

User-Tailored Web Accessibility Evaluations

Markel Vigo
University of the Basque
Country
Informatika Fakultatea
20018 Donostia, Spain
+34 943 015113
markel@si.ehu.es

Alfred Kobsa
University of California,
Irvine
Department of Informatics
CA 92697, USA
+1 949 485-5020
kobsa@uci.edu

Myriam Arrue
University of the Basque
Country
Informatika Fakultatea
20018 Donostia, Spain
+34 943 015160
myriam@si.ehu.es

Julio Abascal
University of the Basque
Country
Informatika Fakultatea
20018 Donostia, Spain
+34 943 018067
julio@si.ehu.es

ABSTRACT

This paper presents a framework and system to evaluate the accessibility of web pages according to the individual requirements of users with disabilities. These requirements not only consist of users' abilities, but also users' assistive technologies and the delivery context. In order to ascertain interoperability with other software components, user requirements are specified taking advantage of the extensibility of the W3C CC/PP recommendation and other feature-specification vocabularies. An evaluation tool capable of understanding these specifications generates evaluation reports that are tailored to the user's individual needs. Quantitative accessibility measures resulting from personalized evaluation reports can be used to improve the web browsing experience for users with disabilities, such as through adaptive navigation support and by sorting the results of search engines according to users' personal requirements. In addition, developers benefit from personalized evaluations when developing websites for specific audiences.

Categories and Subject Descriptors

H.1.2 [User/Machine Systems]: Human factors. H.5.2. [User Interfaces]: Evaluation. H.5.4. [Hypertext/Hypermedia]: User issues. K.4.2 [Social Issues]: Assistive technologies for persons with disabilities.

General Terms

Human Factors, Standardization, Measurement.

Keywords

Web accessibility, user-tailored evaluation, guidelines, personalization, assistive technologies.

1. INTRODUCTION

Web pages that comply with general accessibility guideline sets can still fail to be accessible since some users may still

experience problems. For instance, Correani et al. [10] point out that vision-impaired users may still encounter the following problems even when web pages meet the WCAG 1.0 Guidelines [8] and the Section 508 Standards [25]:

- *Lack of context.* The overall context is lost when visually impaired users employ screen readers or screen magnifiers to scan the content.
- *Information overload.* Navigation is slowed down when navigation bars, menus or banners are read time and again.
- *Excessive sequencing when reading the information.*

In addition, Theofanos and Redish [27] state that meeting accessibility guidelines does not ensure the usability of a site, and enumerate some usability guidelines in order to build usable accessible sites.

Conversely, web pages not fulfilling accessibility guidelines can nevertheless still be accessible for certain user groups. For example, neither deaf nor physically handicapped users require alternative descriptions of pictures. General accessibility guidelines are an invaluable tool for guiding web designers in developing sites that are most likely accessible. From an end user's point of view, however, evaluation reports obtained from tools that evaluate against general accessibility guidelines, and accessibility metrics based thereon, are not very useful since they do not cater to users' individual needs and abilities.

To ameliorate the situation, Brajnik [5] grouped the WCAG 1.0 guidelines by their impact on particular user groups, namely blind, low-vision, deaf, colour blind and physically handicapped users, as well as people with cognitive disabilities. A similar approach can be applied to accessibility metrics: metrics can be developed taking into account just those guidelines that have potential impact on a specific group of users.

Selecting the guidelines that have an impact on one's specific user groups would in general seem sufficient for evaluating the accessibility of a website for oneself. However, there are several cases in which grouping guidelines by their impact on user groups is not enough, neither for evaluation purposes nor for assigning a quality rating in an accessibility metrics:

- *The guideline contains unresolved references to the user's delivery context.* In such a case, the guideline cannot be properly evaluated without concrete information about the usage situation. For instance, Mobile Web Best Practices 1.0 [23] proposes not to use images wider than the width of the screen, so as to avoid horizontal scrolling. In this particular

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

HT'07, September 10–12, 2007, Manchester, United Kingdom.
Copyright 2007 ACM 978-1-59593-820-6/07/0009...\$5.00.

case, it is necessary to know the screen size of the access device so that evaluations can be adjusted to each case.

- *The guideline is incomplete.* In such a case, additional information is required for an accurate evaluation of the guideline. For instance, WCAG 1.0 guidelines 12.1 and 12.2 identify accessibility problems regarding the use of frames and define guidelines for how to deal with them. However, even when designers adhere to these guidelines, some blind users may still have accessibility problems due to the assistive technology they use (e.g., old versions of the JAWS screen reader that are unable to detect frames). This problem is faced quite often since people with disabilities tend to be reluctant to change or upgrade their assistive technologies [9].
- *Users' individual needs are not well covered in group guidelines.* Group guidelines capture "typical" needs of their respective group members, but individual needs may deviate considerably (e.g., individual motor-impaired people having more residual physical abilities than the group guidelines foresee).
- *Multiple group membership is not supported by evaluation tools.* Users can be members of different user groups (e.g., deaf users tend to also have cognitive disabilities, and hence they additionally require some guidelines for users with cognitive impairments). A similar situation occurs when physically handicapped users interact with handheld devices: guidelines for both mobile web devices and physical disabilities have to be taken into account.

The conclusion from these problems is that assistive technologies, access device features and shared group membership have to be considered in user-specific accessibility evaluation and measurement. This paper proposes a fine-grained approach for this purpose. It uses *personal accessibility profiles* containing individual disabilities, assistive technologies and access device constraints to dynamically generate personalized accessibility evaluation reports, while the user is browsing a website. *Accessibility scores* inferred from personalized evaluations can be employed to signal to the user the degree of accessibility of web pages in the current use context, even before the user visits these pages.

2. SCENARIOS FOR PERSONAL ACCESSIBILITY

Personal accessibility evaluations can be performed in various situations to improve the web experience of users with disabilities, and to help in the website development process.

