User Individuality Management in Websites based on WAI-ARIA Annotations and Ontologies

Xabier Valencia, Myriam Arrue, J. Eduardo Pérez and Julio Abascal

EGOKITUZ: Laboratory of HCI for Special Needs, University of the Basque Country (UPV/EHU)
Informatika Fakultatea
20018 Donostia, Spain

[xabier.valencia, myriam.arrue, juaneduardo.perez, julio.abascal]@ehu.es

ABSTRACT
In this paper, we describe a system for adapting websites based on an annotation approach. The system adapts annotated websites according to the user characteristics. The core of the system is composed of a repository containing a comprehensive set of adaptation techniques. 99 techniques have been identified and classified, 48 of them have been fully implemented in the system. This wide range of implemented techniques provides the system with the necessary mechanisms for adapting diverse aspects of website interfaces, such as the content to display, the layout and the structure. The techniques to apply for a given user are inferred according to the information stored in an ontology. This defines the bases for associating user characteristics with the adaptation techniques to apply for specific interaction elements annotated in a website. An annotation language based on the roles and properties of WAI-ARIA has been elaborated. Therefore, any website annotated in this language can be adapted. This adaptation system has been applied to two different websites and adapted interfaces have been obtained for three different user groups.

Categories and Subject Descriptors
H.5.4 [Information Interfaces and Presentation]: Hypertext/Hypermedia—User Issues; K.4.2 [Computers and Society]: Social Issues—Assistive technologies for persons with disabilities

General Terms
Human Factors

Keywords
Website adaptation techniques, Ontologies, ARIA, accessibility.

1. INTRODUCTION
In the last few years, much research effort has been focused on improving the user’s experience when accessing the Web. Several tools and techniques have been implemented in order to help users to perform tasks in an efficient and satisfactory way. However, there are still several groups of users that experience barriers in their interactions. In addition, Web 2.0 technologies are creating new types of barriers [28].

The WAI-ARIA [http://www.w3.org/WAI/intro/aria] language allows the addition of semantic annotations to HTML code so that assistive technologies can better manage interaction elements. However, there is a lack of awareness of this language in the designer community. HTML5 [http://www.w3.org/TR/html5/] has been released with the aim of covering some of the lack of semantics in previous HTML versions. It defines interaction elements by their roles and assistive technologies can benefit from this feature in order to provide more accessible websites to users with disabilities. However, this new version has only recently been launched and most of the existing websites are formatted in previous versions. For this reason, adapting websites is of interest to many users as it would allow them to perform tasks and find information more quickly, with less interaction steps and in a more satisfactory manner [10].

There are several implementations for adapting websites to users’ characteristics. Most of these are focused on a single user group, for instance: blind people [21], people with limited mobility [12] or elderly people [24]. Moreover, these systems have only restricted transcoding functionality. E-learning systems can be thought of as more complete adaptation systems; nonetheless, techniques used by such systems are based more on the level of knowledge or experience of users in a specific topic rather than on their characteristics [8].

The objective of this paper is to present a comprehensive adaptation system which considers techniques for different groups of users. The system adapts any website annotated using a language based on WAI-ARIA roles and properties. The annotation language developed in this work extends the roles and properties defined in WAI-ARIA. An ontology has been defined for modelling user characteristics, adaptation techniques, annotated websites and the relationships between them. The adaptation techniques applied are inferred from this ontology, based on several rules that associate users’ characteristics with adaptation techniques for specific interaction elements in the website. The annotation is based on the elements defined in the CSS files of websites.

The system has been used to obtain adapted interfaces for two different websites: Discapnet and Bidasoa Turismo. These cases studies are presented in section 6. The rest of the paper is
organized as follows: section 2 describes the related work; section 3 presents the adaptation techniques repository; section 4 is devoted to the ontology created for user modelling, adaptation techniques modelling and website annotation; section 5 shows the architecture of the system; and, finally, conclusions are drawn and future work indicated in section 7.

2. RELATED WORK
Transcoding allows Web content to be modified on the fly without having to get website designers to add/change/tag its content. Transcoders are located in the client, in a proxy or in a web server. Some of them require annotated websites in order to perform adaptations. This annotation adds the necessary semantic information to transform content more accurately [5]. Web annotations can provide a better access to the web by offering alternative content, presentation adaptations and navigation support to user.

One interesting example of such transcoding systems is described in [4]. It is a proxy-based tool that tries to solve navigation problems for blind users. It rearranges the content and adds missing alt texts to images, based on annotations provided by volunteers. HTML element annotations are specified with XPath expressions, which require web pages to be analyzed one by one. The annotation process was improved by adding a Dynamic Annotation Matching algorithm (DAM) and an authoring tool called Site Pattern Analyzer (SPA) [25]. However, the maintenance of the system can be difficult and annotation is a tedious process even when the annotation tool is utilized.

