected at runtime.

Currently there have been developed several software frameworks for supporting component oriented programming and implement relevant design patterns discussed in the following subsections. The most acknowledged (and open source) have been the Apache Avalon Project[53] and Spring Framework[54] that have been the basis for popular developments like Apache Cocoon (http://cocoon.apache.org) and Apache Jetspeed-2 (http://portals.apache.org/jetspeed-2).

Inversion of Control Pattern (IoC) – Dependency Injection

Inversion of Control (IoC) is the concept that a component is always externally managed. Everything a component needs including contexts, configurations, and loggers is given to the component. In fact, every stage in the life of a component is controlled by the component's creator. Using this pattern proves a secure method of component interaction in the system under development.

Martin Fowler [Fowler, 2003] introduced an alternative term for inversion of control (more precisely for a form of IoC); dependency injection. The pattern seeks to establish a level of abstraction via a public interface, and to remove dependency on components by (for example) supplying a plug-in architecture.

The advantage of inversion of control is that it decouples objects from specific lookup mechanisms and implementations of the objects it depends on. As a result, more flexibility is obtained for production applications as well as for testing. In particular, dependencies on a particular deployment environment can be removed from the code making it much easier to test

53 *Avalon (closed): http://avalon.apache.org/. Replaced by Excalibur (http://excalibur.apache.org/)*

54 *Spring Framework: http://www.springframework.org/*

functionality in a simple standalone environment. A consequence of this is that it becomes faster and easier to test so that in the end quality of the software is improved[55].

Separation of Concerns Pattern

One of the challenging issues for facing by software engineering is the incremental complexity that often is responsible for low quality software developments. From the very early days – 1972 (see refs in [Ossher & Tarr, 2000]) there were attempts to decompose systems in modules for eliminating the complexity. As Ossher & Tarr discuss, in the past there have been discussed several dimensions of concern including features (like printing, persistence, and display capabilities), aspects (like concurrency control and distribution), roles, viewpoints, variants, and configurations. Thus they [Ossher & Tarr, 2000] propose a multi-dimensional separation of concerns that allows simultaneous separation according to multiple, arbitrary kinds (dimensions) of concerns, with on-demand re-modularization. Concerns can overlap and interact.

Several programming paradigms allow developers to apply SoC. For example, object-oriented programming languages such as Java can separate concerns into classes and methods, and a design pattern like MVC (Model View Controller) can separate content from presentation and data-processing from content. Service-oriented design (following section) can separate concerns into services and operations. Procedural programming languages such as C and Pascal can separate concerns into procedures. Aspect-oriented programming languages can separate concerns into aspects. In the era of the web, Client/Server and N-tier models are all examples of SoC.

5.2.2.3 Open source

Talking for software design engineering the open source movement[56] role cannot be ignored. Generally speaking, Open source is a set of principles and practices that promote access to the production and design process for various goods, products, resources and technical conclusions or advice. Specifically, to the field of software engineering, open source programs are programs whose licenses give users the freedom to run the program for any purpose, to study and modify

55 For an introduction : http://en.wikipedia.org/w/index.php?title=Dependency_injection&oldid=132245921

56 http://www.opensource.org/

the program, and to redistribute copies of either the original or modified program (without having to pay royalties to previous developers) [Wheeler, 2003]. In addition Wheeler research proves that open source has significant market share in many markets, is often the most reliable software, and in many cases has the best performance. From the software design point of view open source paradigm has introduced new models and tools and even new software engineering culture.

Furthermore Viorres et al [Viorres et al, 2007] discusses the impact of open source on HCI factors. They conclude that if open source products are to exploit their full potential in terms of widespread acceptance, they need to systematically address HCI concerns into their design process. Current research focuses on finding more suitable ways for involving end-users in the development process and supporting constructive analysis and resolution by developers. The support of collaboration (communication, coordination, cooperation) among OSS participants needs to be further investigated, mainly in terms of community (tool) support. The influence of established communities which provide ready-made collaboration support via a plethora of tools, is deemed as very important in ensuring a smooth start-up process and critical mass generation. Lastly, accessibility is a significant aspect for open source, which currently from a technological and standards point of view, exhibits unrealized potential. The requirement for a unified approach is imperative, an approach that would both strengthen the open source's appeal and bring it even closer to its philosophical roots.

5.2.2.4 Services (SOA)

Service Oriented Approach / Architecture (SOA) is a way of thinking about building software components. This is independent of underlying technologies and is actually a means for building distributed systems. Fundamental to the service model is the separation between the interface and the implementation. An architecture for service-based applications has three main parts: a provider, a consumer and a registry. The consumer of a service need only (and should only) understand the producer's interface; the implementation can evolve over time, without disturbing the clients of the service. The same interface can be offered by many implementations; several key benefits of service orientation derive from this abstraction of the capability from how the capability is delivered. Thus the primary benefit of SOA is the inherited ability of reusing services in new contexts. This means that for SOA, standardization and interoperability are key issues. Then, complex services involve the selection of appropriate services for the given context and the orchestration of those services into a composite or complex service to the requirements of that context [Brodie et al, 2005].

According to OASIS SOA reference model [MacKenzie et al, 2006], any design for a system that adopts the SOA approach shall (see Illustration 28): Have entities that can be identified as services as defined by this Reference Model; Be able to identify how visibility is established between service providers and consumers; Be able to identify how interaction is mediated; Be able to identify how the effect of using services is understood; Have descriptions associated with services; Be able to identify the execution context required to support interaction and It will be possible to identify how policies are handled and how contracts may be modeled and enforced.

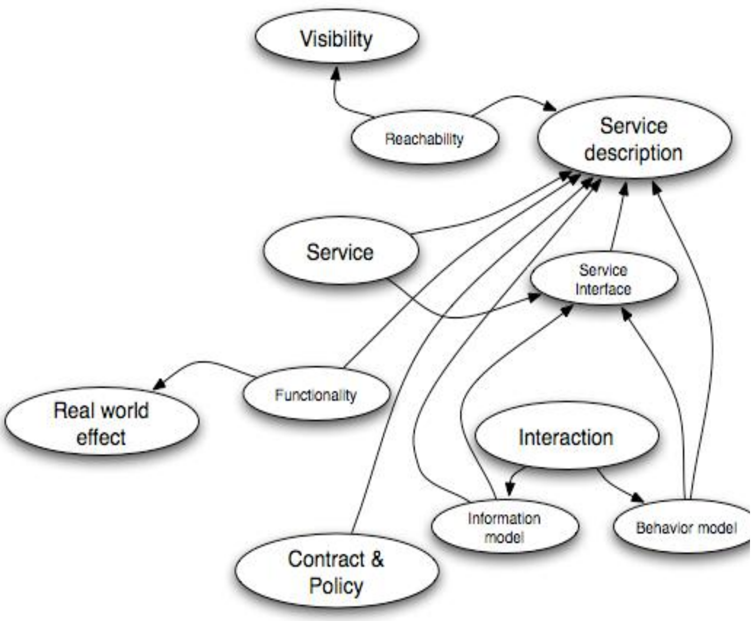


Illustration 28: Reference model of SOA [MacKenzie et al, 2006]

Several technologies have been appeared for applying SOA but the most widely used and standardized is web services.

Web Services paradigm

W3C [Booth et al, 2004] defines web services as:"*A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the*

Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards". - [Booth et al, 2004]

Web Services are platform and language independent, and are based on three key technologies: services are defined with the Web Services Description Language (WSDL), services are published and found using the Universal Description, Discovery, and Integration (UDDI) specification, and the transport protocol is based on the Simple Object Access Protocol (SOAP) over the standards HTTP or SMTP. Illustration 29 presents the typical web service architecture .

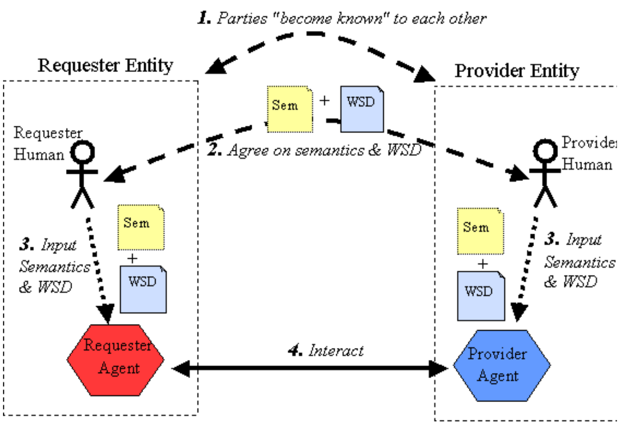


Illustration 29: The General Process of Engaging a Web Service (http://www.w3.org/TR/ws-arch/)

More specifically the Web Services basic key technologies are:

- **UDDI (Universal Description, Discovery, and Integration)**. UDDI is a protocol for describing available Web Services components. This standard allows businesses to register with an Internet directory that will help them advertise their services, so companies can find one another and conduct transactions over the Web. The online yellow pages directory that UDDI provides is a key part of how Web services plans such as Sun ONE and Microsoft .NET will work together. This registration and lookup task is done using XML and HTTP(S)/SMTP-based mechanisms. The UDDI project is working to provide a common access method for the metadata needed to determine if a piece of previously developed code will suffice and, if so, how to access it.

- **SOAP (Simple Object Access Protocol)**. SOAP is a protocol for initiating conversations with an UDDI service. SOAP makes object access simple by allowing applications to invoke object methods, or functions, residing on remote servers. A SOAP application creates a request block in XML, supplying the data needed by the remote method as well as the location of the remote object itself.

- **WSDL (Web Service Description Language)**, the proposed standard for how a Web service described is an XML-based service IDL (Interface Definition Language) that defines the service interface and its implementation characteristics. WSDL is referenced by UDDI entries and describes the SOAP messages that define a particular Web service.

Semantic web services

According to the “manifesto and paradigm shift in computer science” [Brodie et al, 2005], the two core challenges that need to be addressed by SOA are *search* and *integration*. SOA provides the potential of a global registry in which to search for services anywhere in the network (referred as service discovery). The results of invoked services are combined so that a new complex service / application is emerged. This requires that the services interoperate or integrate with respect to their respective data, protocol, and process syntax and semantics. Brodie et al calls these actions as service orchestration (or composition) and adaptation (or integration) and further mention that service-orientation does not address the challenges of automating discovery, orchestration/composition, or adaptation/integration.

Service Discovery matches the requirements for a service against the capabilities of all candidate services to find a single service, or composite service, which meet the requirements. The requirements can either be functional or non-functional including performance and economic factors. *Service selection* is even more complex since it must also consider orchestration, the composition of services to form a composite service. *Service integration or adaptation* maps service protocols, processes, and data so that meaningful interaction of discovered and orchestrated services is itself meaningful. Currently integration is done largely manually between large software components. Brodie et al suggest that integration must be resolved largely dynamically between millions of services and this is the critical role of semantic in SOA.

They also mention that the semantically enabled solutions will dramatically increase the level of automation of discovery, selection, orchestration, and adaptation so that they can operate dynamically to support dynamic business flexibility and adaptation.

Furthermore, the W3C Semantic Annotations Working Group[57] states that the Web Services Description Language (WSDL) specifies a way to describe the abstract functionalities of a service and concretely how and where to invoke it. The WSDL 2.0 specification does not include semantics in the description of Web services. Therefore, two services can have similar descriptions while meaning totally different things. Resolving this ambiguity in Web services descriptions is an important step toward automating the discovery and composition of Web services — a key productivity enabler in many domains including business application integration.

The lack of semantic capabilities in SOA is referred by Brodie et al as *semantic gap*. To make use of a Web service, a software agent needs a computer-interpretable description of the service, and the means by which it is accessed. An important goal for Semantic Web markup languages, then, is to establish a framework within which these descriptions are made and shared. For SESA – such a semantically enabled SOA - the following technologies are proposed [Brodie et al, 2005]. *Web Service Modeling Ontology (WSMO)*[58] provides a conceptual model for adding semantics to service-oriented solutions. It includes elements for user, service definitions of providers, and ontologies and mediators as declarative and procedural means to facilitate interoperability at the level of data, protocols and processes. *Web Service modeling language*[59] *(WSML)* is a family of languages providing formal semantics for WSMO models. *Web Service Execution Environment*[60] *(WSMX)* is a reference implementation of an SESA that is compliant with the semantic specifications of WSMO.

57 http://www.w3.org/2002/ws/sawsdl/

58 WSMO: http://www.wsmo.org

59 WSML: http://www.wsmo/wsml/

60 WSMX" http://www.wsmx.org

An alternative framework is proposed by Martin et al [Martin et al, 2005]. They propose OWL-S and they have also submitted it to W3C[61]. Their structuring of the ontology of services is motivated by the need to provide three essential types of knowledge about a service (shown in Illustration 30), each characterized by the question it answers. What does the service provide for prospective clients which is answered by the "profile", how is it used, which is answered by the "process model" and how does one interact with it, which is answered by the "grounding".

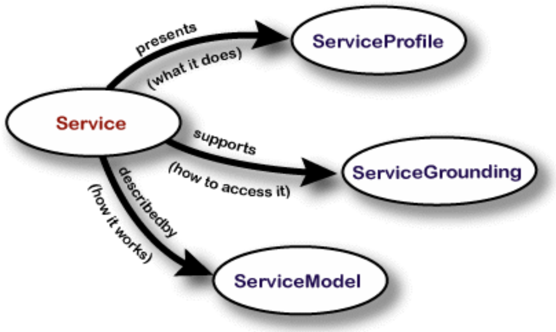


Illustration 30: OWL-S: Top level of the service ontology [Martin et al, 2005]

5.3 Summary and Discussion

This chapter has provided a methodological base for designing adaptive web information systems. This included appropriate HCI design methods (SSM and User-Centered design), software design methods (waterfall, spiral and agile) and paradigms (objects, components, services and open source).

Adaptive web information systems are highly interactive systems, fact that makes the use of HCI methodologies during design phase more than an essential requirement. The presented methods are the fundamental ones and more widely used that match the viewpoint of the proposed framework (systemic). Over and above, such methods could be used as a source of inspiration for the process of adaptation, as adaptation can be seen as an continuous, iterative

61 http://www.w3.org/Submission/OWL-S

design process with changing requirements. It could be said that an adaptive system is a system that can continuously design itself (self-designable).

In addition, software design methods and paradigms have been investigated to provide the adaptive WIS designer with appropriate provisions for the design and implementation of adaptive WIS. From such an investigation it comes out that the tendency is to move from complex, solution-specific monolithic developments to modularized / distributed and open architectures that enables reusability and separation of concerns. A major "fruit" of such open architectures is the service oriented paradigm. The use of service oriented architectures and web services seem to offer a good basis for a distributed architecture for adaptive web information systems that comes up from earlier conclusions.

Special interest turns up the emerging field of semantic web services. The investigation to this field can conclude that in SOA the architecture proves to be an emergent property and this is actually the property that could offer adaptivity in a WIS though the self searching, retrieval and orchestration of services. Finally, such services would be also characterized by capabilities for undertaking roles and evaluating their performance based on certain criteria.

Chapter 6. Designing Adaptive WIS – The DAWIS framework

6.1 Framework requirements

The aim of such a framework is to support a web information system's designer to consider the notion of adaptivity as a feature that would crucially improve her design. The investigation of the notion of adaptivity in the filed of general systems shown that such a notion has been applied to completely different fields, from different approaches but at the same time came out that in all cases there have been identified commonalities that consist the abstract characteristics and concepts of adaptivity. The aim of this thesis is to formulate such concepts and methods and further incorporate them to a reference framework for adaptivity focusing on the field of WIS. Analytically, such a framework is developed aiming at:

- providing axioms and definitions related to the design of adaptive web information systems. Specifically, it should clarify the key notion (adaptation) and further provide indicative models that could quantify the critical parameters which constitute adaptivity in order to evaluate the latter's effect based on the designer's requirements;
- supporting the designer in the very early design phase to evaluate a problem situation in order to decide if and, if so, in what extend the upcoming design should characterized by the feature of adaptivity;
- providing the designer with a conceptual model of an adaptive WIS for indicating the abstract design principles;

- supporting the designer to identify the adaptive system's environment and the model of interaction that would need to be developed by providing generic models and supporting tools;
- providing indicative object model, architecture, technologies and state of the art software frameworks for the implementation of an adaptive web information system.

The proposed framework aims at responding to the above mentioned requirements by obeying the following fundamental principle; the framework should be at the same time:

- robust and abstract enough in order maximize its compatibility with a wide range of current and future WIS applications and technologies, and
- specific enough and extensible in order to lead to realizable WIS

The following sections are an attempt to respond to such requirements and principles.

6.2 Adaptivity notion and metrics

This section aims to introduce the required definition and axioms for the development of the framework under discussion from this thesis' point of view.

From such a point of view, an adaptive system “lives” in a certain changing environment and interacts with entities so that it can continuously fulfills its purpose regardless of the changing circumstances. In other words, an adaptive system must be able to preserve its internal equilibrium in a constant environment and at the same time be able to adapt/reconfigure itself to a changing environment so that its existence is ensured. Doing that it is requisite that the system always acts in order to create itself which in its turn premises that the information which determines the system (code) is preserved and thus distributed to its components. In order to reconfigure itself, an adaptive system need to be able to increase its own internal complexity, expressed in form of internal variety. Such a system could be characterized as dynamic, massively entangled, scale independent, transformative and emergent. These characteristics are the basic factors considered by a designer for the analysis of a system in order to design its adaptivity. They are the basic criteria for the design space analysis, the space where the designer has to take design decisions for creating adaptivity.

Such characteristics should be now transferred to the domain based on which the framework would be developed. Grasping such characteristics to the domain of WIS it seems useful to introduce an oversimplified scenario that would be further used as a reference point.

Thus, consider a WIS as a simple dynamic web page with the capability to change its background color according to the user's color preferences. The specific system has the capability to deliver the web page in N different colors. From this framework's point of view this initial capability provided by the system's designer that can be quantified by the number N is the WIS' internal variety (V). In case that the system has the color that the user requires it is said that the system has the potentials or in other words, the requisite variety, in order to adapt. However having the potential (the required color) does not mean that the system would satisfy the user. The system need to have a mechanism to select between alternatives, or in other words, decide. At such an interaction is is said that the WIS is able to reconfigure itself / adapt to the user's needs as it both has the requisite variety and the selection mechanism. However, it can also happen that a user prefers a color that the system does not have, has not the requisite variety. In that case the system would need to generate such variety (i.e. color). Finally, it is also possible that a user even if she has a specific color preference, she is also color blind and it is impossible for her to work with that. This is a case where the system would need to increase its internal complexity and thus develop alternative inference mechanisms in order to, for instance, take into account more parameters.

Based on such acceptances and scenarios the following sections introduce three adaptivity metrics that would be useful for the development of the conceptual model and consequently the design and development of adaptive systems.

6.2.1 Adaptivity measurement (Am)

The notion of “adaptivity measurement” is introduced in this framework in terms of the system and environment's variety and selection / inference capability. Such a metric is intended to be used to indicate how much adaptation took place in a certain time frame and this is completely unaware of adaptation effect (how successful the adaptation was).

The adaptivity measurement seems to be somehow proportional to the ratio of system variety change to the environment's variety change. Note that the possible system's changing requirements are considered as changing environment and thus this is taken into account. Am is also proportional to the capability of the system to select between available structures (again this is not aware of its effect but at least be able to conclude to a selection). It seems interesting to mention that 1) when system's variety is not changing and thus the numerator is zero this does not mean that adaptation did not happen and 2) when the system cannot select then adaptation cannot take place.

6.2.2 Adaptivity effectiveness (Ae)

Maybe the most important metric for the evaluation of an adaptive system is the adaptivity effectiveness. This is exclusively defined in terms of the fitness situation improvement of the system in a certain time and most of the times where the interacting party is a user this could be expressed in term of user satisfaction.

The user satisfaction at a certain time can be computed in terms of several interaction criteria that could be fed back to the system. Such criteria could be further weighted according to their importance.

6.2.3 Adaptivity capacity (Ac)

This metric is introduced aiming at measuring the ability for adaptivity (not to confuse with "adaptability") of a system; how adaptive a designed system can be. In other words, measuring how long the system can survive (fit) while the complexity of the problem situation increases or/and the environment (system's requirements) continuously changes.

Through the general systems investigation it has been concluded that the system's adaptation ability should be proportional to its variety and its ability/intelligence to manage its variety in its interaction with other systems. Going that a bit further it can be said that variety is a measure of the number of distinct states a system can be in and could be defined as the number of elements in the state space. The "ability/intelligence" refers to the system's capability to both select/inference and generates its structures. It is the interaction between system and environment that makes adaptation both necessary and possible (the "order from noise" principle [49]. Interaction can be represented as the changing system's environment; the variety of its environment (Ve). The system should have the requisite variety (Vs) to destroy the variety of its environment and thus adapt (the law of requisite variety). A system has an inborn variety (given by the designer) that might be increased through interaction. Illustration 31 shows the adaptivity capacity key parameters.

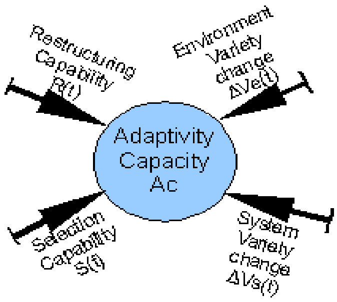


Illustration 31: Adaptivity Capacity

Alternatively, system's autonomy could equivalently replace system's restructuring and selection capability. In that sense system's adaptivity is proportional to system's autonomy.

6.3 To adapt or not to adapt?

In literature presented, adaptivity notion have been interpreted in different ways and applied to different fields and levels; recall for instance adaptivity in adaptive hypermedia (e.g. Brusilovsky) and intelligent agents (e.g. Zimke). But before starting the design of an AWIS, the following questions should be answered:

- When (under what conditions) a designer should worry about designing a system characterized by adaptivity?
- In what extend (sub-systems) the system should be adaptive?
- In what degree an adaptation is enough?

In other words, the designer should primarily evaluate the facing problem situation and according to some criteria decide if and in what extend and degree the system (problem solution) should be adaptive. In this thesis, an indicative empirical design guide chart is being proposed (Illustration 32) in order to assist the designer during such a very initial design phase.

This is a two dimensional chart aiming not at focusing to a plethora of criteria which often vary from one problem situation to the other but instead at providing a visualization of the primary abstract concepts that should affect the designer 's decision. Follows a description of the chart:

- **Y axis:** represents the adaptivity capacity. The minimum value means that the system is not at all adaptive. This means that for a certain user action, under different conditions, the system will always interact in exactly the same way. In other words the behavior of the system is exclusively determined by the triggering event (external user/system action). If its behavior can also be affected by some variables while initialization, then the system can be characterized as configurable or customizable. We talk about personalization if, at runtime, some system parameters are changed (implicitly or explicitly communicated to the system) by either the system or the user to the direction of satisfying the end user requirements. Now, if this is driven by the user then the system reached adaptability threshold. Adaptation goes hereafter this point where the system can said to have a kind of autonomy to drive the change. It would be helpful here to introduce three levels of adaptivity capacity (single, double or triple A), similarly to Zimke's [Ziemke, 1998] levels, starting from “engineering autonomy/adaptivity capacity” to utopian “biological autonomy/adaptivity capacity”.

- **X axis:** represents the complexity of the problem situation (as defined in SSM - [Checkland, 1999]). A minimum value near zero would represent a very well defined problem with very specific / frozen requirements (static environment or do not care about it). On the other end, a completely ill defined problem situation (like those faced by SSM) could be appeared acting in continuously changing environment by which the system would be affected. Between those extreme situations the designer should be able to identify the problem situation under consideration and accordingly decide how adaptive the future system should be.

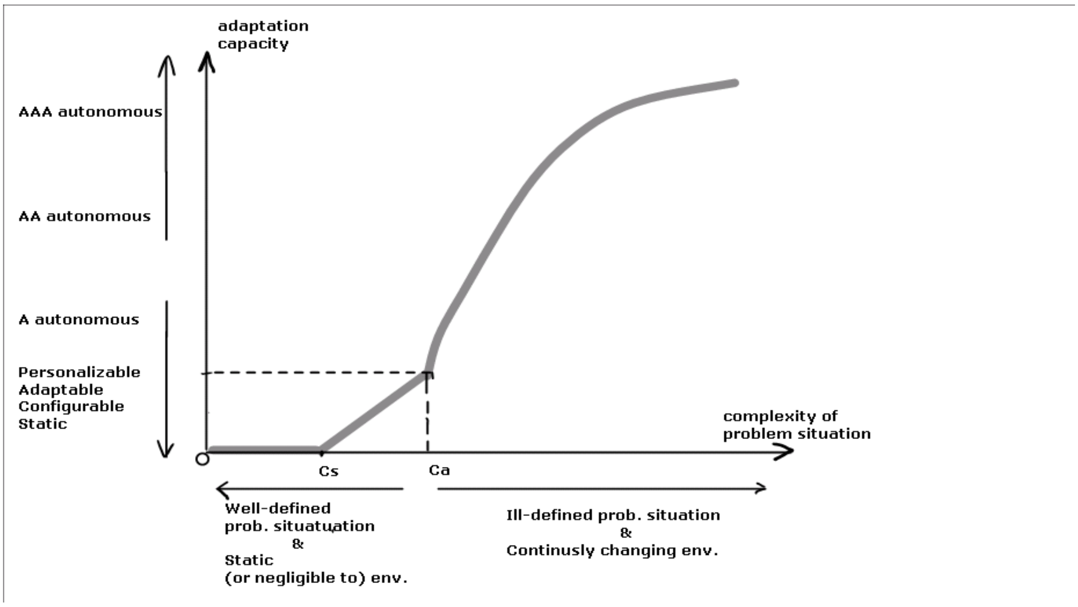


Illustration 32: How to decide how adaptive a system should be

In the graph there can be identified three regions:

- A static one: This is a parallel line to X axes and represents the space where the complexity value of the problem situation remains so low that any kind of adaptation is not required.
- A linear one: This is a line of form ax+b=0 and this represents a linear relation between the complexity of the problem situation and the required adaptation. In this space the complexity of the problem situation has passed a certain complexity threshold (Cs) that makes adaptation a requirement. However, while Ca complexity threshold has not been reached not real autonomy is required and this is actually the region that most of the current implementations can be placed.
- A logarithmic one: This aims at visualizing the fact that from a certain threshold of complexity a system adaptation requirements gets to growing very sharply and this also applied to the autonomy (towards real autonomy). Theoretically when real autonomy can be reached however the complexity of the problem situation would grow the system would be able to survive.

6.4 DAWIS conceptual model and design principles

Inspired and not actually caused by general systems investigation, this section aims to exploit such research developments for building up a conceptual model, "*a simplification of reality intended to promote understanding*"[62], of an adaptive WIS that could push the research and development of adaptive WIS further.

Thus, recalling natural systems section, self-organization seems to be the Utopian adaptive system's main attribute. Having that in mind and looking such a system from a systems thinking point of view, an adaptive WIS (AWIS) could be considered as a system consisting of subsystems. In the case of an AWIS, its control and aim should be distributed to its autonomous subsystems. Such architecture, AI interpreted it as multi-agent system (MAS). AWIS could be considered as a MAS whose members (agents) should be aware of their role both at the order of individual and at the higher order of MAS (how affect the whole). Such an approach is inspired by Luhmann's work in the context of social systems. These agents need to be capable of evaluating themselves by communicating their perceptions in their internal network. The resulting actions would probably lead the whole WIS to a new equilibrium and thus to new behavior. The above mentioned concepts need to be deducted and adapted to the current technological status quo in order to provide a realizable framework.

An adaptive system could either be considered as a system / subsystem architecture or as a multi-agent system with several characteristics which are going to be discussed below by distinguishing two abstraction levels.

6.4.1 The higher order model

This section aims at describing the conceptual model at a higher level of abstraction. At this level a single distinction between the AWIS and its environment is assumed.

From the viewpoint of DAWIS model, an adaptive system "lives" in a certain changing environment and interacts with entities so that it can continuously fulfills its purpose regardless of

62 *http://www.systems-thinking.org/simulation/model.htm*

the changing circumstances. Alternatively it can be said that the AWIS Agent needs to percept its environment and act accordingly so that it fulfills its purpose. This can be seen as two streams of communication; perception and action. At such a level, the agent needs to select the interesting perturbations of its environment, interpret them, evaluate them according to its purpose, decide and act accordingly. This will be repeated till adaptation has taken place. In other words the adaptive system should primarily be able to sense, plan, learn, generate variety and act.

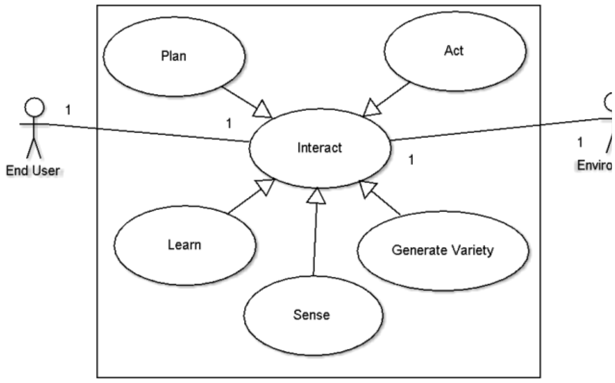


Illustration 33: AWIS Abstract Use Case diagram

Illustration 33 presents such an abstract use case diagram.

Follows a brief description of these use cases. Illustration 34 is a visualization of the perceived conceptual model as it results from this thesis' point of view and literature investigation.

- **Sense**: Sense both the interacting party's actions and the ambient environment of the system undergoing adaptation.

  A system for being adaptive needs to continuously observe (perception) its surroundings and mutate itself in order to sustain the required equilibrium for its effective operation. This means that such a system would need to have sensors for a set of interesting for it, changing characteristics of its surroundings in order to construct an internal model for its environment.

  In other words, under the context of a system operation the designer needs to either predefine or give the system the ability to recognize the certain characteristics that would affect its substance. For such a set of those characteristics the system needs to have a set of sensor components (sub-systems) and also an effective way to communicate/signal the change. However, the resulting set of signals need to be transformed to an overall information in an abstract form that would be appropriate for communicating it. This introduces a requirement for a multiplexing mechanism that, provided a domain knowledge and taking into account the system's purpose, would be able to multiplex a number of input signals, concerning different sensing aspects, to a unique, overall signal. Such a mechanism could refer to inferencing mechanisms having its root to the field of AI. Such a system needs to have a communication protocol capable of enabling interoperability between sub-systems so that the sensed changes would be communicated in such a way that would enable a successful mutation. Such a protocol could be a set of ontologies.

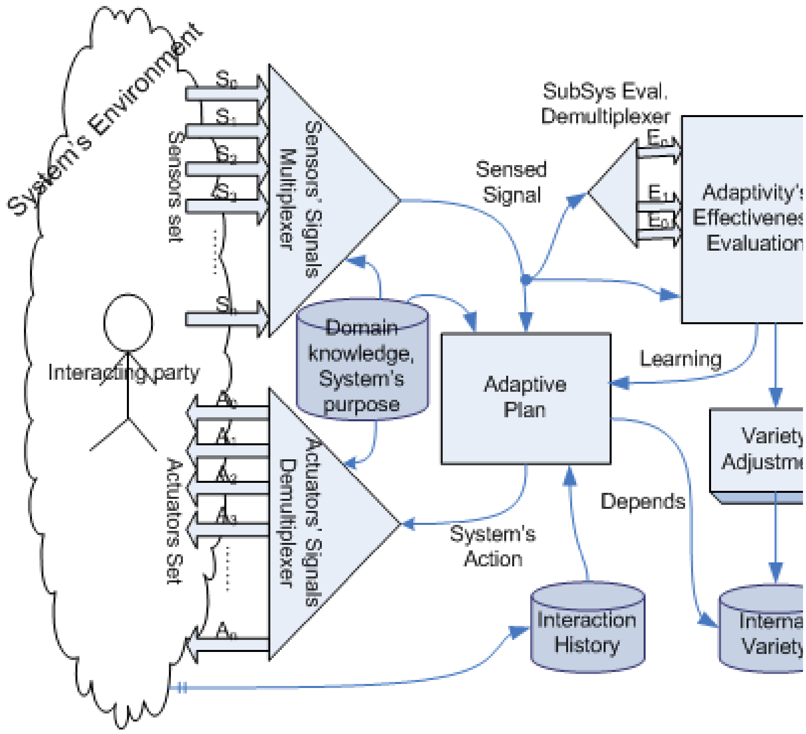


Illustration 34: DAWIS Conceptual Model

- **Plan**: Have always an appropriate adaptive plan.

The adaptive plan is responsible for controlling the system's behavior in response to both the environment's change and system's purpose. The plan is both distributed to subsystems and continuously changing (adaptive).

- **Learn:** Adjust its adaptive plan.

  The adaptive plan is being changed according to the adaptation performance i.e. the fitness of the structures for the environment. This introduces the requirement for measuring the performance (i.e. adaptivity effectiveness) of adaptation. The output of the adaptation plan would be a signal, with information regarding the actions necessary to be taken by the system, so that it will be able to adapt its interaction to the new environmental conditions.

- **Generate variety**: In cases where higher level adaptation is to be performed (increased autonomy) the adaptive plan is responsible to produce structures (system's variety) which perform well (fit) in the environment. In other words it should be able to reorganize its adaptive plan and internal structures (variety) so that it can perform better in future interaction.

- **Act**: Act effectively, according to the adaptive plan.

  Similarly to the sensors' subsystems, a system consists also of several actuators that acts to different aspects of interaction. The abstract actuation signal coming from the adaptive plan now needs to be de-multiplexed so that the different actuators would undertake their responsibilities and affect system's environment.

Note that all the components of the conceptual model are not required in all problem situations. The conceptual model needs to be aligned to the design chart (Illustration 32). Thus, two levels of the conceptual model can be identified corresponding to the two different regions of the chart that identifies the kind of adaptation is required. The already presented model (Illustration 34) is a superset and corresponds to the logarithmic region of the chart; increased autonomy. The model that corresponds to the linear region is characterized by the absence of the capability to alter its internal variety (structures) (see gray region in Illustration 35).

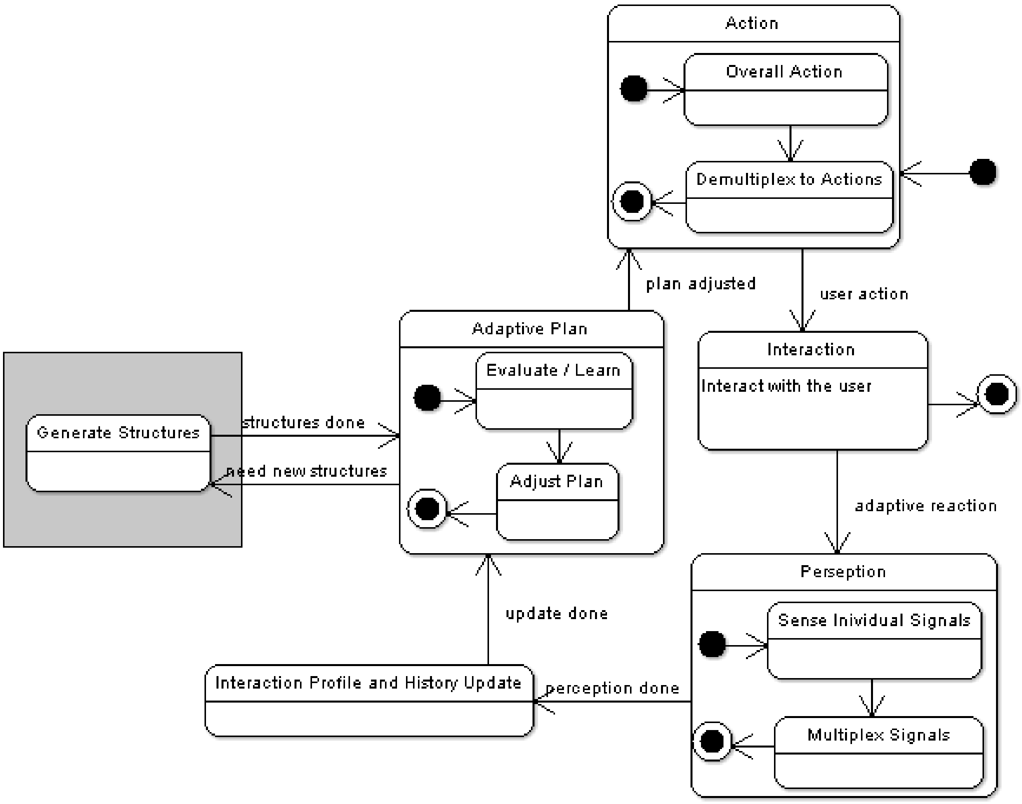


Illustration 35: State diagram of DAWIS conceptual model

Illustration 35 presents the life cycle in form of a state diagram of an adaptive system through the prism of the proposed model. According to that, after an AWIS having accomplished an initial action (e.g. Web page GET) and sensed a change into its environment (essential variables) it has to compose a number of input signals to an abstract overall one. The different signals can consist of both the actual user action / request (interaction channel) but also signals expressing complementary information about the state of its environment. Follows an update of the interaction profile and a recording of the interaction history that will be used as knowledge for the application domain. This is used in combination with the adaptivity effectiveness in order to adjust the adaptive plan. If it is necessary, appropriate structures and mechanisms would need to be generated. The adaptive plan would conclude to an abstract action as a response to the user action. During the last step of a cycle, the abstract action need to be decomposed to several actions that would have equivalent result.