2.1 Adaptive Navigation Support

Orientation and navigation are closely related since both refer to the user's navigational environment. Navigation is part of web browsing and consists in moving around in an electronic environment, deciding at each step where to go next [19]. Orientation is the user's understanding of current movements and the navigation context. The former answers the question "where can I go?" while the latter replies to "where am I?" According to Goble et al. [11], visually impaired users need to be explicitly warned of obstacles since their reliance on environmental cues is higher than for sighted users. Harper et al. [14] found that detecting and notifying users about barriers beforehand improves users' orientation at a website.

Local Guidance [6] can be provided with this aim in mind, by indicating the most accessible links at the current web page according to the user's accessibility profile. *Adaptive sorting* can be performed by including a list of the most accessible links at the top of every page or by including it in the browser GUI. *Local orientation support* can be obtained by annotating links with the quantitative accessibility rating of the linked web pages, so that users become informed about access barriers beforehand. In all cases, the linked pages of the current page would be pre-fetched and evaluated on the fly.

The set of all accessibility measurements of upcoming links can be understood as a mobility ratio [14], i.e. an indicator of how well people will be able to navigate a web page environment. Depending on the user's disabilities, these "clues" should have different formats: numeric link labels in the case of visually impaired users, and icons or colour codes for all other users with disabilities (or both, since the techniques are complementary).

In addition to *Local Guidance*, *Global Guidance* can also be provided, using e.g. learning strategies such as in [20]. Knowing a user's goal would then be reason enough to suggest "the most accessible path" for this objective.

2.2 Information Retrieval Systems

A study by Ivory et al. [18] with visually impaired users concludes that some of them desire to see additional details in search result lists. The authors therefore recommend to sort or rank search results according to accessibility or usability. We can broaden this claim to include all disabilities. The implementation of this approach is quite straightforward: accessibility scores inferred from *personal accessibility profiles* could be included in IR systems, so that results are sorted by their accessibility, or the original results of the IR system could be re-ranked at the client side.

Sorting of search engine results and adaptive navigation support are complementary and can therefore also be combined. After querying a search engine, the user can select the best result from a list of items that was ranked or re-ranked according to his/her accessibility profile. Once the user visits the selected web page, link annotations indicating the accessibility values of the linked pages are also provided.

2.3 Developing Websites for Specific Audiences

Not only end users can benefit from personalized evaluations, but also web developers. Users' particular disabilities, assistive technologies, access devices and multiple group memberships can be considered when developing a website. From the developer's point of view, these requirements form part of the contextual requirements for accessible sites [26]. Developers can define or retrieve user profiles and evaluate their designs against them, and thereby use our framework in building sites for specific audiences.

3. FRAMEWORK FOR PERSONAL ACCESSIBILITY EVALUATIONS

This section describes the components to automatically create personal accessibility profiles and calculate accessibility scores.

Each component in the architecture can be considered to be autonomous with its own internal logic.

First of all, we need an *accessibility evaluation engine* that is generic enough to be able to evaluate different guideline sets, since we aim at evaluating the accessibility problems of different access devices and users. To this end, we need a *repository of machine-understandable guideline sets*. In order to automatically detect system features, a *module for system features detection* gathers information about installed assistive technologies, hardware constraints, etc. The output of this module is used to create *personal accessibility profiles*. Using these profiles, a *module for guidelines instantiation* can retrieve guidelines regarding the user's delivery context and fill in the unresolved and incomplete requirements that some guidelines have (see Section 1). For this reason, a *common vocabulary* for describing user features becomes necessary, so that all components in the framework can understand the information in profiles. Finally, we need a module that calculates the quantitative *accessibility score* based on reports produced by the evaluation engine.

3.1 Guidelines Repository

Guidelines are stored in an XML repository so that every system component can access them. They should be machine readable since they are going to be evaluated by an autonomous module, the Evaluation Engine. We chose to implement guidelines in XML due to the fact that it is a very flexible standard for data exchange. After studying the state-of-the-art on accessibility guidelines, we developed a Uniform Guidelines Language (UGL) which is able to express a large number of guideline sets [2]¹. UGL is capable of implementing guidelines regardless of their abstraction level (generic or specific), access devices, application type, or any combination thereof. In [3] we present a management system for UGL that is extremely useful for creating or editing guidelines in this repository. Developers who are familiar with the representation of guidelines in UGL can edit them directly, while those who are less experienced have a web interface available that guides them in editing guidelines via XHTML forms.

3.2 System Features Detector

The objective of this module is twofold: retrieving information about the user's access device and detecting installed assistive technologies (ATs). The first task can be carried out using system APIs offered by programming languages, and would consist of obtaining device characteristics such as screen size, keyboard type, image format support, script languages supported by browsers, and colour support. The second task is accomplished by querying the System Registry in the case of Windows, or the `/etc/...` files in some Linux distributions. These queries are used to check whether a particular assistive technology is installed (ATs includes both hardware such as Braille displays or input devices, and software such as screen readers, screen magnifiers etc.). These queries compare database information on available ATs with key values in the System

Registry.² When a match is found, information regarding the AT is obtained from both information sources (database and Registry).

3.3 A Vocabulary for User Profiling

The hardware and software characteristics detected by the System Features Detector become entered into the user's profile. A user profile thus contains information on hardware and software constraints, as well as the user's assistive technologies. This *personal accessibility profile* is implemented in Composite Capabilities/Personal Profiles (CC/PP) [7], which is a W3C standard for modelling user profiles with a common vocabulary based on the Resource Description Framework RDF [24]. CC/PP offers a framework to describe device capabilities and user preferences. It is often used for content negotiation between user agents and web servers. If extended correctly, CC/PP can also be used for adapting to user's disabilities [30]. However, the vocabulary of CC/PP for describing user profiles is limited. In order to enhance the expressiveness of CC/PP descriptions, it is usually necessary to define new vocabularies (the reuse of existing vocabulary extensions is thereby strongly encouraged). In this sense, the IMS Consortium has developed the IMS Accessibility Learner Profile (IMS ACCLIP) [17], an XML-based language focused on content adaptation issues in e-learning environments.