Another remarkable proxy-based system is SADIE [13]. Instead of annotating individual web pages, it annotates the entire website via CSS. This tool includes an ontology that describes the meaning of elements found in CSS. The ontology has one upper ontology that describes element concepts such as menu, footer, etc. It also has another site-specific ontology that describes elements in terms of the upper-level ontology. This approach allows a single ontology to be used for the entire website. However, the adaptations performed by the system are focused on a single user group: blind users. Adaptation techniques applied are: removing elements, adapting navigation menus and reordering elements. Our system uses CSS in a similar way to SADIE but also makes use of XPath, when necessary, for allowing the annotation of websites without CSS or elements that are not attached to a CSS element.

Nowadays, web pages are becoming more like desktop applications. One of the most important points in this type of application is to inform users about dynamic changes [26]. WAI-ARIA guidelines can be used in order to overcome this problem. For instance, the main objective of the system described in [14] is to identify inaccessible content and then provide the necessary WAI-ARIA attributes in order to make it accessible. Transformations are intended to be implemented in the core of the browser. This approach differs from the previously described works where transformations are performed by tools that are independent of the browser. This architecture will enable a fast transformation of the content, but the management of browser versions becomes more complex.

The Google AxsJAX project [11] is another system using WAI-ARIA annotations. This library is used to provide WAI-ARIA support to Web 2.0 applications. Annotation can be added at the time of development (inline) or at run time via a bookmarklet or Greasemonkey script. The use of this system requires a strong knowledge of HTML as well as of the language used to create the CNR files.

In [22] the previously mentioned SADIE system and AxsJAX are combined. The SADIE annotation system is used to insert WAI-ARIA roles and states into websites with AxsJAX, allowing user interaction with the Web via the keyboard. The SADIE website annotation language is used to transform these annotations into a CNR file that is responsible for inserting the ARIA statements.

Although the adaptation techniques integrated in the aforementioned projects are sufficient for people with visual impairments, more techniques are required for other user groups with special needs. One solution that provides personalized content is applying User Modelling techniques for determining the capabilities of each user. This has been an important research field for some time [9].

The goal of the i2Web project1, which is currently being developed, is to help people with special needs, focusing on the strategies they use when using Web 2.0 applications. They are creating a metamodel for modelling users’ characteristics (as well as their strategies and interaction modes), device features (including assistive technology information), and application features [2]. Although this system is comprehensive enough to accommodate the adaptation needs for different user groups, it requires a strong knowledge of diverse languages (such as HTML, UML, etc.) to be able to annotate a website.

3. ADAPTATION TECHNIQUES REPOSITORY
The core of the implemented adaptation system is its adaptation techniques repository. The quantity and quality of techniques integrated in the repository determines the capability of the system to adapt interfaces to specific user characteristics. In this research work, a wide range of different sources has been analyzed in order to gather as many methods and associated techniques as possible.

The survey of adaptive hypermedia methods and techniques by Knutov et al. [17] has been the basis for this analysis work. The authors classify adaptation techniques into three main categories: content, presentation and navigation adaptation techniques. This classification is quite standard, as we have found similar approaches in other studies (such as those presented in [19] and [8]). Although there may be terminology differences between these different works, the meaning remains the same. In most cases techniques are classified into three main groups: the first group is for the techniques involving changes in the content displayed in the interface; the second group is for techniques involving style changes in the interface layout, and the third group is for those techniques dealing with interface structure and behaviour. Examples of techniques in each group include the following:

- Content adaptation technique: “1.1.6 Incorporate specific scrolling icons on each page”
- Presentation adaptation technique: “2.2.3 Place important areas of content near the top of the page”
- Navigation adaptation technique: “3.1.2 Create table of contents for the website”

We have also adopted this classification in our study. Table 1 shows the total number of techniques we have identified for each

---

1 i2Web project. http://i2web.eu
main group and the number of techniques implemented for each group.

As shown in Table 1, 48 of a total number of 99 adaptation techniques gathered in the study are fully implemented in the repository (which is 48%). As can be seen, adaptation techniques related to presentation issues are easier to implement (75%).

Table 1. Total number of techniques gathered and fully implemented, classified by group: content adaptation techniques, presentation adaptation techniques and navigation adaptation techniques.

<table>
<thead>
<tr>
<th>Group</th>
<th>Total Number</th>
<th>Implemented</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content (1)</td>
<td>39</td>
<td>11</td>
<td>28%</td>
</tr>
<tr>
<td>Presentation (2)</td>
<td>37</td>
<td>28</td>
<td>75%</td>
</tr>
<tr>
<td>Navigation (3)</td>
<td>23</td>
<td>9</td>
<td>39%</td>
</tr>
<tr>
<td>Total</td>
<td>99</td>
<td>48</td>
<td>48%</td>
</tr>
</tbody>
</table>

These techniques are related to style issues and the majority can be easily developed by applying CSS techniques. Nevertheless, there are some techniques in this group which have been gathered and classified but not implemented, as we considered them more oriented to performing adaptations to specific devices; for example: “2.7.1 Page segmentation” and “2.7.5 Provide thumbnail view of the page”. These types of techniques will be integrated in future development of the system.