6.4.2 *The order of individual model*

This level of analysis of the proposed model involves the further distinction of the two already mentioned interacting parts above. The AWIS agent now, is seen as a multiagent system or a set of subsystems with the following attributes:

1. **Self-reference:** A subsystem knows itself both at a subsystem and supersystem level. This means that such a subsystem needs to know its purpose and capabilities both as a separate system and as a subsystem of the whole system. Thus, it needs access to emerged properties of the supersystem.

2. **Self-maintenance:** A subsystem can reproduce and reconfigure itself. This means that the system acts in order to create itself. Aiming to avoid its disappearance, because of the continuous entropy increase, the system has to continuously reconstruct itself, by collecting material from its environment and by setting a border with it.

3. **Self-evaluable:** A subsystem can evaluate itself both as a separate subsystem and as resulting system behavior. This means that the subsystem needs to have a way to evaluate itself both in a neural context and in system's context based on the feedback / assessment of emerged system properties.

4. **Communicative:** A subsystem can communicate meaningful with other systems or subsystems through either its subsystem interface or the aggregated supersystem one. This means that the subsystem needs mechanisms for both being able to induct subsystem communication to its supersystem one and meaningfully communicate with other subsystems/systems. Thus, a subsystem could either need to communicate with a human subsystem (in her supersystem) or with a software subsystem (in its whole system). Thus, levels of interfaces abstractions could be defined what would involve context aware transformations. At the top of the hierarchy an Abstract System/subsystem interface could be introduced. This could be expressed in either a specific System/Subsystem interfaces or an Abstract User Interface that this could itself be transformed to several “specific user interfaces”.

5. **Self-optimizing:** A subsystem can optimize or even reject and replace itself. This means that the subsystem needs to be able to either find ways to make itself better in the context of supersystem scope or even find and replace itself with a "better" subsystem.

6.5 WIS Adaptation Aspects – Functional Specifications

The following sections will discuss the key aspects of the conceptual model in order come up with a kind of functional specifications and proposals for the implementation of an adaptive system.

6.5.1 Adapt to what? - The Interaction Profile

An interactive system always operates in an environment and interacts with one or more entities (human or software agents). Such a system would need to adapt both to the interacting entities but also to their surrounding environment. As already mentioned in previous chapters, the system would need to construct internal model of the mixture of its interacting parties and their environments. In other words the system needs to percept its environment though selection of important (for itself) signals. For instance, in case of a human agent, the WIS agent would probably need to take into account her age, her location etcetera.

This section aims at classifying such aspects from literature investigation and research experience, construct an abstract model (interaction profile) and present it in a useful form so that a WIS designer can benefit. This work refers to already extensively discussed fields of user/device modeling, adaptive hypermedia and context aware systems.

Targeting at classifying the aspects, an adaptive system would probably need to adapt, it is important to firstly identify the entities of interest. The following four entities are involved (Illustration 36):

- **the user:** The user is usually the human actor that consumes the system's services.

- **the platform:** The platform is the underlying infrastructure that enables the interaction.
- **the service:** The actual product/service offered by the system for consumption.
- **the delivery context:** This is defined by the session parameters that determine the way a

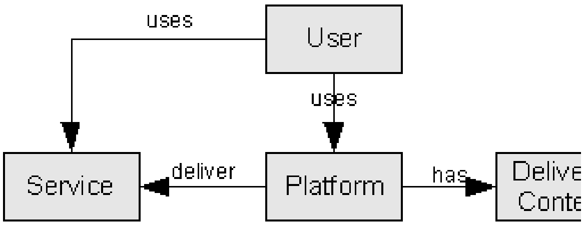


Illustration 36: Interaction entities

service needs to be delivered in order to fulfill its purposes effectively.

Based on such identification the Interaction Profile (visualized in Illustration 37) is developed in two layers:

- An abstract layer that identifies abstract groups of characteristics under four profiles corresponding the above mentioned entities
- A bottom layer that identifies specific characteristics of the interaction entities.

The interaction profile is being modeled using ontologies. Aiming at offering extensibility, there have been developed a single ontology for the abstract layer and several bottom layer ontologies that can be plugged into the abstract one and construct an appropriate profile for a specific problem situation. Regarding the abstract interaction profile layer, four profiles are identified shown in Illustration 38 and discussed in the following sections in more detail.

Note that for each sub-profile's entity there might be identified several entities that can be plugged in. However, the current ones are only indicative and these might be adapted to the requirements of a certain design. Furthermore, it should be noted that in literature and in the web there can be found several ontologies that can either be plugged as they are or adapt to design needs. For instance, in regard to the Personal Info entity Golemati et al 's user profile ontology [Golemati et al, 2007] has been adopted and adapted.

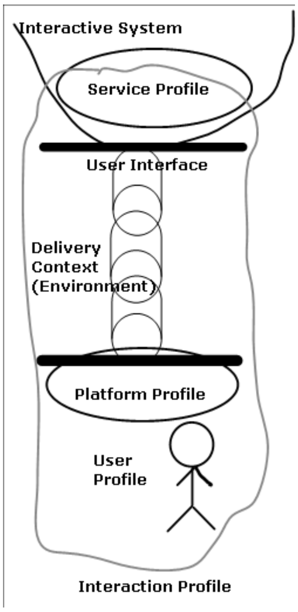


Illustration 37: Interaction Profile

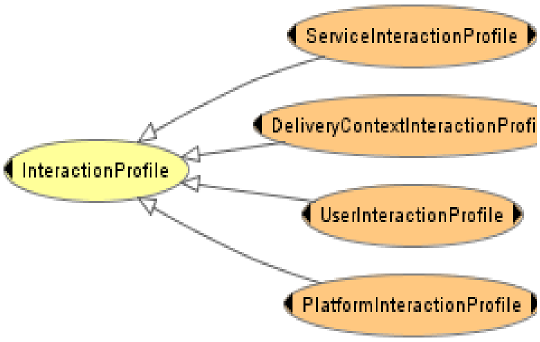


Illustration 38: Interaction Profile – Abstract layer

6.5.1.1 User Interaction Profile

User Interaction Profile concerns characteristics of the user that are neutral to interactive system (application). These are classified as follows and modeled as a second level ontology (Illustration 39), plugged in the abstract interaction ontology.

***Personal Info**:* Contact and user-management information, plus additional demographical data of interest. This entity consists of:

- *Characteristics:* General user characteristics, like eye color, height, weight, etc;
- *Contact*: Other persons, with whom the person is related, including relatives, friends, co-workers;
- *Living Conditions*: Information relevant to the user's place of residence and house type;

- *Person*: Basic User Information like name, date of birth, e-mail plus additional demographical data of interest;
- *Thing*: Living things or Non Living Things the user may posses or otherwise be related to, like a car, a house, a book or a pet.

Ability*:* User abilities and disabilities. Follows a classification based on RNIB classification. An alternative more concrete classification could be TCDL's one[63] containing disabilities like blindness, color vision deficiency, low vision, deafness, hard of hearing, deaf-blindness, dyslexia, dyscalculus, intellectual disability, dexterity impairment, motor impairment, learning disability, functional illiteracy, ADHD, aphasia and speech impairment. However a more widely accepted classification has been adopted for this profile:

- *Cognitive*: Cognitive impairment refers to people with dyslexia and learning difficulties;
- *Eyesight*: This includes people with no vision, or some functional vision;
- *Hearing*: This includes people who are completely deaf or have partial hearing in one or both ears and require the use of a hearing aid;
- *Mobility*: This refers to a wide range of people with varying types of physical disabilities;

Individual Traits: Characteristics that set of individuality. This entity consists of:

- *Interest*: User long term / abstract interest like hobby. For example, “interested in sports”, “interested in cooking”;
- *Learning Style*: Learning style of the user;
- *Personality*: Characteristics like introvert/extravert;
- *Preference*: User preferences, for example “loves cats”, “likes blue color” or “dislikes classical music”.

---

63 http://bentoweb.org/refs/TCDL2.0-20070711.html disabilities list

Illustration 39: User Interaction Profile Ontology Tree

Background: This is defined as all the information related to the user's previous experience outside the subject of the adaptive system, which is relevant enough to be considered. This includes the user's profession, experience of work in related areas, as well as the user's point of view and perspective. This entity consists of:

- Activity: User activities, hobby or work related. For example, collects stamps or investigates the 4th Crusade;
- Education: User education issues, including for example university diplomas and languages;

- Expertise: Includes all kinds of expertise, like computer expertise;
- Profession: The user’s profession;
- Perspective: The user's accumulated perspective;

***Knowledge**:* An adaptive WIS system which relies on user's knowledge has to recognize the changes in the user's knowledge state and update the user model accordingly.

6.5.1.2 Platform Interaction Profile

This sub-profile consists of characteristics of the underlying infrastructure that supports the interaction (Illustration 40). These can be classified to:

- *Software:* Useful information about the operating system and applications run by the device. A distinction between InputSoftware (e.g. speech-recognition, on-screen keyboard, etc.) and OutputSoftware (e.g. Screen-reader) take place. Software can contain sub-classes like User Agent (Web Browser) and Assistive Technology (see TCDL for instance)
- *Hardware:* Hardware description of the platform where the user agent runs. It includes information on CPU type and speed, memory size, network and modem capabilities, Bluetooth and wireless functionalities, etc. Standard and special input devices can be described here within the components InputDevices and OutputDevices.
- *Network:* Global information about the network to access the Internet application: bandwidth, proxies and firewalls, etc (for instance UAProf)

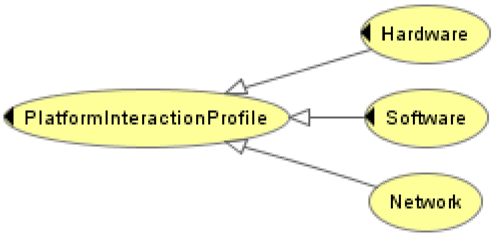


Illustration 40: Platform Interaction Profile ontology tree

Here, CC/PP (ver. 2) seems to be a well established vocabulary that might be plugged in as it is. More on CC/PP and an example provided by W3C can be found in Appendices.

6.5.1.3 Service Interaction Profile

Characteristics of the provided service by the interactive system (Illustration 41). Compilation of properties of the application related to its functionality and interface characteristics.

- *Interaction Specs:* Service properties related to interaction.
- *Functional Specs:* Service properties related to functionality (like web services).

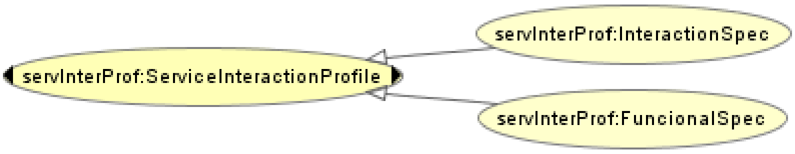


Illustration 41: Service Interaction Profile ontology tree

As an example of these entities' extension would be the web services and WSRP paradigm that explicitly define the specifications of service functionality and interface accordingly.

6.5.1.4 Delivery Context Interaction Profile

This profile contains the additional parameters that characterize an interaction (Illustration 42). These are characteristics that concern all interacting entities during an “interacting session” or, in other words, during the delivery of the requested service.

- *Goals:* These are the properties that define the interaction goals. This could vary from some simple web page access to a learning or task goal (e.g. Buy a cheap book for building web sites).
- *Preferences:* the user can have some preferences for a certain “interaction session”. These may refer to:
    - InputPrefs: User preferences in regard to input modalities.
    - OutputPrefs: User preferences in regard to output modalities.
    - InteractionPrefs: User preferences in regard to interaction issues like navigation, search mechanisms and information highlighting.
- *Experience:* The user's experience in the given service
- *Interests:* short-term / service specific interests aiming at improving information filtering and recommendations
- *Physical State:* Properties that define the physical condition of the interaction (user and system). Such properties refer mostly to properties of location-aware systems and often include information about location and activity.

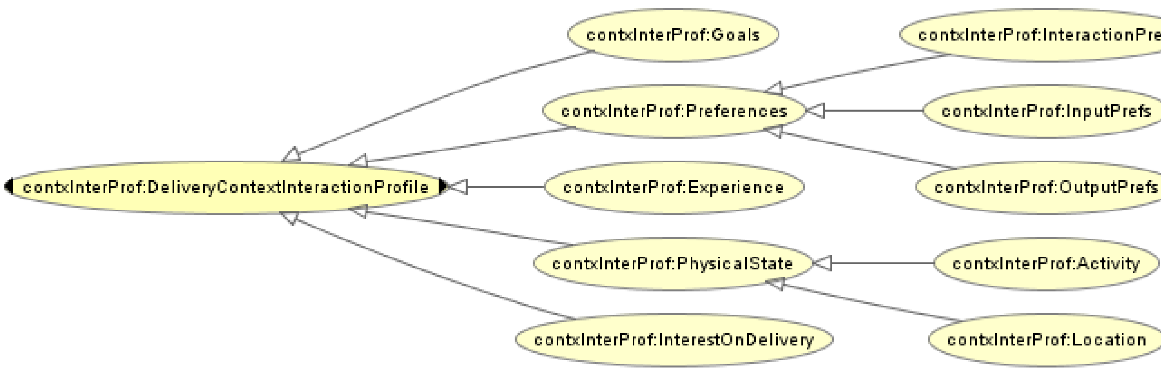


Illustration 42: Delivery Context Interaction Profile ontology tree

6.5.2 *Adapt what?*

The AWIS agent needs to be adapted as a whole according to the changing environment. Last chapter has described the possible characteristics of the environment that can be modeled and interpreted by the AWIS in order to take actions. This section discusses the aspects that such actions can affect. For doing that, a classification of those already discussed in literature investigation section, mainly coming from the Adaptive Hypermedia field, takes place. The proposed classification is being presented in Illustration 43 and its description follows:

- **Semantic Aspects**: Such aspects concern the actual product (information and/or service) offered. This can be further classified to Content and Functionality.
- **Syntactic Aspects**: These aspects concern approaches for communicating the provided product to the user. This can be further classified to Presentation, Interface , Navigation, Modality and Metaphor.

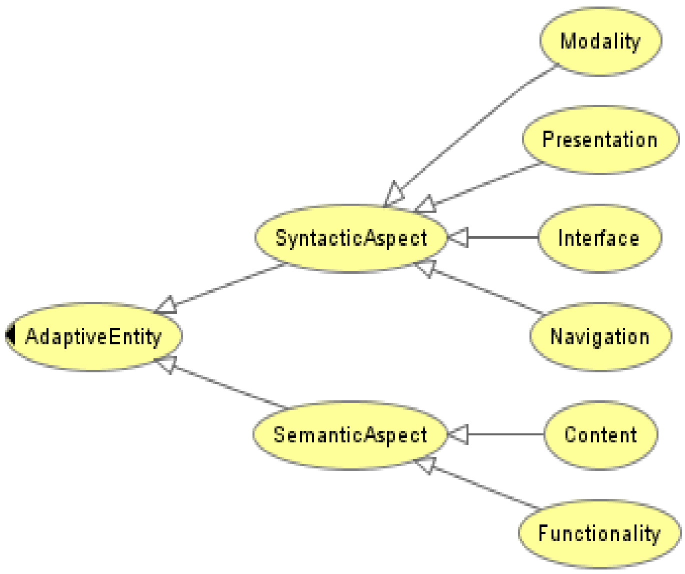


Illustration 43: What to adapt aspects' classification

Comparing with Brusilovsky's classifications ([Brusilovsky, 1996];[Brusilovsky, 2001]) it worths to note that the proposed classification makes a clear distinction between the presentation and content adaptation whether the former one included for instance text adaptation into the presentation group. The following sections discuss the above mentioned aspects.

6.5.2.1 *Adapting the Content*

This aspect consists of techniques that can be applied to either a text or to a multimedia content. These can refer to adaptive hypermedia (see Table 6 - Adaptive Hypermedia Techniques (summarized and classified from [Brusilovsky, 1996])) and also to techniques originated to the web accessibility research field as have been thoroughly investigated in preceding sections.

From the proposed framework point of view a content adaptation takes place when either text or multimedia content is:

- Generated - Retrieved / Deprecated (removed)
- Appeared / Disappeared
- Simplified / Composited
- Summarized / Expanded
- Restructured
- Translated
- Concluded (result of inference process)
- Annotated / Referenced (hyperlinks)

6.5.2.2 *Adapting the Functionality*

This aspect is related to the functionality that is always hidden underneath the system's user interface. This functionality needs to be adapted according the changing requirement so that the system remains effective. From such a point of view a system needs to be able to enable / disable, optimize its and reconfigure its functionality.

6.5.2.3 *Adapting the Presentation*

Adapting the presentation of a WIS means adapt the way the same information (text or multimedia) is being presented to the user using the same modality of interaction. Note that presentation adaptation is considered disjointed from modality adaptation for making these important aspects more concrete. Examples of such adaptations that have been identified in the literature (mainly from web accessibility field) are:

- Enlarging page content—magnifying pages and enlarging specific text or images

- Enhancing text— changing colors, letter and line spacing, and text style
- Enlarging browser controls
- Remove unnecessary formatting
- dim fragments (makes text less visible, instead of hiding)
- remove all the images
- linearize content and HTML tables
- image transforms

6.5.2.4 Adapting the Interface

The user interface refers to the most interacting part of the system that directly communicates with the user. This is often a form or a multimedia / rich user interface.

The design of an application proves very difficult for stakeholders of different background to communicate resulting to the waste of a lot of time with implementations and redesigns as there is no common language for visualizing everyone's ideas in an effective communicative way. Designing for the user, requires the designers to start the design from the user interface ignoring, at least during the first design phase, implementation issues and at the same time allowing for creative design process. For allowing that, the design of the user interface needs to be separated and abstracted from the operations of the system (and of course the technologies) so that it would be widely applicable, either statically or by introducing adaptation mechanisms.

Having that, CAD-like tools could be introduced and enriched with libraries of customizable AUI pattern designs / widgets (e.g. Volume, searching, authentication e.t.c. and their cognitive association of that with for instance up and down arrows.) towards a more creative and effective user interface design and a more user-friendly system. Here we propose the separation of the OSI application layer to three sub-layers towards more user-centered information systems. This could be probably further enriched by extensions to current software engineering methodologies like UML, the introduction of a merged supporting formal methodology or even by the support of such process with design aid tools

Follows the proposed sub layers as presented into the Illustration 44. In this figure it is assumed that point-to-point actors (user or system) should be able to communicate using a specific application but using a completely different interaction pattern/modality/interface

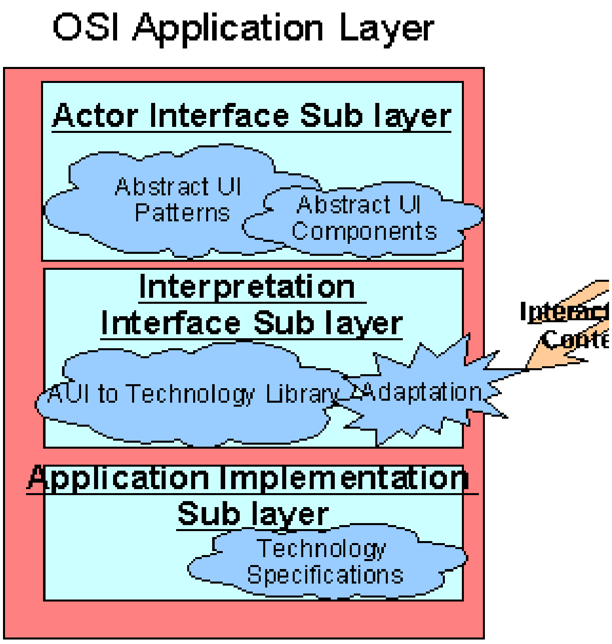


Illustration 44: Proposed sub layer of OSI Application Layer

Actor Interface Sub layer (AUI): This refers to a complete separation of the actor interface from the system. The interface needs characteristics that would enable it to adapt to any kind of "context of application", from a pc keyboard/mouse, to a TV remote control or even nested level of UIs (e.g. A remote control interface that controls a software GUI on my media center device). There seems also to be emerging a requirement for adapting the metaphors used in UI design as the mode of operation frequently changes (desktop – WIMP vs. walking PDA for instance). The interface needs to be completely human-centered and the target subsystem of the Actor Interface Sub layer designer consists of abstract user interface components and users only. The design will be affected by HCI factors such as usability, cognitive process, psychology etc. This design process at this stage could be supported by parameterizable AUI patterns and could be visualized by sketching, prototyping etc. Finally, the work flow needs to be modeled and this also needs to be done abstractly and maybe separately.

Actor interface can be described by an abstract language - application abstraction - one interface to many systems. This can be further divided to navigation e.t.c. Such an abstract user interface language could be used as the representation of knowledge that can be extended and updated.

Interpretation Interface Sub Layer (Mapping and Adapting): The Application Interface sub layer will be a translation layer between the other sub-layers. This can be visualized as a number of cascading filters that the AUI will pass through in order to be delivered to the Implementation sub layer. In this sub layer a mapping of the abstract components and actions will be handled and transformed though a static or dynamic process to actual technology specific correspondences. In case of the dynamic transformation, this actually refers to UI adaptation and the system's designer needs to set the requirements on what and how to adapt.

Application Implementation Sub-Layer (Operations): This includes the actual implementation of the mapped components and actions.

6.5.2.5 Adapting the Navigation

This refers to the way of exploring the user into the product/service offered according to her characteristics in order to help her finding what she is looking for effectively. At this group, techniques that Brusilovsky [Brusilovsky, 2001] has described for adaptive hypermedia applies as they are including direct guidance, adaptive link sorting, adaptive link hiding, adaptive link annotation, adaptive link generation and map adaptation. Furthermore, literature has been identified additional techniques including adaptation of web content based on semantic rather than syntactic constructs – facilitating navigation by streamlining the web interface according to abstract user goals and involvement of software agents & planning technology to assist users in simple and complex query-answering tasks employing machine learning techniques.

6.5.2.6 Adapting the Modality

There have been a lot of years since users were interacting exclusively with hypertext. Nowadays, more and more WIS are based on multimodal technologies that can make them usable under diverging scenarios of use including handheld devices and even users with diminished interacting capabilities (ex. Blind). Multimodal interaction is seen as the communication of the offered product in different types of media that corresponds to different communication channels like video and speech. Recent technological advances like SVG, SMIL and MPEG-21 framework aim at providing the infrastructure for such adaptive multimodal WIS.

6.5.3 Adaptation evaluation

After “acting”, the AWIS agent needs to evaluate its action. This could involve an immediate reaction or a long run feedback. The agent needs to evaluate itself as a whole regarding its purpose. In other words, the agent needs to sense its adaptation effectiveness. More specifically, the agent needs to be capable of sensing the differential satisfaction of the user after every adaptation cycle. This means that the agent needs to have appropriate sensors to cover parameters relevant to the user satisfaction (e.g. Time in site, explicit rating etc). The set of sensors could be expressed in a form of test suite containing several tests that happen to satisfy or not several criteria. Moreover, there raises the requirement of effective communication of the evaluation/ testing results between such subsystems.

The Evaluation and Report Language (EARL) under development by W3C is a framework targeted to express and compare test results in the field of accessibiliy. The concept of test under EARL is taken in its wider acception, and can include anything “identifiable” by a URI. EARL statements contain information about the context information (i.e.. who or what ran the test, the date the test was run, and other information about the test was performed), the test subject (i.e. web pages, tools and user agents), the result (pass or fails and probably certainty factor) and the test criteria. Such a technology could be abstracted and used in the domain of adaptive WIS. An example illustrating how this can happen is being presented in next chapter (7.4).

6.6 DAWIS Implementation

This section starts up by presenting the requirements and basic principles for reaching the implementation of an adaptive system as this emerges from the above presented conceptual model. Such a framework should support the designer by providing a preliminary architecture of an adaptive system through the prism of this thesis' approach. Furthermore, it provides an implementation model and the state of the art technologies and software frameworks that would enable the implementation of foreseen adaptive WIS. The primary principles for designing the architecture and the implementation model are again to keep the formers abstract enough so that the results would be technology neutral and to allow for extensibility.

6.6.1 Architecture design: aspects and indicative API

This section will present the proposed architecture of an adaptive system as this is perceived from both the conceptual model and the current technological advances. In such a problem situation there can be identified three areas of special interest:

- the System-User interface;
- the System-System interface, and
- the adaptive system's internal implementation model

The whole architecture highlighting the above areas is presented in Illustration 45 and discussed in following sections.

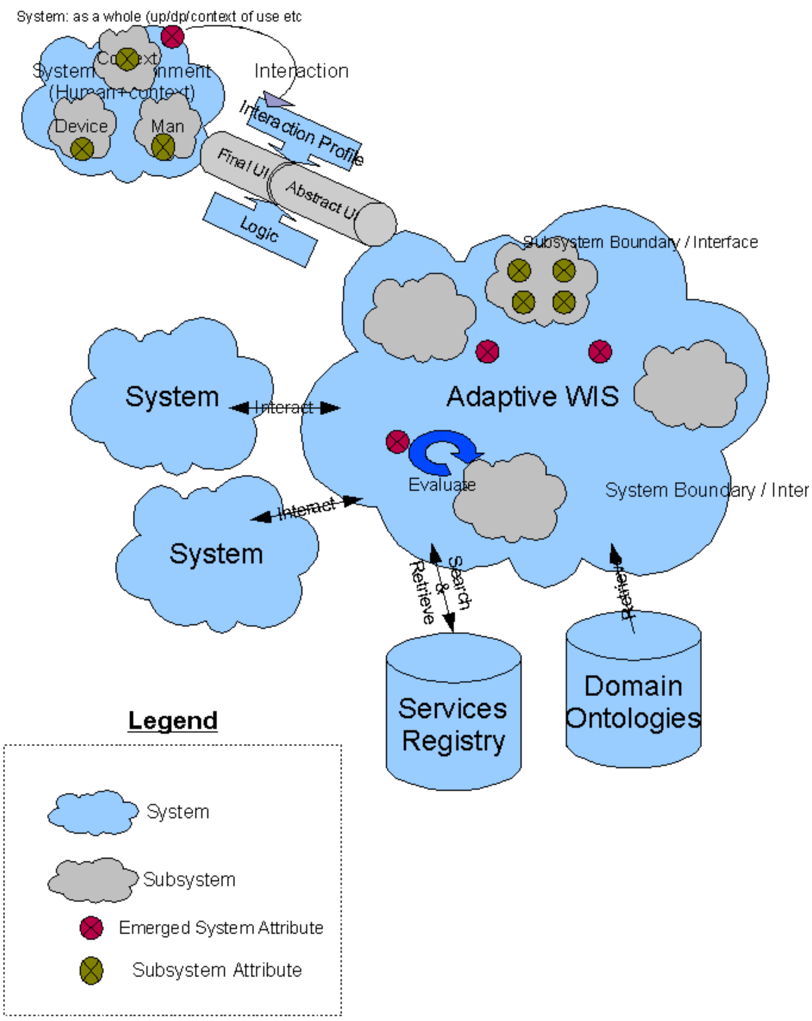


Illustration 45: DAWIS Architecture

6.6.1.1 *The System-User interface*

When a user is interacting with the system, it is not actually only the human being that is interacting but her whole environment. This, as already discussed in previous section, can consists of the device used, the software etcetera. This is actually what the proposed framework names as interaction profile and this had to be taken into account for making the interaction effective. The implementation approach being proposed under this framework is through the use of abstract user interfaces based on the design model discussed in the previous section.

The abstract interfaces could be transformed based for instance on user/system profile and device/context of use to the appropriate modality/metaphor etc. Methods and techniques from adaptive hypermedia such as link hiding, sorting, annotation, direct guidance and hypertext map adaptation could be also combined.

6.6.1.2 The System – System interface

An adaptive system should be also capable of taking advantage from its interaction with its surrounding systems. In our days, this becomes even more critical with the wideness of the world wide web. Maybe, this is our chance to compensate for AI "failure" as the openness would enable huge interaction and hopefully emerging system's variety.

Based on such observations, this framework suggests the adoption of a service oriented architecture for the design and the implementation of adaptive systems. In the emerging digitally open environments, like World Wide Web, a system would be able to decide according to its requirements what kind of services it needs for acquiring that. This involves the following functional requirements:

- self-evaluation / decision what it needs;
- inquiring for appropriate service (thus also capability to check appropriateness);
- interoperability, so that the service can be plugged into the system;
- management of rights / maybe hiring services and service providers authorities.

Considering the requirement for self-optimization, a subsystem could be semantically acquire for better implementation of itself in the World Wide Web, rend/buy such a service and deploy it by replacing itself. All these could be enabled based on semantically enriched web services, described in the previous chapter.

6.6.1.3 Internal implementation model: indicative object model

This section introduces a model for the implementation of the proposed internal adaptive system model. This is a blending of design and development patterns with technological advances in the service of the proposed conceptual model.

Currently, in order to promote extensibility of software systems there have been introduced the so called component frameworks. Such frameworks allow a software designer to easily replace software components implementation with others and further develop more components that can be recruited by referring to an abstract interface (i.e. Simple protocol). These actually implement concepts like Separation of Interface and Implementation, Inversion of Control and Aspect-Oriented Programming that could complement to object oriented approach and provide amazing possibilities for implementation. Most of these are based on the powerful syntax offered by XML.

Thus, adaptive web information subsystems could be defined as software components/services that are interfacing to the whole system through such a framework and provide agreegation. Such frameworks, like Apache excalibur or Spring framework, could probably be extended to provide dynamic reconfiguration of components and used for assembling an adaptive WIS. Then, interfaces (APIs) could be defined to map to the conceptual model attributes and promote adaptivity design and implementation (see Illustration 46 and relating discussion). For instance, Apache Cocoon could be considered as an example framework that on top of generic components has defined specific interfaces like transformers, generators etc. Then pipelines of such components are used to provide certain functionality. In the case of the AWIS, interfaces for perceptors and actuators could be developed in order to provide a standard and extensible way for the system to percept and act.

Furthermore, the components of such a system would need to be accompanied with appropriate metadata (RDF(S)/OWL) that would allow them to know themselves as components. But also, the assembly mechanism of the components framework would provide mechanism for them to access supersystem scope and attributes status. Semantic mechanisms would also need to be used for the self-evaluation of subsystems. For instance, as already mentioned before, EARL could be seen as a standard evaluation feedback protocol.

Furthermore, this section attempts to build up an abstract indicative API for adaptive systems aiming at showing up how the functional requirements that came up from interdisciplinary research could be mapped to a software system for making it adaptive. Illustration 46 is a UML class diagram presenting the basic entities of such a system and how they are related to each other.

- **DAWISComponent** is the most primitive class and consists of attributes and behaviors that all DAWIS components should be capable of. The model of such component comes out mostly from the attributes presented in the order of individual conceptual model.
- **Systema** is meant to be a class of a system. This model applies to any abstraction level and thus can be applied to either super-system or sub-system level.
- **InteractionProfile** has been extensibility described in previous sections.
- **Sensors** are components that can "sense" the environment and communicate the outcome in a formal and semantic way to the inferencing components. For instance, there could be concrete sensors for the elements of the interaction profile.

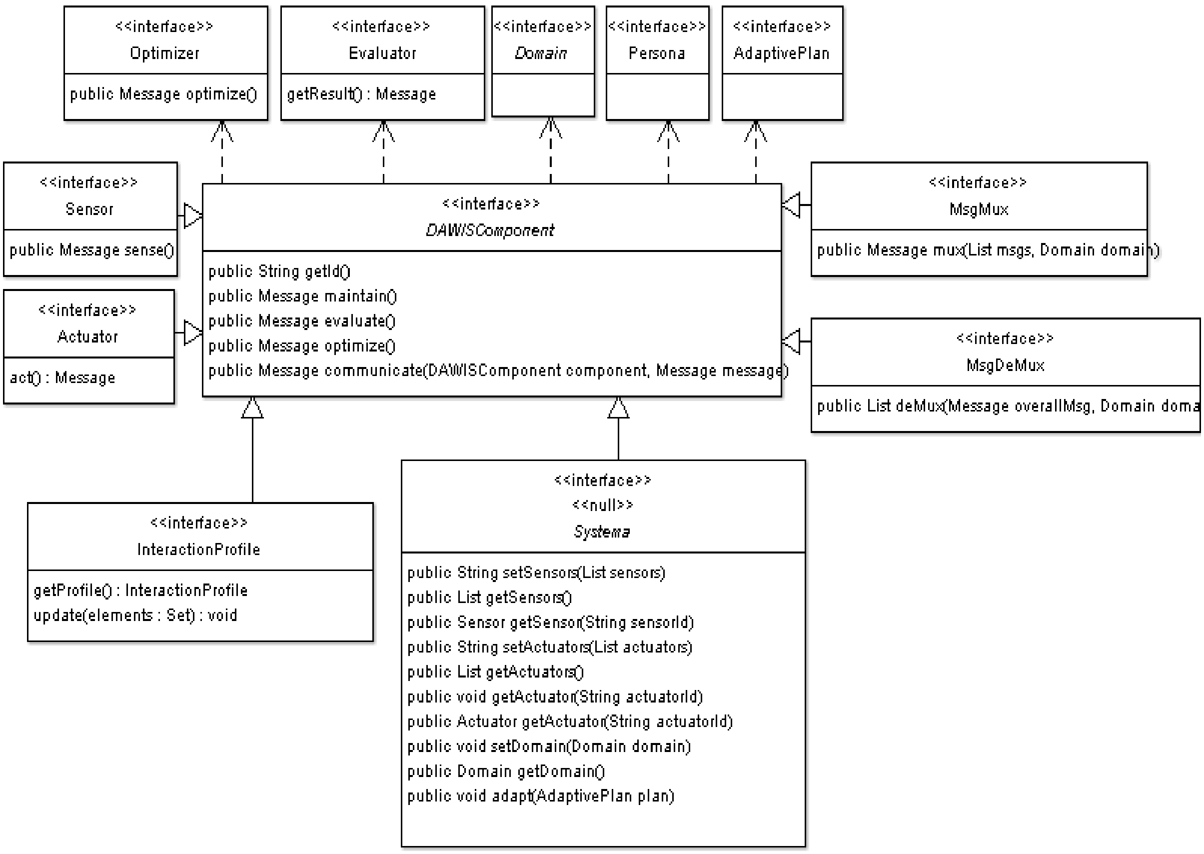


Illustration 46: DAWIS Core Class diagram

- **Actuators** are components that respond in form of interaction with the user to the sensed requirement. Often actuators are actually transformer components that can transform the content, the presentation or even the modality. Speaking in technological terms a big proportion of such transformers could be simply XSL transformation of XML documents that can represent a simple document (e.g. XHTML), a user interface (e.g. XFORMS), multimedia (e.g. SMIL, MPEG-21) or even modality (e.g. VoiceXML).

- **Domain** is the class representing the domain of the application so that an effective communication can happen.

- **Persona** refers to the self-reference attribute as described in the conceptual model and thus aims at modeling the purpose and capabilities both as a separate system and as a subsystem of the whole system so that it will be able to access the emerged properties of the superystem.

- **Evaluator** is a component that according to accumulated knowledge is going to evaluate the current fitness of the adaptation process and communicate results as presented in the conceptual model.

- **Optimizer** is a component that is responsible for optimizing the sub-system it belongs to.

- **AdaptivePlan** is a model of the adaptive plan. This can be either static or dynamically changed.

  The adaptive plan according to Holland [Holland, 1975] is the set of factors that controlling the changing mixture that the system undergoing adaptation is characterized and constitutes the works of the system as far as its adaptive character is concerned. The adaptive plan determines just what structures arise in response to the environment in a continual attempt towards to fitness of the resultant structures with the environment.

  This adaptive plan could be probably implemented using genetic algorithms. This algorithm will be responsible for generating new structures (mutations) but also new relations towards to fitness. Those new structures and relations will be reorganized, grounded to technical characteristics and evaluated by the measure of performance. In order to make such algorithm more effective and avoid logical/inference conflicts we can add constraints with weights maybe implemented as a neural net.

- **MsgMux** is the component that multiplexes a number of sensed messages (signals) and inference an abstract one.
- **MsgDemux** is the component that demultiplexes an abstract action message to concrete actions of the interactive system.

6.6.2 A physical architecture

Even if in the proposed conceptual and implementation model the adaptation control is distributed an implementation scenario of a physical architecture could be based a client / server model as presented in Illustration 47.

The “adaptation server” consists of the Interaction Profile Server (an extended user modeling server) and the Application Server. In addition, IPS is responsible for both generating and evaluating the application structures (mutations). The AS will be responsible for retrieving application specification and actor profile and generate abstract interface according to interaction profile server. Then this abstract UI needs to be grounded to specific technical characteristics.

Every application can have an application specification (e.g. XML) according to standards abstract user interface elements. The UI that wants to connect to this application retrieves this and according to the actor profile it receives the schema of this interface (xsl) from the interaction profile server.

A scenario could be:

1. Client sends request for application
2. Application server responds with its UI application specifications
3. Client sends both interaction profile and application's specification to Adaptation Server
4. Adaptation server responds and updates its knowledge (mutation, evaluation)

Amongst the environments of interest, have been: Apache Jetspeed Portal, a portal that offers a coherent front end application for end-users. It is a hub from which users can locate all their commonly used web content. It makes use of user profiling to offer customisation and personalisation, as well as multi-device adaptation; Apache Cocoon an open source frameworks targeting personalisation and device independent publishing; DELI [Butler, 2002] is an open-source library that allows Java Servlets to resolve HTTP requests containing CC/PP[64] information. The CC/PP (Composite, 2002) specification describes two protocols for transmitting the device profile from the client to the server.