To evaluate the coverage and expressiveness of our CC/PP based *personal accessibility profile*, we analyzed the following accessibility guidelines: WCAG 1.0 [8], MWBP 1.0 [23], the IBM Accessibility Guidelines [16], the Web Design Guidelines for Elderly Users (DGEU) [21], and the IMS Guidelines for Developing Accessible Learning Applications (IMS) [4]. The analysis of these guidelines and their relationship to assistive technology and access devices is crucial to the design of a user-tailored vocabulary which takes into account disabilities, access constraints and the limitations of assistive technologies.

3.3.1 Assistive Technologies

Assistive technologies play an important role in the application of accessibility guidelines. In their evolution over the years, they have increasingly gained control over formerly completely inaccessible content, thus making browsing easier for users with disabilities. Every new version of an assistive technology addresses new accessibility issues. Strict version control is therefore necessary since several guidelines are contingent on different versions of user agents. These dependencies can be grouped into two main categories:

Negative dependencies. Even if guidelines are fulfilled, some versions of ATs may not convey the content of a web page adequately. For instance, even when a change in the language of web content is correctly flagged using the `lang` attribute (see checkpoints 4.1 and 4.3 in WCAG [8]), the content will not be adequately read out to users who are running older versions (<5.0) of the Jaws screen reader since those are not capable of recognizing this HTML attribute.

¹ The XML-Schema is available at <http://sipt07.si.ehu.es/evalaccess3/ugl.xsd>, and its graphical representation at <http://sipt07.si.ehu.es/evalaccess3/ugl.png>

² This database on ATs must currently be maintained manually, which is a major drawback.

Table 1. Features supported by consecutive versions of the Jaws screen reader and their impact on different guideline sets

Version	Features	dep.	WCAG	MWBP	IBM	DGEU	IMS
3.71	Frames navigation	-	1.1, 12.1, 12.2	5.4.2	9	-	6.1, 6.3, 8.2
4.01	Headings (H1-H6) recognition	-	3.5, 12.3	-	8	5.3, 9.5	-
	Recognition of <i>OnMouseOver</i> event	+	6.3, 6.4, 8.1, 9.3	5.4.5	5	-	6.4, 8.2
	<i>AccessKeys</i> supported	-	9.4	5.2.5	-	-	6.3, 8.1, 8.2
	<i>TabIndex</i> supported	-	9.5	5.5.2	-	-	6.1, 6.3, 8.1, 8.2, 8.3, 8.4
	<i>Longdesc</i> supported	-	1.1, 12.2	5.4.5	3	-	5.1.3
4.02	<i>Legend</i> element within <i>Fieldset</i> is recognized	-	12.3	-	7	-	-
4.5	Control over refreshing is supported	+	7.4, 10.1	5.2.8	13	-	-
4.51	Control over <i>iframes</i> is supported	+	1.1, 12.1, 12.2	5.4.2	9	5.2	6.1, 6.3, 8.2
	<i>Title</i> attribute in <i>acronym</i> and <i>abbr</i> is read	-	4.2	-	-	-	-
5	Automatic language detection	-	4.1, 4.3	-	-	-	-
	<i>OnClick</i> supported	+	6.3, 6.4, 8.1, 9.3	5.4.5	5	-	6.4, 8.2
6	Linking of headers to data cells in tables using <i>scope</i> attribute	-	5.1, 5.2	-	10	-	5.1.1, 6.4

Table 2. The most frequently available Assistive Technologies

Product	Users	Assistive Technology
1. Jaws, 2. Window-Eyes	Blind	Screen Reader
1. Zoomtext, 2. Magic	Partially blind	Screen Magnification
1. Dragon Naturally Speaking, 2. Math Talk	Physically disabled	Speech Recognition
1. TextHelp, 2. Co:Writer	Cognitive, Physically handicapped	Word Prediction
1. Kensington Trackballs, 2. Logitech Trackballs	Physically disabled	Mouse Alternatives
1. HeadMaster, 2. HeadMouse	Physically disabled	Hands-free/ Speech-free input
1. Tiger Embosser	Blind	Braille Embosser

Positive dependencies. On the other hand, having a particular AT makes the application of some guidelines obsolete. When WCAG guidelines were published in 1999, it was foreseen that some accessibility problems would be fixed by user agents as technology evolves. Positive dependencies in our framework represent “until user agents...”³ statements in WCAG guidelines that meanwhile became true. As a result, some guidelines do not apply any more when a user employs AT that tackles the accessibility problems described in these guidelines. For instance, version 4.5 of the Jaws screen reader provides mechanisms to control auto refreshing. Thus, checkpoint 7.4 of WCAG is not a problem any more for users who run this or higher screen reader versions, and this guideline should not be taken into account for such users.

Table 1 gives an example of positive and negative dependencies. Specifically, it shows how the above-mentioned guideline sets impact different versions of the Jaws screen reader (or in other words, how each new release deals with accessibility guidelines). The third column specifies whether the version gives rise to a positive (+) or a negative (-) dependency. The remaining columns indicate the guidelines checkpoints that have been newly addressed by the specific release.

Since most guidelines have an impact on particular user groups, it is necessary to know the disability of the potential user of a specific assistive technology. The disability of the current user is used to discard guidelines that do not have a direct impact on him/her. Generic information regarding the type of assistive technology is useful since it gives clues to infer the abilities of the potential users. Table 2 summarizes the most frequently available AT products grouped by type and potential users. This data has been obtained through a survey on the usage of ATs in higher education that was carried out in 2004 [28]. Our extended CC/PP language considers the type of assistive technology, its product name, its version and its potential users. As is explained above, this entails extending the vocabulary of CC/PP in most cases. Figure 1 shows an excerpt of the definition of new words in a vocabulary as CC/PP attributes.