The most difficult techniques to implement seem to be those related to the content of the website. In fact, some of the issues considered in this group require techniques such as natural language processing, which are not in our area of expertise; for example: “1.5.1 Summarization of text” and “1.5.2 Use positive statements”. However, we started to collaborate with natural language processing experts to overcome these types of issues.

As can be seen, each technique has a notation for identifying it in the repository. The numbering assigned to each technique identifies the group and the specific method to which the technique is related. For example, the number 1.5.3 identifies the third technique (3) that is related to natural language processing methods (5) and belongs to the content adaptation techniques group (1).

Different approaches have been considered for developing relationships between adaptation techniques and users’ characteristics: ISO/IEC TR 29138 [15] [16], Brajnik’s Barrier Walkthrough Method2 and WCAG 2.03. This analysis led us to define three general groups of users:

- Users with cognitive impairments (C)
- Users with physical impairments (P)
- Users with sensory impairments (S)

These user groups have been divided into subgroups based on the adaptation techniques we gathered. We defined 4 subgroups for cognitive impairments, 2 subgroups for physical impairments and 7 subgroups for sensory impairments.

The relation of these subgroups is shown in Table 2. More comprehensive user models can be found in the literature, but we have decided to adopt a pragmatic user modelling approach.

Therefore, only if we detect any adaptation technique related to a specific user characteristic will we define a subgroup of users. Note that this approach allows new subgroups to be incorporated when necessary.

Table 2. Relation of subgroups defined in each general group of users: users with cognitive impairments (C), physical impairments (P) and sensory impairments (S).

<table>
<thead>
<tr>
<th>General Group</th>
<th>Subgroup</th>
<th>Number of techniques by group</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>C.1 Decline in maintaining attention</td>
<td>10 5 4 19</td>
</tr>
<tr>
<td></td>
<td>C.2 Learning disabilities</td>
<td>10 1 7 18</td>
</tr>
<tr>
<td></td>
<td>C.3 Language disabilities</td>
<td>5 0 0 5</td>
</tr>
<tr>
<td></td>
<td>C.4 Reduced memory capacity</td>
<td>4 1 7 12</td>
</tr>
<tr>
<td>P</td>
<td>P.1 Limited movement</td>
<td>5 3 7 15</td>
</tr>
<tr>
<td></td>
<td>P.2 Inability to use mouse</td>
<td>2 1 6 9</td>
</tr>
<tr>
<td>S</td>
<td>S.1 low vision</td>
<td>4 10 3 17</td>
</tr>
<tr>
<td></td>
<td>S.2 blindness</td>
<td>3 1 9 13</td>
</tr>
<tr>
<td></td>
<td>S.3 colour blindness</td>
<td>0 3 0 3</td>
</tr>
<tr>
<td></td>
<td>S.4 photosensitivity</td>
<td>3 0 0 3</td>
</tr>
<tr>
<td></td>
<td>S.5 eye strain</td>
<td>0 14 3 17</td>
</tr>
<tr>
<td></td>
<td>S.6 Hearing loss</td>
<td>1 3 0 4</td>
</tr>
<tr>
<td></td>
<td>S.7 Hearing loss</td>
<td>1 0 0 1</td>
</tr>
</tbody>
</table>

The information for classifying adaptation techniques in such user subgroups has been collected from the studied sources, as some of them were focused on adaptations for specific user characteristics [12], [3], [20],[24], [23], [21].

Table 3. Number of techniques in each user type subgroup classified by three main technique type groups (Cont.-Content, Pres.-Presentation and Nav.-Navigation) and the total for each subgroup (Tot.).

<table>
<thead>
<tr>
<th>General Group</th>
<th>Subgroup</th>
<th>Number of techniques by group</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C.2</td>
<td>10 5 4 19</td>
</tr>
<tr>
<td></td>
<td>C.3</td>
<td>10 1 7 18</td>
</tr>
<tr>
<td></td>
<td>C.4</td>
<td>5 0 0 5</td>
</tr>
<tr>
<td></td>
<td>P.1</td>
<td>4 1 7 12</td>
</tr>
<tr>
<td></td>
<td>P.2</td>
<td>5 3 7 15</td>
</tr>
<tr>
<td></td>
<td>S.1</td>
<td>2 1 6 9</td>
</tr>
<tr>
<td></td>
<td>S.2</td>
<td>4 10 3 17</td>
</tr>
<tr>
<td></td>
<td>S.3</td>
<td>3 1 9 13</td>
</tr>
<tr>
<td></td>
<td>S.4</td>
<td>0 3 0 3</td>
</tr>
<tr>
<td></td>
<td>S.5</td>
<td>3 0 0 3</td>
</tr>
<tr>
<td></td>
<td>S.6</td>
<td>0 14 3 17</td>
</tr>
<tr>
<td></td>
<td>S.7</td>
<td>1 3 0 4</td>
</tr>
</tbody>
</table>