The proposed framework has been emerged from both the interdisciplinary investigation but also from the evaluation of relative aspects though the developments of web information systems in different application domains. The application of such aspects on scenarios of a remote accessibility evaluation, web portals and interactive television. These case studies are discussed thoroughly in the next chapter.

64 Composite Capabilities / Preferences Profile (2002), December 15, 2002, from http://www.w3.org/Mobile/CCPP/

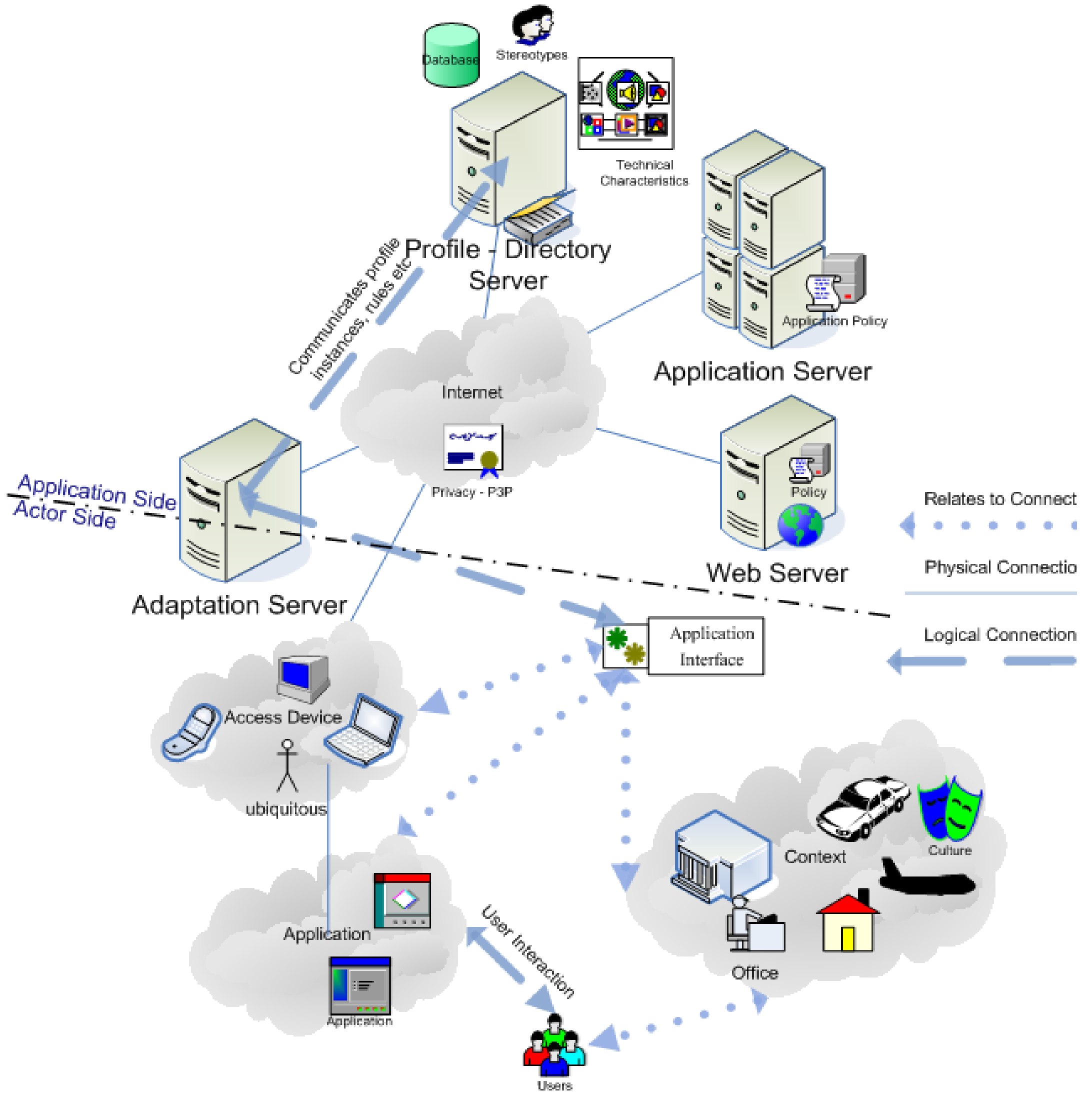


Illustration 47: A possible DAWIS physical Architecture

6.7 Exploitation through DSE: The requirements

The accumulated knowledge and the framework itself as an outcome need to be exploited in a way that would become as useful as possible to the designers of adaptive web information systems. This can take the form of a design support environment (DSE) with the following requirements:

- Provide an introductory material for the introduction of designer to the field of the adaptive web information systems. A web information systems design and development background will be prerequisite.

- Provide an interactive system that would guide the designer to the decision whether adaptive web information design is an appropriate solution for her problem situation. If this is an appropriate solution, the system would need to inference about the level of adaptivity required always from the interaction with the designer (e.g. thought a questionnaire)

- Provide the designer with the conceptual model of an adaptive system and indicate the fundamental principles.

- Guide the designer to identify the domain of the problem and the purpose of the future system.

- Guide the designer to identify the model of perception and the model of actuation of the future system. In other words based on the DAWIS interaction profile ontology, push the designer to identify what need to be sensed by the system and how the system could act.

- In case of a requirement for higher level adaptation provide the designer with methods, tools and references from AI and semantic web so that she can attempt to reach the required adaptation though learning and restructuring.

- Provide an indicative API, examples, technologies and state of the art software frameworks.

The above mentioned requirements, that could be also considered as a method for approaching adaptive web information systems, could be materialized as itself an adaptive web information system probably based on aforementioned technological advances (e.g. Apache Cocoon).

6.8 Summary and Discussion

This chapter has presented the proposed framework for designing adaptive information systems (DAWIS). First of all, the design requirements for such a framework have been identified. Then, according to them, axioms and definitions have been stated aiming at clarifying, in the context of this work, the notion of adaptivity by also providing primitive models / metrics, required for further analysis and design (adaptivity measurement, effectiveness and capacity). The models have been also expressed as mathematical models but these are only in a very primitive state. These have to be evaluated, maybe some though simulation, in order to provide a more appropriate representation. However, such an analysis was beyond of the scope of this research but at least this thesis has provided primitive models for evaluation and further improvement.

Then, an attempt to support the designer on the decision whether and in what extent a problem situation requires to come up with an adaptive system has been presented. This is also a primitive and empirical graph that could be improved through its application to real designs. In addition, this is expressed in term of not straightforward quantified parameters such as “complexity”. This has to be specified maybe by introducing classes of design situations and domains of application.

Furthermore, this chapter introduces a two order (higher and individual order) conceptual model of an adaptive web information system. This has been based on the literature investigation and has its roots to both the natural and artificial systems as presented and discussed in previous thesis' chapters.

Afterwards, an attempt to identify the fundamental functional specifications of an adaptive system took place. Thus, a so named, interaction profile has been proposed based on relating literature, experience and proposed classification. Then, the “adapt to what” question has been discussed and the need for evaluating system's behavior. The benefit of such work is twofold:

- provide a classification of entities based on literature investigation and research experience through DAWIS' prism;
- provide plenty of entities' instances having been used in the past.

Moreover, this chapter has proposed an implementation architecture by identifying key aspect, providing indicative API and proposing useful technologies and software frameworks. Such proposals give the state of the art of the technological landscape but by no means aim to restrict the designer to their use. This chapter closes by discussing how the proposed framework could be exploited through design support environments. It seems that the proposed approach can be also seen as a kind of methodology that could be followed for designing adaptive web information systems.

Chapter 7. Case Studies

7.1 The IRIS Case

The research team participated in the IRIS project, co-financed by the IST Programme of the European Commission (IST-2000-26211). The IRIS project[65] has been an international project with partners in German and Spain. The scope of the IRIS project was to support all designers to design web applications and services that implement a wide variety of accessibility/usability/DfA recommendations, as well as relevant user modeling techniques. The need for the IRIS project was based on the fact that designers are engaged in solving a design problem that requires a DfA (Design for All) approach, which is not supported by existing web development tools. IRIS project was initiated from the lack of familiarization of Web accessibility guidelines from the Internet and ICT industry Community and absence of multimodal services to support Web designers to design inclusive Internet applications. More specifically, that time development/authoring environments offered limited support in terms of built-in functionalities for inclusive design and comply with a small set of relevant guidelines.

It is commonly accepted that Universal Access to information services depends upon the adaptation and customization of content and presentation. This chapter presents an approach that tackles the adaptation process. Parts of these adaptations have been developed under the umbrella of the IRIS[66] project through the combination of user and device profiles, based on the

65 http://www.iris-design4all.org/

66 *Incorporating Requirements of People with Special Needs or Impairments to Internet-based Systems and Services (IRIS). Information Society Technologies Programme (European Commission), IST-2000-26211.*

Composite Capabilities/Preferences Profiles framework [Klyne et al, 2002], via a client-side proxy, together with server-based adaptations. This chapter is adapted from [Velasco et al, 2003] paper aiming at presenting the authors' experiences came out from the IRIS project and further research in outlining and implementing user profiles, as well as possible integration paths with device characteristics.

The underlying principles of such efforts are based upon the following premises:

- Information about user factors, and not just device factors, must be included in the adaptation of service delivery mechanisms in order to obtain access to these services for all people, particularly for people with disabilities.
- Neutral, and preferably standard, terms are required for expressing user interface device characteristics and user needs and preferences in order to allow service adaptation to respond appropriately to the user's needs and preferences when the user is operating in the context of those devices [Zimmermann et al, 2002].
- The CC/PP framework developed by the W3C[67] is a suitable means of communicating user and device demand-side characteristics to adaptable web applications, if augmented with additional vocabulary in the user needs and preferences realm.

The user and the adaptation process need to be able to converse at a meta level, describing what is required, when the process of incremental adjustment from a general purpose initial profile or interface binding fails to work. Incremental adjustment from a general-purpose initial profile will fail to be adequate for people with sufficiently severe functional impairments, and when so many adjustments need to be made that improvement or degradation is not obvious as the several adjustments are individually changed. An example of a functional impairment where customary interface usage will be unusable and hence incremental repair fails, is the situation of persons with attention deficit confronted with a typical web-shopping home page [Brown & Lawton, 2001]. An example where the parametric adjustments are too numerous for the incremental change process to find its way to a feasible set reliably concerns people with severe but not total vision loss using a representative atlas (map) application [Velleman, E. 2000]. A third case where systematic description of needs and preferences is indicated, is where significant morphological changes need to take place to reach the best accommodation. One example of this

67 World Wide Web Consortium, http://www.w3.org/.

has to do with low-vision users of screen magnifiers. Here horizontal scrolling to read lines of text that reach off the screen is a major obstacle to usability. Text should be wrapped within the limits of the actual field of view of the user if at all possible, whether this bound is set by the surrounding content on the layout canvas, the borders of the visible area after screen magnification, or the limits of the user's own field of view being significantly smaller than the extent of the physical display screen. In such a situation it would be better to take the PDA or cell phone layout template as the master topological layout of the material, rather than the more pane-rich layout typically used with a device of a large pixel size. Similarly, users with severe learning disabilities need dialog processes to be boiled down to few choices at a time, after the manner of a voice dialog, and not be assaulted by a welter of attention-seeking sub-displays, as is typical on the web today [Brown & Lawton, 2001]. Here the morphological transformation is in the state transition graph geometry of the dialog, and not the instantaneous geometry of the concurrent information display.

The point of these examples is that there are some people with disabilities who can be served within the un-extended range of adjustment of adaptable services, but not without systematic tools to characterize their needs. There is in addition considerable interest in the general information products and services industry in some way of attaining persistent and portable personal preferences, so that users would be able to deal with newly encountered resources within the comfort of their own climate of preferred adaptations. Such portability could be achieved via Web Services [Velasco & Mohamad, 2002].

7.1.1 The need for neutral terms and the CC/PP framework

Information services, especially as accessed across the Web, are developed by a large number of independent activities. The user's needs may be familiar to their personal equipment, and in particular may be pretty well indicated by the settings on the interfaces that they routinely use. However, the presence of assistive add-on technologies, and the settings of the mainstream and assistive technology components are not covered in the terms defined in the UAProf vocabulary [User Agent Profile, 2002] presently available for use in CC/PP to guide adaptation to mobile devices by web servers.

As the user needs and preferences gathering/assessing applications and the service adapting engines are developed by different parties, a common language is needed to enable the adaptations performed by the server to address the needs and preferences known to the client. That is to say mutual meta-language is required if one is to exceed what can be done by separate trial-and-error training periods with each service. The latter can be tolerated where personalization is a frill, and only used on a few sites that one visits often. For the person with certain disabilities, the meta-dialog is the only way to reach a viable adaptation, and the neutral language for the necessary degrees of freedom is essential.

In the last 20 years several generic user modeling systems have been developed to allow adaptation in different software applications [Kobsa, 2001]. The majority of these developments were academic, never reached the commercial arena, or had very little impact in mainstream software (mainly with very limited customization options in mainstream operating systems). With the explosion of the Web, and e-commerce in particular, several commercial user-modeling tools appeared in the market with the objective of adapting content to user tastes and preferences. There are attempts to characterize user preferences, as for example the CEN standard for smart card encoding of user interface preferences[68]. The emergence of mobile devices has led as well to the appearance of device description vocabularies such as UAProf to provide some basic content adaptation capabilities.

The Composite Capabilities/Preference Profiles framework (CC/PP), offers the possibility to define user and device profiles for an adequate adaptation of content and presentation for Internet services. CC/PP is based upon RDF (Resource Description Framework; [Lassila & Swick, 1999]) a general-purpose metadata description language. RDF provides the framework with the basic tools for both vocabulary extensibility, via XML namespaces, and interoperability. RDF can be used to represent entities, concepts and relationships in the Web. In later sections an overview of some preliminary implementations of these profiles, and the adaptation process within the IRIS project will be presented.

68 http://www.tiresias.org/reports/en1332_4.htm

7.1.2 The Integration of user and device profiles. Content metadata

The motivation for integrating user and device profiles is to create a mechanism that enables users to access information in a way that is best suited to their needs. These needs may permanent, that is, they have a condition that will not change, such as a disability, or may respond to temporary context-based handicapping situations (hands-free operation, noisy environments, inability to access usual input devices, etc.). Within the IRIS context, this information is used to adapt presentation and navigation features. However, the aim of the project is to develop a flexible framework that can allow in the future content adaptation as well.

In regard to content adaptation, there is another strand of work that is contingent to the work described in this paper, that of accessibility metadata. In order for the user and device profile integration to be useful, the content delivered must be also meaningful for the users. However, there is a strong case for saying that users will also want to be aware of content, even if it is something that is not directly accessible to them, according to their device and user profile. After all, they may be able to change access devices, or even ask the author of the content to supply an accessible equivalent or alternative. Therefore, before users embark on downloading, negotiation between the content metadata and the device could take place, to ensure that the resource can be rendered.

Efforts in this area are still very fragmented with several groups working somewhat independently, as well as some confusion over what can be meant by accessibility. For example, for some groups, it is the right of access to documents, or the accessibility of Dublin Core itself. There are also other important issues linked to accessibility, such as the fact that resource discovery can be as important as downloading the resource. The Dublin Core Accessibility Group[69] is working to understand these different contributions, bring their work together[70], and elaborate in the future a metadata profile for accessibility, examples, and guidelines for accessibility metadata. At present there is no well-specified format for a description of the accessibility of a piece of content or 'resource', for general purposes. There is considerable impetus from the IMS

69 http://dublincore.org/groups/access/standards.html

70 http://dublincore.org/groups/access/workshop-20021017.html

project[71] to release specifications for standards of the LOM (Learning Object Metadata). As increasingly, online courses are being accessed by learners with disabilities, it is important that the user profile, known as the Learner Information Profile (LIP), which is designed to be used by the learner as they travel through systems, enroll in classes, participate in courses and undergo assessments, can also register particular needs, and that these needs be accommodated. IMS is still in the process of determining just how the LIP will characterize accessibility needs and how they will be related to, and matched with, accessibility metadata profiles for resources and services.

Therefore, given this state of affairs, it is probably fair to say that as work on user and device profiles continue, these can contribute with their requirements to helping to shape the descriptions needed for matching resources to users and devices, i.e. the accessibility metadata.

7.1.2.1 The Identification of domains

The integration approach recognizes different actors that intervene in the use of Internet services. There have been identified the following: user, access device (including the user agent), content, application to deliver the content and author. The inclusion of the authoring process lies outside the scope of this work, and presents interesting challenges, especially when addressing the issue of device-independent authoring. These actors are represented by:

- User profile. Compilation of different information aspects associated with the user, e.g. personal data, functional characteristics and interaction preferences, etc. It also includes information about the context of use (either automatically acquired such as GPS location, or given by the user).
- Device profile. Summary of the device characteristics: hardware, software, operating system, etc. We foresee a dependence relationship between these two profiles where blending or user and device characteristics take place.
- Application abstraction. Compilation of properties of the application related to its functionality and interface characteristics. This aspect is covered partially under the scope of IRIS as content presentation issues have been only considered. A universal application abstraction mechanism is not foreseen, but it might be feasible to develop some standards for generic types of applications (AIAP-URC).

71 http://www.imsproject.org/

- Content metadata. See previous section.

7.1.2.2 The Device profile

CC/PP does not define by itself a vocabulary to represent user or device profiles. It is a powerful framework to develop such vocabularies. Up to now, the only implementation of CC/PP is or User Agent Profile, by the Open Mobility Alliance (formerly the WAP-Forum) and targeted to mobile devices. UAProf has a very specific scope and cannot cover the whole spectrum of devices that can access nowadays Internet services and applications, nor even considers user characteristics.

UAProf presents two additional problems. First, mainstream operating systems and user agents do not implement any CC/PP profile, which forces to generating some proxy-based implementation of the device profile. Furthermore, the device profile alternative I/O devices (switches, Braille-lines, etc.) and assistive technology software (speech-recognition software, on-screen keyboard, etc.) need to be able to be included.

CC/PP foresees the possibility to integrate in a single vocabulary user and device profiles. Although this approach is feasible, it was opted to separate software/hardware components from user preferences, as the user could access the same service with different devices. This approach will help future implementations based upon Web Services, the introduction of multimodal interfaces, and the standardization of profiling information.

The selected approach is based upon four of the sections of UAProf. The specific sections WAP- and PushCharacteristics are eliminated, as from our point of view they can be integrated within the relevant network characteristics. The novelty lies in extending the basic Hardware- and SoftwarePlatform to emphasize the interaction aspect by adding specific input and output components. These components can then contain information relevant to assistive technologies: switch, head-mouse, Braille-line, on-screen keyboards, etc.

- HardwarePlatform. Hardware description of the platform where the user agent runs. It includes information on CPU type and speed, memory size, network and modem capabilities, Bluetooth and wireless functionalities, etc. Standard and special input devices can be described here within the components InputDevices and OutputDevices.

- SoftwarePlatform. Generic information about the operating system run by the device. We add here two new components: InputSoftware (e.g. speech-recognition, on-screen keyboard, etc.) and OutputSoftware (e.g. screen-reader).

- UserAgent. Information about the browser developer, markup, styling and scripting languages supported, and its MIME-type rendering capabilities.

- NetworkCharacteristics. Global information about the network to access the Internet application: bandwidth, proxies and firewalls, etc.

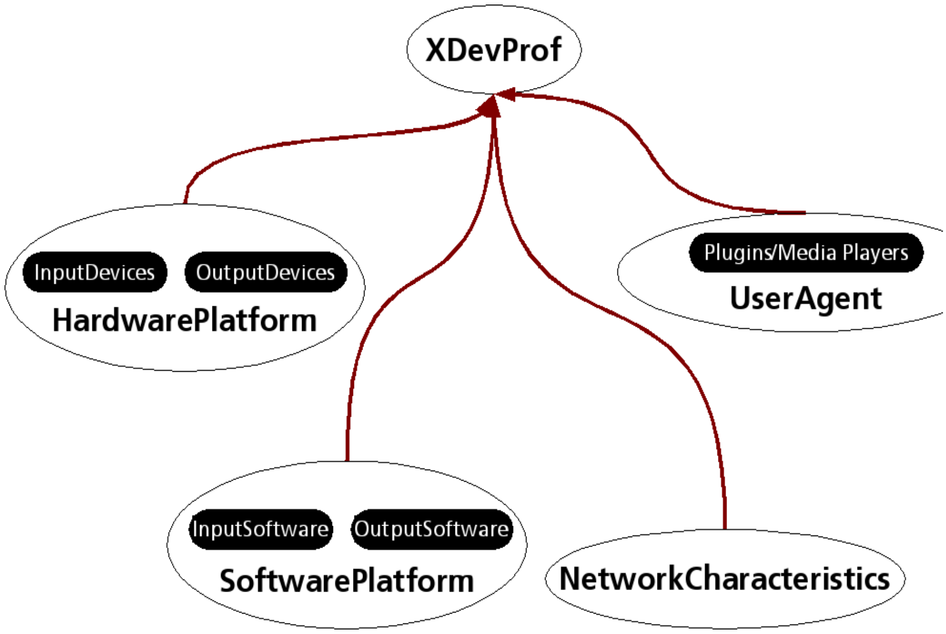


Illustration 48: eXtended Device Profile outline (XdevProf) ([Velasco et al, 2003] working version)

7.1.2.3 The User profile: the blending approach

As mentioned earlier, a user profile based on CC/PP has been defined. The design of the profile considers the following constrains:

- Compatibility with traditional profiles that store personal and demographic data, as well as typical user-management information (username, password, etc.).

- Ability to store information related to the application's functionality.

- Capability to store information about the context delivery (context of use).

- Facility to translate user profile components to device profile components and application abstraction components. It is a key issue that the user profile can override or modify device characteristics. This is what we called a blending process to express priorities [Gilman, 1997].

A generic vocabulary able to consider not only the relation user/device profile, but to include relationships with abstract application models, was beyond our resources. As a test framework, not exempt of commercially appealing value, we focused on designing a user profile that can be matched directly to the device profile, plus some additional components that are equivalent to some parameters typical of Internet applications. The defined components are:

- **PersonalInfo**. Contact and user-management information, plus additional demographical data of interest.
- **InputPrefs**. User preferences in regard to input modes. It includes a parameterized list of modes to be matched with the device profile and the delivery context.
- **OutputPrefs**. User preferences in regard to output modes. It includes a parameterized list of modes to be matched with the device profile and the delivery context.
- **InteractionPrefs**. User preferences in regard to navigation, search mechanisms and information highlighting.
- **DeliveryContext**. It includes location and time awareness. It might include biometrical data.

It is important to highlight that both profiles contain sensitive information in regard to the personal situation of the user and her functional impairments. That could potentially affect user's acceptance for the system. Security and privacy issues shall be addressed.

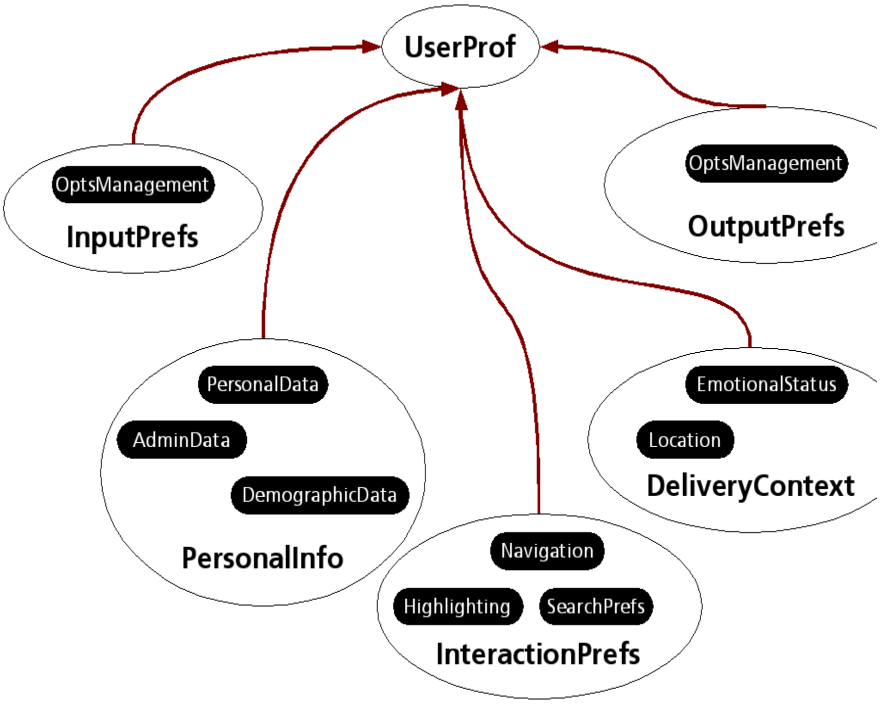


Illustration 49: IRIS user profile (UserProf) main components([Velasco et al, 2003] working version)

As part of the delivery context, the user's emotional status can be considered ([Picard, 2000]; [Mohamad, 2002]). This information can certainly be used to adapt and improve the human-computer communication. However, tracking of emotions is complex, and the authors are investigating the use of biofeedback sensors for this task[72].

7.1.3 Blending user and device profiles in IRIS

The IRIS project has as the goal of assembling various strands of work regarding Internet-based Systems and Services, their usability and accessibility aspects, and to make this knowledge useful and accessible to designers of such services and systems. One of the ways the project is trying to achieve this goal is with the creation of a design support environment, the IRIS DSE. The project aims to achieve this by combining user and device profiles, to ensure that the presentation of the environment is suitable for the designer, whether she has a disability or not. Such an implementation is based on two elements:

72 IPCA (Intelligent Physiological Navigation and Control of web-based Applications) project (IST-2001-37370). http://www.ipca.info/

- client-side proxy that compiles information about the client device and, by interacting with the user, elaborates a user profile; and
- server-side processing of information to render adapted content based upon the aforementioned profiles.

Illustration 50 presents the approach as an iterative process of translating user-related technical characteristics to the communication channels that the application needs to conform to.

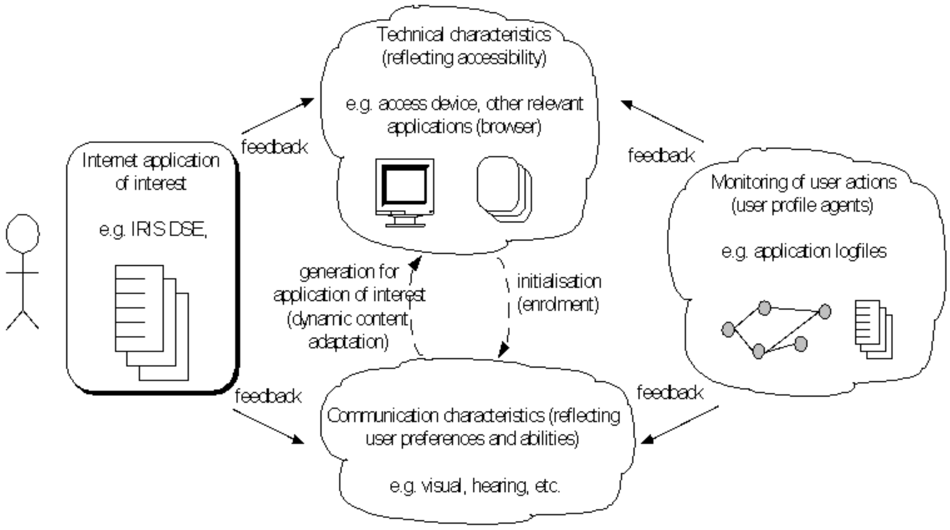


Illustration 50: IRIS approach for adaptation.[IRISD0602int]

The acquisition and generation of profiles is related to the user enrollment process. Typically the enrollment process has two steps, each of them related to acquiring information related to the access device and user-related preferences and characteristics. User enrollment starts as soon as the user connects to the IRIS DSE application. Parts of the device profile are automatically retrieved using the developed CC/PP proxy, located at the client side. A screen shot of the prototype implementation of this step of the enrollment process is shown in Illustration 51.

The user profile is generated:

- explicitly by the user ('manually', with system help);
- by the application in interaction with the user, leading to the inheritance of a predefined stereotype.

It is important to note that the initial profile proposed to the user must start within the feasible region, to allow her the manipulation and fine-tuning of the parameters [Velleman, 2000].

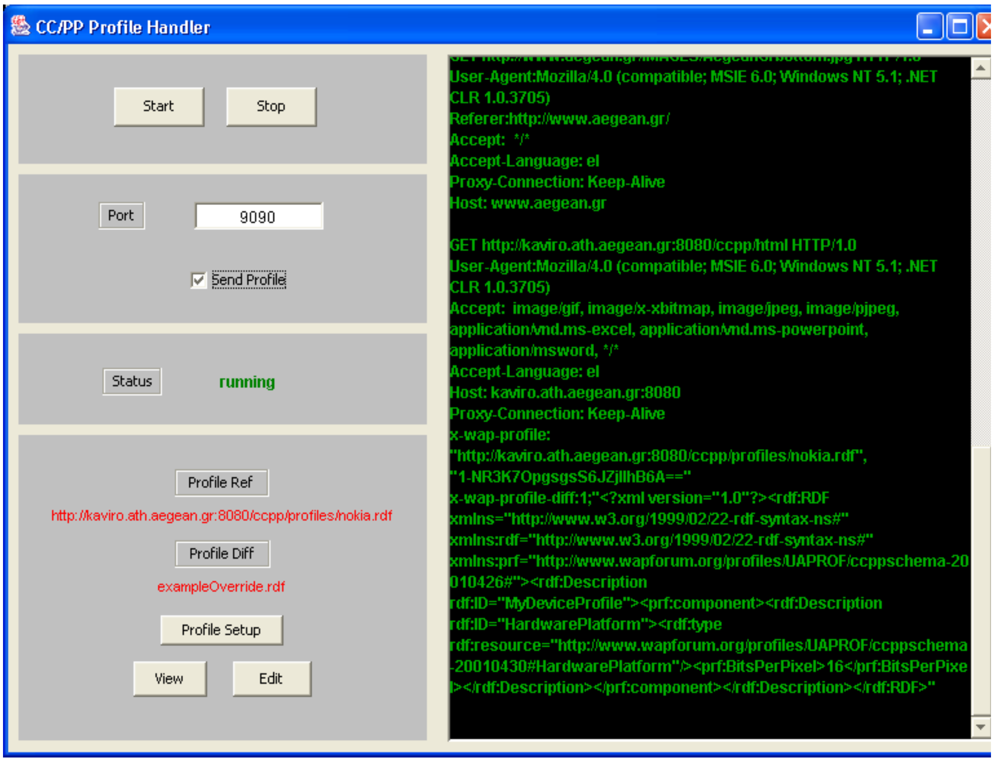


Illustration 51: Prototype implementation of acknowledgment of access device characteristics and preferences during the IRIS DSE enrollment process [Velasco et al, 2003]

For that purpose, stereotypes describing certain user categories are created. Each stereotype describes a certain user profile category, in terms of the characteristics of the respective communication channels that fits better to her, while in parallel, it associates these specific user characteristics with the respective representational parameters of the content: formatting, scaling and timing parameters.

Within IRIS, stereotypes are implemented by sets of templates and stylesheets. The stereotype has to be selected on the basis of the system knowledge about users' preferences. This knowledge is represented in the form of rules that map generic profile characteristics to presentation vehicles. A formal example of these rules could be as follows:

If {(X є [x1, x2]) and (Y є [y1, y2]) and (Z є [z1, z2]) and (…)} then suggest stereotypeN where X, Y and Z are user or device characteristics; x_i, y_i and z_i are different range values or parameters; and stereotypeN is one of the basic stereotypes.

The mapping mechanism is complex, and the decision-making process needs to be validated via user testing. The system must be able to provide a sensible initial presentation from the N set of available stereotypes (i.e. template/stylesheet combinations). User-defined fine-tuning can be implemented, and some further automatic adaptivity of the system can be explored on the basis of user interaction with the system, by inheritance of one of the aforementioned stereotypes. This can be implemented by the usage of accumulated statistical information regarding the profile-type/template-index pair. The actual agent based architecture proposed and partially implemented is shown in Illustration 52.

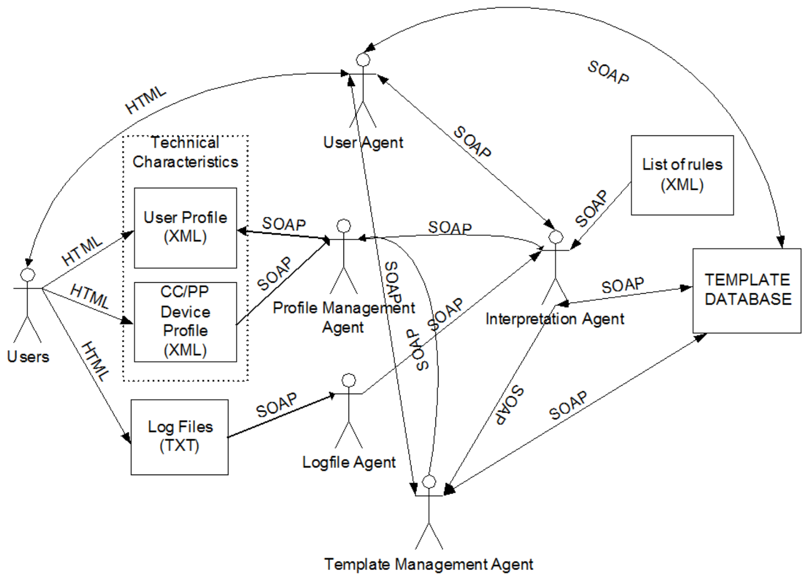


Illustration 52: IRIS agents [IRISD0602int]

This mapping must avoid falling in typical misconceptions about how users interact with the Internet. For example, it might be argued that a user that scans web pages with a "peephole" viewer, such as refreshable Braille display, will not care about font-sizes. However, the case is that many users of these systems set the font-size to a minimum to get the maximum content into a static frame over which they move the Braille line inspection window. In that way, they minimize the number of times they have to redefine the background coordinates with regard to which they move the viewing window. Another typical use case for mapping of user characteristics has been offered in the Introduction section, where it is shown that users with visual impairments are better served by layouts used for PDAs or mobile phones.

7.2 The BenToWeb Case

This section presents the relating work taken place under the umbrella of BenToWeb project that offered tools based on state of the art research and technologies for the web accessibility.

BenToWeb[73] is a project within the Web Accessibility Benchmarking (WAB) Cluster[74] aimed to support the European public and private sector to implement the recommendations of the eEurope 2005 Action Plan by providing new software modules and methodologies that satisfy some of the accessibility recommendations of the Web Accessibility Initiative (WAI) of the World Wide Web Consortium (W3C), which are not analyzed by existing tools due to their inherent complexity.

BenToWeb supports the objectives of the Cluster in regard to the creation of a validated methodology to test Web sites. Furthermore, the project:

- Supported the Web Accessibility Initiative (WAI) to develop the next generation of the Evaluation and Repair Language (EARL) under the umbrella of the Evaluation and Repair Tools Working Group, as well as several related activities targeted to the production of resources to combine reports from different Evaluation and Repair Tools.
- Supported the relevant WAI Working Groups in the development of complementary documents of the second generation of the Web Content Accessibility Guidelines, such as technology-specific techniques documents and accessibility test-suites for key W3C recommendations.
- Investigated further the feasibility of automatic testing procedures that include issues like color-contrast, low-vision, color-deficiency and consistency of navigation elements, and develop implementation modules for user testing.

73 *http://www.bentoweb.org*

74 *WAB Cluster: http://www.wabcluster.org/*

- Developed new testing modules based upon modern language technology from Computational Linguistics, to control the syntactic-semantic properties of documents as required by accessibility guidelines.

Even if BenToWeb involves work focused on tools for web accessibility several research results could prove useful in the domain of adaptive web systems.

During BenToWeb, appeared the requirement for remote evaluation of test cases. The purpose was to develop test cases and have them evaluated by appropriate users based on their user profiles focusing on characteristics related to web accessibility. This involved the design and development of the following software advances:

TCDL: Test Case Description Language (a use is presented as appendix) serves two purposes. First, it allows test suite developers to save the metadata that are necessary for testing accessibility evaluation tools. Each test cases maps to a specific "rule", for example a WCAG 2.0 success criterion or WCAG 1.0 checkpoint, and either passes or fails that rule at specific locations in the code. These metadata can then be compared to the output of an accessibility evaluation tool to check if the tool covers the "rules" defined in the test cases, and to check for false positives and false negatives. Second, TCDL supports the definition of test scenarios, when needed, to validate the test cases during the development process. Each test case needs to be reviewed for obvious quality assurance reasons, but it is also possible to define scenarios for end-user testing. [Strobbe et al, 2006].

Parsifal : Parsifal is a graphical Test Case editor for editing user test descriptions as defined by TCDL specification. Test case description files are written in XML. Since work with XML documents in text editors is not very comfortable and error-prone a graphical editor was implemented to ease editing XML test case description files [Herramhof et al, 2006].

Amfortas: Amfortas is a test case evaluation framework for test suites. Due to its importance and relation to this thesis this is discussed in detail in the next sections.

EARL: Between others, BenToWeb was responsible to support the specification process for the Evaluation and Report Language (EARL) under development by ERT working group of W3C WAI.