It can be observed in Figure 1 how the words *ATtype*, *ATname*, *version* and *user* are defined in a vocabulary called *access*. These words can univocally describe the name, version, potential users, etc. of an Assistive Technology. Figure 2 shows how the new vocabulary is used to enhance the CC/PP file. An Assistive Technology is described as a software feature there.

³ See <http://www.w3.org/TR/WAI-WEBCONTENT/wai-pageauth.html#until-user-agents>

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:ccpp="http://www.w3.org/2002/11/08-ccpp-schema#"
<ccpp:Attribute rdf:ID="ATtype">
  <rdfs:label xml:lang="en">ATtype</rdfs:label>
  <rdfs:comment xml:lang="en">
    Generic group of Assistive Technologies
  </rdfs:comment>
  <rdfs:domain rdf:resource="http://www.w3.org/2002/11/08-ccpp-
schema#Component"/>
  <rdfs:range rdf:resource="http://www.w3.org/2002/11/08-ccpp-
schema#string"/>
</ccpp:Attribute>
<ccpp:Attribute rdf:ID="version">
  <rdfs:label xml:lang="en">version</rdfs:label>
  <rdfs:comment xml:lang="en">
    Version of Assistive Technology
  </rdfs:comment>
  <rdfs:domain rdf:resource="http://www.w3.org/2002/11/08-ccpp-
schema#Component"/>
  <rdfs:range
rdf:resource="http://www.openmobilealliance.org/tech/profiles/UA
PROF/xmlschema-20030226#Number"/>
</ccpp:Attribute>
<ccpp:Attribute rdf:ID="user">
  <rdfs:label xml:lang="en">user</rdfs:label>
  <rdfs:comment xml:lang="en">
    Potential user of a particular Assistive Technology
  </rdfs:comment>
  <rdfs:domain rdf:resource="http://www.w3.org/2002/11/08-ccpp-
schema#Component"/>
  <rdfs:range rdf:resource="http://www.w3.org/2002/11/08-ccpp-
schema#string"/>
</ccpp:Attribute>

```

Figure 1. Excerpt of the vocabulary definition for ATs.

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
ns#"xmlns:ccpp="http://www.w3.org/2002/11/08-ccpp-schema#"
xmlns:access="http://sipt07.si.edu.es/profiles/access/schema#"
<rdf:Description
rdf:about="http://sipt07.si.edu.es/profiles/prof00001">
<ccpp:component>
  <rdf:Description rdf:about="
http://sipt07.si.edu.es/profiles/prof00001#SoftwareFeature_1">
    <access:ATtype>screen reader</access:ATtype>
    <access:ATname>Jaws</access:ATname>
    <access:version>3.0</access:version>
    <access:user>blind</access:user>
  </rdf:Description>
</ccpp:component>

```

Figure 2. Usage of the new vocabulary in a CC/PP profile.

3.3.2 Access Device

Not only traditional accessibility has to be appraised when dealing with web content, but also access device specific features have to be considered in the evaluation process (this is specifically true for the mobile web with its plethora of different access devices). Therefore, our framework not only deals with accessibility mark-up in web documents, but also with the evaluation of physical features and software support limitations in handheld devices that people use to access these documents. Physical limitations of the access device cause accessibility problems specifically for handheld devices. Limitations of the display, the available bandwidth and of input mechanisms (such

as the absence of a pointing device) determine the quality of the interaction. In addition, since computing resources such as CPU and memory are also limited, there exists a lack of support for well-established web mark-up and scripting languages (such as XHTML and JavaScript), styles (such as CSS), and picture formats or encodings. Considering these features is necessary to carry out accessibility evaluations with respect to the current access device. MWBP 1.0 is the guideline set that contains the most best-practice guidelines referring to access devices.

Table 3 shows how some best practices determine the concepts that need to be introduced in the personal accessibility profile. The first column contains the best practice name from MWBP 1.0 and the second column its description. The subsequent columns indicate the concept that needs to be described in the profile to capture the requirements of the best practice, its data type, and the (new) word for it in the vocabulary. If available, words from existing vocabularies are being used as long as they refer to the same concept and type, such as the User Agent Profile⁴ (UAProf) by the Open Mobile Alliance⁵ (OMA). Note that words borrowed from UAProf vocabulary have a `prf` prefix while our vocabulary uses the `access` prefix.

3.4 Enhancing UGL with Metadata

To date, guidelines implemented in UGL are not able to refer to AT features or access device constraints. Therefore, we are currently extending the language so that AT and device features can also be evaluated.

3.4.1 Considering Assistive Technologies

Until now, UGL could express requirements for HTML tags and attributes only. The forthcoming version of UGL will be able to also evaluate AT features, in addition to web content. The dependencies identified in Section 3.3.1 are thereby dealt with in different ways.

Negative Dependencies

For negative dependencies, the guideline and the version of the related assistive technology have to be evaluated (information about the latter can be found in the CC/PP profile). Figure 3 shows an example. The guideline specifies that whenever a frame is encountered, the evaluation engine should check the name of the screen reader. If the name is JAWS, it has to check the version number and report an error if it is less than 3.71.

```

<element>
  <label>FRAME</label>
  <analysis_type>check AT</analysis_type>
  <related_AT>
    <AT type="access:ATname">Jaws</AT>
    <analysis_type>value</analysis_type>
    <content type="access:version" dep="-">3.71</content>
  </related_AT>
</element>

```

Figure 3. Piece of a guideline in UGL dealing with negative dependencies.