In Table 3, the number of techniques in each user subgroup classified by each type (Cont.-Content, Pres.-Presentation and Nav.-Navigation) is shown. It should be noted that some techniques could be replicated in different subgroups and groups. For example, the technique “3.1.2 Provide table of content for the website” is classified as necessary for the following subgroups in the cognitive impairments group: C.2 – Learning disabilities and C.4 – Reduced memory capacity. In addition, it is also included in some subgroups of the other two user groups: physical impairments and sensory impairments.

In some cases techniques are not clearly classified in the literature, so we undertook this task. In other cases, we assumed that some techniques were also of interest for different user groups not considered in the source. These assumptions were made based on knowledge acquired in previous research works.

---


3 WCAG 2.0, http://www.w3.org/TR/WCAG/
For example, the technique “1.4.1 Stretchtext” was gathered from sources focused on general adaptation techniques and not specific to any user group. We assume that it could be applied to users with decline in divided attention as when applying it less information (the most significant one) is shown in the interface.

4. **ONTOLOGY**

The implemented system is aimed at adapting website content, presentation and navigation, so that websites can be accessed by people with cognitive, physical and/or sensory impairments. Ontologies have been broadly applied for user modelling. Some of them [6] develop comprehensive classification of user functionalities, usually based on ICF classification [27]. Others are focused on specific user characteristics [1]. In our work we decided on following a more practical approach so only the necessary user characteristics for adapting interfaces are included. The developed ontology models users’ characteristics, adaptation techniques and annotation language. In addition, it includes the necessary mechanisms to select the most appropriate adaptations for a given user. Figure 1 shows the general concepts considered in the ontology and the following subsections describe each component of the ontology in detail.

![Figure 1. Upper ontology categorization models](image)

**4.1 User Model**

As mentioned in section 3, different user interaction characteristics have been considered in building the user model. Three general user interaction factors were included in the ontology (see Figure 1):

Cognitive: Cognitive skills are involved in human communication and in interaction with technological devices. Attention, learning, language processing and memory are characteristics considered to represent the user within this field.

Physical: Only mobility skills have been modelled as physical user characteristics that have an influence in the interaction with web pages. Limited movement and the physical inability to use a mouse are the two disabilities considered in this model.

Sensory: Visual and hearing senses are included since they are related to interaction abilities. The 7 characteristics defined in section 3 for this group have been included in the user model.

**4.2 Annotation Model**

The annotator can describe the role of interaction elements, based on the annotation model defined in the ontology. This process adds semantic meaning that enables automatic adaptations to be performed. Annotations are completely independent of the original website and adaptations are made through CSS and DOM files of the website.

The annotation model is divided into two abstract classes: one for roles and properties belonging to WAI-ARIA and another for the new roles and properties created. Most of the roles and some of the properties from WAI-ARIA are used. These roles are sufficient to perform most adaptations, but are not adequate for inserting new elements such as contextual help, tutorials and so on. Therefore, we decided to extend its vocabulary.

The new roles created are the following: ContextInfo, Tutorial, FAQ, SiteMap, Captions and GeoMap (see Figure 2). First, three elements were created to provide help to the user about an element, web page or even the entire site. The SiteMap role is used to identify an element that points to the sitemap or to create a new element if necessary. Captions are used to indicate where the captions are and GeoMap is used to provide written directions, instead of a visual map.

The new properties defined are: dimmable, hideable, stretchable, removable and priority. These properties are used to mark information or interaction elements as optional when it is necessary to simplify the interface for a given user. These properties are usually assigned to those interaction elements forming an intelligible chunk of information. They are also annotated with a role such as region, article, area, etc. Techniques such as dimming or stretchtext can be applied to these roles so that part of their content is hidden. For instance, an element tagged with “removable=flashing”, will be removed in the adapted interface for a user with photosensitivity or decline in maintaining attention.

In the website annotation process, CSS elements such as ids or classes are annotated with a semantic meaning by assigning the original WAI-ARIA or the defined new roles and properties. This approach reduces the workload for the annotation process, as it requires less effort and maintenance work.

![Figure 2. Original WAI-ARIA and the new roles and properties added in the ontology.](image)

**4.3 Adaptation Techniques**

Adaptation techniques are also modelled in the ontology. Currently, 48 fully implemented techniques are stored in it.
However, our approach allows the extension of the adaptation knowledge and the easy integration of new techniques in a continuous process.