The Evaluation and Report Language (EARL) is a format to express test results. Test results include bug reports, test suite evaluations, and conformance claims. The test subject might be a Web site, an authoring tool, a user agent or some other entity. Thus, EARL is flexible. It enables any person, entity, or organization to state test results for anything tested against any set of criteria [Abou-Zahra, 2007]. The basic components of EARL are: who (or which tool) runs a test, the resource tested, the result(s) of the test and the tested criterion(-a).

The conclusions of the design, development and user evaluation of the above mentioned tools seem to be reusable to the domain of adaptive web systems. A part of TCDL can be considered as a mechanism for explicitly describing the requirements of a certain digital resource. In particular, scenario elements can be used to specify the requirements of the resource from a disability point of view. For making TCDL usable in the domain of adaptive systems the interesting part (at least) need to be transformed to RDF format and probably extend this by incorporating other vocabularies like CC/PP. Finally, like Amfortas, a mechanism is required for inferencing and for adapting a resource to a user. In this way a dynamic quality assurance mechanism is designed. The above mentioned proposal could either apply to the narrower domain of web accessibility or to the wider one of web adaptive systems.

7.2.1 Amfortas – A Test Case Evaluation Framework

Amfortas is a Java-based Web application for remotely evaluating accessibility of test suites. It covers the whole management process from creating and handling user and testing profiles, to storing the test results in a database (Illustration 53). Usually user evaluation processes are monitored by an expert, who in a first step presents a task and then gathers information about the process and the result by asking relevant questions. Obviously, this makes testing fairly expensive.

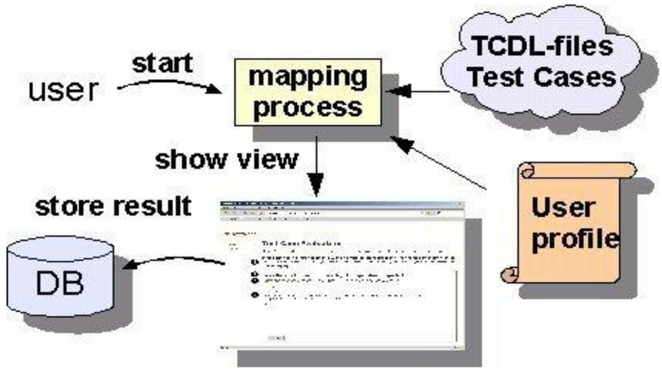


Illustration 53: Amfortas life cycle ([BenToWeb_Delv3.7a]

The evaluation framework should ease user testing procedures in a way that evaluators can proceed with the tests by themselves without any human support. Amfortas aims to manage in an effective way test cases for different web technologies, expert users and end users so that an appropriate feedback would be extracted. User profiling mechanisms are involved in combination with matching algorithms so that every user is allocated appropriate test and a quality feedback is insured. ([Herramhof et al, 2006];[BenToWeb_Delv3.7a])

7.2.1.1 Process Work-flow

The evaluation task starts with a recruitment procedure. The recruit is guided through a series of questions in order to gather information about his personal constitution (disability, age range, internet experience) and technical equipment (assistive technology, browser, device used to access web content). The answers of the recruitment procedure determine one testing profile. The evaluator should be able to set-up additional profiles, if for some reason the equipment changes or he uses more than one set of equipment to access web content.

The administrator can view the status and profile of the registered participants. All participants with adequate profile will be granted access to the evaluation framework by activating their accounts. The users then are able to access the log-in area of the web-application, but cant start testing unless the test profile is admitted to a particular test suite.

The testing process starts by activating the corresponding link in the web application. The mapping algorithm first looks up the database, selects profile information for the actual user and tries to match it with the TCDL description files. The matches are stored in a pooling table, in blocks of 20 test cases (called test run). Testing can be repeated as long as there are matching test cases available. The evaluators are expected to have at least moderate English skills. As all test cases are in English, the log-in area of the web-application is also kept in English.

The framework guides the evaluator through all the allocated test cases. A test case is finished when the user answers the question about the test case. The answer is stored in the database. After evaluating the test suite the data is extracted from the database for later analysis.

7.2.1.2 System Architecture

Amfortas is built on top of the XML publishing framework Cocoon2 . The core of Cocoon's object-oriented architecture is based on the Apache Avalon project. The overall architectural view of Amfortas consists of three components: a Java web server containing the application, a MySQL database and a resource containing the test files and test description files. Usually, the files are provided via a web interface, but any other providing mechanism, e.g. CVS, would also be appropriate.

Due to Cocoon's internal architecture, the evaluation framework is composed out of 3rd party components; own components, Javascript files, XML files and certain additional resources.

Database Layer: Instead of creating a custom persistency layer, we decided to use Hibernate3. Hibernate not only provides a powerful and easy-to-use object relational bridge for Java applications, but also offers a rich query language to retrieve objects from the database.

The evaluation framework uses 41 tables to store persistent data. Persistent data is data needed to build up the application view, data to accomplish the mapping procedure, data to conduct application management procedures and data which composes the evaluation result. The database model is straightforward: it is actually a normalized view on the users personal condition and technical equipment. Amfortas stores the mapping-related data for assistive technologies, user agents, devices and disability in different tables, which in the end are consolidated in the table test profiles. One entity set in test profile determines one test profile.

Presentation Layer: The initial version of the user evaluation framework presents a very simple and intuitive user interface, as it is going to be accessed by users with a huge variety of interaction requirements. For later extension of functionality a clear separation of content and presentation is needed.

Amfortas’ content has been completely authored in XML reusable entity documents. This process actually involves three sitemap components. If there is no need for aggregation, a Cocoon generator simply loads XML from the file system or web resource and generates SAX events which are handled by consecutive XSLT transformers and finally a serializer (e.g. HTML for Browsers). In most cases, an Cocoon aggregator is required which offers additional functionality by aggregating more than one XML files (e.g. Header, Content and Footer) inside a root element.

Amfortas’ public access area is implemented in the languages English and Dutch, as the potential evaluators are recruited in England and Belgium. Cocoon offers the i18n-Transformer component to implement internationalization features. Language-dependent text is stored in an XML file and referenced by the application through a unique key.

CForms -Forms are important for interaction but at the same time raise a lot of accessibility issues. This is mainly due to the need for direct and responsive interaction, which is usually implemented with client-side technologies -in most cases Javascript -which may cause serious accessibility barriers. These problems have been already addressed by W3C, which proposes the next generation of web forms named XForms. XForms seems ideal for Amfortas’ forms implementation, but it’s not applicable as most user agents have not yet implemented this technology. A good alternative that merits goods from both current and future world are Cocoon forms (CForms). CForms are XML forms that introduce the separation of the model (form model) and instance (form template) of the form that can be implemented separately. The so-called form widgets can be developed and include their own server-side validation. In Amfortas, the form instances are controlled with Cocoon flow. A further advantage of this approach is the ability to move to XForms by simply applying XSL transformation.

Application Logic: While ’action components’ have been the dominant method to encapsulate application logic in Cocoon, this position has been taken over by the Control Flow. A flow script is implemented in Javascript notation. A considerable part of application logic, like the recruitment process, the application administration or the saving routine of the evaluation results is implemented using flow scripts. Application logic which directly influences the view of web application is implemented using the Cocoon JX-Transformer. Higher-level logic, such as the mapping procedure, is implemented in Java classes.

Profile Mapping: Only test cases marked with status ’accepted for end user evaluation’ are pooled. Each user request for new test cases triggers the mapping algorithm. Mapping involves comparing TCDL disabilities and experience elements (i.e. user agents, assistive technologies and devices) with the user’s test profile stored in Amfortas’ database. The mapping algorithm first filters out the test cases that are ’done’ and those that the user has already evaluated. For a test case to be allocated, the following conditions needs to be satisfied. For disabilities, if in TCDL there is a disability, the test profile needs to have at least one of them. For user agents, assistive technologies and devices, the test profile must have all the types that appear in TCDL file. If the TCDL specifies a product as well, the profiles need to have at least one of the specified products for each type. Further, if minimum level and product version are also specified these need to be equal or less than the profile's one. Finally, for better matching, a complementary grading mechanism is involved that enables a better selection after sorting by grade and getting the required number designated by the test suite configuration.

Test Presentation: The test case evaluation is a cyclic process which can be invoked as long as test cases can be allocated for the actual evaluator. The whole test presentation is created within the Cocoon sitemap, by passing a number of parameters from the Cocoon flow, for example, the URI of the actual test case description file, the URI to the test file, and the scenario. The default sitemap generator is used to fetch the test description file from the web resource and page header, footer and navigation from the local file system. The standard cocoon aggregator bundles these XML trees to a single XML tree which is handed over to XSLT transformation and finally serialization. The result is an XHTML page with user guidance information, a link to the test file and a question with corresponding answer type (see Illustration 53).

7.2.1.3 Application View

Illustration 53 shows the user interface for the evaluation process. The ’user guidance’ section (1) requests the evaluator to adjust special settings or behave in a certain manner in order to complete the test. The question (2) is presented before the link to the test file (3), to give a first idea what to mention when evaluating the test file. Finally the answer section (4) presents one of the answer categories to be replied. On submit, the answer with references to the accomplished test is stored.

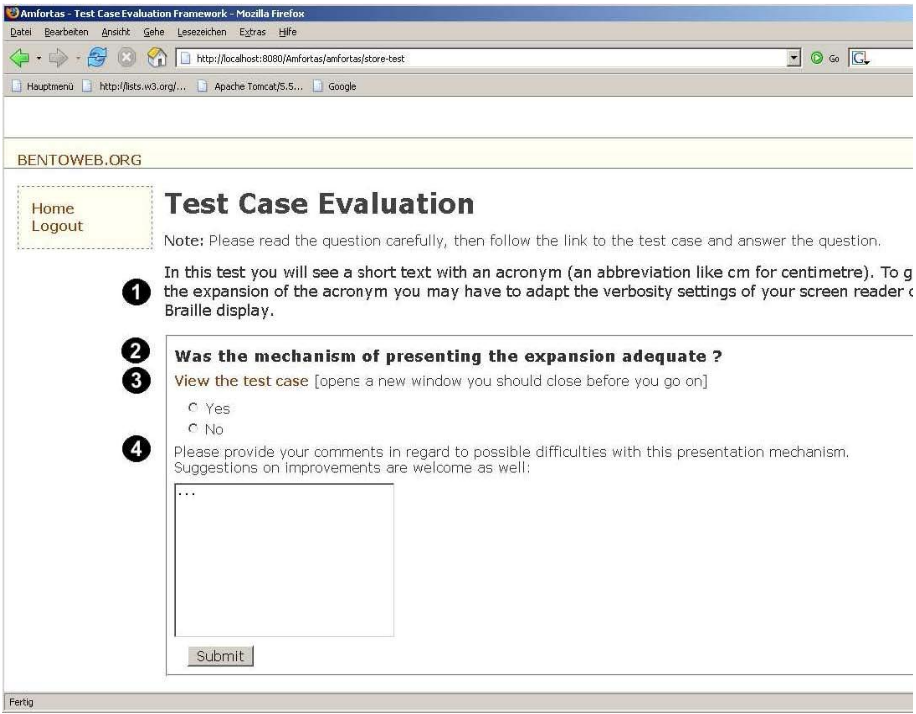


Illustration 54: Amfortas test case evaluation [BenToWeb_Delv3.7a]

7.3 Web Portals Cases

Web portal seems an appropriate application for doing a first empirical evaluation to figure out how the proposed framework could be applied to real world applications. Their appropriateness is referred to their primary requirement for personalization due to their wide and diverse audience with often different roles / profile behaviors. As mentioned in previous sections, portal paradigm and infrastructures can be used in different problem situations. Follows a number of projects that make use of portal paradigm and infrastructure:

- *EQUAL – SYMPOLITIA*[75] a portal infrastructure is being used as a virtual incubator for communication and learning between educators, consultants and interested parties, while in

- *EQUAL-EUNETYARD*[76] portal is used as the information and communication tool between the project's partners.

- *IRIS*: A portal has been used as the user interface module of a web-based design support environment for web accessibility design that was capable of being adapted to a disabled designer needs though mechanisms of user and device profiling.

- *e-University*: Design and development of a portal infrastructure aiming to be used by Greek universities for both internal and external services with emphasis to accessibility. The infrastructure has been developed based on Open Source (Apache Jetspeed2) and Open standards (XML-based technologies) allowing for interoperability.

From a DAWIS point of view, portals and portlets seem an interesting case study. Portlets can be seen as subsystems that could be self-referenced, distributed and able to reproduce and replace themselves by sensing the supersystem emerged attributes by semantically search services on the web and after evaluation reconstruct themselves and the whole system/portal. Thus, an implementation proposal would be an extension of portlets' specifications and WSRP so that they fulfill the conceptual model's requirements.

Follows an analysis the case study by introducing four level of abstraction as presented in Illustration 55.

75 http://simpolitia.syros.aegean.gr

76 http://www.eunetyard.net

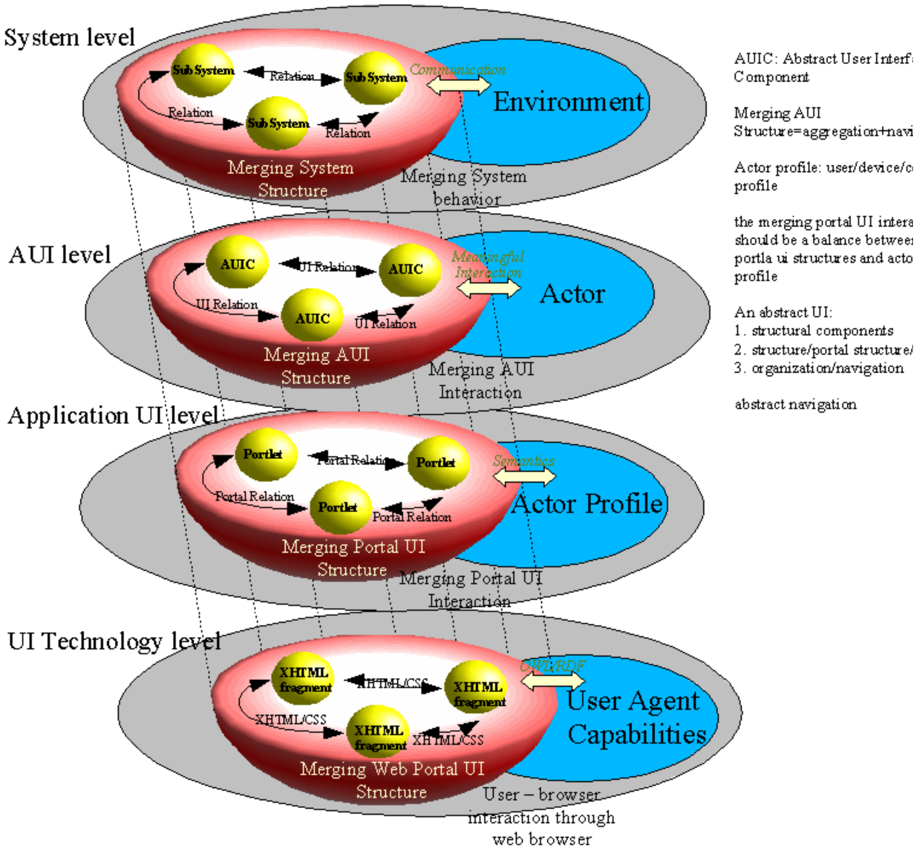


Illustration 55: DAWIS Portal/Portlet Use Case

System level: At this level there is a brief illustration of general systems' relevant points. This presents the subsystems that are the structural elements of system defining its structure. The internal structures, through communication, are triggered by the changes of the environment that defines the merging system behavior. A process of adaptation takes place until a balance between the system structures and the environment takes place.

AUI level: At this level, the idea of Abstract User Interface is applied; in other words, a way to represent a system's interface and its interaction in a technology and application independent way. In this context, the actor is any kind of system that composes the interaction environment of the user AUI.

Application UI level: At this level the focus is at the portal application, but as a "generic" application. This means that portals are proscribed as a type of application that is not limited only as browser-based one, but as a "virtual operating system". At this level, we talk about portlets by abstracting their definition. The sum of the portlets and their interrelationships (aggregation+navigation) composes the portal UI structure. The former combined with the actor profile in same sense, as upper levels, gives us the portal UI interaction. This interaction becomes more effective by blending semantics.

UI technology level: This is the well-known technological level that appears in browsers, which interaction depend on the web portal structures (fragments, navigation, aggregation) and the user agent (browser, actor etc) capabilities/preferences. Semantic web techniques such as RDF and OWL could be employed for the system to be more effective.

The portlets can then be self-evaluated and though semantically enriched web services optimize or even replace their selves. Note, that portlet specifications and reference implementations support the on-demand download and hot deploy of portlets. The last one is a fruit of standardization.

7.3.1 Accessibility of Portals

In the most remarkable work found on the field of portals' accessibility evaluation, recently, the accessibility of portals has been investigated [Gappa & Nordbrock, 2004]. That work was aiming "to gather information and issue suggestions and recommendations for the improvement of portal designs and services, as well as standard strategies for portal customization" using questionnaires, a standard task, log files and observational protocols applied to disabled users of diverse profiles. It concluded to accessibility and usability barriers on the design of web portals. This work concludes with its findings and suggestions for accessible portals.

At the same time, during the design, conception and development of the portal for the Government of Mauritius a document for raising accessibility issues was created that also raises some important issues.

Finally, certain documents on best practices ([Hepper & Lamb, 2004]; [Aiken & Sullivan, 2002];[Hepper et al, 2005]) for designing and developing portals contain or can be extracted portal's accessibility related issues that can be incorporated into the proposed framework.

A web page hypertext structure, compared with that of portals, can be much more easily accessible following the Web accessibility guidelines and techniques. On the other hand, simple web pages often do not offer adaptation according to the actor. A critical parameter of accessibility that becomes even more acute due to the portal's complexity is navigation. Navigation in the context of portals is multilevel as discussed in next sections. Additionally, the number of people needed to be involved for maintaining a portal increases, while often their computing background differs greatly. In addition as we will describe in a later section, the interaction takes place in multiple levels as well. Moreover, the customization functionality of portals adds a requirement for accessible page customization but at the same time offer capability to customize.

On top of these, keeping in mind that a portal can also contain fragments of more than one web technology and also that their designers could be completely unaware of the whole portal design (i.e. WSRP), controlling accessibility of a portal is a difficult and multifaceted task that currently follows a piecemeal approach that result to serious accessibility problem. It is more than having multiple accessible web pages. As Lausen H [Lausen et al, 2004] state that "*portals are a special breed of web offering a blend of information, applications and services. Thus, a portal's usability is more than the usability and design of its parts. It has also to care of more general issues like packaging, structuring, integrating and organizing information and knowledge provided to their user community*".[Lausen et al, 2004]

From aforementioned findings merges the requirement for a framework that would face portals' accessibility issues. It aims to respond to questions such as: 1) what makes portals a different design case in terms of accessibility? 2) How this case could be faced in terms of accessibility? 3) Are the existing guidelines and frameworks addressing those requirements? 4) What kind of abstract and technological solutions could be provided? 5) Could the design of the portal offer new ways / approaches towards accessible web? 6) How could all these be integrated to a framework for designing accessible portals?

Answers to such questions was attempted to be given through a whole/parts approach prism ([Vlachogiannis et al, 2005];[Vlachogiannis et al, 2007]). The whole/parts approach has been inspired from system's thinking basic principles. Systems thinking [Checkland, 1999] go beyond classical analytical methods and face problems in a non reductive way. The sum of the properties of subsystems / components does not define the properties of the whole system, but is something more. The whole system also has emerging properties that determine its behavior (as already discussed in the previous section).

Thus, in case of a web page, the whole page can be inaccessible even though its concrete component is fully accessible. And, this is basically the case with portals that aggregate diverse entities named portlets. So, for having accessible portals we should investigate both portlets accessibility but also portals as a whole.

Under this prism, this work proposes a whole/parts perspective that considers portal systems as a whole and the portlets as parts that need to have some attributes, behaviors and organization for accomplishing their purpose. If one adds the capabilities of the software and the device that the user makes use of to access the web (e.g. text browser, mobile phone, etc) also known as user agent by the W3C and also the context of use, which is an ultimate factor in intelligent environments, then the problem space might be expressed as:

Accessible Portal Interaction = accessible communicating of [accessible aggregation of (accessible portlets + accessible navigation)]

In other words, accessible portlets and navigation are essential but not enough for composing an accessible portal as the emerging properties would probably result to an inaccessible whole / portal. Thus, portlets and navigation should have such properties capable of sensing their "aggregated" effect on the portal as a whole. "Accessible communicating" of the aggregated content refers to personalization/adaptation features involving issues such as user/device modeling [Velasco et al, 2003] and location/context of use awareness. The following subsections will briefly discuss portals accessibility issues under such a prism.

7.3.1.1 Authoring management

Portal's design and maintenance complexity is analogous to the number of portlets and the number of content providers (authors). As in every task applies, organizing humans is much more difficult than organizing machines. At least this involves setting authoring specifications and principles and training authors both on the authoring tool and the portal's authoring principles. At this point a balance should be kept so that authors follow the specifications but at the same time do not loose anything from their authoring creativity.

It is neither very feasible to separate the information providers from the editor nor to educate them on accessible portal design. The best solution is possibly a "What You See What You Get" editor with controlled functionality, so that, for instance, an author cannot change the font style but can underline and automatically add needed portlet related markup as described above.

Of course this approach cannot satisfy the need for accessible content (e.g. for cognitive disabled people). The authors' managers need to train authors to use appropriate / simple language for their audience / target group.

Finally, a further step could involve the use of semantic annotations based on a portal's domain ontology [Reeve & Han, 2005] blended with actor profiles and the integration of these on the editor.

7.3.1.2 Portal Navigation

For working on the navigation aspect of accessible portal systems, a good test case is made by investigating the case of mobile portal systems [Godwin & Haenel, 2002]. Mobile devices can be a very good simulation application for the designers of web portals. This is because they simulate different access problems due to their limited screen size and input capabilities that many categories of people may meet and at the same time give the marketing push to the portal customers and consequently their vendors to design with accessibility in mind.

For such portals, navigation is the key to success. The users need to access the information they are looking for with not too much of cognitive navigation overhead. Artificial techniques have been used [Smyth & Cotter, 2003] for reducing click-distance and providing successful navigation. Similarly, generally speaking, in accessible web portals, users need to interact with portlets in a "portal transparent" way and also have at their disposal powerful and useful search engines: the more the information, the more the complexity of its organization and the navigation complexity.

Authors distinguish portal navigation in three kinds: 1) main portal navigation, 2) inter-portlet navigation and 3) intra-portlet navigation. Further we could distinguish two kinds of hyperlinks: 1) user interface links and 2) semantic links. Semantic links are links that can be used for a kind of navigation but semantically. That is, in a document that talking about a subject need to refer to a subject relating another portal page instance. On the other hand user interface links are considered as the repeatable links that are provided in a toolbar paradigm and adds overhead to the actual content.

In general, for a portal page instance to be accessible, it is important to make sure that when the page is serialized by an actor it will produce an acceptable result. By that, we mean that the windowed (portlets windows) version needs to be effectively transformed to good structural user interface. In other words, portal systems need a mechanism to semantically communicate the portal page structure to the user agents. Consider for instance PSML or an automatically generated portlet navigation or even a separation of navigation concern using semantic web technologies.

In other words, navigation should be metadata and not data. Thus a separation of the control of a web resource from the resource itself is needed. This will allow the actors to semantically extract the navigation information and use it to guide themselves in the web resource. The metadata can now be more that simple hyperlinks. It can also contain more information about the structure and the content of the web resource (i.e. pages descriptions or pages relations). Furthermore, this approach could offer much towards the semantic web. The navigation would then be much more easily being adapted both to the actor, and to the purpose of the content (e.g. learning). Consider for instance the scenarios that aim to be personalized navigation guides. At the same time the proposed approach offers to the actors a means of global navigation by allowing them build their personalized navigation according to their cognitive and presentation requirements / interaction capabilities.

The proposed abstract navigation language would allow to fill such a gap and provide more opportunities. For example, one might consider a portal that can be accessed via a radio set while driving, tuned through the navigation and then making use of screen reader for listening the content or even animating on a screen. Present-day adaptive navigation techniques [Brusilovsky, 2004] could be used such as link hiding or link generation. This might allow “virtual portals” (aggregation of web resources in a form of navigation from a set of web resources sites) and much more. A small view of this could be seen through RSS , which can automatically syndicate information from resources of interest and thereby create a new resource.

7.3.1.3 Portal Aggregation and interaction layers

Portal systems consist of portlet applications that can provide completely different functionalities and serve completely different aims. Portlet applications consist of a number of portlets that have a common aim but at the same time they are reusable and autonomous components. Portlet applications can even be situated on different servers or different web domains WSRP and consist of fragments of third party resources.

In simple terms, aggregation is actually markup that creates the “windows” for the portlets and puts them in page. This markup, following web accessibility guidelines, should contain neither tables or even worse nested tables nor frames for layout (a common practice). That layout needs to be undertaken by CSS. Recently there is such a tendency for instance in Jetspeed-2 layout decorators.

The aggregation of the content on an interactive environment such a portal involves an “aggregated interaction” as well. We can distinguish page interaction and portlet interaction while the first one refers to the interaction that happens on portal as a whole page and the last one that happens on a specific portlet without affecting the rest of the portal. This difference need to be taken into account towards a more accessible interaction. This can be seen both on a client side and on a server side manner. For instance a client script might cause change of behavior of another portlet and consequently to the resultant portal page instance. Similarly, in case of server side this is expressed with get and post requests of the portlets that can for instance contradict. Finally, the AJAX upcoming technology is an interesting case that its accessibility is under investigation.

7.3.1.4 Portlets

Portlet interfaces may consist of hypertext and/or multimedia content. To be accessible, these need to follow web accessibility guidelines, just as for web pages. Here a distinction between portlets and web pages needs to be made because of the former’s “page fragment” nature.

Already, above mentioned portal specifications define portlets' attributes that affect portal as a whole. An example is the CSS classes including portlets' title and more. In the context of intelligent and distributed environments context-aware attributes should be introduced for extending portlets capabilities and as result portals' services and information accessibility.

7.3.1.5 Portal Accessibility Guidelines as Extensions (PAGE)

Following the proposed approach, related literature conclusions, portlet specifications JSR-168, WSRP specifications, best practices (Hepper & Lamb, 2004) and authors experience on portals and accessibility (IRIS, BenToWeb, EQUAL - EUNETYARD, EQUAL - SIMPOLITIA) an indicative work is extracted aiming to provide principles and guidelines for designing and developing accessible portals. It should be mentioned that this attempt does not claim to be an exhaustive investigation of the requirements, but aims to show up the need for such work by an organization like the W3C WAI and thus tries to provide a roadmap and an initial input.

Web Content Portal Accessibility Guidelines as Extentsions (WC-PAGE)

The WC-PAGE organization aims to be complementary to WCAG 2.0. Further general techniques and specific technology techniques can be provided. Thus the concluding guidelines/success criteria are:

Guideline 1.3

- Follow a serializable multi-layered layout design approach
- Provide a clear and simple design with identifiable headers and distinctive paragraphs
- Use portlet style classes according to JSR-168 and WSRP instead of markup formatting attributes.
- Use color coding for highlighting and differentiating between different type of information but also provide markup differentiation for vision impaired [Level 2]

Guideline 2.1

- Make portlet controls operable through a keyboard interface. Do it in a non obstructive way.

- Make customization link operable through a keyboard interface.
- Make site map link operable through a keyboard interface.
- Make main portal navigation elements operable through a keyboard interface.
- Personalization and customization actions should be operable via keyboard interface.

Guideline 2.2

- Portlets must not refresh the portal page without the confirmation of the user.

Guideline 2.3

- Provide a site map
- Do not use more than 6 main navigation buttons on a screen
- Take into account the distance between navigation elements and/or make it customizable so that motor impairment users can access it
- Make sure every main information can be accessed on most 2 layers deep
- Use a sophisticated search engine and add the search input form on top center
- When a link is about to open a new browser, window must warn the user for that and how she can go back to the working portlet (closing window). Also provide visual cues (small icon) and adequate title attribute
- Provide inter-portlet navigation on the top of each page
- Use the link element of HTML for “portal main navigation” so that it can be extracted by user agents
- When it is avoidable the need for screen scroll, provide float navigation
- Provide mechanism that user remains on her working portlet even if paged is reloaded by action

- Provide way to go to inter-portlet-navigation from every portlet

Guideline 2.5

- Provide ability to your search engine to suggest alternative keywords (i.e. from synonyms)

Guideline 3.1

- Keep small (4-6 lines) initial portlet information using representative keywords in text. Otherwise a summary of the content should be provided.
- Use clear and simple language
- Jargon should be explained in a glossary
- Avoid passive voice.
- Provide meaningful portlet and sections titles
- Provide an understandable way to offer customization (A lot of people do not know how to customize user agents' font size)
- When possible, use alternative modes in terms of modalities (i.e. multimedia) to offer further understandability
- Provide internationalization
- Provide metadata for important information to allow content adaptation

Guideline 3.2

- Make the link of the site map perceivable on top (preferable centered) of every the page
- Make the link of the customization perceivable on top (preferable centered) of every the page
- Make the placement of the portlet controls consistent

- Make the search results directly accessible to the user (avoid banners before irrelevant links and information before)
- Remain in portal environment and on working portlet.
  - Provide a kind of mark at the beginning of a portlet so that it can be used to directly be accessed
- When a link is about to open a new browser, window must warn the user for that and how she can go back to the working portlet.
- For all external links provide visual cues (small icon) and adequate title attribute

Guideline 4.1

- Global page information is not permitted
- Keep every portlet entity (instance) unique on a page.

Authoring Tool Portal Accessibility Guidelines as Extentsions (WC-PAGE)

Similarly an extension of ATAG 2.0 is provided:

Guideline 2:

- In case of authoring portlets do not provide ability to include global page information
- Incorporate portal styles according to JSR-168 and WSRP and promote their use when appropriate.
- Do not provide markup formatting functions. Instead, provide semantic markup.
- Allow for semantic annotation

7.3.2 *Accessibility of commerce Portals*

The design and development of e-commerce portals entails collaboration of people from business and marketing and from the information systems world. The first one sets the business requirements and the second one designs and develops. In our days, the shift from the systems centred design approach to user centred one, required the involvement of Human Computer Interaction (HCI) people aiming to better user satisfaction. Designing not for the average user but for all (design for all – Universal design) becomes a challenge. This becomes even more critical when “all” users are actually the potential customers. Thus, developing a successful e-commerce portal has been a multidisciplinary task that necessitates a supporting framework.

At the same time coping with e-commerce portals provides an opportunity for designing accessible portals as the former seem to be more “structured”. In other words the range of services that e-commerce portals offer is often narrower comparing with a generic information portal. This fact can be seen as an opportunity to develop a framework for designing and developing accessible e-commerce portals by building up a taxonomy of e-services and systematically investigate their accessibility requirements [Vlachogiannis et al, 2007 [a]].

Such an approach attempts to provide to the e-commerce designer the theoretical foundations and implementation guides for an effective design. A rich picture of such a problem space is developed by discussing portal notion in different domains. This can be summarised into the following premises:

- A portal is an appropriate pattern for e-commerce as it offers add-on services that has huge impact on customer decision
- Accessibility of e-commerce portal is an important factor for its success; thus the designer should follow a universal design approach.
- Portals have different characteristics compared to web pages, thus applying accessible design to portals has complementary requirements. General portals accessibility is difficult to face due to wide range of services, which cannot be easily modeled /abstracted.

- E-commerce universal design requirements cohere with m-commerce design requirements;

- Classifying e-commerce portal services provides an opportunity to systematically investigate services accessibility requirements.

- The fundamental requirement for an accessible portal is the user's ability to fulfil her goal of consuming the core service(s).

Through such remarks, the design of e-commerce substances could be faced from an alternative / integrative viewpoint illustrated in Illustration 56.

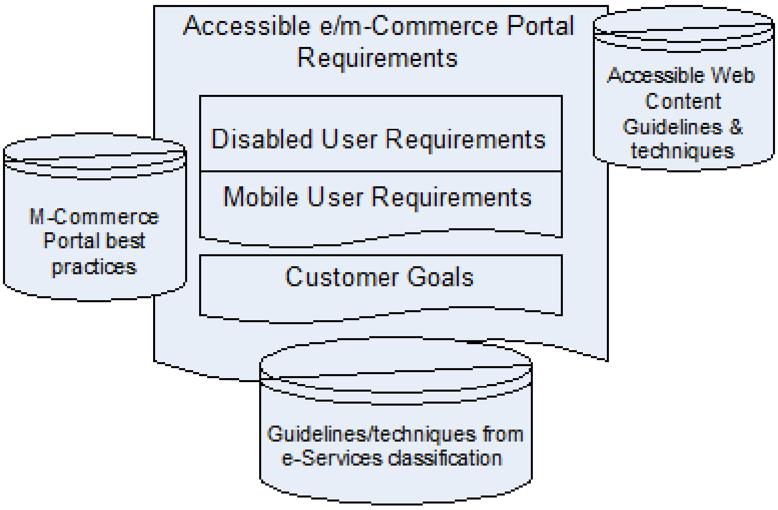


Illustration 56: Perspective of Integrative approach

According to this approach, e-commerce portal accessibility is considered as a layer above a general accessible portal. In this layer the services offered are the entities that are under investigation. These could be classified according to the adopted conceptual model of e-commerce services illustrated in Illustration 57. The resulting abstract services' accessibility requirements should be determined and guidelines and techniques facing such issues should be developed and associated with them.

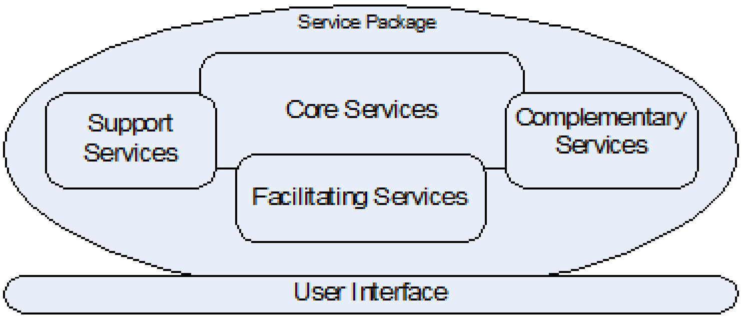


Illustration 57: Portal e-services classification

An e-commerce portal designer should have three major sources of information for making accessible e-services and also her portal accessible:

- Accessible web content guidelines and techniques
- Generic portals accessibility guidelines
- m-commerce techniques and best practices

During the design and development process of e-commerce portal, due to limited resources, it is often useful to prioritize design requirements. Following the priority paradigm of WAI WCAG this paper proposes a prioritization of accessible e-services based on their returning satisfaction from customers. So, for instance, a search service might be of higher priority than a “news” one.

Finally it is important to set up the basic principles that such an approach should obey. From the authors’ viewpoint such an approach:

1. Should be abstract enough to be able to include future requirements and technologies but at the same time specific enough to be applicable (a multi-layered approach).
2. Should be written for a diverse audience.
3. Should clearly identify who the stakeholders are and how they benefit from it.

Using the above mentioned sources of information though the proposed conceptual model by obeying the aforementioned principles, “e-service oriented” guidelines and accompanied implementation techniques could be developed aiming at offering a more successful e-commerce portal design.

7.4 The Interactive Web Television Case

A challenging application for adaptive web information systems research has recently been the interactive television. More and more this field adopts techniques and technologies initially developed for the World Wide Web. This is more apparent in the case of IP-TV, but this generally applies to all kinds of interactive TV. If somebody considers also that the number of TV sets is really considerable bigger than the number of personal computers into our world it is obvious that the interaction requirements and specifically the need for adaptation is crucial. For instance, an iTV user now is in front of a huge number of services (term used for TV channels) with amazing possibilities. A similar “explosion” has happened in the past in the world wide web and search engines and then it were the portals (with search engines) and the adaptation mechanisms that made the huge information manageable. Carmichael et al [Carmichael et al, 2006] discover similarities between the directions of interactive television with that of web and further note that the gained experience from the later has to be transferred to the domain of interactive television to avoid similar mistakes. However it seems that till now these have not been avoided. Even since 1997, RNIB has provided recommendations for the accessibility of interactive television [Darby, 1997]. Carmichael et al conclude that the accessibility characteristics that has not yet been given necessary emphasis are subtitles, captions and audio description.

In literature there have been identified several attempts to incorporate accessibility issues into the MPEG-21. It comes out that most of them are focused into visual disabilities. Rice [Rice, 2004] presents the difficulties that visually disabled users face while they consume interactive television services. This work gives emphasis into parameters like screen size, font size and color, icons' identification and screen layout. The conclusion of this work is that the best facing approach of the problem situation is personalization due to the diverging requirements. Yang et al [Yang et al, 2004] proposes a technique for the accessibility of interactive television for people with visually deficiency, especially color blindness. This technique consists of both the incorporation of MPEG-

21 with relating descriptive metadata and the design of an adaptive system. Berglund & Johansson [Berglund & Johansson, 2004] study the benefits of the usage of speech – dialog in the domain of interactive television. The evaluation of such a system concludes to several design considerations.

In this chapter the case study is a possible architecture of iTV through the DAWIS prism. The use case is based on an in-progress Greek national project that aims at using MPEG-21 infrastructure for adapting interactive TV's content to disabled children.