⁴<http://www.openmobilealliance.org/tech/profiles/UAPROF/ccp-pschema-20030226>

⁵<http://www.openmobilealliance.org/>

Table 3. Concepts and related words to describe device dependencies in MWBP 1.0

Best practice name	Description	Required information in accessibility profile	Type	Word
<i>Navigation and Links</i>				
ACCESS_KEYS	When pointing device support is lacking, access keys provide mechanisms for efficient browsing with the keyboard.	pointing device support?	boolean	access:pntSupport
		keyboard support?	boolean	access:kbdSupport
LINK_TARGET_FORMAT	A link may lead to a non-supported object: web page, picture or file.	supported formats	resource	access:formatSupport
IMAGE_MAPS	Server-side images can only be accessed with a pointing device. Check client-side maps support	pointing device support?	boolean	access:pntSupport
		client-side map support?	boolean	access:csmSupport
POP_UPS	Check multi-window support	multi-window support?	boolean	access:mwSupport
<i>Page Layout and Content</i>				
PAGE_SIZE_LIMIT	Check that the memory size is greater than the web page size	available memory size	number	access:memSize
SCROLLING	Check that pictures are smaller than the screen size	available screen size	dimension	prf:ScreenSize
LARGE_GRAPHICS	Check whether the graphics resolution is higher than is displayable	available screen resolution	dimension	prf:ScreenSize
<i>Page Definition</i>				
NO_FRAMES	Check frames support	frames supported?	boolean	prf:FramesCapable
TABLES_SUPPORT	Check tables support	tables supported?	boolean	prf:TablesCapable
OBJECTS_OR_SCRIPT	Check script or other object support (flash, applets and so on)	supported formats	resource	access:formatSupport
			boolean	prf:JavaScriptEnabled
			boolean	prf:JavaAppletEnabled
STYLE_SHEETS_USE	Check CSS support	CSS support	resource	access:formatSupport
CHARACTER_ENCODING_SUPPORT	Check character encoding support	supported encodings	literal	prf:OutputCharSet

Note that the guideline definition includes words from the access vocabulary such as ATname and version. The evaluation engine knows how to match these fields with their correspondent values in CC/PP since they have the same name.

Positive Dependencies

Positive dependencies decide whether or not a guideline will be retrieved from the guidelines repository. Guidelines related to positive dependencies of a particular version of an AT will not be added to the final guideline set since they are redundant for this version.

```
<element>
  <label>IFRAME</label>
  <analysis_type>check AT</analysis_type>
  <related_AT>
    <AT type="access:ATname">Jaws</AT>
    <analysis_type>value</analysis_type>
    <content type="access:version" dep="+">4.51</content>
  </related_AT>
</element>
```

Figure 4. A guideline fragment in UGL that handles a positive dependency.

The example in Figure 4 specifies that when an iframe is encountered, the evaluation engine should check the name of the screen reader. If the name is JAWS, it should check the version

and not retrieve the guideline if the version number is 4.51 or higher.

3.4.2 Considering Features of the Access Device

Since the features of the access device cannot be foreseen, guidelines need to deal with any kind of access device. Figure 5 shows an excerpt of a guideline definition in UGL with “hard-coded” characteristics. It gives roughly the following specification: “report an error when images whose width is greater than 250px are found”. This guideline would be useful for all displays whose width is 250 pixels or less.

```
<element>
  <label>IMG</label>
  <analysis_type>check attribute</analysis_type>
  <related_attribute>
    <atb>WIDTH</atb>
    <analysis_type>value</analysis_type>
    <content test=">">250px</content>
  </related_attribute>
</element>
```

Figure 5. Piece of guideline code in UGL that deals with device features.

However, our system must be able to evaluate any access device no matter what its features are, and these features are not known until the time of the evaluation. This requires the introduction of

slots into guidelines definitions so that elements in the guidelines can represent any value. Since the value of the slot will be found in the personal accessibility profile, the same names are being used in the profile and the guidelines. Using the same vocabulary has one big advantage: guidelines in UGL and CC/PP files can “talk” in the same language and therefore their interoperability is enhanced. Figure 6 shows how the word `ScreenSize` from the vocabulary `prf` has been included. This guideline will report an error every time a picture is wider than the current width of the screen, and the evaluation engine will be able to evaluate this guideline for any device, no matter what its screen width is. As a result, a typed language enhanced with semantic information is obtained, such as in [15].

```

<element>
  <label>IMG</label>
  <analysis_type>check attribute</analysis_type>
  <related_attribute>
    <atb>WIDTH</atb>
    <analysis_type>value</analysis_type>
    <content test="" type="prf:ScreenSize"/>
  </related_attribute>
</element>

```

Figure 6. Piece of code of an enhanced guideline in UGL.

3.5 Guidelines Instantiator

The task of the Guidelines Instantiator (GI) is to retrieve those guidelines that have an impact on the current user, and to complete missing specifications in these guidelines based on information in the personal accessibility profile.

Three criteria are used to determine whether a guideline is relevant for the current user:

- Since CC/PP profiles store the handicap categories of users (in a field named `access:user`), the GI just matches these values with values of the same field in the UGL guidelines (e.g., `<access:user> partially blind </access:user>`).
- Using access device information in the CC/PP file (see Table 3), the GI infers whether the user is accessing with a handheld device. In this case, mobile web accessibility guidelines are also retrieved.
- Guidelines related to positive dependencies are found by comparing users' AT versions with the `access:version` field in each guideline. If the AT version is higher than what the guideline specifies, this guideline becomes removed from the set of guidelines that were retrieved in the previous two steps.

The second task of the GI is to complete missing specifications in guidelines with concrete values from the CC/PP. In this vein, the name of the variable in the guideline is matched with the field names in the CC/PP file. An example can be found in Figure 7, where the value of `prf:ScreenSize` (namely 320) is copied into the empty slot in the guideline definition.

Once the guidelines have been retrieved and complemented, they are ready to be evaluated by the Evaluation Engine.