Adaptation techniques are organized into three main groups: content, presentation and navigation adaptations. Methods and techniques are incorporated into their corresponding group. For instance, techniques related to changing font style are grouped in the FontStyle method inside the presentation adaptations group (see Figure 3). The technique identification, name and an index to the implementation function are stored for each technique. Thus, there is a direct indexing of each technique stored in the ontology and the function implementing it in the system.

In addition, each adaptation technique identifies the roles and/or properties of the interaction elements it is related to. For example, techniques regarding navigation bar issues will identify the role GeoMap, and so on.

4.4 Reasoning Rules, Constraints and Stereotypes

In order to infer adaptation techniques from the characteristics of a given user, some reasoning rules have to be included in the ontology. These rules can be classified into two groups: rules that infer adaptation techniques to apply for a given user and rules for defining constraints between techniques. In fact, there could be inconsistencies when inferring adaptation techniques for a given user. The situation becomes worse when users characteristics are probably a combination of different subgroups. For example, one user can experience low vision and also limited movement.

Adaptation techniques set for this type of user include the following two techniques: “2.3.2 Enlarge graphical buttons” and “2.7.2 Place content in web page to avoid scrolling”. The application of the first technique will enlarge the overall web page so that more scrolling will be needed. These types of situations are very common in adaptation processes and even more so in this case, considering that 48 adaptation techniques are implemented. Therefore, constraints have to be defined for avoiding these inconsistencies. Sometimes, it could be enough to apply other techniques as well as the specific ones. In this particular case applying techniques such as stretchtext or hiding some optional information could improve the adaptation results.

We decided to use stereotypes [18] in order to overcome these situations and obtain a preliminary version of the system, which could then be validated. Then, the inference rules were simplified and they are now intended to select one predefined group of adaptation techniques (well-suited stereotype) instead of selecting adaptation techniques one by one. We have detected and resolved inconsistencies between the set of adaptation techniques in each stereotype.

Therefore, stereotypes have been integrated into the ontology. In this first version, 3 stereotypes have been used. Each stereotype is defined in the ontology as a set of adaptation techniques.

A detailed description of the generic reasoning rules and stereotypes defined for the system can be found later in this paper (see Section 6.2).

5. DEVELOPMENT OF ADAPTATION SYSTEM

5.1 System Architecture

Asakawa and Takagi [5] identify three different architectures for adaptation systems: server side, intermediary and client side. In the first case, adaptations are performed in the server where the original web page is located. In the second case, the adaptation process is performed by a machine that sits between the user machine and the web page server. Finally, in the third case, the adaptation process is performed locally in the user machine by an application positioned in front of the browser, between the browser and the user.

Server side adaptations only can be implemented by website owners and adaptations are limited to the websites located in the server. Our approach has been a hybrid one combining the intermediary and client side approaches. The advantages of this approach are that it allows adapting any annotated website as well as including user tracking functionalities in the client side part. The adaptation system developed is composed of three components:

- Presentation Module
- Adaptation Module
- Knowledge Base

The Knowledge Base and the Adaptation Module are located in a remote machine. The Adaptation Module is implemented as a Web Service whereas the Presentation Module is implemented as a Firefox Add-on installed locally on the user machine. This architecture is shown in Figure 4.

![Figure 4. General architecture of the web interface adaptation system.](image)

The Presentation Module is in charge of starting and completing the adaptation process performed by the system. It gets a web page to adapt (steps 1 and 2 in Figure 4) and sends the source code with the user’s credentials to the remote Web Service (step 3 in Figure 4). This module presents the adapted version of the web page returned by the Web Service (step 6 in Figure 4). In addition, it has been designed to track the user’s interactions and obtain accurate data in specific user testing. Our plan is to utilize the results obtained to improve the system’s ontology.

The Adaptation Module implements two main functionalities. The first is to query the Knowledge Base about adaptations corresponding to the current user and the web page (step 4 in Figure 4). The second is to perform adaptations resolved by the Knowledge Base (step 5 in Figure 4) and return the adapted web page to the Presentation Module (step 6 in Figure 4). For this purpose, this module contains a repository of the techniques implemented and invokes the corresponding functions when necessary.
The Knowledge Base contains an ontology to store, update and maintain the models about user characteristics, adaptation techniques, annotation models and web page annotation instances. The results inferred by executing the defined rules are returned to the Adaptation Module. These results specify the adaptation techniques to apply for a given user.

5.2 Design considerations
When designing and developing the system, Brajnik’s guidelines [7] were followed. These guidelines provide the minimum requirements a transcoding system should meet.

Efficiency: There is an overload when adapting web pages. Users must wait for the finalization of the adaptation process before accessing a web page. This adaptation process is not trivial due to the Knowledge Base and Adaptation Module have to collaborate in order to get the adaptation techniques to apply for a given user and perform them based on the annotated roles of the interaction elements. Our initial tests show that the delay for this adaptation process is about 300ms.