7.4.1 MPEG-21 role in terms of DAWIS

MPEG-21 is, between others, an attempt to give to the iTV designer a framework that can provide a big – integrative picture of an iTV system. Based on that, an indicative scenario has been developed, including production and consumption of the digital content, and aiming at identifying the primary entities and the way these are involved in the overall design outcome (see Illustration 58). According to that:

- The content designer (CD) identifies the target groups
- CD, supported by MPEG-21 metadata, describes the target groups using their characteristics (e.g. blindness) and associate interaction mode (e.g. aural description) using an appropriate authoring tool.
- CD develops the required content components (digital items) based on the above-decided interaction modes. These are integrated into the metadata using the authoring tool.
- End user A, say blind, wants to consume developed content. She has already stored her profile. The context of use is accomplished with access device capabilities, audio configuration, time and location of the end user.
- The context of use is delivered to the serving system accompanied by the user request.
- The system inferences and maps the user's context of use with an appropriate composition of the components of the content. If, while consuming, the context of use is being modified, the system need to be aware so that it can adapt to new requirements.

- Except for the content, the navigation of the content and also the navigation of device should be adaptive.

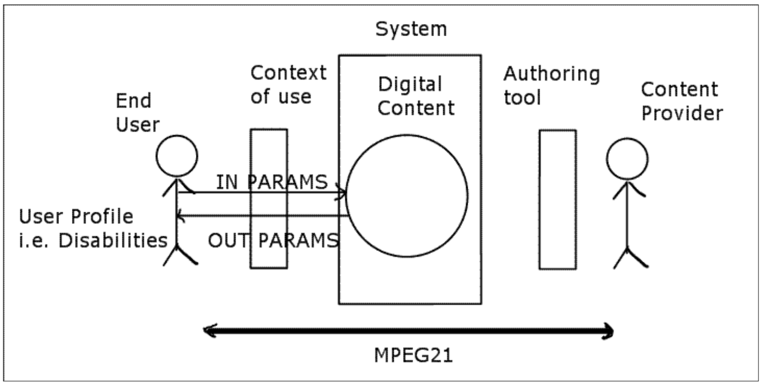


Illustration 58: MPEG-21 Involvement. A possible scenario

Even if MPEG-21 (as presented before) contains considerations for adaptation and specifically accessibility, it seems that on its own this cannot ensure the accessibility of delivered content. Instead, this is a fundamental presupposition for allowing accessibility output of the involving systems. In other words, it should be able to provide the required infrastructure so that a digital content would be able to obtain the requisite variety for both the user to be able to design accessible content and the involved systems to have the required information to deliver an accessible result (Illustration 59). Briefly, MPEG-21 seems to be capable of contributing to the accessibility of digital content though:

- *Alternative content:* MPEG-21 offers metadata that enables the content provider to provide the content in one or more alternative ways. The ways often refer to different modalities and thus there can included captions, aural descriptions etcetera.

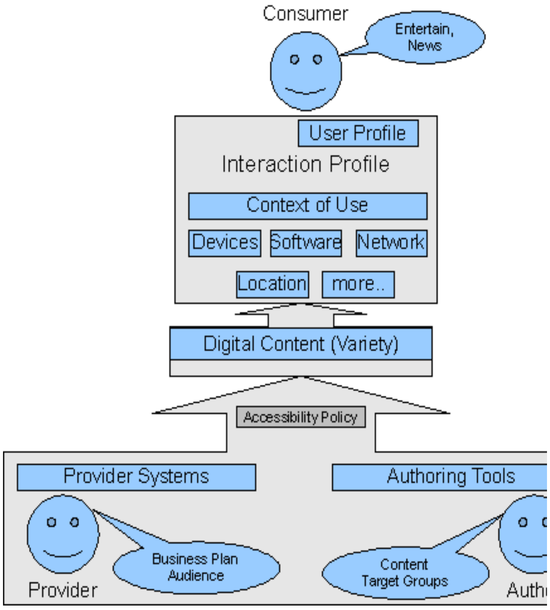


Illustration 59: Multimedia Delivery Stakeholders

- *Digital Content Navigation*: In the case of interactive television, the so called Electronic Program Guide (EPG) is responsible for the navigation through the provided content. This is actually the interactive portion of the system that offers the required functionality to the user including service (channel) selection / retrieval, informing about program and scheduling, profiling / personalizing, rating and/or even acting on the content.

- *Description of context of use (IN PARAMS):* The context of use actually refers to all the information required to take into account in order to adapt the digital content to the user's requirements. In terms of DAWIS, we are talking about the interaction profile. Appendix II provides a prototype implementation of the interaction profile in the corpus of the project.

- *Description of presentation parameters of digital content (OUT PARAMS):* This responds to the “adapt what” DAWIS question. Table 8 is a kind of possible specialization of some of DAWIS “adapt to what” aspects in the domain of iTV are presented to demonstrate the usefulness of the framework. Note here that an important implementation consideration was the transformation of MPEG-21 to SMIL as an intermediate solution to ensure media players compatibility. This involves the mapping between those two infrastructure implemented by XSLT.

Content	Service	Presentation	Interface
Generate – Retrieve / remove a movie from a virtual channel, Subtitles / audio descriptions	Enable / disable its functionalities: Electronic program guide	Enlarging page content— magnifying and enlarging specific captions or screens	Adapt metaphor to TV remote control
Appeared / Disappeared : Switch between channels	Optimize its functionality	Enhancing text— changing colors, letter and line spacing, and text style	
Simplified / Composited: simplification of video	Reconfigure its functionality	Enlarging EPG controls ex. Enlarge back and fw for motor disabled	
Summarized / Expanded: summarization techniques		Remove unnecessary formatting	
Restructured: like a trailer		Dim fragments (makes text and controls less visible, instead of hiding)	
Translated		Remove fragments that can cause epilepsy	
Concluded (result of inference process): e.g. a child not over 13 cannot watch.		Linearize content and HTML tables	
Annotated / Referenced : ex. A star rating		image transforms	

Table 8: Adapt what: the case of iTV

- *Content provider accessibility policy*: Probably, an important contribution to the field of accessibility of MPEG-21 is the capability of applying and claiming for an accessibility policy. In other words, the content provider needs to be capable of applying some kind of accessibility policy based on the target consumer group and the former' s requirements for quality assurance. So, for instance, such a policy could assign that every digital content must be accompanied by subtitles of two languages (e.g. en, el) and every image with an alternative text could be between two and ten words. Applying such policies requires a mechanism for validating a digital content to a policy description and could be for instance implemented based on Schematron[77], an XML structure validation language for making assertions about the presence or absence of patterns in trees.

7.4.2 An architecture in terms of DAWIS

An empirical evaluation of the DAWIS is attempted through the development of architecture for an IP-TV scenario. The logical architecture is being presented in Illustration 60 and a preliminary technological architecture is presented in Illustration 61.

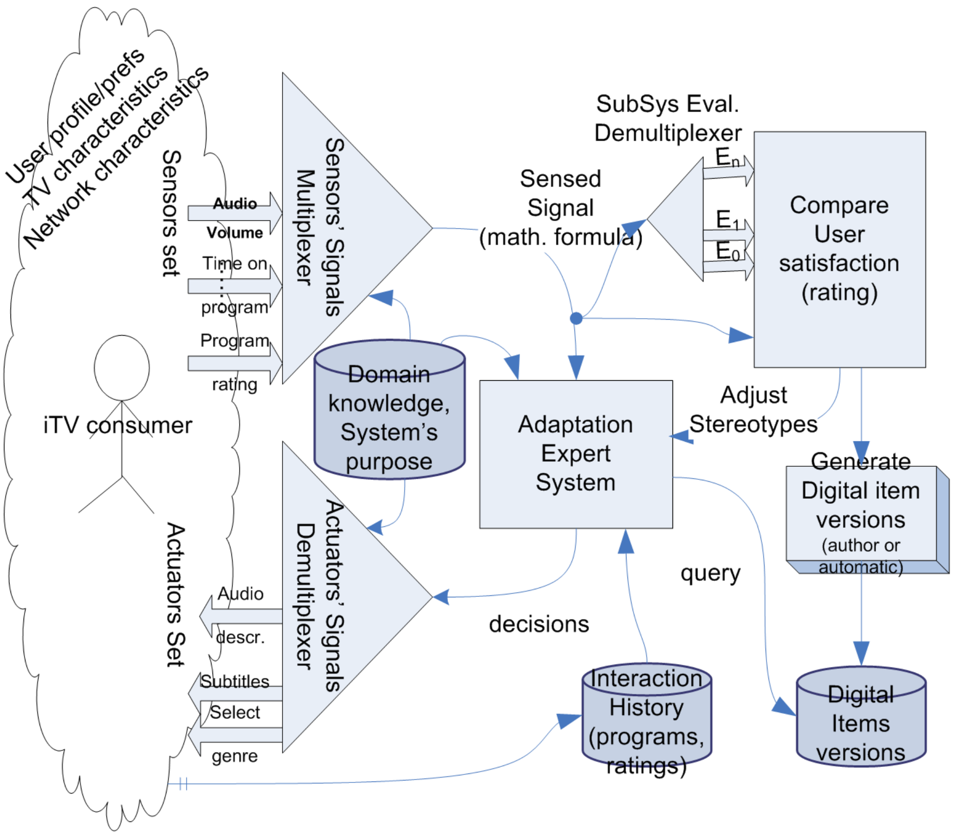


Illustration 60: DAWIS model applied to iTV scenarios

According to the first one, which is actually a specialization of DAWIS conceptual model, a number of sensors are recruited for sensing the changing environment of the iTV consumer. There are used both implicit (time on program and audio volume) and explicit (program rating) types of sensors which are placed on the customer side (Applet). Through a kind of mathematical formula,

77 http://www.schematron.com/

these are multiplexed to an overall signal and sent to the adaptation expert system using web services and an appropriate XML Schema. According to both the overall sensed signal and the user's request, the stereotypes are being adjusted though clustering techniques and this can happen to conclude that more variety is required. In such case, there are two possible scenarios: Either automatic mechanisms are employed for generating such variety (e.g. Automatic summarization, translation etc) or a digital content author is being recruited to generate appropriate digital content. The interaction is being logged so that in the long run of the system the inferences would be safer. At the end of an interaction cycle, the adaptation expert system sends a decision to the customer side accompanied with appropriate resources where this is translated to several user interface actions.

From a technological point of view, web services would be involved offering communication between the subsystems and XML vocabularies would enable metadata serialization. The XML documents are stored into an XML native database that enables retrieval through XQuery. The client side is a Java applet due to the high interactivity requirements.

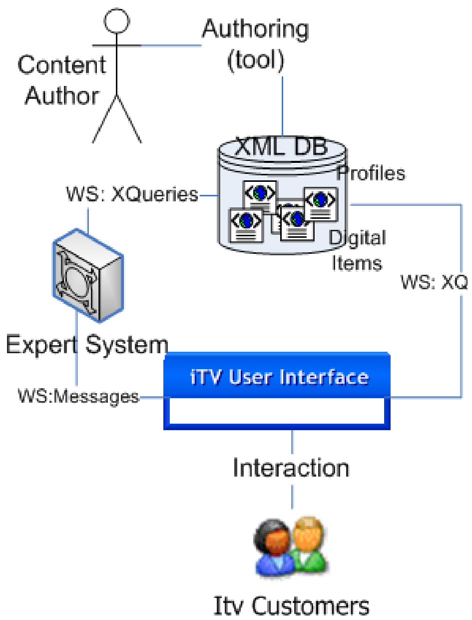


Illustration 61: iTV adaptation architecture

In addition, according to the framework, the adaptive system needs to evaluate its actions in order to adjust its adaptive plan. This means that the system needs to have appropriate sensors to cover parameters relevant to the user satisfaction (e.g. Time in site, explicit rating etc). The fed back values from the sensors can be expressed in a form of test suite containing several tests that happen to satisfy or not several criteria. DAWIS has proposed EARL as a possible infrastructure for accomplishing the requirement for effective communication of the evaluation / testing results between such subsystems. An example of EARL usage in this application can be seen in Table 9:

Context	Mr. X is viewing the Y movie and has just stopped that
Test Subject	The content of the movie
Test Result	the user is interested on this movie
Test performed	What proportion of the movie did he saw ?

Table 9: An example of EARL application on iTV

Concluding this section, the prototype client-side user interface of the application (JAVA applet) is being discussed (Illustration 62). There are three main regions: the TV screen, where the digital item is being presented, the EPG panel where the user can see the program, schedule a reminder and set up her profile and finally the Logger panel where feedback is being provided to the user for the tasks taken.

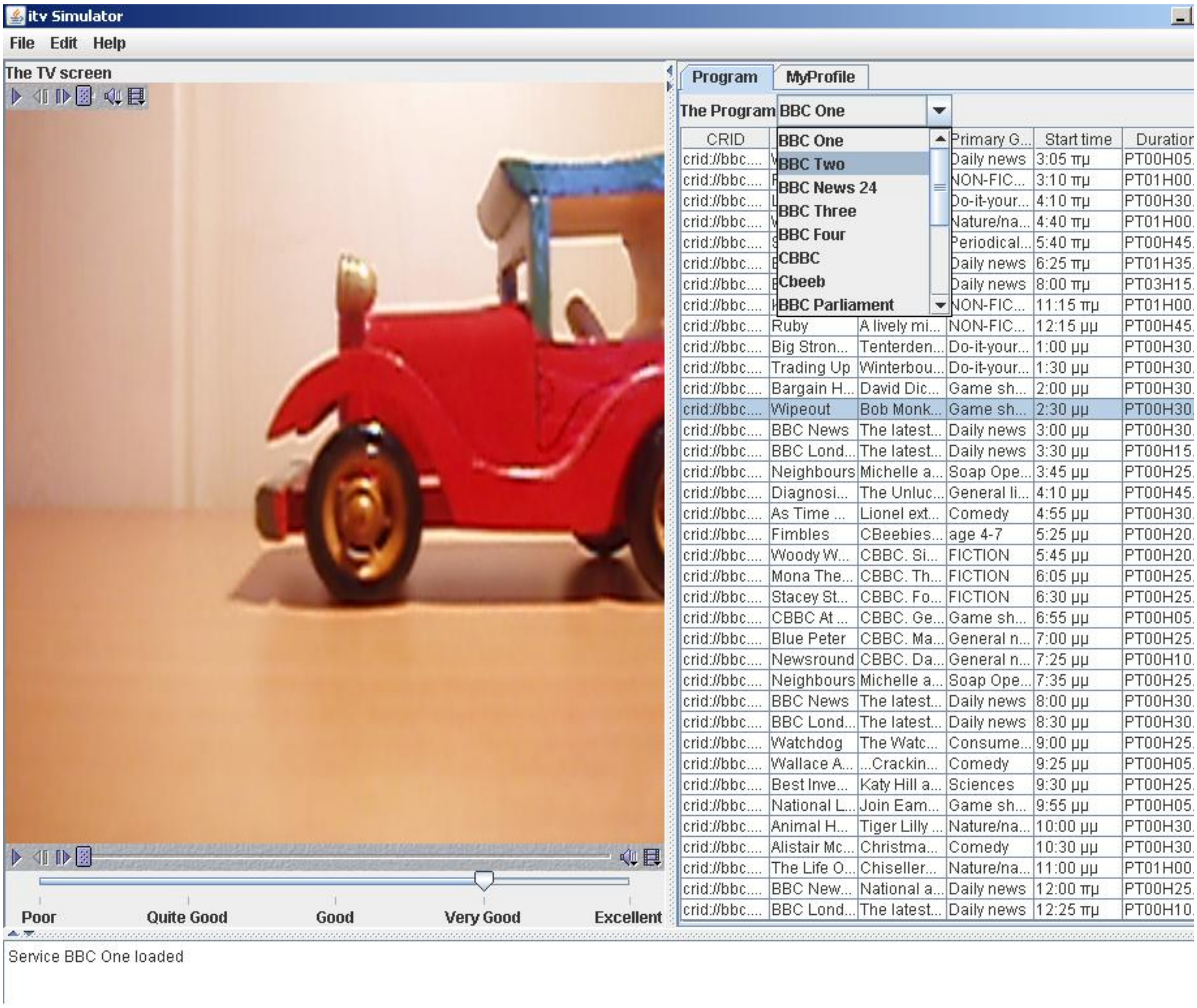


Illustration 62: iTV Simulator prototype Applet

7.5 Summary and Discussion

During the iterative design process of the aforementioned framework several methods, techniques, technologies and software frameworks have been evaluated. This chapter has presented four case studies that have contributed to the development and the preliminary evaluation of the proposed framework.

In the corpus of the IRIS project, web accessibility has been approached through adaptation based on the blending of user and device profiles. The application of this approach has provided a basis for modeling a user-system interaction on top of which the interaction profile has been built. IRIS approach has also provided an interesting software architecture based on a proxy.

BenToWeb has also contributed to the design of adaptive WIS by providing tools based on gained knowledge from real users. During the design and the development of Amfortas several important issues have been investigated relating to the design and development of WIS:

- Experience the interaction of disabled users with specific web technologies and techniques though the evaluation of the test cases developed in the corpus of BentoWeb project;
- Evaluate user profiling techniques focusing on disables users;
- Evaluate state of the art technologies and software frameworks that could be used for implementing adaptive web information system.

Furthermore, there has been done a lot of work on making portal web infrastructure accessible, either as a generic portal, portal in e-commerce and m-commerce. Such a layered approach has been proposed and as a result design guidelines have been emerged.

The last case study discussed in this chapter has been the case of interactive television. Technological advances like MPEG-21 have been studied that on one hand provided information for the proposed framework and on the other hand contributed to the proposal of an accessibility approach through adaptation. This case has been seen as a preliminary evaluation of the proposed framework. Thus, several components of the framework have been evaluated including the conceptual model and the interaction profile. Such a preliminary evaluation has shown the importance and the usefulness of the proposed framework and at the same time introduced new implementation architectures. However, in this case study, several simplifications have taken place and the complexity of the problem situation has been bounded enough so that a prototype development would be realizable in the given short time period by also avoiding to focus only into certain aspects. Consequently, during this attempt it was given the opportunity to approach the whole system through the prism of the proposed framework and evaluated it as a whole. Even the time limitation was itself an opportunity to figure out the usefulness of the framework in terms of required design time and what came out was that the framework can decrease the required design time of an adaptive WIS by assuring its quality at the same time.

Overall Summary and Conclusions

This section aims to provide an overall summary of this thesis by identifying important aspects and conclusions and further propose a road map for future research on the field.

The thesis' problematic area is being emerged from the evolution of the Information and Communication Technological (ICT) developments that has introduced augmenting requirements concerning their symbiosis with human. The notion of adaptivity, the central notion of this thesis, has proved to be the key attribute for the design of complex interactive systems, which seeks to improve the quality of everyday's life. Such a requirement is even more apparent in the case of people with disabilities or more generally in cases of diverging environments - requirements.

Even if so far the research in relating scientific fields has provided many concepts, techniques and tools to enable, in an extend, the adaptation of web information systems it was identified the absence of a reference framework, that could incorporate the mixture of the above mentioned. The aim of such a framework is twofold: to offer a design aid environment to the WIS designer towards to more user-friendly designs and to push the research further by providing a common base and identifying surrounding aspects of adaptation. This thesis argues the possibility to construct such a framework as it is being emerged from the investigation of general adaptive systems though a systemic point of view.

Aiming at investigating systems with different characteristics, and thus ensuring an interdisciplinary research, a classification of general systems has been built as a combination of pre-existing classifications met in literature. As a result, the investigation included natural, social and artificial systems. Issues concerning adaptivity attribute have been discussed in such interdisciplinary fields but this is not meant to be an exhaustive one, but one that gives a rich picture of the notion of adaptivity in general systems and inspire new developments both in conceptual and implementation level.

Starting from the investigation of natural systems, the mechanism of evolution has been expressed in the form of natural selection, according to which the adaptive behavior of a system results from purely physical environmental factors that lead to the inheritable characteristics in populations of organisms. Based on that:

- ✓ A WIS needs not to be adaptive due to its designer but due to its capability of taking advantage of its interaction and interaction history with its environment.

To the same direction, based on social systems literature, it comes out that:

- ✓ the adaptation must arise in a dynamic internal way as the system evolves and thus cannot be externally imposed.

Furthermore, from the viewpoint of self-organization principle,

- ✓ an adaptive WIS can be modeled as an open dynamic system that during its life cycle is being pushed far from its equilibrium due to environmental changes.

In such conditions, in order for a WIS to be adaptive this needs to be self-organized and thus depends on the coupling of the system with the environment and not externally imposed factors, which means that:

- ✓ it is the interaction between system and environment that makes adaptation both necessary and possible.

Such a conclusion is further elaborated from the theory of autopoiesis which states that in order such a system to be adaptive, this must be structure-depended. As such, this should always tend to evolve towards an attractor (a state that fits better to environmental changes**).** Based on the “order of noise principle” it seems that the more the interaction, the more quickly the system will self-organize. According to this and having in mind that a system is “intelligent” if it has a large internal variety of behaviors,

- ✓ a WIS adaptation ability should be proportional to its variety and its ability/intelligence to manage its variety in its interaction with other systems.

Furthermore, this chapter presents Ashby's ultra-stable system which seems that can be used as a basis for a model of a general adaptive system emphasizing on the concept of “essential variables” that determines the model of the system. This can also be compared with human's nervous system as have been investigated through the prism of both cognitive science and biology, which can sense certain changes in its environment and trigger appropriate actions.

Closing the investigation in the field of natural systems, it came out that even in the case of ecosystems there are attempts that seek to face problems through the notion of adaptivity. Specifically, in the case of climate change such an investigation emphasized the requirement:

- ✓ for continues evaluation of the adaptation process and introduced a point of view for a generic framework based on three questions: adapt to what, who or what adapts and how does adaptation occurs.

From a brief investigation of social systems it also came out that:

- ✓ a WIS can be seen as a social network of self-referenced subsystems that are characterized by both their individual and their emerging - holistic attributes.

An important notion that social systems emphasize and at the same time raise its importance to the field of WIS as a social network of self-referenced subsystems is information.

- ✓ The concept of information as defined in the field of social systems (internal process of selection) is foreseen as a vehicle towards the “self-evaluation” of both subsystems and system.

The idea of a social network of subsystems has been formulated in the past in the corpus of Artificial Intelligence and more specifically in the subfield of intelligent agents as multi-agent systems. More specifically, this thesis has presented engineering approaches like neural nets, evolutionary approaches, and more, even from classic AI, and also from cybernetics and control systems engineering that have been proved successful in the past, as foreseen solution to specific problems (e.g. learning and structure generation) in the field of WIS. At the end of this chapter, the narrow field of autonomic computing has been investigated in the domain and scope of adaptive WIS.

Chapter 3 moves the thesis focus to the field of hypertext and hypermedia, the ancestor of the web. From a flashback it came out that the need for developing the infrastructures had disoriented the research to more system-centric solutions and the initial design requirements had been neglected. However, recently HCI factors has been given paramount importance by introducing more user-centric systems mainly expressed with the introduction of user modeling that enabled personalized systems.

Similarly to the evolution of user modeling, adaptation feature has been appeared into many systems as part of their logic. Speaking for adaptation, the separation is seen possible only in the form of aspects (aspect oriented manner). In other words,

- ✓ the adaptation as a mechanism needs to be separated as "logic" but at the same time it must be distributed to the sub-systems.

Furthermore, in this chapter, it has been identified that the nature of hypertext fits to already developed human reading and writing models. This is a fact that need to be considered as an advantage and utilized in favor of the user.

- ✓ It seems that the comparison of the human's cognitive model with the nature of hypertext can prove very interesting and its conclusion can be applied to future adaptive WIS aiming at enabling the cognitive style adaptation. This is even more interesting in the case of cognitive disabled users.

This chapter ends up with the presentation of methods, models and techniques having been used in the field of adaptive hypermedia that need to be abstracted and adapted in order to fit to the WIS domain.

The next chapter, chapter 4, employed the investigation of the role of adaptivity in the field of Web Information Systems with strong emphasis to the web accessibility requirements, seen as an interesting case of adaptive WIS. A survey in the field of web accessibility and a resulting taxonomy of the facing approaches came up with several techniques that are incorporated to the techniques for adaptive WIS.

Portal paradigm and infrastructure is considered as a special – interesting case for the subject of this thesis thanks to its primary requirement; personalization. An investigation of its role to the e-commerce and m-commerce shows off its importance and at the same time the requirement to make that as accessible as possible. However, at the same time, this chapter sets off portal's special characteristics that could contribute towards to such purpose.

Even from the early hypermedia days it was apparent the requirement for modeling the system's environment. Such a requirement has been recently transformed to a demand due to the incursion of the "disappearing computer" to our everyday life. WIS come to respond to such a demand by incorporating context-aware parameters. This chapter has presented such interesting developments of context modeling approaches mainly based on semantic web concepts and technologies. From such a point of view semantic web is seen as a complementary mechanism that would enable more intelligent systems, capable of concluding to more useful results, closer to human needs.

This chapter ends up with the investigation of the use of multimedia technologies and abstract user interfaces intended for highly interactive systems. Multimedia technologies seem to be the upcoming hot subject for adaptation for instance through the spreading of the use of interactive television. Technological advances like MPEG-21 have been studied providing useful information for the proposed framework. Last, but not least, the concept and developments of abstract user interfaces has seen as a stopover to the adaptive WIS as this would allow an abstracted designed interface to be capable of adapting its presentation according to the interaction context.

Chapter 5 has provided a methodological base for designing adaptive web information systems including HCI design methods, software design methods and paradigms. It came out that such systemic design methods can been used as a source of inspiration for the process of adaptation. Consequently,

- ✓ adaptation can be seen as an continuous, iterative design process with changing requirements. In that sense, an adaptive system is a system that can continuously design itself (self-designable).

From a software engineering perspective, it came out that the tendency is to move from complex, solution-specific / monolithic developments to modularized / distributed and open architectures that enables reusability and separation of concerns. A major “fruit” of such open architectures is the service oriented paradigm.

- ✓ The use of service oriented architectures and web services seem to offer a good basis for the necessary distributed architecture for adaptive web information systems.

Special interest turns up to be the emerging field of semantic web services. The investigation to this field concluded that,

- ✓ in SOA the architecture proves to be an emergent property and this is actually the property that could offer adaptivity in a WIS though the self searching, retrieval and orchestration of services.

Finally, such services would be also characterized by capabilities for undertaking roles and evaluating their performance based on certain criteria.

The actual proposed framework for designing adaptive information systems (DAWIS) has been presented in chapter 6. The framework consists of:

- Axioms and definitions aiming at clarifying the notion of adaptivity by also providing primitive models / metrics, required for further analysis and design (adaptivity measurement, effectiveness and capacity). The models have been also expressed as mathematical models but these are only in a very primitive model. As a future work, these should be evaluated, maybe some though simulation, in order to provide a more appropriate representation.
- A primitive and empirical graph aiming at supporting the designer on the decision, whether and in what extent a problem situation requires to come up with an adaptive system.

Further evaluation need to be done by applying it to real designs. In addition, not straightforward quantified parameters such as "complexity" need to be specified more by probably introducing classes of design situations and domains of application.

- A two order (higher and individual order) conceptual model of an adaptive web information system based on literature investigation in the field of General Systems.
- Fundamental functional specifications of an adaptive system. An interaction profile has been proposed responding to the "adapt what" question. The "adapt to what" response and the proposal for the system's behavior evaluation complements the functional specifications.
- An implementation architecture that identifies the key aspect, providing indicative API and proposing useful technologies and software frameworks. Such proposals give the state of the art of the technological landscape but by no means aims to restrict the designer to their use.
- Exploitation scenario through design support environments which can also be used as a kind of methodology for designing adaptive web information systems.

The last chapter presented four case studies that have contributed to the development and the preliminary evaluation of the proposed framework.

- Under the umbrella of IRIS project, web accessibility has been approached through adaptation based on the blending of user and device profiles. The application of this approach has provided a basis for modeling a user-system interaction on top of which the interaction profile has been built. IRIS approach has also provided an interesting software architecture based on a proxy.
- BenToWeb has also contributed to the design of adaptive WIS by providing tools based on gained knowledge from real users. During the design and the development of Amfortas several important issues were investigated relating to the design and development of WIS. This included the experience of the interaction of disabled users with specific web technologies and techniques, though the evaluation of the test cases developed in the

corpus of BentoWeb project, the evaluation of user profiling techniques focusing on disables users, and finally, the evaluation of the state of the art technologies and software frameworks that could be used for implementing adaptive web information system.

- The case of a web portal from a point of view on making portal web infrastructure accessible, either as a generic portal, portal in e-commerce and m-commerce has been discussed. A layered approach has been proposed and as a result design guidelines have been emerged showing off the usefulness of DAWIS perspective.

- The case of interactive television that considers technological advances like MPEG-2,1 for both providing information for the proposed framework and for contributing to the accessibility of iTV through adaptation. During such a preliminary evaluation of the proposed framework, several components of the framework have been evaluated including the conceptual model and the interaction profile, which showed off the importance of the proposed framework and at the same time introduced new implementation architectures.

In the case studies presented, always the focus was on the evaluation of concepts, techniques and technologies of certain aspects of adaptation and only in the case of iTV the problem has been approached as a whole from a DAWIS perspective. Thus it is apparent the need for further evaluation of the conceptual model and the framework as a whole into different complexity levels of problem situations.

It should be mentioned that this work does not claim to provide all the methods and tools that would be capable of offering “true” autonomy to WIS as Steels [Steels, 1993] describes the top level of behavioral autonomy. However, it can be claimed that this thesis has provided an abstract enough framework addressing the important aspects that need to be faced for establishing such an adaptation, as came out from an interdisciplinary research, and further offered tools and techniques for implementing a certain design. Finally, several directions for further research have been identified that could contribute to the design and implementation of web information adaptive systems giving more emphasis to the investigation of applicability of classical and modern AI such as neural nets, evolutionary algorithms etcetera and to the evaluation of the framework.

Index of Illustrations and Tables

Abbreviations

AI	Artificial Intelligence
API	Application Programming Interface
AWIS	Adaptive WIS
BenToWeb	Benchmarking Tools for the Web (IST project)
CC/PP	Composite Capabilities/Preferences Profile
DAWIS	Design Adaptive WIS (the name of the proposed framework)
DfA	Design for all
DSE	Design Support Environment
EARL	Evaluation And Report Language
GAS	General Adaptive Systems
GST	General Systems Theory
HCI	Human Computer Interaction
ICT	Information and Communication Technologies
IRIS	Incorporating Requirements of People with Special Needs or Impairments to Internet-based Systems and Services (IST project)
ITV	Interactive Television
MPEG	Moving Pictures Experts Group
OWL	Web Ontology Language
RDF	Resource Description Framework
SSM	Soft Systems Methodology
TV	Television

UA	User Agent (web browser)
WIS	Web Information Systems
WWW	World Wide Web
XML	extensible Markup Language

References

[Abdelnur & Hepper, 2003]: Abdelnur, A.;Hepper, S.(2003), Java Portlet Specification (Version 1.0). In JSRs: Java Specification Requests .

[Abou-Zahra, 2007]: Abou-Zahra, S.(2007), Evaluation and Report Language (EARL) 1.0 Schema. In W3C Working Draft 23 March 2007 .

[Aiken & Sullivan, 2002]: Aiken M.;Sullivan D.(2002). Best Practices in Enterprise Information Portal adoption: 5 key drivers. In DMReview in Noember 2002 .

[Alexandraki et al, 2004]: Alexandraki, C.;Paramythis, A.;Maou, N.;Stephanidis, C., (2004). Web Accessibility through Adaptation. In 9th International Conference, ICCHP 2004, Paris, France, July 7-9, 2004.

[Allan et al, 2006] Allan, D.;Kling, G.;Pluijm, B., Evolution and Natural Selection, 2006, http://www.globalchange.umich.edu/globalchange1/cu

[Antoniou & Harmelen, 2004]: Antoniou, G.;van Harmelen, F., A Semantic Web Primer,(2004). In MIT Press, 2004, ISBN 0-262-01210-3.

[Armstrong, 2006]: Armstrong, D. J. (2006). The quarks of object-oriented development. In Commun. ACM 49, 2 (Feb. 2006), 123-128. DOI=http://doi.acm.org/10.1145/1113034.1113040

[Ashby, 1947]: Ashby, W. R.(1947). Principles of the Self-Organizing Dynamic System. In Journal of General Psychology (1947), volume 37, pages 125--128.

[Ashby, 1952]: Ashby, W. R., Design for a Brain - The Origin of Adaptive Behavior, (1952).In Chapman and Hall.

[Ashby, 1956]: Ashby, W. R., Introduction to Cybernetics,(1956). In Chapman & Hall, 1956, ISBN 0-416-68300-2 (also available in electronic form as a PDF from Principia Cybernetica.

[Balasubramanian et al,2002]: Balasubramanian, S.;Peterson, R.A.;Jarvenpaa, S.L.(2002). Exploring the Implications of M-commerce for Markets and Marketing. In Journal of the Academy of Marketing Science, Vol.30, No.4:348-361, 2002.

[Bausch, 2001]: Bausch C. K. (2001), The Emerging Consensus of Social Systems Theory, In New York: Kluwer Academic/Plenum Publishers.

[Beck, 2000]: Beck, K. (2000), Extreme Programming Explained: Embrace Change,. In Addison-Wesley, ISBN 0201616416.

[Bellas et al, 2004]: Bellas, F.;Daniel, F.; Abel, M., (2004). A flexible framework for engineering "My" Portals. In The 13th international conference on World Wide Web table of contents New York, NY, USA Pages: p. 234 - 243.

[BenToWeb_Delv3.7a]: Evaluation and validation reports for test suites (First evaluation), 2006. In BenToWeb Project, Deliverable 3.7a .

[Berglund & Johansson, 2004]:Berglund, A.; Johansson, P.(2004). Using speech and dialogue for interactive TV navigation. In Universal Access Inf Soc 3(3–4):224–238.

[Berners-Lee et al, 2001]: Berners-Lee, T.;Hendler, J.;Lassila, O.(2001). The Semantic Web. In Scientific American, May 2001.

[Boehm, 1986]: Boehm, B.(1986). A spiral model of software development and enhancement. In SIGSOFT Softw. Eng. Notes 11, 4 (Aug. 1986), 14-24. DOI= http://doi.acm.org/10.1145/12944.12948.

[Booth et al, 2004]: Booth, D.;Haas, H.;McCabe, F.;Newcomer, E.;Champion, M.;Ferris, C.;Orchard, D.(2004), Web Services Architecture. In W3C Working Group Note 11 February 2004 .

[Boulding, 1956]: Boulding, K. E.(1956). General systems theory - the skeleton of science. In Management Science, 2 (3).

[Boulding, 1985]: Boulding, K. E., The World as a Total System,(1985). In SAGE Publications, 1985.

[Bray et al, 2006]: Bray, T.;Paoli, J.;Sperberg-McQueen, C.; Maler, E.;Yergeau, F.(2006), Extensible Markup Language (XML) 1.0 (Fourth Edition). In W3C Recommendation 16 August 2006, edited in place 29 September 2006 .

[Brickley & Guha, 2004]: Brickley, D.;Guha, R.(2004), RDF Vocabulary Description Language 1.0: RDF Schema. In W3C Recommendation 10 February 2004 .

[Brodie et al, 2005]: M., Brodie;C., Bussler;J., de Brujin;T., Fahringer;D., Fensel;M., Hepp;H., Lausen;D., Roman;T., Strang;H., Werthner; M. Zaremba(2005), Semantically Enabled Service-Oriented Architectures: A Manifesto and a Paradigm Shift in Computer Science. In DERI-TR-2005-12-25, Dec. 2005 .

[Broekstra et al, 2002]: Broekstra, J.;Kampman, A.; Harmelen, F.(2002). Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema. In Towards the Semantic Web: Ontology-Driven Knowledge Management, J. Davis, D. Fensel, and F. v.Harmelen, Eds.: John Wiley and Sons Ltd, 2002.

[Brooks, 1991]: Brooks, R. A.(1991). Intelligence without reason. In Proceedings of the twelfth International Joint Conference on Artificial Intelligence (IJCAII-91), 569-595.

[Brown & Lawton, 2001]: Brown, D. J.;Lawton, J.(2001), Design Guidelines and Issues for Web Site Production for Use by People with a Learning Disability. In (Draft 3.0 January 2001) Foundation for People with Learning Disabilities .

[Brusilovsky & Maybury, 2002]: Brusilovsky, P.;Maybury, M. T. (2002). From adaptive hypermedia to adaptive Web.. In P. Brusilovsky and M. T. Maybury (eds.), Communications of the ACM 45 (5), Special Issue on the Adaptive Web, 31-33.

[Brusilovsky, 1996]: Brusilovsky, P.(1996). Methods and techniques of adaptive hypermedia. In User Modeling and User-Adapted Interaction, 6 (2-3), pp. 87-129.

[Brusilovsky, 2001]: Brusilovsky, P.(2001). Adaptive hypermedia. In User Modeling and User Adapted Interaction, Ten Year Anniversary Issue (Alfred Kobsa, ed.) 11 (1/2), 87-110.

[Brusilovsky, 2004]: Brusilovsky, P.(2004). Adaptive navigation support: From adaptive hypermedia to the adaptive web and beyond. In PsychNology Journal 2004, Volume 2 Number 1, 7-23.

[Bush, 1945]: Bush, V.(1945). As We May Think. In The Atlantic Monthly, 176(1),101-108, July 1945.