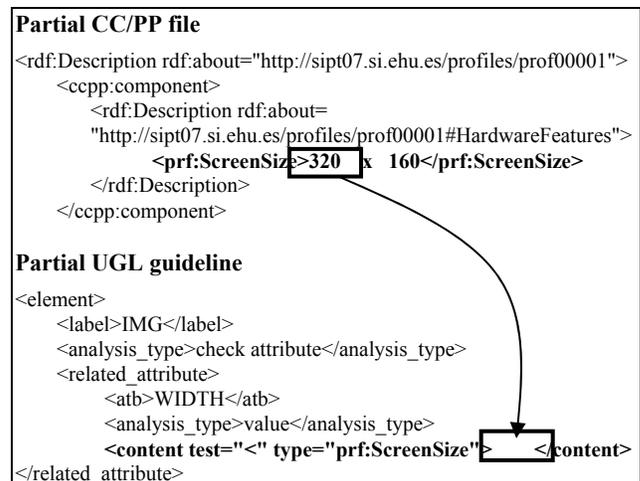


Figure 7. Filling in guideline slots with values from the CC/PP profile.

3.6 Evaluation Engine

The evaluation engine evaluates web pages against the *ad-hoc* created guidelines and the personal accessibility profile. Evaluating different sets of guidelines requires a tool in which the evaluation logic is separated from the implementation of the guidelines. This is a valuable measure to achieve modularity, which also entails more straightforward updates since new guideline sets or upgrades of existing guidelines will not require changing the evaluation engine. Previous research in this regard has been carried out by [1], [29] and [22].

The evaluation engine produces an evaluation report with detailed information on accessibility problems. Potentially, it can evaluate the following types of accessibility issues:

- issues with elements, attributes and the content of hypertext documents,
- issues with Assistive Technologies which may raise a barrier, and
- problems with access device features in combination with elements and attributes.

3.7 Metrics Calculation Module

For every web page, the Metrics Calculation Module computes a numerical accessibility score from the personalized reports. This value between 0 and 100 will tell users how accessible a web page is according to their individual disabilities and access environments. A metric has been defined that takes into account: the number of guidelines in the guideline set, their priority within the guideline set, the number of guidelines whose evaluation resulted in error notices, and the number of guidelines whose evaluation resulted in warnings that still have to be manually verified. The algorithm has been described in more detail in [31] for the WCAG guidelines, but was meanwhile enhanced to take more than one guideline set as well as priorities between guidelines into account, to generate accessibility scores dynamically.

3.8 Putting all Together

Figure 8 depicts the modules described above, their locations, their interaction, as well as the processing steps in more detail.

Step 1. The AT Detector retrieves data from the AT data base.

Step 1'. The Access Device Features Detector obtains the characteristics of the access device.

Step 2. The AT Detector queries the System Registry as to whether the user employs any AT that was found in the AT data base.

Step 3. Information regarding the discovered ATs, their potential users and access device features are put together in a CC/PP file and sent to the server.

Step 4. The Guidelines Instantiator retrieves guidelines for the end users described in the CC/PP file from the Guidelines Repository. Information in the CC/PP file will also be used to fill in the slots in the guidelines. Additional meta-data about the selected guidelines are also supplied.

Step 5. A web page will be evaluated against the instantiated guidelines. A report of discovered errors and warnings is produced as a result.

Step 6. This evaluation report and metadata about the evaluated guidelines set are used by the Metrics Calculation Module for computing a personal accessibility score.

Step 7. A user agent (i.e. the user's browser) obtains the numerical score of the accessibility of all links in a web page. Thereafter, the user agent can apply all necessary user-adaptive modifications to the document object model of a page.

Steps 1-4 are carried out only once when the browser is launched. Steps 5-7 are carried out each time a web page is accessed.

3.9 Case Study

In order to demonstrate the validity of the proposed system and the necessity of our approach we created four different profiles. These profiles refer to four web interaction contexts and consider disabilities, assistive technologies and access devices:

- Profile 1 (P₁): a blind person using Jaws with version > 6.
- Profile 2 (P₂): a blind person using Jaws with version < 4.01.
- Profile 3 (P₃): a user accessing the web using a Nokia 9210 mobile phone.
- Profile 4 (P₄): a user accessing the web using a Blackberry 7750 with the RIM 3.7.3 browser.

The home pages of 10 sites that received Webby Awards⁶ in 2007 were evaluated using the above-mentioned profiles. The columns in Table 4 show the number of automatically reported errors for each profile. There is a noticeable difference between the results obtained with different profiles, and specifically between P₁ and P₂ versus P₃ and P₄. The main reason for this difference is that the former were evaluated against WCAG 1.0 [8] and the latter against MWBP 1.0 [23], and that the coverage of these guideline sets is different.

⁶ <http://www.webbyawards.com/index.php>

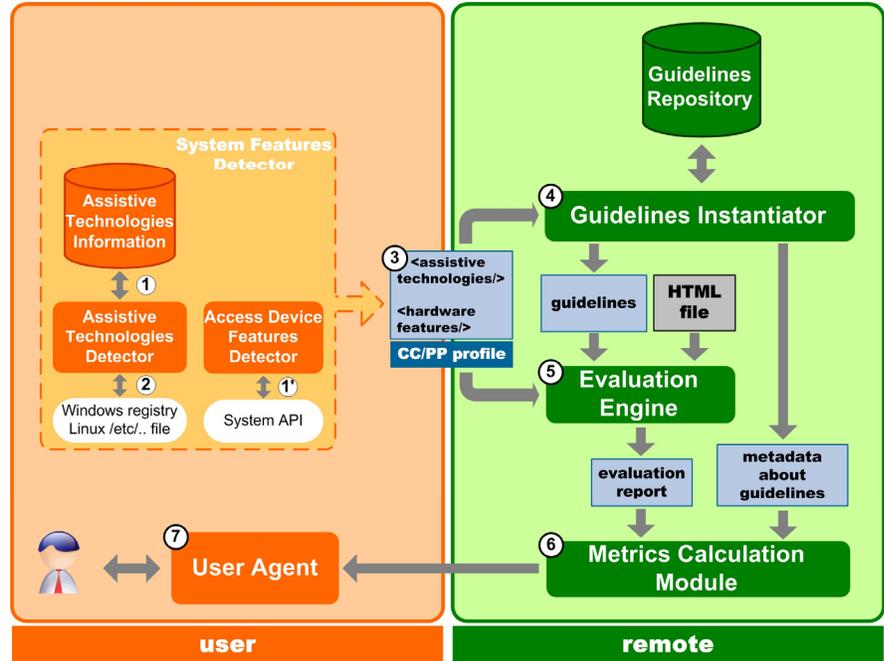


Figure 8. Architecture of the system.