Decoupling of Servers: Since our aim is to be able to perform adaptations in any website, the adaptation system is located outside the original web page server. Currently, the Knowledge Base and Adaptation Modules are implemented in the same Web Service in order to minimize the overload. However, they could also work in different servers.

Repurpose of Content: Most of the adaptations are made to the original content, based on the annotations created for the website. In some cases, new content is added to the original web page, but only with the purpose of providing help.

Appropriate Output: The adaptation system is currently based on stereotypes. However, the objective in the future is to produce personalized web interfaces based on users’ specific characteristics.

Customizability: The annotator is able to specify which elements to adapt by annotating them with the appropriate roles and properties. But in order to guarantee the customizability, other requirements have to be satisfied, such as: decoupling of content, robustness, modularity, re-usability and content preservation. These issues are described in the following subsections.

- Decoupling of Content: The system is able to store annotations in a different machine from the original web page server and it refers only to the web page through CSS elements such as class or id and/or XPath when necessary.
- Robustness: Since most website annotations are made by CSS elements, it is possible to update the website and the system will still work correctly unless specific class or id elements are changed.
- Modularity: Different adaptation techniques can be applied to the same element. Techniques are applied one by one and the combination of a set of techniques produces the adapted version of a web page.
- Reusability: Annotating class or id elements in CSS files allows reusability as the required adaptations can be applied in different web pages by defining only one annotation.
- Content Preservation: The original web page is modified using its content. As stated before, the only new content added is for the purpose of providing help.

Adaptability: At present the user is not able to customize the resulting adapted website. This issue will be considered in future versions of the system.

Adaptivity: The system has been designed in order to allow user interaction data to be obtained by the Firefox Add-on. This feature is useful for updating user modelling and adaptation techniques to apply to specific users. Hence, adaptivity could be integrated in future versions.

Interoperability: The adaptation system architecture design allows users to operate with websites as usual. The adaptation system can interoperate with other technology such as cookies, filling forms, etc.

6. USE CASE
The adaptation system has been applied to two websites: Discapnet and Bidasoa Turismo. Discapnet4 provides services and content related to social integration and employment of people with disabilities. Bidasoa Turismo5 provides information related to tourism, such as locations of interest, events, tourism facilities, etc. aimed at the general public. These two websites have been selected as they have been also utilized in other research projects. The objective of these projects was to apply web mining techniques to server logs data for automatically obtaining different user profiles.

Both websites have been annotated and adapted versions were obtained for 3 different user groups. Due to the limited space in the paper, this section thoroughly describes the annotation process and adapted versions for Discapnet. The annotation process for Bidasoa Turismo is similar and only the resulting adapted interfaces are shown at the end of this section.

It is worth noting the lack of consistency in the design and the information overload in the Discapnet website, especially since it is supposed to be designed with handicapped users in mind. The web pages are well tagged for blind people but present some barriers for other types of users; for instance, people with a cognitive impairment who may be disoriented due to inconsistency in navigation and information overload.

6.1 Discapnet Website Annotation Process
An HTML code and CSS style inspector (Firebug6) has been used to analyze the source code of the website and annotate the required interaction elements with their roles and properties. The entire process is currently manual, which is quite tedious and difficult. We plan to implement an annotation editor in the near future.

This manual process led us to identify different areas in the web page. The areas identified in the homepage are marked in Figure 5. The different areas identified and role assignments are the following:

Area 1: this area is devoted to advertisement. The only annotation applied to elements in this area is to identify them as low priority elements; the priority property is set to 10. Therefore, if sorting adaptation techniques have to be applied these elements will be placed at the bottom of the web page.

4Discapnet, http://www.discapnet.es
5Bidasoa Turismo, http://www.bidasoaturismo.com
Area 2: this area is identified as the banner of the website and is composed of different components. It is repeated in all web pages. We annotated it with the “banner” role. In this case, there is not a unique div for this element and, thus, the different elements composing the banner have to be identified (elements in the CSS files identified for this area are: classes: logo, annunciate and cabecera_home; ids: menu_rapido and registro). In addition, this area also includes two elements to annotate: the sitemap (2a) and the search functionality (2b). We annotated them with the “sitemap” and “search” roles respectively (elements in the CSS files identified for these roles are the second element of menu_rapido which is expressed by XPath and id with value buscador).

Area 3: this area is composed of navigation elements. There are three main sections and subsections inside each element (see subareas 3a, 3b and 3c in Figure 5). Each subarea is annotated with “navigation” role (elements in the CSS files identified for this role are divs: areaa, comuni and actualidad). The title of the section (elements tagged as H2) is inserted in the annotation “labelledby” property and the properties “stretchable” and “dimmable” are also inserted in order to facilitate the development of special navigation menus in adapted web pages.