[Butler, 2002]: Butler, M.H. (2002). DELI: A DElivery context LIbrary for CC/PP and UAProf. In External Technical Report HPL-2001-260.

[Canali et al, 2005]: Canali, C.;Cardellini, V.;Lancellotti, R.(2005). Content Adaptation Architectures Based on Squid Proxy Server. In World Wide Web, 2006, VOL 9; NUMBER 1, pages 63-92.

[Carmichael et al, 2006]: Carmichael, A.;Rice, M.;Sloan, D.(2006). Inclusive Design and Interactive Digital Television: Has an Opportunity been Missed?. In 3rd Cambridge Workshop on Universal Access and Assistive Technology. Fitzwilliam College , Cambridge, 10-12 April 2006.

[Chan et al, 2002]: Chan, S.;Fang, X.;Brzezinski, J.;Zhou, Y.;, Xu, S.; Lam, J. (2002). Usability for Mobile Commerce Across Multiple Form Factors. In Journal of Electronic Commerce Research, Pg 181-199, VOL. 3, NO. 3, 2002.

[Checkland, 1976]: Checkland, P.(1976). Towards a systems-based methodology for real-world problem solving. In Systems Behaviour, J. Beison and G. Peters, eds. Harper and Row, London, pp 51-77.

[Checkland, 1990]: Checkland, P., Soft Systems Methodology in Action,(1990). In Wiley, John and Sons, Inc., 1990.

[Checkland, 1999]: Checkland, P., Systems Thinking, Systems Practice (Includes a 30-year retrospective),(1999). In .

[Chen et al, 2004]: Chen, H.;Finin, F.;Joshi, A., (2004). An ontologyfor context-aware pervasive computing environments. In Spe-cial Issue on Ontologies for Distributed Systems, KnowledgeEngineering Review, 2004.

[Chen et al, 2005]: Chen, M.;Zhang, D.;Zhou, L.(2005). Providing web services to mobile users: the architecture design of an m-service portal. In Int. J. Mobile Communications, Vol. 3, No. 1, pp.1–18.

[Chin, 1993]: Chin, D. N(1993). Acquiring user models. In Artificial Intelligence Review 7:185-197.

[Clarke & Flaherty, 2003]: Clarke, I.;Flaherty, T. B.(2003). Web-based B2B portals. In Industrial Marketing Management, 32 (2003) 15– 23.

[Conklin,1988]: Conklin, J.(1998). Hypertext: an introduction and survey (Reprint). In Computer-Supported Cooperative Work: A Book of Readings, I. Greif, Ed. Morgan Kaufmann Publishers, San Francisco, CA, 423-475.

[Corcho et al, 2003]: Corcho, O.;Fernandez-Lopez, M.;Gomez-Perez, A.(2003). Methodologies, tools andlanguages for building ontologies: Where is their meeting point?. In Data & KnowledgeEngineering, Vol. 46, 2003, 41-64.

[Cranor et al, 2002]: Cranor, L.;Langheinrich, M.;Marchiori, M.; Presler-Marshall, M.;Reagle, J.(2002). The Platform for Privacy Preferences 1.0 (P3P1.0) Specification. In W3C Proposed Recommendation, 28 January 2002.

[Csanyi & Kampis, 1991]: Csanyi, V.;Kampis, G.(1991). Modeling biological and social change: Dynamic replicative network theory. In Lazlo E. (1991b), 77-92.

[Darby, 1997]: Darby, S.(1997). Introduction to Enhancing the Accessibility of Digital Television. In RNIB.

[Darwin, 1859]: Darwin C., Origin of Species,(1859). On the Origin of Species. John Murray, London.

[De Bra et al, 2004]: De Bra, P.;Aroyo, L.;Chepegin,V.(2004). The Next Big Thing: Adaptive Web-Based Systems. In Journal of Digital Information, Volume 5 Issue 1Article No. 247, 2004-05-27.

[De Bra, 2002]: De Bra P.(2002)(2002). Adaptive educational hypermedia. In Web Communications of the ACM, Volume 45, Number 5 Pages 60-61.

[De Nicola et al, 2005]: De Nicola, A.;Missikoff, M.;Navigli, R., (2005). A Proposal for a Unified Process for Ontology Building: UPON. In Proceedings of the 16th International Conference on Database and Expert Systems Applications (DEXA 2005).

[DeRose et al, 1999]: DeRose, S.;Orchard, D.;Trafford, B.(1999), XML Linking Language. In W3C Specification .

[Detlor, 2002]:Detlor, B.(2002). The corporate portal as information infrastructure: Towards a framework for portal design. In International Journal of Information Management. 20, 2 (2000), 91–101.

[Dey & Abowd, 2000]: Dey, A. K.;Abowd, G. D., (2000). Towards a bet-ter understanding of context and context-awareness. In Workshop on the What, Who, Where,When and How of Context-Awareness, New York. ACMPress.

[Dholakia & Rask, 2002]: Dholakia, N.;Rask, M.(2002). M-commerce is all about personalization, permision and specification . In book, Dynamic elements of emerging mobile portal strategies.

[Ding et al, 2004]: Ding, L.;Finin, T.;Joshi, A.;Pan, R.;Cost, R. S.;Peng, Y.; Reddivari, P.;Doshi, V.;Sachs, J., (2004). Swoogle: a search and metadata engine for the semantic web. In the Thirteenth ACM international Conference on information and Knowledge Management (Washington, D.C., USA, November 08 - 13, 2004). CIKM '04. ACM Press, New York, NY, 652-659. DOI= http://doi.acm.org/10.1145/1031171.1031289.

[Fairweather et al, 2002]: Fairweather, P.;Hanson, V.;Detweiler, S.;Schwerdtfeger, R.(2002). From Assistive Technology to a Web Accessibility Service. In The fourth International ACM Conference on Assistive Technologies, p. 4–8. ACM Press.

[Fensel et al, 1998]: Fensel, D.;Decker, S.;Erdmann, M.;Stude, R., (1998). Ontobroker : The very high idea.. In 11th International Flairs Conference (FLAIRS-98), Sanibal Island, Florida, USA.

[Fernández-López et al, 2002]: Fernández-López, M. et al (2002), Survey on Methodologies for Developing, Maintaining,Evaluating and Reengineering Ontologies. In OntoWeb Deliverable 1.4 .

[Fernández-López, et al, 1997]: Fernández-López, M.;Gómez-Pérez, A.;Juristo, N., (1997). Methontology: from ontological art towards ontological engineering. In Proc. Symposium on Ontological Engineering of AAAI (1997).

[Fink & Kobsa, 2000]: Fink, J.;Kobsa A.(2000). A Review and Analysis of Commercial User Modeling Servers for Personalization on the World Wide Web. In User Modeling and User-Adapted Interaction Journal, 10, 209-249.

[Fink et al, 1996]: Fink, J.; Kobsa, A; Nill A.(1996). User-oriented Adaptivity and Adaptability in the AVANTI Project. In Designing for the Web: Empirical Studies, 1996.

[Foerster, 1960]: von Foerster, H.(1960). On self-organising systems and their environments. In in Self-Organising Systems, M.C. Yovits and S. Cameron (eds.), Pergamon Press, London, pp. 30-50.

[Fowler, 2003]: Fowler, M., Inversion of Control Containers and the DependencyInjection pattern,(2003). (http://www.martinfowler.com/articles/injection.html)

[Frank, 2006]: Frank, B.(2006). Driving devices: lessons learned in the business of designing mobile UIs. In Interactions 13, 4 (Jul. 2006), 14-15.

[Fred & Lindesmith, 2003]: Fred, A.;Lindesmith, S.(2003). The case for portlets. How to decide if portlets are your best option. In IBM DeveloperWorks.

[Gappa & Nordbrock, 2004]: Gappa,H.;Nordbrock,G.(2004). Applying Web accessibility to Internet portals. In Universal Access in the Information Society, 3, 1, 3/1/2004, Pages 80-87.

[Geyer, 1995]: Geyer, F.(1995). The Challenge of Sociocybernetics. In Kybernetes24(4), 6-32.

[Gilman, 1997]: Gilman, S.(1997), Note on CSS/ACSS requirements. In .

[Goble et al, 2001]: Goble, C.;Bechhofer, S.;Carr, L.;De Roure, D.;Hall W., (2001). Conceptual open hypermedia = the semantic web?. In In Proceedings of the Second International Workshop on the Semantic Web - SemWeb'2001, Hongkong, China, May 1, 2001. CEUR Workshop Proceedings, 2001.

[Godwin & Haenel, 2002]: Godwin, R.;Haenel, W., (2002).), Portal Aggregation for Pervasive Devices. In IBM WebSphere Developer Technical Journal.

[Golemati et al, 2007]: Golemati, M.;Katifori, A.;Vassilakis, K.;Lepouras, G.;Halatsis, C., (2007). Creating an Ontology for the User Profile: Method and Applications. In The First IEEE International Conference on Research Challenges in Information Science (RCIS), Morocco 2007.

[Grewal et al, 2004]: Grewal, D.;Iyer, G.R.;Levy, M.(2004). Internet retailing: enablers, limiters and market consequences. In Journal of Business Research, Elsevier, vol. 57(7), pages 703-713, July.

[Grosso et al, 2003]: Grosso, P.;Maler, E.;Marsh, J.; Walsh, N.(2003), XPointer Framework. In W3C Recommendation 25 March 2003 .

[Gu et al, 2004]: Gu, T.;Wang, X.H.;Pung, H.K.;Zhang, D.Q., (2004). An Ontology-based Context Model in Intelligent Environments. In The Communication Networks and Distributed Systems Modeling and Simulation Conference, San Diego, USA.

[Halasz & Schwartz, 1990]: Halasz, F. & Schwartz,M., (1990). The Dexter Hypertext reference Model. In Proceedings of the Hypertext Standardization Workshop January 16-18, 1990, National Institute of Standards and Technology.

[Halsey & Anderson, 2000]: Halsey, B.,;Anderson, K. M.(2000). XLink and Open Hypermedia Systems: A Preliminary Investigation. In In Proceedings of Hypertext 2000, pp. 212-213.

[Hanson et al, 2005]: Hanson,L.;Brezin, J. P.; Crayne, S.;Keates, S.;Kjeldsen, R.;Richards, J. T.; Swart, C.;Trewin, S.(2005). Improving Web accessibility through an enhanced open-source browser.. In IBM System Journal, 44 (3):573--588, 2005..

[Hariri et al, 2006]: Hariri;Khargharia;Chen;Yang;Zhang;Parashar;Liu (2006). The Autonomic Computing Paradigm. In Journal of Cluster Computing, Vol. 9, Num 1, p.5-17, 2006.

[Harper & Bechhofer, 2005]: Harper, S.;Bechhofer, S.(2005). Semantic Triage for Accessibility. In IBM SystemsJournal, 44(3):637–648, 2005.

[Harris et al, 2005]: Harris, P.;Rettie, R.; Kwan, C.(2005). Adoption and Usage of M-Commerce: A Cross-Cultural Comparison of Hong Kong and the United Kingdom. In Journal of Electronic Commerce Research Vol. 6, num. 3, 2005. Special Issue: Mobile Commerce Research.

[Hazra, 2002]: Hazra T. K.(2002). Building enterprise portals: principles to practice. In The 24th International Conference on Software Engineering table of contents (2002).

[Heckmann et al, 2005]: Heckmann, D.;Schwartz, T.;Brandherm, B.;Schmitz, M.;Wilamowitz-Moellendorff, M., (2005). GUMO – The General User Model Ontology. In The 10th International Conference on User Modeling, pages 428–432, Ed-inburgh, Scotland, Jun 2005. LNAI 3538: Springer, BerlinHeidelberg.

[Heflin et al, 1999]: Heflin, J.;Hendler, J.; Luke, S.(1999), A Knowledge Representation Language for Internet Applications. In Dept. of Computer Science,University of Maryland, College Park 1999 .

[Held et al, 2002]: Held, A.;Buchholz, S.;Schill, A., (2002). Modeling of Context Information for Pervasive Computing Applications. In the 6th World Multiconference on Systemics, Cybernetics and Informatics (SCI), Orlando, FL, July 2002.

[Hepper & Lamb, 2004]: Hepper, S. & Lamb M.(2004), Best practices for developing portlets using JSR 168 and WebSphere Portal V5.02. In IBM, developerworks .

[Hepper et al, 2005]: Hepper, S.;Fischer, P.;Hesmer, S.;Jacob, R.;Taylor D.S., Portlets and Apache Portals,(2005). In Manning Publications Co..

[Herramhof et al, 2006]: Herramhof, S.;Petrie, H.;Strobbe, C.; Vlachogiannis, E.;Weimann, K.;Weber, G.;Velasco, C A, (2006). Test Case Description Language (TCDL): Test Case Metadata for Conformance Evaluation. In Lecture Notes in Computer Science, Volume 4061, Jul 2006, Pages 164 - 171, DOI 10.1007/11788713_25.

[Heylighen & Joslyn, 2001]: Heylighen, F.;C. Joslyn(2001). Cybernetics and Second Order Cybernetics. In in: R. A.Meyers (ed.), Encyclopaedia of Physical Science & Technology, Vol. 4 (3rd ed.), AcademicPress, pp. 155-170.

[Heylighen & Joslyn, 2001b]: Heylighen, F.;Joslyn,C.(2001). The Law of Requisite Variety. In Principia Cybernetica Project .

[Heylighen et al, 1999]: F. Heylighen, C. Joslyn, V. Turchin,(1999). What are Cybernetics and Systems Science?. In Principia Cybernetica Web (Principia Cybernetica, Brussels).

[Heylighen, 1989]: Heylighen, F, Representation and Change,(1989). In Communication & Cognition, Ghent, Belgium (1989).

[Heylighen, 1997] Heylighen, F., Self-organization, 1997, http://pespmc1.vub.ac.be/SELFORG.html

[Holland & Morse, 2001]: Holland, S.;Morse, D. R., (2001). Audio GPS: Spatial Audio in a Minimal Attention Interface. In Human Computer Interactionwith Mobile Devices, Lille, France, 2001.

[Holland, 1975]: Holland, J. H.(1975). Adaptation in Natural and Artificial Systems,. In University of Michigan Press. (Second edition: MIT Press, 1992).

[Houben & De Bra, 1998]: Houben, Wu H.;De Bra, P., (1998). AHAM: A Reference Model to SupportAdaptive Hypermedia Authoring. In Proceedings of InfWet 98.

[Huang, 2000]: Huang, A. W., (2000). Aurora: A Conceptual Model for Web-Content Adaptation to Support the Universal Usability of Web-based Services. In .

[IBM Autonomic] IBM Autonomic Research, , http://www.research.ibm.com/autonomic

[IRISD0602int]: Software Agents Design (2002). IRIS Internal Deliverable D0602int.

[Issarny et al, 2005]: Issarny, V.;Sacchetti, D.;Tartanoglu, F.;Sailhan, F.;Chibout, R.;Levy, N.;Taloma, A.(2005). Developing ambient intelligence systems: A solution based on Web services. In Journal of Automated Software Engineering, 2005. .

[Jarvenpaa et al, 2003]: Jarvenpaa, S.L.; Lang, K.R.;Takeda, Y.;Tuunainen, V.K.(2003). Mobile Commerce At Crossroads. In Association forComputing Machinery: Communications of the ACM, Vol. 46, No. 12:41, 2003.

[Jones & Marsden, 2006]: Jones, M.; Marsden, G.(2005). Mobile Interaction Design. In John Wiley & Sons.

[Jones & Marsden, 2006]: Jones, M.;Marsden, G., Mobile Interaction Design,(2006). In John Wiley.

[Jordan, 1968]: Jordan, N., Themes in Speculative Psychology,(1968). In Tavistock, London.

[Kephar & Chess, 2003]: Kephart, J., O.;Chess, D. M.(2003). The Vision of Autonomic Computing. In IEEE Computer, 36(1), pp. 41-50, 2003.

[Khalifa & Shen, 2006]: Khalifa M.;Shen K., (2006). Determinants of m-commerce adoption: An integrated approach. In Conference on Information Systems (EMCIS) 2006, 6-7 July.

[Kimball, 2003] Kimball, W. J. (2003), Organization of the Nervous System, http://users.rcn.com/jkimball.ma.ultranet/BiologyP

Kintsch, 1988: Kintsch, W., The Role of Knowledge in Discourse Comprehension :A Construction-Integration Model, 1988

[Klyne et al, 2002]: Klyne, G;Reynolds, F.;Woodrow, C.;Ohto, H.;Butler, M. H.(2002). Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies. In , W3C Working Draft 08 November 2002. World Wide Web .

[Kobsa et al, 1999]: Kobsa, A.;Koenemann, J.;Pohl, W.(1999), Personalized hypermedia presentation tech-niques for improving online customer relationships. In Technical report No. 66 GMD,German National Research Center for Information Technology, St. Augustin, Germany .

[Kobsa, 2001]: Kobsa, A(2001). Generic User Modeling Systems. In User Modeling and User-Adapted Interaction 11: 49-63.

[Kobsa, 2002]: Kobsa, A.(2002). Personalized hypermedia and International Privacy. In Communication of the ACM, May2002/Vol 45 No5. SPECIAL ISSUE: The adaptive web.

[Koch & Wirsing, 2002]: Koch, N.;Wirsing, M., (2002). The Munich Reference Model for adaptive hypermedia applications. In Adaptive Hypermedia and Adaptive Web-Based Systems, Second International Conference, AH 2002, Vol. 2347, Lecture Notes in Computer Science, Springer, pp. 213–222.

[Koch, 2001]:Koch, N., (2001). Software Engineering for Adaptive Hypermedia Systems. In PhD Thesis.

[Kottapally et al, 2003]: Kottapally, K.;Ngo, C.;Reddy, R.;Pontelli, E.;Son, T. C.;Gillan, D.(2003). Towards the creation of accessibility agents for non-visual navigation of the web. In The 2003 Conference on Universal Usability (Vancouver, British Columbia, Canada, November 10 - 11, 2003). CUU '03. ACM Press, New York, NY, 134-141..

[Kovacic, 2005]: Kovacic, S.F.(2005). General Taxonomy of System[ic] Approaches for Analysis and Design. In Systems, Man and Cybernetics, 2005 IEEE International Conference on , vol.3, no.pp. 2738- 2743 Vol. 3, 10-12 Oct. 2005.

[Koza, 1997]: Koza, J. R.(1997). Genetic Programming. In Encyclopedia of Computer Science and Technology.

[Kropp et al, 2003]: Kropp, A.;Leueu, C.;Thompson, R.(2003), Web Services for Remote PortletsSpecification. In Approved as an OASIS Standard August 2003 .

[Kruchten, 2000]: Kruchten, P., The Rational Unified Process: An Introduction,(2000). In Addison-Wesley - ISBN 0201707101.

[Lassila & Swick, 1999]: Lassila, O.;Swick, R.(1999), Resource Description Framework (RDF) Model and Syntax Specification. In W3C Recommendation 22 February 1999. World Wide Web Consortium .

[Lausen et al, 2004]: Lausen, H.;Stollberg, M.;Hernández, R. L.;Ding, Y.;Han, S. K.;Fensel D.(2004), Semantic Web Portals - state of the art survey. In Technical Report TR-2004-04-03, DERI(www.deri.org), 2004 .

[Ledbetter, 1999]: Ledbetter J.(1999). Some pitfalls in portals. In Columbia Journalism Rev 1999, 38:22– 23.

[Lee & Benbasat, 2003]: Lee, Y.E.;Benbasat, I.(2003). Interface design for mobile commerce. In Commun. ACM 46, 12 (Dec. 2003), 48-52. DOI= http://doi.acm.org/10.1145/953460.953487 .

[Lee & Benbasat, 2004]: Lee, Y.E.;Benbasat, I.(2004). A framework for the study of customer interface design for mobile commerce. In International Journal of Electronic Commerce. 8(3) 79-102.

[Liao et al, 2005]: Liao, L.;Kaiquan, Xu; Liao, S.(2005). Constructing intelligent and open mobile commerce using a semantic web approach. In Journal of Information Science 2005 31: 407-419.

[Luhmann, 1995]: Luhmann, N, Social Systems,(1995). In Stanford UniversityPress, Stanford, CA.

[MacKenzie et al, 2006]: MacKenzie, C. M.;Laskey, K.;McCabe, F.;Brown P. F.;Metz, R.(2006), Reference Model for Service Oriented Architecture 1.0. In OASIS .

[Maedche et al, 2001]: Maedche, A.;Staab, S.;Stojanovic, N.;Studer, R.;Sure, Y., (2001). SEAL -- A Framework for Developing SEmantic Web PortALs. In Advances in Databases : 18th British National Conference on Databases, BNCOD 18 Chilton, UK, July 9-11, 2001, Proceedings.

[Maedche et al, 2002]: Maedche, A.;Staab, S.;Stojanovic, N.;Studer, R.; Sure, Y., (2002). SEmantic portAL - The SEAL approach. In Creating the Semantic Web. D. Fensel, J. Hendler, H. Lieberman, W. Wahlster (eds.) MIT Press, Cambridge, MA.

[Maguire, 2001]: Maguire, M.(2001). Methods to support human-centred design. In International Journal of Human-Computer Interaction, 55, 587-634.

[Martin et al, 2005]: D., Martin;M., Paolucci;S., McIlraith;M., Burstein;D., McDermott;D., McGuinness;B., Parsia;T,, Payne;M., Sabou;M., Solanki;N., Srinivasan;K., Sycara, (2005). Bringing Semantics to Web Services: The OWL-S Approach. In 1st International Workshop on Semantic Web Services and Web Process Composition (SWSWPC 2004) 6-9, 2004, San Diego, California, USA.

[Masuwa-Morgan & Burrell, 2004]: Masuwa-Morgan, K. R.;Burrell, P.(2004). Justification of the need for an ontology for accessibility requirements (Theoretic framework). In Interacting with Computers, Volume 16, Issue 3, Universal Usability Revisited, June 2004, Pages 523-555.

[Mathiassen, 1991]: Mathiassen, L.;Munk-Madsen, A.;Nielsen P. A.;Stage, J.(1991). Soft Systems in Software Design. In Systems Thinking in Europe, M. C. Jackson et al. (Eds.) Plenum Press, 1991.

[Maturana & Varela, 1980]: Maturana, H.;Varela, F., Autopoiesis and Cognition: The Realization of the Living,(1980). In Reidel, Dordrecht.

[Maturana & Varela, 1987]: MATURANA, H.R.; Varela, F.J., The Tree of Knowledge:The Biological roots of Human Understanding,(1987). In Shamhala Press,Boston.

[Maturana, 1981]: Maturana, H. M.(1981). Autopoiesis. In Zeleny, M. (ed.), Autopoiesis: A Theory of Living Organization, Elsevier North-Holland, New York.

[Meister et al, 2000]: Meister F.;Patel J.;Fenner J.(2000). E-commerce Platforms Mature. In Informationweek (Issue 809), 99–108.

[Mingers, 1997]: Mingers, J(1997). Systems Typologies in the Light of Autopoiesis: A Reconceptualization of Boulding'sHierarchy, and a Typology of Self-Referential Systems. In Systems Research and Behavioural Science, Vol. 14,5, pp.303-313.

[Mitchell, 1996]: Mitchell, M.(1996). An Introduction to Genetic Algorithms. In MIT Press,Cambridge, Massachusetts.

[Mohamad, 2002]: Mohamad, Y.; Hammer, S.;Haverkamp, F.;Noeker, M.;Tebarth, H., (2002). First Evaluational Results in the Development of Training by Animated Pedagogical Agents (TAPA). In Proceedings of "Mensch und Computer 2002", pp. 145–154. Teubner (Stuttgart).

[Monohan, 1999]: Monohan J. (1999). Portal puzzle. In Banking Strategies 1999, 75:148–158.

[Murray et al, 2003]: Murray, R.M.; Astrom, K.J.; Boyd, S.P.; Brockett, R.W.; Stein, G., (2003). Future directions in control in an information-rich world. In Control Systems Magazine, IEEE , vol.23, no.2pp. 20- 33, Apr 2003.

[Mylonakis, 2004]: Mylonakis, J.(2004). Can mobile services facilitate commerce? Findings from the Greek telecommunications market . In International Journal of Mobile Communications. Vol. 2, no. 2, pp. 188-198. 2004.

[Nahl, 1996]: Nahl, D.(1996). The User-Centered Revolution: 1970-1995. In Encyclopedia of Microcomputers Volume 19, (New York: Marcel Dekker, Inc.), 1996, pp. 143-199.

[Nelson, 1965]: Nelson, T., (1965). A File Structure for the Complex, the Changing, and the Indeterminate. In ACM National Conference. 1965. pp. 84-100.

[Nelson, 2003] Nelson, T. (2003), Keynote speaking at the fourteenth conference on Hypertext and Hypermedia, 2003,

[Newell, 1980]: A. Newell, Physical Symbol Systems,(1980). In CognitiveScience, 4 (1980) 135-183.

[Newell, 1982]: Newell, A.(1982). The Knowledge Level. In Articial Intelligence Journal. Vol 18. p. 87-127.

[Nielsen, 1993]: Nielsen, J., Usability Engineering,(1993). In Academic Press, Inc., San Diego. 1993.

[Nikolis & Prigogine, 1997]: Nikolis, G.;Prigogine, I., Self-Organization in Nonequilibrium Systems: From Dissipative Structures to Order through Fluctuations. ,(1997). In John Wiley & Sons.

[Norfolk, 1986]: Norfolk, V. A., A Dictionary of Cybernetics,(1986). In The American Society for Cybernetics.

[Norman, 1988]: Norman, D, The Psychology of Everyday Things,(1988). In New York, Basic Books, pp.87-92.

[Noy & McGuinness, 2001]: Noy, N. F.;McGuinness, D. L.(2001), "Ontology Development 101: A Guide to Creating Your First Ontology. In Stanford Knowledge Systems Laboratory Technical Report KSL-01-05, March 2001 .

[Okazaki, 2005]: Okazaki, S(2005). New Perspectives on M-Commerce Research. In Special Issue on Mobile Commerce Research. Journal of Electronic Commerce Research VOLUME 6, NUMBER 3, 2005. p160 - 164.

[Ossenbruggen et al, 1998]: van Ossenbruggen, J., Eliëns, A., and Rutledge, L.(1998). The Role of XML in Open Hypermedia Systems. In In Proceedings of OHS'4 (held at Hypertext'98), pp. 63-67..

[Ossher & Tarr, 2000]: Ossher, H.;Tarr, P., (2000). Multi-Dimensional Separation of Concerns and The Hyperspace Approach. In the Symposium on Software Architectures and Component Technology: The State of the Art in Software Development. Kluwer, 2000.

[Paciello, 2000]: Paciello M., G., Web Accessibility for People with Disabilities,(2000). In CMP Books.

[Parmanto et al, 2005]: Parmanto, B.;Ferrydiansyah, R.;Saptono, A.;Song, L.;Sugiantara, I. W.;Hackett, S., (2005). AcceSS: accessibility through simplification & summarization. In In Proceedings of the 2005 international Cross-Disciplinary Workshop on Web Accessibility (W4a) (Chiba, Japan, May 10 - 10, 2005). W4A '05, vol. 88. ACM Press, New York, NY, 18-25. DOI= http://doi.acm.org/10.1145/1061811.1061815.

[Pérez et al, 2002]: Pérez et al(2002), A survey on ontology tools. In OntoWeb Deliverable 1.3 .

[Perrochon & Kennel, 1995]: Perrochon, L.;Kennel, A., (1995). W3-Access for Blind People. In Symposium on Data Highway, October 1995, Bern..

[Picard, 2000]:Picard, R. W.(2000). Toward computers that recognize and respond to user emotion. In IBM Systems Journal, Volume 39, Nos. 3 & 4, pp. 705–719.

[Preuveneers et al, 2004]: Preuveneers, D.;Bergh, J.;Wagelaar, D.;Georges, A.;Rigole, P.;Clerckx, T.;Berbers, Y.;Coninx, K.;Jonckers, V.;Bosschere, K.(2004). Towards an extensible context ontology for Ambient In-telligence. In Markopoulos, P., Eggen, B., Aarts, E., Crowley, J.L., eds.: SecondEuropean Symposium on Ambient Intelligence. Volume 3295 of LNCS., Eindhoven,The Netherlands, Springer (2004) 148 – 159.

[Pulido et al, 2006]: Pulido, J.;Ruiz, M.;Herrera, R.;Cabello, E.;Legrand, S.;Elliman, D.(2006). Ontology languages for the semantic web: A never completely updated review. In Knowledge-Based Systems, Volume 19, Issue 7, Creative Systems, November 2006, Pages 489-497.

[Rabin & McCathieNevile, 2006]: Rabin, J.;McCathieNevile, C.(2006), Mobile Web Best Practices 1.0. In W3C Proposed Recommendation 2 November 2006 .

[Rayport & Jaworski, 2002]: Rayport, J. F.;Jaworski, B. J., Introduction to E-commerce,(2002). In McGraw Hill. Boston.

[Reeve & Han, 2005]: Reeve, L.;Han, H., (2005). Survey of semantic annotation platforms. In The 2005 ACM Symposium on Applied Computing (Santa Fe, New Mexico, March 13 - 17, 2005). L. M. Liebrock, Ed. SAC '05. ACM Press, New York, NY, 1634-1638. DOI= http://doi.acm.org/10.1145/1066677.1067049.

[Reichheld & Schefter, 2000]: Reichheld, F. F.;Schefter, P.(2000). E-Loyalty: Your Secret Weapon on the Web. In Harvard Business Review (78:4), 2000, pp.105-113.

[Rice, 2004]: Rice, M.(2004). Personalisation of interactive television for visually impaired viewers. In 2nd Cambridge Workshop on Universal Access and Assistive Technology. Fitzwilliam College , Cambridge, 22-24 March 2004.

[Roman et al, 2000]: Roman, G.-C.;Picco, G.;Murphy, A(2000). Software engineering for mobility: A roadmap.. In In Proceedingsof the 22nd International Conference on Software Engineering.

[Royce, 1970]: Royce, W. W. 1987. Managing the development of large software systems: concepts and techniques. In Proceedings of the 9th international Conference on Software Engineering (Monterey, California, United States). International Conference on Software Engineering. IEEE Computer Society Press, Los Alamitos, CA, 328-338.

[Russell & Norvig, 1995]: Russell, S,;Norvig, P., Artificial Intelligence: A Modern Approach, Prentice Hall, Englewood Cliffs, NJ (1995).

[Sadeh, 2002]: Sadeh, N., M-Commerce: Technologies, Services, and Business Models,(2002). In New York: John Wiley & Sons, 2002.

[Saha, 1999]: Saha, A.(1999), Application Framework for e-business: Portals. In IBM DeveloperWorks .

[Scott, 2002]: Scott, R., Organizations, Rational, Natural and Open Systems,(2002). In Prentice Hall, 2002.

[Seeman, 2004]: Seeman, L., (2004). The Semantic Web, Web Accessibility, and Device Independence. In The ACM International Cross-Disciplinary Workshop on Web Accessibility (2004), pp. 67–73.

[Shadbolt et al, 2006]: Shadbolt, N.;Berners-Lee, T.;Hall, W.(2006). The Semantic Web Revisited. In IEEE Intelligent Systems 21(3) pp. 96-101, May/June 2006.

[Shapiro & Niederhauser, 2004]: Shapiro, A.;Niederhauser, D., Learning from hypertext: Research issues and findings,(2004). In Handbook of researxh on educational communications and technology. 2nd Edition. Edited by David H. Jonassen, Chapter 23.

[Smit et al]: Smit, B.;Burton, I.; Klein, R.J.T., Street, R. (1999). The Science of Adaptation: A Framework for Assessment, 1999

Smith, 2003: Smith, B., Ontology and Information Systems, 2003

[Smith, 2004]: Smith, M. A.(2004). Portals: Towards an application framework for interoperability. In Communications of the ACM, October 2004/Vol. 47, No. 10.

[Smyth & Cotter, 2003]: Smyth, B.; Cotter, P., (2003). Intelligent Navigation for Mobile Internet Portals. In In Proceedings of 2003 AI Moves to IA: Workshop on Artificial Intelligence, Information Access, and Mobile Computing .

[Souchon & Vanderdonckt, 2003]: Souchon, N.;Vanderdonckt, J.(2003). A review of xml-compliant user interface description languages. In Conference on Design, Specification, and Verification of Interactive Systems. Volume 2844 of Lecture Notes in Computer Science., Springer (2003) 377–391.

[Spiro et al, 1992]: Spiro, R.;Feltovitch, P.;Jacobson, M.;Coulson, R., Cognitive flexibility, constructivism, and hypertext: Random access instruction for advanced knowledge acquisition domains,(1992). In Hand.

[Spyrou et al, 2003]: Spyrou, T;Vlachogiannis, E.;Arnellos, A.;Darzentas, J, (2003). Towards a framework for creating design support environments for adaptive systems. In HCII2003 , Crete, Greece.

[Srinivasan et al, 2002]: Srinivasan, S.S.;Anderson, R.;Ponnavolu, K.(2002). Customer loyalty in e-commerce: an exploration of its antecedents and consequences. In ournal of Retailing, 78, 41–50.

[Steels, 1993]: Steels, L.(1993). The articial life roots of articial intelligence. In Articial Life Journal, Vol 1,1. MIT Press, Cambridge..

[Steels, 1996]: Steels, L.(1996). The origins of intelligence. In In Proceedings of the Carlo Erba Foundation Meeting on Artificial Life. Berlin: Springer-Verlag.

[Stephanidis, 2001]: Stephanidis, C.(2001). Adaptive techniques for Universal Access. In User modeling and User-adapted Interaction 11 159-179.

[Stojanovic et al, 2003]: Stojanovic, N.;Gonzalez, J.;Stojanovic, L., (2003). Ontologer – A system for usage-driven management of ontology-based information portals. In L-CAP '03 Conf., 2003.

[Stolze, 2002]: Stolze M.(2002), Domain-Oriented Recommender Applications: A Framework for Intimate Recommending. In Proceedings Adaptive Hypermedia WOrkshop on Recommendation and Personalization in eCommerce, F. Ricci & B. Smith (Eds.), University of Malaga, Computer Science, Technical Report, pp. 124-131 .

[Strang & Linnhoff–Popien, 2004]: Strang, T.;Linnhoff–Popien, C., (2004). A Context Modeling Survey. In Workshop on Advanced Context Modellin;g, Reasoning and Management associated with the Sixth International Conference on Ubiquitous Computing (UbiComp 2004), Nottingham/England, September. (2004).

[Strauss, 2001]: Strauss, H, (2001). A homepage doth not a portal . In EDUCAUSE 2001.

[Strobbe et al, 2006]: Strobbe, C.;Herramhof, S.;Vlachogiannis, E.;Koch, J.;Velasco, C A, (2006). The BenToWeb XHTML 1.0 Test Suite for the Web Content Accessibility Guidelines 2.0. In Lecture Notes in Computer Science, Volume 4061, Jul 2006, Pages 172 - 175, DOI 10.1007/11788713_26.

[Sure et al, 2004]: Sure, Y.;Staab, S.;Studer, R., On-To-Knowledge Methodology (OTKM),(2004). In Handbookon Ontologies, Springer, pp. 117-132.

[Szyperski, 1998]: Szyperski, C., Component Software: Beyond Object-Oriented Programming,(1998). In Addison-Wesley, 1998.

[Tarasewich, 2003]: Tarasewich, P.(2003). Wireless devices for mobile commerce: User interface design and us-ability. In Mennecke, B.E., and Strader T.J. (eds.) Mobile Commerce – Technology, The-ory and Applications, Hershey: Idea Group Publishing, 26-50.

[Tedesco et al, 2006]: Tedesco, R.;Dolog, P;Nejdl, W.;Allert, H., (2006). Distributed Bayesian Networks for User Modeling. In ELearn'2006: World Conference on E-Learning in Corporate, Government, Health Care, and Higher Education. Hawai, USA, October 2006.

[Trewin et al, 2004]: Trewin, S.;Zimmermann, G.;Vanderheiden, G.(2004). Abstract representations as a basisfor usable user interfaces. In Interacting with Computers 16. May 2004, pp.477-506.

[User Agent Profile, 2002]: (2002), User Agent Profile Version 1.1. In 1 (Version 22-Oct-2002). Open Mobile Alliance .

[van Riel et al, 2001]: van Riel, A. C. R.;Liljander, V.;Jurriens, P.(2001). Exploring consumer evaluations of e-services: aportal site. In International Journal of Service IndustryManagement, Vol. 12 No. 4, pp. 359-77.

[Velasco & Mohamad, 2002]: Velasco, C. A. & Mohamad, M., (2002). Web Services and User/Device Profiling for Accessible Internet Services Provision. In Center on Disabilities, Proceedings of CSUN's Seventeenth Annual International Conference "Technology and Persons with Disabilities" (Los Angeles, 2002). Northridge: California State University Northridge.

[Velasco et al, 2003]: Velasco, C.A.;Mohamad, Y.;Gilman, A.S.;Viorres, N.;Vlachogiannis, E.;Arnellos, A.;Darzentas, J. S.(2003). The Need for User Information and its Relationship to Device Profiles. In Journal of Universal Access to Human-Computer Interaction.

[Velleman, 2000]: Velleman, E., (2000). Pre-testing for user-profiles to enhance accessibility of digital cartography for the partially sighted. In Center on Disabilities, Proceedings of CSUN's Fifteenth Annual International Conference "Technology and Persons with Disabilities" (Los Angeles, 2000). Northridge, California State University Northridge.