Table 4. Websites and accessibility errors produced by different profiles

website	P ₁	P ₂	P ₃	P ₄
http://www.guardian.co.uk/	5	7	73	73
http://www.theonion.com/content/index	6	13	91	94
http://online.wsj.com/public/us	42	130	390	405
http://www.davidbowie.com/	1	1	15	15
http://dealbook.blogs.nytimes.com/	25	26	169	173
http://www.howstuffworks.com/	87	150	338	338
http://www.ehealthinsurance.com/	71	88	253	274
http://www.poetryfoundation.org/	97	131	289	300
http://www.mgmgrand.com/	127	209	271	328
http://photography.si.edu/	14	24	93	100

There is an obvious difference between P₁ and P₂ due to the different AT versions used (see Table 1). Newer versions of the Jaws screen reader provide better support for scripting, content refresh, updates and pop-ups. Lack of HTML support (such as table rendering and display limitations) makes the results for P₄ worse than for P₃.

If fewer errors are reported for one profile than for others this currently does not necessarily mean that a web page is more accessible for the respective user. This is due to the fact that the requirements of some user groups are often not so well defined or quite fuzzy (e.g. those for people with learning disabilities in WCAG 1.0), even when a guideline sets aims for full coverage. As a result, these users would obtain fewer errors due to guideline gaps or difficulties with the automatic evaluation of guidelines (such as the WCAG 14.1 guideline “Use the clearest and simplest language”). Moreover, obtained results actually constitute an upper bound for the degree of accessibility since

evaluation reports are based on automatic evaluation and must still be verified through manual evaluation.

4. FUTURE WORK AND CONCLUSIONS

Our system can be deployed in various ways: by building a new browser from scratch, adapting an open source browser, by using a browser plug-in, by using a server-side proxy that re-codes the hypertext document, or by combining some of these approaches (e.g., a proxy and a dedicated browser). Using proxies is problematic due to numerous problems that may arise, such as a lack of correctness of the transformations, and security, browser configuration, copyright and scalability problems [12]. As stated in [13], users prefer a standard browser over special browsers, hence creating a new browser is not going to fly. We made the decision to encapsulate and integrate our system as a browser plug-in. Every time a user accesses a site, the web pages behind every link are being rated according to the user's profile. Afterwards, by manipulating the DOM structure of the page, all links would be labelled with their accessibility scores and a list of the most accessible links could be also added. However, the more links a web page contains, the slower will be the transition between pages since evaluations and measurements may take a while. A solution could be to pre-fetch pages by more than just one level.

Regarding the modelling of users' abilities and available assistive technologies, the most meaningful information can be obtained from settings such as the user's language, the magnification level of the screen magnifier they use, or the speed of their screen reader. However, accessing those parameters is not always possible. An environment to annotate ATs with metadata about their current settings would help to accomplish this task. To this end, there should be an agreement between web developers and AT manufacturers to create a vocabulary or ontology for the description of ATs. In addition, we also consider using question answering dialogs or web forms to obtain complementary information about user settings if they cannot be obtained from the Registry.

RDF and more specifically CC/PP have proven to be useful for the definition of users' characteristics and the features of their access devices. From its early onset, the language of CC/PP was meant to include users' preferences, in order to transform content according to users' likings. Future releases of CC/PP could include the delivery context desired by users, independent of their abilities and access device.

Currently, accessibility scores are obtained from evaluation reports that are based on accessibility guidelines. We are aware that guidelines are sometimes insufficient to adapt web pages to user requirements. User needs can be so specific that the necessary adaptations to accommodate a given user are ones that apply to his/her *individual* abilities only and cannot be deduced from user disability group membership. However, users who are member of a group share similar restrictions/problems when browsing. Therefore something can be done in order to facilitate navigation sessions. The inclusion of guidelines or preferences that go beyond stereotypical user profiles needs however also be considered.

5. ACKNOWLEDGMENTS

Markel Vigo's work is funded by the Department of Education, Universities and Research of the Basque Government.

