# Towards the Universal Semantic Assessment of Accessibility

Rui Lopes[*], Konstantinos Votis[† ‡], Luís Carriço[*], Dimitrios Tzovaras[†], Spiridon Likothanassis[‡]
{rlopes,lmc}@di.fc.ul.pt, {kvotis,Dimitrios.Tzovaras}@iti.gr, likothan@ceid.upatras.gr

## ABSTRACT

The ever increasing adoption of software technologies has bring closer technology to users with disabilities and users that interact with devices other than a PC. This diversification poses a real challenge to developers when creating software that has to cope with a myriad of interaction situations, as well as specific directives for ensuring an accessible interaction. In this paper we present SAAF, the Semantic Accessibility Assessment Framework. SAAF provides a set of constructs to describe the semantics of accessibility assessment procedures that can cope with different users, devices, and software technologies in a coherent and formalised way. SAAF affords the description of users and devices, their constraints and requirements, in the light of different accessibility assessment procedures. We exemplify the usage of SAAF by applying it in the context of the Web and related accessibility best practices such as WCAG.

## Categories and Subject Descriptors

H.1.2 [**Models and Principles**]: User/Machine Systems—*Human factors*; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*Theory and methods*

## General Terms

Human Factors.

## Keywords

Universal Accessibility, Semantic Accessibility Assessment

## 1. INTRODUCTION

Designing for people with disabilities is becoming an important topic today. This idea is strongly supported by the fact that an increasing number of countries are legislating towards promoting and enforcing the rights of people with disabilities. Consequently, accessibility is one more aspect that has to be taken into account in the development of software applications, especially in user interfaces.

Furthermore, accessibility can be perceived in different angles [13]. What is accessible to one person, might not be accessible to another one. The different requirements to access and interact within applications pose a significant challenge on how these should be developed [11]. This task often includes the development of different user interfaces to support a particular user group [12], as well as an augmentation of user interfaces with accessibility concerns that can be interpreted by assistive technologies, such as ARIA [10].

However, the development of accessible software requires a strong effort from developers. With the additional encumbrance of taking into account different kinds of accessibility requirements, guidelines and best practices, and different user interface implementation technologies (which by themselves might pose severe problems of delivering accessible applications), developers are faced with a daunting task. Therefore, the highly specialised skills required for developing accessible software sets aside most developers.

To mitigate these problems, developers should be guided in their development process about accessibility concerns within user interface development. This includes the definition of target users (e.g., their requirements, disabilities, etc.), which aspects should be taken into account to meet users' accessibility expectations, and how it reflects on user interfaces of software applications (thus coping with the particularities of different technologies).

To overcome the gap between developers knowledge on accessibility issues and the development of accessible and tailored software applications, we present in this paper the Semantic Accessibility Assessment Framework (SAAF). This framework provides a set of constructs to describe users, accessibility guidelines, as as well how these two concepts can be integrated to form the semantic accessibility assessment of software applications. Through this framework, developers can be guided on application development without having to manually verify if user interfaces are properly implemented to take into account the modelled accessibility constraints. We apply this framework into a particular domain, the Web, showing how it can be used in the context of specific software technologies.

[*]LaSIGE/University of Lisbon, Campo Grande, Edíficio C6, 1749-016 Lisboa, Portugal

[†]CERTH/ITI, Thermi-Thessaloniki, GR-57001 Greece

[‡]Pattern Recognition Laboratory, Computer Engineering and Informatics, University of Patras


*SAC'09* March 8-12, 2009, Honolulu, Hawaii, U.S.A.

## 2. RELATED WORK

With the introduction of accessibility-centric legislation, several governmental bodies around the world have been devising inclusive laws for software applications. These laws are centred on a set of accessibility standards and best practices that are well-known to developers. Authoritative examples are WCAG [2] for Web related technologies, ISO/TS 16071:2003 [6] for general practices on providing accessible software applications, or even application domain-specific guidelines such as for Web browsers [7].

The existence of different standards, guidelines, legislation, and target technologies, put an enormous onus on developers when creating accessible software applications. Each one might have different abstraction levels (i.e., high level *vs.* formalisms *vs.* implementation-specific), hence requiring different approaches on accessibility assessment procedures. Furthermore, when coping with different user requirements, developers might have to follow more than one standard or guideline. Consequently, the wild landscape of accessibility standards and guidelines is challenging. This often results in a disregard from developers and managers as a complex and cost-prohibitive problem. To mitigate these issues, accessibility assessment must be rethought from the start [8]: a mix of user requirements, technologies, and guidelines, framed in the same development context to facilitate the design and development of accessible applications.

### 2.1 Ontologies for Accessibility Knowledge

There are several efforts towards the direction of the definition of ontological concepts and taxonomies for disabilities. These efforts try to cover adequately the personal requirements of the end users, including the person's disabilities and individual preferences.

A central reference for classifying disabilities concerns the World Health Organisation's International Classification of Functioning, Disability and Health (ICF) [14], particularly tailored to impairment qualification on medical diagnosis tasks. Consequently, it stresses on profound disabilities, leaving out several impairments such as colour blindness. In [9], the authors have leveraged ICF concepts into an accessibility description framework to help designers and developers discuss and describe multi-modal interaction issues, providing interesting clues on how to specify the mapping between disabilities and user interfaces.

Other works are centred on how to use ontologies to describe context-awareness for people with disabilities [3, 4]. This type of approaches tries to combine contextual information like personal aspects (e.g., disabilities, preferences), technical aspects (e.g., equipment, services, network) and natural aspects (e.g., location, time) in a way that software applications can cope with such diversities. Similar approaches have also been devised in the ASK-IT project [1], specially centred on multimodal contents.