[Velleman, E. 2000]: Velleman, E., (2000). Pre-testing for user-profiles to enhance accessibility of digital cartography for the partially sighted. In Center on Disabilities, Proceedings of CSUN's Fifteenth Annual International Conference "Technology and Persons with Disabilities" (Los Angeles, 2000). Northridge, California State University Northridge.

[Venkatesh et al, 2003]: Venkatesh, V.;Ramesh, V.;Massey, A. P.(2003). Understanding usability in mobile commerce. In Commun. ACM 46, 12 (Dec. 2003), 53-56. DOI= http://doi.acm.org/10.1145/953460.953488.

[Vertegaal, 2003]: Vertegaal, R.(2003). Introduction. In Communications of the ACM, 46(3), 30-33.

[Vetro, 2004]: Vetro, A. (2004). MPEG-21 digital item adaptation: enabling universal multimedia access. In Multimedia, IEEE, Vol.11, Iss.1, Jan.-March 2004. Pages: 84- 87.

[Viorres et al, 2007]: Viorres, N.;Xenofon, P.;Stavrakis, M.;Vlachogiannis, E.;Koutsabasis, P.;Darzentas, J., (2007). Major Challenges for Open Source Software Adoption and Development. In Proceedings of HCII2007 Conference in Thematic Area of Online Communities and Social Computing.

[Vlachogiannis et al, 2005]: Vlachogiannis, E.;Darzentas, J.S.;Arnellos, A.;Spyrou, T.;Darzentas, J., (2005). The accessibility of web applications: the case of portals and portlets. In HCI International 2005, Las Vegas.

[Vlachogiannis et al, 2007]: Vlachogiannis, E.;Velasco, C. A.;Gappa, H.;Nordbrock, G.;Darzentas, J. S., (2007). Accessibility of Internet Portals in Ambient Intelligent scenarios: re-thinking their design and implementation. In HCII2007 Conference in Thematic Area of Universal Access in Human-Computer Interaction.

[Vlachogiannis et al, 2007 [a]]: Vlachogiannis, E.;Viorres, N.;Darzentas, J. S.(2007). E-commerce and portals: the need for a universal design approach. In (forthcoming) electronic Journal for E-Commerce Tools and Applications (eJETA).

[Volz et al, 2003]: Volz, R.;Oberle, D.;Staab, S.;Motik, B., (2003). A Semantic Web Management System. In Alternate Track Proceedings of the Twelfth International World Wide Web Conference, WWW2003, Budapest, Hungary, 20-24 May 2003. ACM, 2003.

[Wardrip-Fruin], Wardrip-Fruin, N. 2004. What hypertext is. In Proceedings of the Fifteenth ACM Conference on Hypertext and Hypermedia (Santa Cruz, CA, USA, August 09 - 13, 2004). HYPERTEXT '04. ACM, New York, NY, 126-127. DOI= http://doi.acm.org/10.1145/1012807.1012844

[Wheeler, 2003]: D. A., Wheeler (2003). Why Open Source Software / Free Software (OSS/FS, FLOSS, or FOSS)? Look at the Numbers!. In http://www.dwheeler.com/oss_fs_why.html (Revised as of April 16, 2007).

[WIKIPEDIA] , The Wikipedia Encyclopedia, 2006, http://www.wikipedia.org

[Winkler, 2006] Winkler, R., Portals – The All-In-One Web Supersites: Features, Functions, Definitions, Taxonomy, 2006, http://www.sapdesignguild.org/editions/edition3/portal_definition.asp

[Xu, 2000]: Xu L.(2000). The contribution of systems science toinformation systems research. In Systems Research and Behavioral Science 17: 105–116.

[Yang et al, 2004]: Yang, A.;Ro, Y.;Nam, J.;Hong, J.,Choi, S.;Lee, J.(2004). Improving Visual Accessibility for Color Vision Deficiency Based on MPEG-21. In ETRI Journal, vol.26, no.3, June 2004, pp.195-202.

[Zajicek & Edwards, 2004]: Zajicek, M.;Edwards, A.(2004). Universal usability revisited. In Interacting with Computers, Volume 16, Issue 3, Universal Usability Revisited, June 2004, Pages 403-410.

[Ziemke, 1998]: Ziemke, T. (1998). Adaptive Behavior in Autonomous Agents, In Agents, Adaptive Behaviors and Distributed Simulations ' journal.

[Zimmermann et al, 2002]: Zimmermann, G.;Vanderheiden, G.;Gilman, A. S.(2002). Remote Console - Prototyping for the Alternate Interface Access Standard. In N. Carbonell and C. Stephanidis (eds.), Universal Access: Theoretical perspectives, practice and experience - 7th ERCIM UI4ALL Workshop, October 2002, Paris, France - Selected Papers. Lecture Notes in Computer Science, Springer-Verlag.

Appendix I: Interaction Profile Ontology

This section presents the ontologies that consist the Interaction Profile.

Abstract Interaction Profile

```
<?xml version="1.0"?>
<rdf:RDF

xmlns:servInterProf="http://www.syros.aegean.gr/users/evlach/refs/ServiceInteractionPr
ofile.owl#"

xmlns:contxInterProf="http://www.syros.aegean.gr/users/evlach/refs/DeliveryContextInte
ractionProfile.owl#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns="http://www.syros.aegean.gr/users/evlach/refs/InteractionProfile.owl#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"

xmlns:userInterProf="http://www.syros.aegean.gr/users/evlach/refs/UserInteractionProfi
le.owl#"

xmlns:platfInterProf="http://www.syros.aegean.gr/users/evlach/refs/PlatformInteraction
Profile.owl#"
  xml:base="http://www.syros.aegean.gr/users/evlach/refs/InteractionProfile.owl">
  <owl:Ontology rdf:about="">
    <owl:imports
rdf:resource="http://www.syros.aegean.gr/users/evlach/refs/UserInteractionProfile.owl"
/>
```

```
    <owl:imports
rdf:resource="http://www.syros.aegean.gr/users/evlach/refs/ServiceInteractionProfile.o
wl"/>
    <owl:imports
rdf:resource="http://www.syros.aegean.gr/users/evlach/refs/PlatformInteractionProfile.
owl"/>
    <owl:imports
rdf:resource="http://www.syros.aegean.gr/users/evlach/refs/DeliveryContextInteractionP
rofile.owl"/>
  </owl:Ontology>
  <owl:Class rdf:ID="ServiceInteractionProfile">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="InteractionProfile"/>
    </rdfs:subClassOf>
    <owl:equivalentClass
rdf:resource="http://www.syros.aegean.gr/users/evlach/refs/ServiceInteractionProfile.o
wl#ServiceInteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="DeliveryContextInteractionProfile">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#InteractionProfile"/>
    </rdfs:subClassOf>
    <owl:equivalentClass
rdf:resource="http://www.syros.aegean.gr/users/evlach/refs/DeliveryContextInteractionP
rofile.owl#DeliveryContextInteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="UserInteractionProfile">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#InteractionProfile"/>
    </rdfs:subClassOf>
    <owl:equivalentClass
rdf:resource="http://www.syros.aegean.gr/users/evlach/refs/UserInteractionProfile.owl#
UserInteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:about="#InteractionProfile">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:DatatypeProperty rdf:ID="frequencyRange"/>
        </owl:onProperty>
        <owl:maxCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >1</owl:maxCardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
```

```
        <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >0</owl:minCardinality>
        <owl:onProperty>
          <owl:DatatypeProperty rdf:about="#frequencyRange"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
  </owl:Class>
  <owl:Class rdf:ID="PlatformInteractionProfile">
    <owl:equivalentClass
rdf:resource="http://www.syros.aegean.gr/users/evlach/refs/PlatformInteractionProfile.
owl#PlatformInteractionProfile"/>
    <rdfs:subClassOf rdf:resource="#InteractionProfile"/>
  </owl:Class>
  <owl:DatatypeProperty rdf:about="#frequencyRange">
    <rdfs:domain rdf:resource="#InteractionProfile"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
  </owl:DatatypeProperty>
</rdf:RDF>
```

User Interaction Profile

```
<?xml version="1.0"?>
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns="http://www.syros.aegean.gr/users/evlach/refs/UserInteractionProfile.owl#"
  xml:base="http://www.syros.aegean.gr/users/evlach/refs/UserInteractionProfile.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Eyesight">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Ability"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >This includes people with no vision, or some functional vision</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Person">
```

```
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Basic User Information like name, date of birth, e-mail plus additional 
demographical data of interest.</rdfs:comment>
    <rdfs:subClassOf>
      <owl:Class rdf:ID="PersonalInfo"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="IndividuaTraits">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="UserInteractionProfile"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >This is a feature that is either stable or change rarely. These features define 
the user as an individual and include like personality factors (e.g. 
introvert/extravert), cognitive factors and learning styles.</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Profession">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Background"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >The userβ€™s profession</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Contact">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#PersonalInfo"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Other persons, with whom the person is related, including relatives, friends, co-
workers.</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Mobility">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Ability"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >This refers to a wide range of people with varying types of physical 
disabilities</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Hearing">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >This includes people who are completely deaf or have partial hearing in one or 
both ears and require the use of a hearing aid</rdfs:comment>
    <rdfs:subClassOf>
```

```
      <owl:Class rdf:about="#Ability"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Expertise">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Includes all kinds of expertise, like computer expertise</rdfs:comment>
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Background"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Activity">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Background"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >User activities, hobby or work related. For example, collects stamps or
investigates the 4th Crusade</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Preference">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >User preferences, for example β€□ loves catsβ€□ , β€□ likes blue colorβ€□  or
β€□ dislikes classical musicβ€□ </rdfs:comment>
    <rdfs:subClassOf rdf:resource="#IndividuaTraits"/>
  </owl:Class>
  <owl:Class rdf:ID="Characteristic">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#PersonalInfo"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >General user characteristics, like eye color, height, weight, etc.</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:about="#PersonalInfo">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Contact and user-management information, plus additional demographical data of
interest (based on Golemati et al, 2007)</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#UserInteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="Cognitive">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Cognitive impairment refers to people with dyslexia and learning
difficulties</rdfs:comment>
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Ability"/>
```

```
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="LivingConditions">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Information relevant to the userβ€™s place of residence and house
type.</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#PersonalInfo"/>
  </owl:Class>
  <owl:Class rdf:ID="Knowledge">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >An adaptive WIS system which relies on user's knowledge has to recognize the
changes in the user's knowledge state and update the user model
accordingly.</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#UserInteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="Personality">
    <rdfs:subClassOf rdf:resource="#IndividuaTraits"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >(e.g. introvert/extravert)</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:about="#Background">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >This is defined as all the information related to the user's  previous experience
outside the subject of the adaptive system, which is relevant  enough to be
considered. This includes the user's profession, experience of work  in related areas,
as well as the user's point of view and perspective.</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#UserInteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="Thing">
    <rdfs:subClassOf rdf:resource="#PersonalInfo"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Living things or Non Living Things the user may posses or otherwise be related
to, like a car, a house, a book or a pet</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="LearningStyle">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Learning style of the user</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#IndividuaTraits"/>
  </owl:Class>
  <owl:Class rdf:ID="Interest">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >User hobby or work-related interests. For example, β€ interested in sportsβ€ ,
β€ interested in cookingβ€ </rdfs:comment>
    <rdfs:subClassOf rdf:resource="#IndividuaTraits"/>
```

```
  </owl:Class>
  <owl:Class rdf:about="#Ability">
    <rdfs:subClassOf rdf:resource="#UserInteractionProfile"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >User abilities and disabilities (Based on RNIB classification. You may also look
to http://bentoweb.org/refs/TCDL2.0-20070711.html disabilities list)</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Education">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >User education issues, including for example university diplomas and
languages</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#Background"/>
  </owl:Class>
</rdf:RDF>
```

Service Interaction Profile

```
<?xml version="1.0"?>
<rdf:RDF

xmlns="http://www.syros.aegean.gr/users/evlach/refs/ServiceInteractionProfile.owl#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"

xml:base="http://www.syros.aegean.gr/users/evlach/refs/ServiceInteractionProfile.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="FuncionalSpec">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="ServiceInteractionProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="InteractionSpec">
    <rdfs:subClassOf rdf:resource="#ServiceInteractionProfile"/>
  </owl:Class>
</rdf:RDF>
```

Delivery Context Interaction Profile

```
<?xml version="1.0"?>
<rdf:RDF

xmlns="http://www.syros.aegean.gr/users/evlach/refs/DeliveryContextInteractionProfile.
owl#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"

xml:base="http://www.syros.aegean.gr/users/evlach/refs/DeliveryContextInteractionProfi
le.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Goals">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="DeliveryContextInteractionProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Location">
    <rdfs:subClassOf rdf:resource="#DeliveryContextInteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="Experience">
    <rdfs:subClassOf rdf:resource="#DeliveryContextInteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="DeliveryInterest">
    <rdfs:subClassOf rdf:resource="#DeliveryContextInteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="Preferences">
    <rdfs:subClassOf rdf:resource="#DeliveryContextInteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="OutputPrefs">
    <rdfs:subClassOf rdf:resource="#Preferences"/>
  </owl:Class>
  <owl:Class rdf:ID="InteractionPrefs">
    <rdfs:subClassOf rdf:resource="#Preferences"/>
  </owl:Class>
  <owl:Class rdf:ID="PhysicalState">
    <rdfs:subClassOf rdf:resource="#DeliveryContextInteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="InputPrefs">
```

```
    <rdfs:subClassOf rdf:resource="#Preferences"/>
  </owl:Class>
</rdf:RDF>
```

Platform Interaction Profile

```
<?xml version="1.0"?>
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns="http://www.syros.aegean.gr/users/evlach/refs/InteractionProfile.owl#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.syros.aegean.gr/users/evlach/refs/InteractionProfile.owl">
  <owl:Ontology rdf:about="">
    <rdfs:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Interaction Profile</rdfs:label>
  </owl:Ontology>
  <owl:Class rdf:ID="Hearing">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Ability"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >This includes people who are completely deaf or have partial hearing in one or
both ears and require the use of a hearing aid</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Location">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="DeliveryContextProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="InputPrefs">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Preferences"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Activity">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Background"/>
    </rdfs:subClassOf>
```

```
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >User activities, hobby or work related. For example, collects stamps or
investigates the 4th Crusade</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Hardware">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="PlatformProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="IndividuaTraits">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="UserProfile"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >This is a feature that is either stable or change rarely. These features define
the user as an individual and include like personality factors (e.g.
introvert/extravert), cognitive factors and learning styles.</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Mobility">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >This refers to a wide range of people with varying types of physical
disabilities</rdfs:comment>
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Ability"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="PhysicalState">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#DeliveryContextProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="InteractionPrefs">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Preferences"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="PersonalInfo">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#UserProfile"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Contact and user-management information, plus additional demographical data of
interest (based on Golemati et al, 2007)</rdfs:comment>
```

```
  </owl:Class>
  <owl:Class rdf:ID="Expertise">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Includes all kinds of expertise, like computer expertise</rdfs:comment>
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Background"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="InteractionSpecs">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="ServiceProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Ability">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#UserProfile"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >User abilities and disabilities (Based on RNIB classification. You may also look
to http://bentoweb.org/refs/TCDL2.0-20070711.html disabilities list)</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="DeliveryInterest">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#DeliveryContextProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Background">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#UserProfile"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >This is defined as all the information related to the user's  previous experience
outside the subject of the adaptive system, which is relevant  enough to be
considered. This includes the user's profession, experience of work  in related areas,
as well as the user's point of view and perspective.</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Preference">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >User preferences, for example β€ loves catsβ€ , β€ likes blue colorβ€  or
β€ dislikes classical musicβ€ </rdfs:comment>
    <rdfs:subClassOf rdf:resource="#IndividuaTraits"/>
  </owl:Class>
  <owl:Class rdf:ID="Network">
```

```
    <rdfs:subClassOf>
      <owl:Class rdf:about="#PlatformProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="FunctionalSpecs">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#ServiceProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Software">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#PlatformProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Experience">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#DeliveryContextProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="LearningStyle">
    <rdfs:subClassOf rdf:resource="#IndividuaTraits"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Learning style of the user</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Education">
    <rdfs:subClassOf rdf:resource="#Background"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >User education issues, including for example university diplomas and
languages</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:about="#DeliveryContextProfile">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="InteractionProfile"/>
    </rdfs:subClassOf>
    <owl:disjointWith>
      <owl:Class rdf:about="#UserProfile"/>
    </owl:disjointWith>
    <owl:disjointWith>
      <owl:Class rdf:about="#PlatformProfile"/>
    </owl:disjointWith>
    <owl:disjointWith>
      <owl:Class rdf:about="#ServiceProfile"/>
```

```
    </owl:disjointWith>
  </owl:Class>
  <owl:Class rdf:ID="Firewall">
    <rdfs:subClassOf rdf:resource="#Network"/>
  </owl:Class>
  <owl:Class rdf:ID="DeviceAssistiveTechnology">
    <rdfs:subClassOf rdf:resource="#Hardware"/>
  </owl:Class>
  <owl:Class rdf:about="#PlatformProfile">
    <owl:disjointWith>
      <owl:Class rdf:about="#UserProfile"/>
    </owl:disjointWith>
    <owl:disjointWith>
      <owl:Class rdf:about="#ServiceProfile"/>
    </owl:disjointWith>
    <owl:disjointWith rdf:resource="#DeliveryContextProfile"/>
    <rdfs:subClassOf>
      <owl:Class rdf:about="#InteractionProfile"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Characteristic">
    <rdfs:subClassOf rdf:resource="#PersonalInfo"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >General user characteristics, like eye color, height, weight, etc.</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:about="#Preferences">
    <rdfs:subClassOf rdf:resource="#DeliveryContextProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="LivingConditions">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Information relevant to the userβ€™s place of residence and house
type.</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#PersonalInfo"/>
  </owl:Class>
  <owl:Class rdf:ID="OutputPrefs">
    <rdfs:subClassOf rdf:resource="#Preferences"/>
  </owl:Class>
  <owl:Class rdf:ID="Personality">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >(e.g. introvert/extravert)</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#IndividuaTraits"/>
  </owl:Class>
```

```
  <owl:Class rdf:ID="Contact">
    <rdfs:subClassOf rdf:resource="#PersonalInfo"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Other persons, with whom the person is related, including relatives, friends, co-workers.</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:about="#InteractionProfile">
    <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:DatatypeProperty rdf:ID="frequencyRange"/>
        </owl:onProperty>
        <owl:minCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >0</owl:minCardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:DatatypeProperty rdf:about="#frequencyRange"/>
        </owl:onProperty>
        <owl:maxCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >1</owl:maxCardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Knowledge">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#UserProfile"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >An adaptive WIS system which relies on user's knowledge has to recognize the changes in the user's knowledge state and update the user model accordingly.</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:about="#ServiceProfile">
    <owl:disjointWith>
      <owl:Class rdf:about="#UserProfile"/>
    </owl:disjointWith>
    <owl:disjointWith rdf:resource="#PlatformProfile"/>
    <owl:disjointWith rdf:resource="#DeliveryContextProfile"/>
    <rdfs:subClassOf rdf:resource="#InteractionProfile"/>
```

```
  </owl:Class>
  <owl:Class rdf:about="#UserProfile">
    <owl:disjointWith rdf:resource="#PlatformProfile"/>
    <owl:disjointWith rdf:resource="#ServiceProfile"/>
    <owl:disjointWith rdf:resource="#DeliveryContextProfile"/>
    <rdfs:subClassOf rdf:resource="#InteractionProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="Profession">
    <rdfs:subClassOf rdf:resource="#Background"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >The userβ€™s profession</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Interest">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >User hobby or work-related interests. For example, β€ interested in sportsβ€ ,
β€ interested in cookingβ€ </rdfs:comment>
    <rdfs:subClassOf rdf:resource="#IndividuaTraits"/>
  </owl:Class>
  <owl:Class rdf:ID="UserAgent">
    <rdfs:subClassOf rdf:resource="#Software"/>
  </owl:Class>
  <owl:Class rdf:ID="Proxy">
    <rdfs:subClassOf rdf:resource="#Network"/>
  </owl:Class>
  <owl:Class rdf:ID="Cognitive">
    <rdfs:subClassOf rdf:resource="#Ability"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Cognitive impairment refers to people with dyslexia and learning
difficulties</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="NetworkChannel">
    <rdfs:subClassOf rdf:resource="#Network"/>
  </owl:Class>
  <owl:Class rdf:ID="Goals">
    <rdfs:subClassOf rdf:resource="#DeliveryContextProfile"/>
  </owl:Class>
  <owl:Class rdf:ID="Eyesight">
    <rdfs:subClassOf rdf:resource="#Ability"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >This includes people with no vision, or some functional vision</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Thing">
```

```
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Living things or Non Living Things the user may posses or otherwise be related
to, like a car, a house, a book or a pet</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#PersonalInfo"/>
  </owl:Class>
  <owl:Class rdf:ID="SoftAssistiveTechnology">
    <rdfs:subClassOf rdf:resource="#Software"/>
  </owl:Class>
  <owl:Class rdf:ID="Person">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Basic User Information like name, date of birth, e-mail plus additional
demographical data of interest.</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#PersonalInfo"/>
  </owl:Class>
  <owl:DatatypeProperty rdf:about="#frequencyRange">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
    <rdfs:domain rdf:resource="#InteractionProfile"/>
  </owl:DatatypeProperty>
</rdf:RDF>
```

Graphical overall Representation

Appendix II: iTV XML Profile Sample

The XML document below is a sample profile used for the iTV.


```xml
<?xml version="1.0" encoding="UTF-8"?>
<n:InteractionProfile
	xmlns:n="http://www.syros.aegean.gr/users/evlach/refs/dawis/InteractionProfile.xsd"
	xmlns:delivery="http://www.syros.aegean.gr/users/evlach/refs/dawis/DeliveryContextInteractionProfile.xsd"
	xmlns:platform="http://www.syros.aegean.gr/users/evlach/refs/dawis/PlatformInteractionProfile.xsd"
	xmlns:service="http://www.syros.aegean.gr/users/evlach/refs/dawis/ServiceInteractionProfile.xsd"
	xmlns:user="http://www.syros.aegean.gr/users/evlach/refs/dawis/UserInteractionProfile.xsd"
	xmlns:types="http://www.syros.aegean.gr/users/evlach/refs/dawis/datatypes.xsd"
	xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
	xsi:schemaLocation="http://www.syros.aegean.gr/users/evlach/refs/dawis/InteractionProfile.xsd
../../design/user_profile/InteractionProfile/InteractionProfile.xsd">
	<n:UserInteractionProfile>
		<user:Background>
			<user:Language types:native="true" types:level="5">en-us</user:Language>
		</user:Background>
		<user:Ability>
			<user:Disability>blindness</user:Disability>
			<user:Disability>low vision</user:Disability>
		</user:Ability>
		<user:IndividialTraits />
		<user:PersonalInfo>
			<user:Person>
				<user:FirstName>SomeFirstname</user:FirstName>
				<user:LastName>SomeLastname</user:LastName>
```

```
                <user:Email>somebody@some.com</user:Email>
            </user:Person>
            <user:Characteristic>
                <user:Age>2</user:Age>
                <user:Sex>male</user:Sex>
            </user:Characteristic>
            <user:Thing />
            <user:Contact />
        </user:PersonalInfo>
        <user:Knowledge />
    </n:UserInteractionProfile>
    <n:PlatformInteractionProfile>
        <platform:Hardware>
            <platform:AssistiveTechnology platform:minimumLevel="0"
            platform:type="alternative input devices" />
            <platform:Device platform:minimumLevel="0"
                platform:type="PC"/>
        </platform:Hardware>
        <platform:Software>
            <platform:AssistiveTechnology platform:minimumLevel="0"
                platform:type="screenreader" platform:version="1.0"
                platform:product="JAWS" />
            <platform:UserAgent platform:minimumLevel="0"
                platform:type="browser" platform:version="1.5"
                platform:product="Firefox" />
        </platform:Software>
        <platform:Network>
            <platform:AvailableBandwidth platform:maximum="2000000"
                platform:average="1000000" platform:minimum="400000" />
        </platform:Network>
    </n:PlatformInteractionProfile>
    <n:ServiceInteractionProfile>
        <service:FunctionalSpec />
        <service:InteractionSpec />
    </n:ServiceInteractionProfile>
    <n:DeliveryContextInteractionProfile>
        <delivery:Location />
        <delivery:Experience />
        <delivery:DeliveryInterest />
        <delivery:PhysicalState />
        <delivery:Preferences>
```

```
                    <delivery:InteractionPrefs>
                          <rate>5</rate>
                    </delivery:InteractionPrefs>
                    <delivery:OutputPrefs>
                          <delivery:Subtitles delivery:level="2" />
                          <delivery:AudioDescription delivery:level="0" />
                    </delivery:OutputPrefs>
                    <delivery:InputPrefs />
              </delivery:Preferences>
              <delivery:Goals />
        </n:DeliveryContextInteractionProfile>
</n:InteractionProfile>
```

Appendix III: Test Case Description Language Sample

This is test sample from the set of XHTML + CSS test cases mapped to the 27 April 2006 Last Call Working Draft of WCAG 2.0. This refers to:

- Guideline 1.2: Provide synchronized alternatives for multimedia
- Success criterion 1.2.2: Audio descriptions of video, or a full multimedia text alternative including any interaction, are provided for prerecorded multimedia.

```
<?xml version="1.0" encoding="UTF-8"?>
<testCaseDescription xmlns:html="http://www.w3.org/1999/xhtml"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:dc="http://purl.org/dc/elements/1.1/" id="sc1.2.2_l1_002" xml:lang="en"
xsi:schemaLocation="http://bentoweb.org/refs/TCDL1.1
http://bentoweb.org/refs/schemas/tcdl1.1.xsd http://purl.org/dc/elements/1.1/
http://dublincore.org/schemas/xmls/simpledc20021212.xsd http://www.w3.org/1999/xhtml
http://www.w3.org/2004/07/xhtml/xhtml1-strict.xsd http://www.w3.org/1999/xlink
http://bentoweb.org/refs/schemas/xlink.xsd" xmlns="http://bentoweb.org/refs/TCDL1.1">
  <!--
  Copyright &#169; BenToWeb Consortium 2004-2007

  Licensed under the BenToWeb License, Version 1.0 (the "License"); you may
  not use this file except in compliance with the License. You may obtain
  a copy of the License at http://bentoweb.org/refs/LICENSE.html

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
```

```
-->
  <formalMetadata>
    <description>Document containing a video <html:code>object</html:code> with audio
description.</description>
    <title>A video with transcript.</title>
    <dc:creator>evlach@aegean.gr</dc:creator>
    <dc:language>en</dc:language>
    <dc:rights>Copyright BenToWeb</dc:rights>
    <date>2005-08-28</date>
    <status>accepted for end user evaluation</status>
  </formalMetadata>
  <technology>
    <recommendation xlink:href="http://www.w3.org/TR/xhtml1/">
      <label>
        <html:acronym>XHTML</html:acronym>™ 1.0 The Extensible HyperText Markup
Language (Second Edition)</label>
      <testElements>
        <testElement>
          <elementName localname="object" namespace="http://www.w3.org/1999/xhtml" />
          <specReference xlink:href="http://www.w3.org/TR/1999/REC-html401-
19991224/struct/objects.html#h-13.3.1">Rules for rendering objects</specReference>
        </testElement>
      </testElements>
    </recommendation>
  </technology>
  <testCase complexity="atomic">
    <purpose>
      <p>The test case is intended to pass as there is audio description
<html:code>object</html:code>.</p>
    </purpose>
    <requiredTests>
      <testModes>
        <testMode>oneExpert</testMode>
      </testModes>
      <scenario id="s01" name="Scenario s01 blind speech">
        <userGuidance xml:lang="en">
          <p>Make sure you have the sound on your computer turned on.</p>
        </userGuidance>
        <userGuidance xml:lang="nl">
          <p>Zorg ervoor dat u het geluid op uw computer hebt aanstaan.</p>
        </userGuidance>
        <questions>
          <yesNoQuestion>
```

```xml
        <questionText xml:lang="en">
          <p>Did you hear an audio description of the video?</p>
        </questionText>
        <questionText xml:lang="nl">
          <p>Hoorde u een audiobeschrijving van de video ?</p>
        </questionText>
        <optionYes value="+1" />
        <optionNo value="-1" />
        <optionOther xml:lang="en"><p /></optionOther>
        <optionOther xml:lang="nl"><p /></optionOther>
      </yesNoQuestion>
    </questions>
    <experience>
      <AssistiveTechnology minimumLevel="2" type="screenreader" />
      <UserAgent minimumLevel="2" type="browser" />
    </experience>
    <disabilities>
      <disability>blindness</disability>
      <disability>low vision</disability>
    </disabilities>
  </scenario>
  <scenario id="s02" name="Scenario s02 blind braille">
    <userGuidance xml:lang="en">
      <p>Make sure you have the sound on your computer turned on.</p>
    </userGuidance>
    <userGuidance xml:lang="nl">
      <p>Zorg ervoor dat u het geluid op uw computer hebt aanstaan.</p>
    </userGuidance>
    <questions>
      <yesNoQuestion>
        <questionText xml:lang="en">
          <p>Did you hear an audio description of the video?</p>
        </questionText>
        <questionText xml:lang="nl">
          <p>Hoorde u een audiobeschrijving van de video ?</p>
        </questionText>
        <optionYes value="+1" />
        <optionNo value="-1" />
        <optionOther xml:lang="en"><p /></optionOther>
        <optionOther xml:lang="nl"><p /></optionOther>
      </yesNoQuestion>
```

```
        </questions>
        <experience>
          <AssistiveTechnology minimumLevel="2" type="Braille display" />
          <UserAgent minimumLevel="2" type="browser" />
        </experience>
        <disabilities>
          <disability>blindness</disability>
          <disability>low vision</disability>
        </disabilities>
      </scenario>
      <scenario id="s03" name="Scenario s02 lowvis mag">
        <userGuidance xml:lang="en">
          <p>Make sure you have the sound on your computer turned on.</p>
        </userGuidance>
        <userGuidance xml:lang="nl">
          <p>Zorg ervoor dat u het geluid op uw computer hebt aanstaan.</p>
        </userGuidance>
        <questions>
          <yesNoQuestion>
            <questionText xml:lang="en">
              <p>Did you hear an audio description of the video?</p>
            </questionText>
            <questionText xml:lang="nl">
              <p>Hoorde u een audiobeschrijving van de video ?</p>
            </questionText>
            <optionYes value="+1" />
            <optionNo value="-1" />
            <optionOther xml:lang="en"><p /></optionOther>
            <optionOther xml:lang="nl"><p /></optionOther>
          </yesNoQuestion>
        </questions>
        <experience>
          <AssistiveTechnology minimumLevel="2" type="screenreader with magnification"
/>
          <UserAgent minimumLevel="2" type="browser" />
        </experience>
        <disabilities>
          <disability>blindness</disability>
          <disability>low vision</disability>
        </disabilities>
      </scenario>
    </requiredTests>
```

```
    <files>
      <file xlink:href="../testfiles/sc1.2.2_l1_002.html" hrefLang="en" />
    </files>
  </testCase>
  <rules>
    <rule id="http://bentoweb.org/refs/rulesets.xml#WCAG2_20060427_1.2_media-equiv-
audio-desc" primary="yes">
      <locations expectedResult="pass">
        <location line="12" column="10" xpath="/:html/:body/:p/:div/:object" />
      </locations>
      <functionalOutcome>
        <p>There is audio description.</p>
      </functionalOutcome>
      <techComment>
        <p>The <html:code>object</html:code> element does contain audio
description.</p>
      </techComment>
    </rule>
    <rule id="http://bentoweb.org/refs/rulesets.xml#WCAG2_20050630_1.1_text-equiv-
text-doc" primary="no">
      <locations expectedResult="pass">
        <location line="12" column="10" xpath="/:html/:body/:p/:div/:object" />
      </locations>
      <functionalOutcome>
        <p>There is no link for a descriptive transcript of the video.</p>
      </functionalOutcome>
      <techComment>
        <p>The <html:code>object</html:code> element does not contain a <html:code>a
href</html:code> to a descriptive transcript of the video.</p>
      </techComment>
    </rule>
  </rules>
  <namespaceMappings>
    <namespace nsPrefix="" nsURI="http://www.w3.org/1999/xhtml" />
  </namespaceMappings>
</testCaseDescription>
```

Appendix IV: Bibliography

[C01] Vlachogiannis, E. , Hliadhs Vangelis , Tzefridou Aleksia , Tzimopoulos Nikos (2003). Design and implementation of South Aegean's educational portal - Epyna.gr (Greek). Presented at Panehellinic conference of Secondary education.

[C02] Darzentas J, Arnellos, A, Darzentas, J. S., Koutsabasis P., Spyrou T., Viorres Nikos , Vlachogiannis, E , Velasco C A, Mohamad Y, Abascal J, Jorge Tomas Guerra , Myriam Arrue Recondo , Nikitas Tsopelas , Nikos Floratos (2003). IRIS: An Open Environment that Supports Inclusive Design of Internet Applications. - Ninth IFIP TC13 International Conference on Human-Computer Interaction (Interact2003), Zurich.

[C03] Spyrou T, Vlachogiannis, E, Arnellos, A, Darzentas J (2003). Towards a framework for creating design support environments for adaptive systems. - HCI International 2003 , Crete, Greece

[C04] Vlachogiannis, E., Darzentas, J S, Arnellos, A, Spyrou T, Darzentas J (2005). The accessibility of web applications: the case of portals and portlets - HCI International 2005, Las Vegas

[C05] Herramhof, S, Petrie, H, Strobbe, C, Vlachogiannis, E, Weimann, K, Weber, G, Velasco, C A (2006). Test Case Management Tools for Accessibility Testing. In: Miesenberger K et al. (eds). Proceedings of the 10th International Conference ICCHP 2006 (Linz, Austria, July 2006), LNCS 4061, pp. 215—222. Berlin-Heidelberg: Springer-Verlag. DOI: 10.1007/11788713_32.

[C06] Strobbe C, Herramhof S, Vlachogiannis E, Koch J & Velasco C A (2006). The BenToWeb XHTML 1.0 Test Suite for the Web Content Accessibility Guidelines 2.0. In: Miesenberger K et al. (eds). Proceedings of the 10th International Conference ICCHP 2006 (Linz, Austria, July 2006), LNCS 4061, pp. 172-175. Berlin-Heidelberg: Springer-Verlag. DOI: 10.1007/11788713_26.

[C08] Viorres, N. Papadopoulos, X. Stavrakis, M. Vlachogiannis, E. Koutsabasis, P. Darzentas, J. (2007) Major HCI Challenges for Open Source Software

Adoption and Development, 12th International Conference on Human-Computer Interaction, Beijing, China, 2007, Lecture Notes in Computer Science, Vol. 4564, Springer.

[C09] Strobbe C, Engelen J, Koch J, Velasco C A, Vlachogiannis E, Ortner D (2007). The BenToWeb XHTML 1.0 Test Suite for the Web Content Accessibility Guidelines 2.0 - Last Call Working Draft. In: Stephanidis C (ed). Universal Access in Human-Computer Interaction. Applications and Services (4th International Conference on Universal Access in Human-Computer Interaction, UAHCI 2007, HCI International 2007 Beijing, China, July 22—27, 2007 Proceedings, Part III), LNCS 4556/2007, pp. 160—166. Berlin Heidelberg: Springer-Verlag. DOI: 10.1007/978-3-540-73283-9_19

[C10] Vlachogiannis E, Velasco C A, Gappa H, Nordbrock G, Darzentas J S (2007). Accessibility of Internet Portals in Ambient Intelligent Scenarios: Re-thinking Their Design and Implementation. In: Stephanidis C (ed). Universal Access in Human-Computer Interaction. Ambient Interaction (4th International Conference on Universal Access in Human-Computer Interaction, UAHCI 2007, HCI International 2007 Beijing, China, July 22—27, 2007 Proceedings, Part II), LNCS 4555/2007, pp. 245—253. Berlin Heidelberg: Springer-Verlag. DOI: 10.1007/978-3-540-73281-5_26.

[J01] Velasco, A., Mohamad, Y., Gilman, S., Viorres, N., Vlachogiannis, E., Arnellos, A., and Darzentas, S. 2004. Universal access to information services—the need for user information and its relationship to device profiles. Univers. Access Inf. Soc. 3, 1 (Mar. 2004), 88-95. DOI= http://dx.doi.org/10.1007/s10209-003-0075-5

Appendix V: Περίληψη στην Ελληνική Γλώσσα

Η προβληματική περιοχή της διδακτορικής διατριβής προκύπτει από τις ολοένα αυξανόμενες ανάγκες ενσωμάτωσης των επιτευγμάτων της πληροφορικής και των τηλεπικοινωνιών στην καθημερινότητα του ανθρώπου. Δεδομένου ότι η αυξανόμενη πολυπλοκότητα της αναδυόμενης ψηφιακής κοινωνίας έχει ως αποτέλεσμα τη γνωστική και πληροφοριακή υπερφόρτωση του χρήστη, εξουσιοδοτούμενοι πράκτορες επιστρατεύονται για να προσφέρουν τις υπηρεσίες τους.