Area 4: this area is where main content is placed. It has to be annotated with “main” role. As it can be seen in Figure 5, this area is composed by two subareas 4a and 4b. In this case, there is a class defined in the CSS files named secciones related to this main content. However, this class also includes Area 5 (see Figure 5). We assume that Area 5 is related to different content than 4a and 4b. Therefore, an XPath statement is necessary to identify 4a and 4b with the role “main”. These elements are also annotated with the “stretchable” and “dimmable” properties, inserting the H2 tagged elements “Hoy en Discapnet” and “Noticias y eventos” in the “labelledby” property.

Area 5: this area is composed of a set of images pointing to external locations and it is annotated with the “complementary” role. In this case, as stated previously for Area 4, an XPath statement is necessary to identify the third section element for the class secciones with the “complementary” role.

Area 6: this area contains some links to information regarding “what is Discapnet”, the date last updated and so on. We annotated it with the “contentinfo” role. The annotation is made for the CSS file element with id value equal to pie.

We have also annotated each chunk of information regarding news. These are composed of one title (which is a link to the complete piece of information), several lines of text and a link (see Area 4b” in Figure 5). This structure is annotated using the “article” role and it is defined in the CSS files with the class noticiaPortada.

The annotation process described previously has been repeated for different web pages in Discapnet, those having a different “look & feel”.. Most elements were already annotated as the same types of elements are tagged with the same div and classes in the CSS files. However, some new elements were detected in this process and annotated correspondingly.

The same annotation process was followed for the Bidasoa Turismo website. The different areas detected in homepage are shown in Figure 6. The process was easier for this website due to the consistency in the design of web pages. In this case, once we had annotated the homepage few elements had to be added to complete the annotation of the whole website.

6.2 Stereotypes
As mentioned above, we have defined 3 stereotypes in order to simplify the rules and constraints for adaptation techniques. The adaptations set defined for the 3 stereotypes are: Narrow Style, Index Style and Sorted Style. The matching process for assigning one stereotype to a specific user is shown in the next algorithm:

<table>
<thead>
<tr>
<th>Algorithm for Stereotype Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF (user cannot_use_mouse AND not more impairments) THEN</td>
</tr>
<tr>
<td>Index Style adaptationStereotype</td>
</tr>
<tr>
<td>ELSE IF (user blind OR low_vision OR colour_blindness OR photosensitivity OR eyestrain) AND not more impairments THEN</td>
</tr>
<tr>
<td>Sorted Style adaptationStereotype</td>
</tr>
<tr>
<td>ELSE</td>
</tr>
<tr>
<td>Narrow Style adaptationStereotype</td>
</tr>
</tbody>
</table>

As stated in the algorithm, if the user only suffers from impairments for using the mouse correctly the Index Style stereotype will be assigned. If the user only suffers visual impairments (blindness, low vision, colour blindness, photosensitivity or eyestrain) the Sorted Style stereotype will be assigned. The last case is used for users with a combination of impairments of the 3 general groups: cognitive, physical and sensory impairments. The Narrow Style stereotype will be assigned in this last case. This stereotype covers adaptation techniques for the 3 different general user groups. We assume that this stereotype is also appropriate for users with impairments not covered by the other two ones as well as when a user cannot be correctly classified (for example, when we have insufficient information about user characteristics).

The following sections describe each stereotype in detail.

6.2.1 Narrow Style
This stereotype is intended to solve problems related to movement, vision, attention and memory. It includes adaptation techniques of different subgroups: P.1 Limited movement, S.S.1 Low Vision, C.1 Decline in maintaining attention, C.4 Reduced memory capacity, S.S.5 Eye strain.
Table 4 shows: the total number of techniques considered in this stereotype, classified by group (Content adaptations, Presentation adaptations and Navigation adaptations); the number of techniques implemented in the repository; and the number of techniques applied for the Discapnet website.

A total number of 32 techniques have been applied for developing this stereotype. Most of these are related to presentation issues (21). If we analyze the adaptation techniques included in these subgroups we can find several techniques related to simplifying the interface in terms of structuring content in a hierarchical way, dimming insignificant content, minimizing content for limiting the use of scrolling, using the stretchtext technique, hiding insignificant links, etc. Another group of techniques is related to the layout of the web page such as linearizing content, presenting content centrally, providing a table of contents, navigation bars, etc. Finally, there is a third group of techniques related to the style of elements such as adding hotarea to links (this enables clicking in a wider area), only using underlining for links, always displaying links in blue, adding icons to links, etc.