6. REFERENCES

- [1] Abascal, J., Arrue, M., Fajardo, I., Garay, N. and Tomás, J. The use of guidelines to automatically verify Web accessibility. *Universal Access in the Information Society* 3(1), 71-79, Springer, 2004.
- [2] Arrue M., Vigo M. and Abascal J. Including Heterogeneous Web Accessibility Guidelines in the Development Process. *Engineering Interactive Systems conference 2007* (Salamanca, Spain, March 22-24, 2007). To be published in LNCS, Springer, 2007.
- [3] Arrue, M., Vigo, M., Aizpurua, A. and Abascal, J. Accessibility Guidelines Management Framework. In C. Stephanidis (Ed.). *Universal Access in HCI, Part III, HCI International 2007* (Beijing, China, July 22-27, 2007). LNCS 4556, 3-10, Springer, 2007.
- [4] Barstow, C., McKell, M., Rothberg, M. and Schmidt, C. (2002, June). IMS Guidelines for Developing Accessible Learning Applications 1.0. <http://www.imsglobal.org/accessibility/accessiblevers/index.html>
- [5] Brajnik, G. Web Accessibility Testing: When the Method is the Culprit. In K. Miesenberger et al. (Eds.). *Computers Helping People with Special Needs, ICCHP 2006* (Linz, Austria, July 12-14, 2006). LNCS 4061, 156-163, Springer, 2006.
- [6] Brusilovsky, P. Methods and Techniques of Adaptive Hypermedia. *User Modeling and User-Adapted Interaction: The Journal of Personalization Research* 6 (2-3), 87-129, Springer, 1996.
- [7] CC/PP Information Page. Available at <http://www.w3.org/Mobile/CCPP/>
- [8] Chisholm, W., Vanderheiden, G. and Jacobs, I. (Eds.). (1999, May 5). Web Content Accessibility Guidelines 1.0. Available at <http://www.w3.org/TR/WAI-WEBCONTENT/>
- [9] Clark, J. *Building Accessible Websites*. New Riders, Indianapolis, IN, 2002.
- [10] Correani, F., Leporini, B. and Paternò, F. Supporting Web Usability for Vision Impaired Users. In C. Stary and C. Stephanidis (Eds.). *User-Centered Interaction Paradigms for Universal Access in the Information Society, UI4All 2004* (Vienna, Austria, June 28-29, 2004). LNCS 3196, 242-253, Springer, 2004.
- [11] Goble, C., Harper, S. and Stevens R. The Travails of Visually Impaired Web Travellers. *11th ACM Conference on Hypertext and Hypermedia, Hypertext 2000* (San Antonio, TX, USA, May 30-June 3, 2000). 1-10, ACM Press, 2000.
- [12] Hanson, V.L. and Richards, J.T. A Web Accessibility Service: Update and Findings. *ACM SIGACCESS Conference on Computers and Accessibility, ASSETS 2004*

- (Atlanta, GA, USA, October 18-20, 2004). 169-176, ACM Press, 2004.
- [13] Hanson, V.L., Brezin J.P., Crayne, S., Keates, S., Kjeldsen, R., Richards, J.T., Swart, C. and Trewin S. Improving Web accessibility through and enhanced open-source browser. *IBM Systems Journal* 44(3), 573-588, 2005.
- [14] Harper, S., Goble, C. and Stevens, R. Augmenting the mobility of profoundly blind Web travellers. *New Review of Hypermedia and Multimedia* 11(1), 103-128, Taylor & Francis, 2005.
- [15] Hunter, J. and Lagoze, C. Combining RDF and XML Schemas to Enhance Interoperability Between Metadata Application Profiles. *10th International World Wide Web Conference, WWW 10* (Hong Kong, China, May 1-5, 2001). 457-466, ACM Press, 2001.
- [16] IBM Corporation. IBM Accessibility Center: Developer guidelines for Web Accessibility. Available at <http://www-306.ibm.com/able/guidelines/web/accessweb.html>
- [17] IMS Global Learning Consortium. IMS Learner Information Package Accessibility for LIP. Available at <http://www.imsglobal.org/accessibility/>
- [18] Ivory, M., Yu, S. and Gronemyer, K. Search result exploration: a preliminary study of blind and sighted users' decision making and performance. *CHI Extended Abstracts 2004* (Vienna, Austria, April 24-29, 2004). 1453-1456, ACM Press, 2004.
- [19] Jul, S. and Furnas, G.W. Navigation in electronic worlds: a CHI 97 Workshop. *ACM SIGCHI Bulletin*, 44-49, ACM Press, 1997.
- [20] Kaplan, C., Fenwick, J. and Chen, J. Adaptive Hypertext Navigation Based on User Goals and Context. *User Modeling and User-Adapted Interaction: The Journal of Personalization Research*, 3 (3), 193-220, Springer, 1993.
- [21] Kurniawan, S. and Zaphiris, P. Research-derived web design guidelines for older people. *ACM SIGACCESS Conference on Computers and Accessibility, ASSETS 2005* (Baltimore, MD, USA, October 9-12, 2005). 129-135, ACM Press, 2005.
- [22] Leporini, B., Paternò, F. and Scordia, A. Flexible tool support for accessibility evaluation. *Interacting with Computers* 18(5), 869-890, Elsevier, 2006.
- [23] Rabin, J. and McMathieNevile, C. (Eds.). (2006, November 2). Mobile Web Best Practices 1.0. (W3C Proposed Recommendation). Available at <http://www.w3.org/TR/mobile-bp/>
- [24] Resource Description Framework (RDF) / W3C Semantic Web Activity. Available at <http://www.w3.org/RDF/>
- [25] Section 508. Section 508 Standards, Web-based intranet and internet information and applications. Available at <http://www.section508.gov/index.cfm?FuseAction=Content&ID=12#Web>
- [26] Sloan, D., Heath, A., Hamilton, F., Kelly, B., Petrie, H. and Phipps, L. Contextual Web Accessibility - Maximizing the benefit of Accessibility Guidelines. *International Cross-Disciplinary Workshop on Web Accessibility, W4A 2006* (Edinburgh, UK, May 22, 2006). 121-131, ACM Press, 2006.
- [27] Theofanos, M.F. and Redish, J. Bridging the gap: between accessibility and usability. *ACM Interactions* 10(6), 36-51, ACM Press, 2003.
- [28] Thompson, T. (2004). 2004 ATHEN Survey: Assistive Technology Products. Available at <http://staff.washington.edu/tft/athen/results10.html>
- [29] Vanderdonckt, J. and Bereikdar, A. Automated Web Evaluation by Guideline Review. *Journal of Web Engineering* 4(2), 102-117, Rinton Press, 2005.
- [30] Velasco, C., Mohamad, Y., Gilman, A., Viorres, N., Vlachogiannis, E., Arnellos, A. and Darzentas, J. Universal access to information services-the need for user information and its relationship to device profiles. *Universal Access in the Information Society* 3(1), 88-95, Springer, 2004.
- [31] Vigo, M., Arrue, M., Brajnik, G., Lomuscio, R. and Abascal, J. Quantitative Metrics for Measuring Web Accessibility. *International Cross-Disciplinary Workshop on Web Accessibility, W4A 2007* (Banff, Canada, May 7-8, 2007). 99-107, ACM Press, 2007.