Το τεχνολογικό επίτευγμα που αποτελεί το κύριο αίτιο μιας τέτοιας πολυπλοκότητας είναι ο παγκόσμιος Ιστός (WWW), ο οποίος θεωρείται, πλέον, όχι απλά μια υπηρεσία αλλά μια βάση ενός τεράστιου αριθμού υπηρεσιών που προσφέρονται μέσω των Πληροφοριακών Συστημάτων Ιστού (ΠΣΙ). Ο παγκόσμιος Ιστός παρουσιάζει κάποια χαρισματικά χαρακτηριστικά στα οποία οφείλεται η γρήγορη εξάπλωσή του. Τα συγκεκριμένα χαρακτηριστικά κληρονομούνται από τα ΠΣΙ.

Από την άλλη μεριά, τα ΠΣΙ είναι συστήματα που απευθύνονται σε μεγαλύτερο και ποικιλότερο κοινό καθώς και τρόπους πρόσβασης. Κατά τη σχεδίαση τέτοιων πολύπλοκων αλληλεπιδραστικών συστημάτων, η έννοια της «προσαρμοστικότητας», η κεντρική έννοια της διατριβής, αποδεικνύεται να είναι το χαρακτηριστικό/ιδιότητα κλειδί. Η απαίτηση για προσαρμοστικά συστήματα καθίσταται περισσότερο επιτατική στην περίπτωση των ανθρώπων με αναπηρίες (ΑμεΑ) ή/και γενικότερα κατά την αλληλεπίδρασή κάτω από αποκλίνοντα περιβάλλοντα - ανάγκες.

Ακόμη και αν μέχρι σήμερα η έρευνα στα σχετικά επιστημονικά πεδία έχει παράσχει πολλές αρχές, τεχνικές και εργαλεία που καθιστούν δυνατή, σε ένα βαθμό, την προσαρμοστικότητα των πληροφοριακών συστημάτων του παγκόσμιου ιστού, αναγνωρίστηκε η έλλειψη κάποιου πλαισίου αναφοράς το οποίο θα μπορούσε να ενσωματώσει αποδοτικά τα προαναφερθέντα. Ο ρόλος ενός τέτοιου πλαισίου είναι διττός. Πρώτον, να προσφέρει ένα βοηθητικό σχεδιαστικό περιβάλλον στο σχεδιαστή των ΠΣΙ με απώτερο στόχο τους περισσότερο φιλικούς προς το χρήστη σχεδιασμούς. Δεύτερον, την περαιτέρω ώθηση της έρευνας, προσφέροντας μια κοινή βάση και αναγνωρίζοντας τα περιβάλλοντα πεδία που συνθέτουν την προσαρμοστικότητα. Η διατριβή υποστηρίζει την πιθανότητα ανάπτυξης ενός κατάλληλου πλαισίου όπως αυτό προκύπτει από την έρευνα των «γενικών προσαρμοστικών συστημάτων» μέσω μιας συστημικής προσέγγισης.

Σκοπεύοντας στη διερεύνηση συστημάτων με διαφορετικά χαρακτηριστικά για τη διασφάλιση μιας διεπιστημονικής έρευνας, αναπτύχθηκε μια ταξινόμηση των γενικών συστημάτων βασισμένη σε προϋπάρχουσες ταξινομήσεις που συναντήθηκαν στη βιβλιογραφία. Η διερεύνηση περιέλαβε φυσικά, κοινωνικά και τεχνητά συστήματα τα οποία αποτέλεσαν αντικείμενο μελέτης σε θέματα σχετικά με την ιδιότητα της προσαρμοστικότητας. Η μελέτη πραγματοποιήθηκε σε βαθμό όχο εξαντλητικό αλλά ικανό, αφενός να δημιουργήσει μια πλούσια εικόνα για την έννοια της προσαρμοστικότητας στα γενικά συστήματα και αφετέρου να εμπνεύσει νέα επιτεύγματα, τόσο σε εννοιολογικό επίπεδο όσο και σε επίπεδο υλοποίησης.

Ξεκινώντας από τη διερεύνηση των φυσικών συστημάτων, ο μηχανισμός της εξέλιξης εκφράστηκε υπό τη μορφή της φυσικής επιλογής. Η προσαρμοστική συμπεριφορά ενός συστήματος απορρέει από απολύτως φυσικούς περιβαλλοντικούς παράγοντες οι οποίοι είναι υπεύθυνοι για τα κληρονομικά χαρακτηριστικά στους πληθυσμούς των οργανισμών. Συμπερασματικά, ένα ΠΣΙ δεν χρειάζεται να είναι προσαρμοστικό λόγω του σχεδιαστή του αλλά λόγω της δυνατότητάς του να επωφελείται από την αλληλεπίδρασή και το ιστορικό αλληλεπίδρασής του με το περιβάλλον του. Προς την ίδια κατεύθυνση, βάσει της βιβλιογραφίας στα κοινωνικά συστήματα, γίνεται καταφανές ότι η προσαρμοστικότητα πρέπει να ανέρθει με ένα δυναμικό εσωτερικό τρόπο καθώς το σύστημα εξελίσσεται, επομένως δεν μπορεί να επιβληθεί εξωτερικά.

Επιπλέον, από τη σκοπιά της αρχής της αυτό-οργάνωσης, ένα προσαρμοστικό ΠΣΙ μπορεί να μοντελοποιηθεί σαν ένα ανοιχτό δυναμικό σύστημα το οποίο κατά τη διάρκεια του κύκλου ζωής του ωθείται μακριά από την θέση ισορροπίας του λόγω των περιβαλλοντικών αλλαγών.

Υπό τις δεδομένες συνθήκες, για να είναι ένα ΠΣΙ προσαρμοστικό θα πρέπει να είναι αυτό-οργανωμένο και συνεπώς εξαρτάται από την σύζευξη του συστήματος με το περιβάλλον του και όχι από εξωτερικούς επιβαλλόμενους παράγοντες. Η αλληλεπίδραση ανάμεσα στο σύστημα και το περιβάλλον είναι η υπεύθυνη η οποία καθιστά την προσαρμοστικότητα τόσο απαραίτητη όσο και εφικτή.

Αυτό το συμπέρασμα στηρίζεται ακόμη περισσότερο στη θεωρεία της αυτοποίησης, σύμφωνα με την οποία ένα σύστημα για να είναι προσαρμοστικό θα πρέπει να εξαρτάται από την δομή του. Αυτό θα πρέπει πάντα να τείνει να εξελιχθεί προς έναν ελκυστήρα, δηλαδή μια κατάσταση η οποία ταιριάζει καλύτερα στις εκάστοτε περιβαλλοντικές αλλαγές. Βάσει της αρχής «βαθμός του θορύβου» διαφαίνεται ότι όσο περισσότερη η αλληλεπίδραση με το περιβάλλον τόσο γρηγορότερα το σύστημα θα αυτό-οργανωθεί. Σύμφωνα με αυτό και δεδομένου ότι το σύστημα είναι «ευφυές» αν έχει μεγάλη εσωτερική ποικιλία συμπεριφορών, απορρέει το συμπέρασμα ότι η ικανότητα ενός ΠΣΙ να προσαρμόζεται θα πρέπει να είναι ανάλογη της ποικιλίας του και της ικανότητάς του / ευφυΐας του να διαχειρίζεται την ποικιλία του κατά την αλληλεπίδρασή του με άλλα συστήματα.

Το «υπερ-σταθερό» (ulta-stable) σύστημα του Ashby φαίνεται ότι μπορεί να αποτελέσει μια βάση για την μοντελοποίηση ενός προσαρμοστικού συστήματος δίνοντας έμφαση στην έννοια των «θεμελιωδών μεταβλητών» η οποία καθορίζει το μοντέλο του συστήματος. Αυτό μπορεί συγκριθεί με το ανθρώπινο νευρικό σύστημα, όπως έχει διερευνηθεί μέσω του πρίσματος τόσο της γνωστικής επιστήμης όσο και της βιολογίας, το οποίο μπορεί να αντιλαμβάνεται κάποιες αλλαγές στο περιβάλλον του και να προκαλεί κατάλληλες ενέργειες.

Κλείνοντας τη διερεύνηση στα φυσικά συστήματα, αποδείχτηκε ότι ακόμη και στην περίπτωση των οικοσυστημάτων υπάρχουν έρευνες οι οποίες επιζητούν την αντιμετώπιση των προβλημάτων μέσα από την έννοια της προσαρμοστικότητας. Συγκεκριμένα, στην περίπτωση της κλιματικής αλλαγής μια τέτοια διερεύνηση έδωσε έμφαση στην ανάγκη για συνεχή αξιολόγηση της διεργασίας της προσαρμοστικότητας και εισήγαγε μια σκοπιά για ένα γενικευμένο πλαίσιο βασισμένο σε τρία βασικά ερωτήματα: προσαρμογή σε τι, ποιος ή τι προσαρμόζεται και πως πραγματοποιείται μια τέτοια προσαρμογή.

Από μια σύντομη διερεύνηση στο χώρο των κοινωνικών συστημάτων προκύπτει ότι ένα ΠΣΙ μπορεί να θεωρηθεί ως ένα κοινωνικό δίκτυο από αυτό-αναφερόμενα υποσυστήματα τα οποία χαρακτηρίζονται τόσο από τις ατομικές όσο και από τις αναδυόμενες – ολιστικές τους ιδιότητες. Η μελέτη των κοινωνικών συστημάτων δίνει ιδιαίτερη έμφαση στην σημασίας της έννοιας της πληροφορίας. Βλέποντας τα ΠΣΙ σαν κοινωνικά δίκτυα από αυτό-αναφερόμενα υποσυστήματα καθιστά την έννοια της πληροφορίας και στο χώρο αυτό εξίσου σημαντική. Πιο συγκεκριμένα, η έννοια της πληροφορίας όπως ορίζεται στο πεδίο των κοινωνικών συστημάτων, δηλαδή ως εσωτερική διεργασία επιλογής, καθίσταται όχημα στην πορεία προς την «αυτό-αξιολόγηση» των υποσυστημάτων και των συστημάτων.

Η ιδέα ενός κοινωνικού δικτύου από υποσυστήματα έχει διατυπωθεί στο παρελθόν στους κόλπους της Τεχνητής Νοημοσύνης και ειδικότερα στην περιοχή των έξυπνων πρακτόρων με τη μορφή των πολύ-πρακτορικών συστημάτων. Συγκεκριμένα, η διατριβή παρουσιάζει μηχανιστικές προσεγγίσεις συμπεριλαμβάνοντας νευρωνικά δίκτυα, εξελικτικές προσεγγίσεις και προσεγγίσεις ακόμη και από την κλασσική ΤΝ, την κυβερνητική, ικανές να αντιμετωπίσουν επί μέρους προβλήματα (π.χ. μάθηση και παραγωγή δομών) στο χώρο των ΠΣΙ.

Κλείνοντας το σχετικό κεφάλαιο, το πεδίο των «αυτόνομων υπολογιστικών συστημάτων» (autonomic computing) διερευνάται. Το AC αποτελεί μια προσέγγιση σύμφωνα με την οποία, τα υπολογιστικά συστήματα αυτό-διαχειρίζονται με την ελάχιστη ανθρώπινη παρεμβολή. Μια τέτοια προσέγγιση φαίνεται να έχει παραπλήσιες αρχές με το αναδυόμενο πλαίσιο της παρούσας έρευνας αλλά με τη βασική διαφορά ότι η παρούσα έρευνα εστιάζει στην αλληλεπίδραση με το χρήστη και όχι στην εξασφάλιση των απαραίτητων πόρων.

Το κεφάλαιο 3 εστιάζει στην έρευνα στο χώρο του Υπερκειμένου και των Υπερμέσων, τους “προγόνους” του Ιστού. Μια βιβλιογραφική αναδρομή στα σχετικά επιτεύγματα, συνυπολογίζοντας το περιβάλλον ανάπτυξής του, κατέληξε ότι η ανάγκη για την ανάπτυξη απαραίτητων υποδομών είχε αποπροσανατολίσει την έρευνα προς «σύστημοκεντρκές» λύσεις με αποτέλεσμα οι αρχικές σχεδιαστικές ανάγκες να έχουν παραμεριστεί. Πρόσφατα, έχει δοθεί η δέουσα έμφαση σε τέτοιες παραμέτρους, οι οποίες αφορούν στην αλληλεπίδραση ανθρώπου – υπολογιστή εισάγοντας ανθρωποκεντρικά συστήματα τα οποία εκφράστηκαν κατά βάση με την εισαγωγή της μοντελοποίησης χρήστη και τα προσωποποιημένα συστήματα.

Ομοίως με την εξέλιξη της μοντελοποίησης χρήστη, η ιδιότητα της προσβασιμότητας έκανε την εμφάνισή της σε συστήματα αποτελώντας μέρος της λογικής τους. Στην παρούσα εργασία προτείνεται ο διαχωρισμός σε διαστάσεις (aspect oriented manner). Με άλλα λόγια, ο μηχανισμός της προσαρμοστκότητας ενός ΠΣΙ χρειάζεται να διαχωριστεί σαν λογική αλλά ταυτόχρονα να είναι κατανεμημένο στα υποσυστήματά του.

Επιπλέον, στο εν λόγω κεφάλαιο αναγνωρίστηκε ότι η φύση του υπερκειμένου ταιριάζει σε αναπτυγμένα μοντέλα ανθρώπινης γραφής και ανάγνωσης. Το γεγονός αυτό θα πρέπει να αξιοποιηθεί σαν πλεονέκτημα και να χρησιμοποιηθεί προς χάρη του χρήστη. Συνεπώς, φαίνεται πως η σύγκριση του γνωστικού μοντέλου του ανθρώπου με την φύση του υπερκειμένου μπορεί να αποβεί πολύ ενδιαφέρουσα και να τα συμπεράσματα που απορρέουν να μπορούν να εφαρμοστούν σε μελλοντικά ΠΣΙ καθιστώντας ικανή την προσαρμογή του στο γνωστικό προφίλ αλληλεπίδρασης του χρήστη. Κάτι τέτοιο έχει ιδιαίτερο ενδιαφέρον στην περίπτωση των χρηστών με διανοητικές δυσκολίες.

Έμφαση κατά τη διερεύνηση, δίνεται στο πεδίο των προσαρμοστικών υπερμέσων. Πρόκειται για υπερμέσα τα οποία αντανακλούν κάποια χαρακτηριστικά του χρήστη σε κάποιο κατάλληλο μοντέλο με στόχο την προσαρμογή κάποιων αισθητών (στο χρήστη) διαστάσεων του συστήματος. Είναι κρίσιμος ο διαχωρισμός ανάμεσα στα προσαρμόσιμα (adaptable) συστήματα, τα οποία χαρακτηρίζονται από προκαθορισμένη παραμετροποίηση που γίνετε από το χρήστη σε αντιδιαστολή με τα προσαρμοστικά (adaptive) συστήματα, όπου η προσαρμογή είναι αυτόματη. Δηλαδή, το σύστημα αποφασίζει μέσα από την αλληλεπίδραση βάσει της γνώσης του. Το κεφάλαιο καταλήγει με την παρουσίαση μεθόδων, μοντέλων και τεχνικών που έχουν χρησιμοποιηθεί στο πεδίο των προσαρμοστικών υπερ-μέσων τα οποία έχρηζαν αφαίρεση και προσαρμογή στο πεδίο των ΠΣΙ.

Το επόμενο, τέταρτο, κεφάλαιο ασχολείται με την διερεύνηση του ρόλου της προσαρμοστικότητας στο χώρο των ΠΣΙ δίνοντας έμφαση στο χώρο της προσβασιμότητας του Ιστού, βλέποντάς τον σαν μια ιδιαίτερα απαιτητική περίπτωση των προσαρμοστικών συστημάτων. Μια ανασκόπηση στο πεδίο της προσβασιμότητας του Ιστού καθώς και η απορρέουσα ταξινόμηση των σχετικών προσεγγίσεων συνεισέφερε διάφορες τεχνικές στην σχεδίαση των προσαρμοστικών ΠΣΙ.

Το παράδειγμα της πύλης (Portal) και οι υποδομές του θεωρούνται σαν μια ειδική και ενδιαφέρουσα περίπτωση για το θέμα της διατριβής λόγω της βασικής απαίτησης, την προσωποποίηση. Η έρευνα του ρόλου της στο χώρο τόσο του ηλεκτρονικού όσο και του κινητού επιχειρείν επιδεικνύει τη σπουδαιότητα και την ανάγκη της προσβασιμότητάς του. Η διατριβή, μέσω της προτεινόμενης προσέγγισης, καταλήγει σε εξειδικευμένα χαρακτηριστικά που θα πρέπει να διέπουν τέτοια συστήματα για να είναι προσβάσιμα.

Ακόμη και από τις πολύ πρώιμες ημέρες έρευνας των υπερ-μέσων ήταν εμφανή η ανάγκη για μοντελοποίηση του περιβάλλοντος του συστήματος. Μια τέτοια ανάγκη στις μέρες μας εξελίχθηκε σε απαίτηση λόγω της εισβολής του «εξαφανιζόμενου υπολογιστή» (disappearing computer) στην καθημερινή μας ζωή. Τα ΠΣΙ καλούνται να ανταποκριθούν σε μια τέτοια απαίτηση ενσωματώνοντας παραμέτρους μοντελοποίησης του πλαισίου δράσης (context-aware). Αυτό το κεφάλαιο παρουσίασε σχετικά ενδιαφέροντα επιτεύγματα ακολουθώντας προσεγγίσεις μοντελοποίησης του πλαισίου δράσης. Οι περισσότερες προσεγγίσεις είναι βασισμένες σε αρχές και τεχνολογίες του σημασιολογικού ιστού. Από μια τέτοια σκοπιά, ο σημασιολογικός ιστός αντιμετωπίζεται σαν έναν συμπληρωματικός μηχανισμός ο οποίος θα μπορούσε να ενδυναμώσει περισσότερο τα έξυπνα συστήματα καθιστώντας τα ικανά να καταλήγουν σε πιο χρήσιμα αποτελέσματα και να προσεγγίζουν περισσότερο τις ανάγκες του ανθρώπου.

Το κεφάλαιο καταλήγει με τη διερεύνηση των τεχνολογιών που χρησιμοποιούνται στα πολυμέσα και των αφηρημένων διεπαφών χρήστη για ιδιαίτερα αλληλεπιδραστικά συστήματα. Οι τεχνολογίες των πολυμέσων φαίνεται να αποτελούν ένα ανερχόμενο πεδίο για την προσαρμοστικότητα μέσα από την διάδοση εφαρμογών σαν την αλληλεπιδραστική τηλεόραση. Τεχνολογικά επιτεύγματα σαν το MPEG-21 μελετήθηκαν τα οποία προσέφεραν χρήσιμη πληροφορία για το προτεινόμενο πλαίσιο. Τέλος, η έννοια και τα επιτεύγματα των αφηρημένων διεπαφών χρήστη προσβλέποντάς τις σαν ενδιάμεσο σταθμό για την σχεδίαση και ανάπτυξη προσαρμοστικών ΠΣΙ προσφέροντας τη δυνατότητα τέτοιες διεπαφές να μπορούν να προσαρμόζουν την παρουσία τους ανάλογα με το αλληλεπιδραστικό πλαίσιο.

Το 5ο κεφάλαιο παρέχει μια μεθοδολογική βάση για την σχεδίαση προσαρμοστικών ΠΣΙ συμπεριλαμβάνοντας HCI σχεδιαστικές προσεγγίσεις καθώς και μεθόδους και παραδείγματα σχεδίασης λογισμικού. Τέτοιες μέθοδοι φαίνεται τελικά να μπορούν να χρησιμοποιηθούν σαν πηγή έμπνευσης κατά την διαδικασία της προσαρμογής. Συνεπώς, η πρσαρμοστικότητα μπορεί να θεωρηθεί ως μια συνεχή, επαναληπτική σχεδιαστική διαδικασία με μεταβαλλόμενες απαιτήσεις. Από μια τέτοια σκοπιά, ένα προσαρμοστικό σύστημα είναι ένα σύστημα το οποίο είναι σε θέση να σχεδιάζει συνεχώς τον εαυτό του (αυτο-σχεδιαζόμενο).

Από την σκοπιά της μηχανικής λογισμικού, φάνηκε ότι η τάση οδηγεί από τις πολύπλοκες, εξειδικευμένες λύσεις προς τα αρθρωμένα / κατανεμημένα συστήματα και τις ανοιχτές αρχιτεκτονικές, οι οποίες επιτρέπουν την επαναχρησιμοποίηση και το διαχωρισμό των εμπλεκομένων. Καρπός αυτών είναι οι λεγόμενες Υπηρεσιο-στραφής (SoA) αρχιτεκτονικές.

Η χρήση Υπηρεσιο-στραφών (SoA) αρχιτεκτονικών και υπηρεσιών Ιστού διαφαίνεται ως μια καλή βάση για την απαραίτητη - διαφαινόμενη κατανεμημένη αρχιτεκτονική για τα προσαρμοστικά ΠΣΙ. Ιδιαίτερο ενδιαφέρον παρουσιάζει το αναδυόμενο πεδίο των σημασιολογικών υπηρεσιών Ιστού. Η αρχιτεκτονική αποδεικνύεται να είναι μια αναδυόμενη ιδιότητα, γεγονός που την καθιστά ικανή να προσφέρει προσαρμοστικότητα σε ένα ΠΣΙ μέσα από την αυτό-αναζήτηση, ανάκτηση και ενορχήστρωση των υπηρεσιών. Αυτές οι υπηρεσίες θα είναι σε θέση να αναλαμβάνουν ρόλους και να αξιολογούν την απόδοσή τους βάσει ορισμένων κριτηρίων (βλέπε SESA Framework, OWL-S).

Το κεφαλαίο 6 παρουσιάζει το εν λόγο πλαίσιο (DAWIS). Το πλαίσιο διέπεται από δύο βασικές αρχές. Πρώτον, να είναι εύρωστο και αφηρημένο αρκετά έτσι ώστε να έχει αυξημένη συμβατότητα με ένα ευρύ φάσμα τωρινών και μελλοντικών εφαρμογών και τεχνολογιών ΠΣΙ. Δεύτερον, είναι αρκετά εξειδικευμένο και εκτατό ώστε να είναι εφαρμόσιμο.

Σύμφωνα με το DAWIS, ένα προσαρμοστικό σύστημα «ζει» σε ένα μεταβαλλόμενο περιβάλλον και αλληλεπιδρά έτσι ώστε συνεχώς να πληροί το σκοπό του, ανεξαρτήτως από τις μεταβαλλόμενες συνθήκες. Ένα προσαρμοστικό σύστημα πρέπει είναι σε θέση να διατηρεί την εσωτερική του ισορροπία σε ένα σταθερό περιβάλλον αλλά ταυτόχρονα να μπορεί να προσαρμόζεται / ανασχηματίζεται σε ένα μεταβαλλόμενο περιβάλλον για να εξασφαλίσει την ύπαρξή του. Συνεπώς, το σύστημα θα πρέπει πάντα να δρα με στόχο να δημιουργήσει τον εαυτό του, το οποίο με τη σειρά του σημαίνει ότι η πληροφορία η οποία προσδιορίζει το σύστημα (κώδικας) διατηρείται και άρα είναι κατανεμημένη στα συστατικά του. Για να μπορεί ένα σύστημα να ανασχηματίζεται θα πρέπει να είναι σε θέση να αυξάνει την εσωτερική του πολυπλοκότητα, σε μορφή εσωτερικής ποικιλίας. Για τη μεταφορά των παραπάνω στα πλαίσια των ΠΣΙ μελετάται ένα απλουστευμένο σενάριο προσπέλασης μιας δυναμικής ιστοσελίδας από χρήστες με διαφορετικές απαιτήσεις. Σε ένα τέτοιο σενάριο αναγνωρίζονται οι σχετικές έννοιες:

Προκαθορισμένη ποικιλία: Υποθέτουμε ένα ΠΣΙ, μια δυναμική ιστοσελίδα, με τη δυνατότητα να αλλάζει το χρώμα του φόντου βάσει αντίστοιχης προτίμησης του χρήστη. Το συγκεκριμένο σύστημα έχει τη δυνατότητα να παραδώσει τη σελίδα σε Ν διαφορετικά χρώματα. Από τη σκοπιά του DAWIS αυτή η αρχική δυνατότητα που προκαθορίστηκε από τον σχεδιαστή και μπορεί να ποσοτικοποιηθεί με τον αριθμό Ν είναι η εσωτερική ποικιλία του ΠΣΙ (V).

Απαιτούμενη ποικιλία: Κατά την αλληλεπίδραση, όταν το σύστημα έχει το χρώμα που ο χρήστης προτιμά λέγεται ότι το σύστημα έχει τη δυναμική ή με άλλα λόγια, την απαιτούμενη ποικιλία για να προσαρμοστεί.

Επιλογή: Ωστόσο, η δυναμική από μόνη της (απαραίτητο χρώμα) δεν σημαίνει ότι το σύστημα θα ικανοποιήσει τον χρήστη. Το σύστημα χρειάζεται έναν μηχανισμό για να είναι σε θέση να επιλέξει ανάμεσα στα εναλλακτικά, δηλ. να αποφασίσει. Σε αυτή τη περίπτωση λέγεται ότι το σύστημα είναι ικανό να προσαρμοστεί (απαραίτητη ποικιλία + επιλογή).

Παραγωγή ποικιλίας: Είναι επίσης πιθανό ένας χρήστης να προτιμά ένα χρώμα το οποίο το σύστημα δεν το έχει, δηλαδή δεν έχει την απαραίτητη ποικιλία Σε αυτή την περίπτωση το προσαρμοστικό σύστημα θα πρέπει παράγει ποικιλία για να ικανοποιήσει τον χρήση.

Αύξηση πολυπλοκότητας: Ένα χρήστης παρότι έχει δηλώσει προτίμηση σε ένα χρώμα τυγχάνει να έχει αχρωματοψία και επομένως δεν μπορεί να δουλέψει με ένα τέτοιο χρώμα. Το ΠΣΙ θα πρέπει να αναπτύξει εναλλακτικούς μηχανισμούς για να μπορέσει να αντιμετωπίσει μια τέτοια σύγκρουση, δηλαδή να λάβει υπόψη του περισσότερες - νέες παραμέτρους.

Στη συνέχεια διευκρινίζεται η έννοια κλειδί «προσαρμοστικότητα» και περαιτέρω παρέχονται ενδεικτικά μοντέλα τα οποία μπορούν να ποσοτικοποιήσουν τις κρίσιμες παραμέτρους που συνθέτουν την προσβασιμότητα έτσι ώστε να καθίσταται δυνατή η αξιολόγησή της βάσει της επίδρασής της στο χρήστη:

Διάσταση Προσαρμοστικότητας (Measurement) – Am: Ορίζεται βάσει της ποικιλίας του συστήματος (Vs), του περιβάλλοντος (Ve) και της δυνατότητας επιλογής / συμπερασματολογίας (S). Με άλλα λόγια, εκφράζει το «πόση» προσαρμογή έλαβε χώρα σε ένα χρονικό διάστημα και είναι εντελώς ανεξάρτητο από το αποτέλεσμα της προσαρμογής (πόσο επιτυχής ήταν η προσαρμογή).

Αποτελεσματικότητα Προσαρμοστικότητας (effectiveness) – Ae: Ορίζεται αποκλειστικά ως συνάρτηση της βελτίωσης του ταιριάσματος της κατάστασης του συστήματος μια δεδομένη χρονική στιγμή. Τις περισσότερες φορές που η οντότητα που αλληλεπιδρά είναι κάποιος χρήστης, η Ae μπορεί να εκφραστεί σε σχέση με την ικανοποίηση του χρήστη.

Χωρητικότητα Προσαρμοστικότητας (capacity) – Ac: Η ικανότητα για προσαρμογή ενός συστήματος, δηλαδή πόσο προσαρμοστικό μπορεί να καταστεί το σύστημα. Πόσο καιρό το σύστημα μπορεί να επιβιώσει (ταιριάζει) όσο η πολυπλοκότητα της προβληματικής κατάστασης αυξάνεται ή/και το περιβάλλον (απαιτήσεις του συστήματος) συνεχώς μεταβάλλεται.

Περαιτέρω, το DAWIS υποστηρίζει το σχεδιαστή κατά τις πρώτες φάσεις της σχεδίασης και ειδικότερα κατά την αξιολόγηση της προβληματικής κατάστασης. Τον υποστηρίζει να αποφασίσει αν, και αν ναι, σε τι βαθμό το προϊόν της σχεδίασής του θα πρέπει να χαρακτηρίζεται από την ιδιότητα της προσαρμοστικότητας. Γι αυτό το λόγο, το πλαίσιο παρέχει ένα ενδεικτικό / εμπειρικό καθοδηγητικό γράφημα στην υπηρεσία του σχεδιαστή. Το γράφημα δεν είναι εστιασμένο σε πληθώρα κριτηρίων που ποικίλλουν ανάλογα με την προβληματική κατάσταση. Τουναντίον, πραγματεύεται τις βασικές αφηρημένες αρχές που καθορίζουν την απόφαση του σχεδιαστή.

Επιπροσθέτως, το DAWIS παρέχει στον σχεδιαστή ένα εννοιολογικό μοντέλο ενός ΠΠΣΙ δύο βαθμών. Το μοντέλο χαμηλότερου βαθμού καθορίζει το βασικό κύκλο ζωής (Αισθάνεται, Κάνει πλάνο, Μαθαίνει, Παράγει ποικιλία, Ενεργεί) ενός προσαρμοστικού ΠΣΙ και τις λειτουργικές απαιτήσεις μελετώντας το σαν “όλον”. Το μοντέλο υψηλότερου βαθμού καθορίζει βασικές αρχές που πρέπει να διέπουν τα υποσυστήματα ενός προσαρμοστικού ΠΣΙ. Έτσι ένα υποσύστημα θα πρέπει να είναι: *Αυτό-αναφορικό*, να γνωρίζει τον εαυτό του τόσο σε επίπεδο υποσυστήματος όσο σε επίπεδο υπερσυστήματος (αναδυόμενες ιδιότητες); *Αυτό-διατηρούμενο*, να αναπαράγει και να ανασχηματίζει τον εαυτό του; *Αυτό-αξιολογήσιμο*, να αξιολογεί τον εαυτό του τόσο σε επίπεδο υποσυστήματος όσο και σε επίπεδο απορρέουσας συμπεριφοράς του υπερσυστήματος; *Επικοινωνιακό*, να επικοινωνεί σημασιολογικά με άλλα συστήματα ή υποσυστήματα μέσω της διεπαφής, είτε του υποσυστήματος, είτε του υπερ-συστήματος; *Αυτό-βελτιστοποιήσιμο*, να βελτιστοποιεί ή ακόμη και να απορρίπτει - αντικαθιστά τον εαυτό του

Επιπροσθέτως, το πλαίσιο αναγνωρίζει τις βασικές οντότητες αλληλεπίδρασης: Ο χρήστης, η πλατφόρμα, η προσφερόμενη υπηρεσία και το πλαίσιο διανομής. Βάσει αυτών, το πλαίσιο εισάγει ένα προφίλ αλληλεπίδρασης δύο επιπέδων: Το αφαιρετικό, το οποίο καθορίζει αφηρημένες ομάδες από χαρακτηριστικά κάτω από τέσσερα πιο εξειδικευμένα προφίλ που αντιστοιχούν σε κάθε οντότητα και το κατώτερο επίπεδο το οποίο παρέχει πιο συγκεκριμένα χαρακτηριστικά των οντοτήτων. Ένα τέτοιο προφίλ μπορεί να υποστηρίζει τον σχεδιαστή κατά την αναγνώριση του περιβάλλοντος του προσαρμοστικού συστήματος και την ανάπτυξη κατάλληλου μοντέλου αλληλεπίδρασης για το σύστημα του. Ακόμη περισσότερο, το DAWIS παρέχει ενδεικτικό μοντέλο οντοτήτων (object model), αρχιτεκτονική, τεχνολογίες και λογισμικά πλαίσια για την υλοποίηση ενός προσαρμοστικού ΠΣΙ.

Κατά το τελευταίο κεφάλαιο της διατριβής παρουσιάζονται τέσσερις μελέτες περίπτωσης οι οποίες έχουν συμβάλλει κατά την ανάπτυξη και την προκαταρκτική αξιολόγηση του προτεινόμενου πλαισίου:

Ανάμιξη Προφίλ Χρήστη / Συσκευή – IRIS: Κάτω από την ομπρέλα του έργου IRIS η προσβασιμότητα στον ιστό προσεγγίστηκε μέσα από την προσαρμοστικότητα βάσει της ανάμιξης προφίλ χρήστη / συσκευής. Η εφαρμογή μιας τέτοιας προσέγγισης παρείχε μια βάση για τη μοντελοποίηση μιας αλληλεπίδρασής χρήστη/σύστημα και αποτέλεσε τη βάση για την ανάπτυξη του προφίλ αλληλεπίδρασής. Επιπλέον, η προσέγγιση αυτή παρείχε μια ενδιαφέρουσα αρχιτεκτονική λογισμικού βασισμένη σε “πληρεξούσιο”.

Εργαλεία για την προσβασιμότητα του Ιστού – BenToWeb: Το έργο BenToWeb επίσης συνεισέφερε στον σχεδιασμό προσαρμοστικών ΠΣΙ μέσω: αξιολόγησης τεχνικών και τεχνολογιών και από τους ίδιους τους χρήστες; αξιολόγησης τεχνικών μοντελοποίησης χρήστη – έμφαση σε

ΑμεΑ; αξιολόγησης τεχνολογιών και λογισμικών πλαισίων αιχμής σχετικά με την ανάπτυξη προσαρμοστικών ΠΣΙ.

Προσβασιμότητα Πυλών: Από το πρίσμα του DAWIS προσεγγίστηκε επίσης και η προσβασιμότητα των πυλών ιστού. Η διατριβή προτείνει μια προσέγγιση τριών επιπέδων: Προσβασιμότητα περιεχομένου, όπου ισχύουν οι οδηγίες που αφορούν και στις απλές ιστοσελίδες με κάποιες εξαιρέσεις; Προσβασιμότητα Γενικευμένων Πυλών, όπου υποστηρίζεται ότι η σχεδίαση των πυλών αποτελεί ιδιάζουσα σχεδιαστική περίπτωση συγκριτικά με αυτή των ιστοσελίδων με βασική απαίτηση για σχεδίαση διεπαφής εφαρμογιδίων πυλών αγνοώντας την υπόλοιπη πύλη; Προσβασιμότητα Υπηρεσιών Πυλών όπου η προσβασιμότητα μελετάται σε επίπεδο υπηρεσιών και εστιάζεται στα ιδιαίτερα χαρακτηριστικά μιας υπηρεσίας ή μιας ομάδας υπηρεσιών. Η μελέτη αυτή πραγματοποιείται στο χώρο του Ηλεκτρονικού και Κινητού εμπορίου.

Προσβασιμότητα στην Αλληλεπιδραστική Τηλεόραση – MPEG21: Η τελευταία μελέτη περίπτωση αφορά στην προσβασιμότητα της αλληλεπιδραστικής τηλεόρασης με έμφαση στο ρόλο του MPEG-21 προτύπου. Το θέμα προσεγγίστηκε αμιγώς μέσω του πλαισίου DAWIS και συνεπώς η συγκεκριμένη μελέτη προσφέρει μια πρώτη αξιολόγηση της χρήσης του πλαισίου. Βάσει του πλαισίου η σχεδιαστική λύση έπρεπε να αντιμετωπίσει το πρόβλημα μέσω της προσαρμοστικότητας. Έτσι το σύστημα (προσομοιωτής iTV) μοντελοποιήθηκε βάσει του αφηρημένου μοντέλου του DAWIS, αναπτύχθηκε κατάλληλο αλληλεπιδραστικό προφίλ και υλοποιήθηκε βάσει προτεινόμενων τεχνικών και τεχνολογιών.

Κλείνοντας την διατριβή, παρουσιάζονται κάποια χρήσιμα συμπεράσματα που θα πρέπει να ληφθούν υπόψιν για την περαιτέρω έρευνα στο χώρο. Έτσι, μέσα από την παρούσα έρευνα, αναδεικνύεται η ανάγκη για προσαρμοστικά ΠΣΙ και η έλλειψη ενός κοινού πλαισίου αναφοράς. Επιπλέον, αποδεικνύεται ότι μια διεπιστημονική διερεύνηση μέσα από μια συστημική σκοπιά μπορεί να αποφέρει ένα τέτοιο πλαίσιο. Το πλαίσιο που αναπτύχθηκε φαίνεται αφηρημένο αρκετά για να ενσωματώσει ευρύ χώρους και προβληματικές καταστάσεις αλλά και ικανό να βοηθήσει έμπρακτα το σχεδιαστή από τα πρώτα στάδια της σχεδίασης μέχρι και τα τελευταία της υλοποίησης και της διασφάλισης της ποιότητας. Πρέπει να σημειωθεί, ότι το πλαίσιο από μόνο του δεν προσφέρει δυνατότητα σχεδίασης συστημάτων «πραγματικής» αυτονομίας αλλά διαχωρίζει τους διάφορους βαθμούς προσαρμοστικότητας οριοθετώντας τις απαιτήσεις σε αφαιρετικό επίπεδο. Τέλος, το πλαίσιο μέσα από την ίδια του την προσέγγιση αλλά και από την ανάδειξη των αφηρημένων απαιτήσεων ανοίγει το δρόμο προς πολλές ερευνητικές κατευθύνσεις για περαιτέρω έρευνα σε πιο εξειδικευμένα πεδία (π.χ. neural nets, evolutionary algorithms). Κατά την εξέλιξη της έρευνας το πλαίσιο θα πρέπει συνεχώς να επαναξιολογείται και να βελτιώνεται.