### Table 4. Number of adaptation techniques considered, number of adaptation techniques implemented, and total number of techniques applied for the Narrow Style stereotype, classified by group (content, presentation and navigation)

<table>
<thead>
<tr>
<th>Group</th>
<th>No. of Techniques</th>
<th>Implem. Techniques</th>
<th>Applied techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cont.</td>
<td>20</td>
<td>11</td>
<td>6</td>
</tr>
<tr>
<td>Pres.</td>
<td>24</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>Nav.</td>
<td>12</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>Total</td>
<td>56</td>
<td>40</td>
<td>32</td>
</tr>
</tbody>
</table>

All these techniques have been considered in the stereotype developed and some of them are implemented in conjunction. For example, a table of contents is created with all navigation elements and the stretchtext technique is applied for simplifying it so that not all the sections and subsections are shown. The elements in this table of contents are links with an added hotarea and an icon. When clicking on the link or icon, subsection links are extracted and the content is accessed directly by clicking on them. The web page is linearized and the information is presented as close to the centre as possible. The conjunction of these techniques allows the development of a simplified interface, limiting the use of scrolling and with quick access to the important web page content.

In addition, style issues such as only using blue for links, underlining links, enlarging fonts and showing large icons, providing high contrast, double spacing in text, etc. have also been considered. Figure 7 shows a Narrow Style stereotype for the Discapnet website.

### 6.2.2 Index Style

This stereotype is intended to solve problems related to movement. Specifically it is aimed at users who cannot use a mouse. It considers adaptation techniques for the P.2 subgroup.

The following table (Table 5) shows: the total number of techniques considered in this stereotype, classified by group (Content adaptations, Presentation adaptations and Navigation adaptations); the number of techniques implemented in the repository; and the number of techniques applied for the Discapnet website.

The appearance of the web pages in this stereotype is quite similar to the original ones. Only 4 adaptation techniques are applied, 3 of which are related to navigation.

### Table 5. Number of adaptation techniques considered, number of adaptation techniques implemented and total number of techniques applied for the Index Style stereotype, classified by group (content, presentation and navigation).

<table>
<thead>
<tr>
<th>Group</th>
<th>No. of Techniques</th>
<th>Implem. Techniques</th>
<th>Applied techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cont.</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Pres.</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Nav.</td>
<td>6</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>Total</td>
<td>9</td>
<td>6</td>
<td>4</td>
</tr>
</tbody>
</table>

However, they completely change the navigation mode in the website. In fact, the most significant adaptation is the new navigation structure added in the left side of the screen. This navigation structure is always accessible as an add-on sidebar. Navigation menus are removed from web pages and all the options are shown in the sidebar. This acts as a table of contents for the whole website. Therefore, users can easily access any content in the website by just using the keyboard. The element with focus is highlighted and techniques such as AccessKeys have been included to move the focus from the sidebar to the web pages and back. Figure 8 shows the Index Style stereotype for the Discapnet website.

### Figure 8. Index Style stereotype for the Discapnet homepage

### 6.2.3 Sorted Style

This stereotype is intended to solve problems related to sight disabilities. It includes adaptation techniques for different subgroups: S.S.1 Low Vision, S.S.2 Blindness, S.S.3 Colour blindness, S.S.4 Photosensitivity, S.S.5 Eye strain.

Table 6 shows: the total number of techniques considered in this stereotype, classified by group (Content adaptations, Presentation adaptations and Navigation adaptations); the number of techniques implemented in the repository; and the number of techniques applied for the Discapnet website.

![Figure 7. Narrow Style stereotype for the Discapnet homepage](image-url)
In addition, presentation issues such as greater spacing between lines and characters, enlargement of images and fonts, high contrast using negative polarity, etc. have been considered.

In the case of Discapnet (and also for Bidasoa Turismo) there is no need to apply content adaptations for improving the experience of users suffering from photosensitivity, such as removing any flashing or blinking content. Figure 9 shows the Sorted Style stereotype for the Discapnet website. Figure 10 shows these 3 stereotypes for Bidasoa Turismo.

Table 6. Number of adaptation techniques considered, number of adaptation techniques implemented and total number of techniques applied for the Sorted Style stereotype, classified by group (content, presentation and navigation).

<table>
<thead>
<tr>
<th>Group</th>
<th>No. of Techniques</th>
<th>Impel. Techniques</th>
<th>Applied techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cont.</td>
<td>8</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Pres.</td>
<td>25</td>
<td>23</td>
<td>12</td>
</tr>
<tr>
<td>Nav.</td>
<td>9</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>Total</td>
<td>42</td>
<td>33</td>
<td>15</td>
</tr>
</tbody>
</table>

7. CONCLUSIONS AND FUTURE WORK

The authors gratefully acknowledge the support of the Spanish Ministry of Science and Innovation through Project ModelAccess (TIN2010-15549). The EGOKITUZ Research Laboratory is supported by the Department of Education, Universities and Research of the Basque Government (Eusko Jaurlaritza/Gobierno Vasco) through Grant# IT-395-10.

REFERENCES