The aforementioned ontological frameworks emphasise the fact that there is little coupling between ontologies for accessibility and disabilities, and accessibility assessment practices (as they tend just to frame different accessibility scenarios). Nevertheless, they point out several directions - typically in the form of ontological concepts - which are taken into account in SAAF, as described in the next Sections.

## 3. SAAF: A SEMANTIC ACCESSIBILITY ASSESSMENT FRAMEWORK

The main purpose of SAAF, the Semantic Accessibility Assessment Framework, is the division between generic and domain-specific accessibility concepts, and how they can be mapped within accessibility assessment scenarios. Hence, SAAF aims at providing support for the formal and unambiguous definition of accessibility domains, as well as the possible semantic interactions between them. We have specified SAAF to be integrated into accessibility verification environments (such as Integrated Development Environments - IDEs). This will establish a common vocabulary for exchanging and describing the complex information that is related to accessibility assessment of software applications and, ultimately, guide developers on the creation of accessible user interfaces. SAAF aims at formalising conceptual information about: (1) The characteristics of users with disabilities, devices, applications, and other aspects that should be taken into account when describing an audience with disabilities and developing software applications; (2) Accessibility standards and encompassing assessment rules and checkpoints; and (3) Mapping requirements and constraints of users to assessment rules.

One of the main issues in designing the proposed framework is to make it maintainable and extensible, while assuring consistency of accessibility assessment scenarios. Therefore, we have separated SAAF into two dimensions: *meta* and *instance*. Each one of these dimensions is further subdivided into *concepts* and *semantics*. The relationship between all dimensions is depicted in Figure 1:

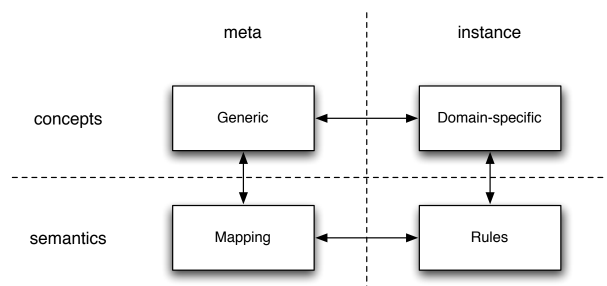


**Figure 1: The four dimensions of the Semantic Accessibility Assessment Framework**

In this Figure we present the four slots where different ontologies are supposed to be integrated within SAAF: *generic*, *mapping*, *domain-specific*, and *rules*. Each one of these slots has specific goals within the overall framework and, consequently, plays a different role in the context of accessibility assessment of software applications: (1) *Generic ontologies* provide a set of *meta concepts* related to accessibility that are independent from technological particularities. Such concepts include the description of users and devices, as well as more general terms to describe software applications domains; (2) *Mapping ontologies* relate to the *meta semantics* of accessibility validation procedures. Mapping procedures should be able to capture the semantic relations between generic concepts. Examples include the definition of semantic constraints, dependencies, or limitations of coping with different generic terms; (3) *Domain-specific ontologies* specify a set of *instance concepts* that are dependent of a particular application domain or technology; and (4) *Rules* *ontologies* provide the specific set of *instance semantics* that allow the translation of specific accessibility assessment procedures into properly verifiable terms that are applied to domain-specific concepts.

The decoupling obtained from separating *meta* ontologies from *instance* ontologies provides the necessary abstraction level that allows the description of accessibility situations that are independent from technologies and, consequently, open the way to the integration of different accessibility assessment procedures that are properly tailored to particular application and technology domains. Next, we describe the generic and mapping ontologies.

## 3.1 Generic Ontology

The Generic ontology describes top-level entities and concepts that are critical for the semantic and universal validation of Web accessibility. This ontology provides knowledge such as general characteristics and disabilities of users, devices, and other aspects that constitute the basis for applying accessibility-based approaches into the accessibility validation field. The main ontological concept available in this ontology is the *saaf:Characteristic* class, a general descriptor for user and device characteristics. To better discern between characteristics belonging to users or devices, we further subclass it into *saaf:UserCharacteristic* and *saaf:DeviceCharaceristic* classes.

We also introduce in this dimension a set of meta-concepts and meta-properties for the instantiation of domain specific ontologies, as depicted in Figure 2. All of the concepts specified by domain specific ontologies will form the core assets for technology-dependent accessibility verification procedures and, through ontologies defined within the rules domain, specify the semantics of accessibility assessment through their combination with user and device concepts, as well as with the mapping ontology.

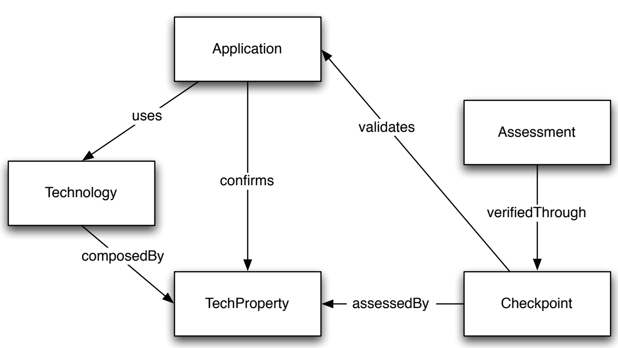


**Figure 2: Domain Specific Ontologies meta level**

The main concepts at this meta level are *saaf:Application*, *saaf:Technology*, *saaf:TechProperty*, *saaf:Assessment*, as well as *saaf:Checkpoint* (the *saaf* prefix has been removed from the Figure for clarity purposes). The *saaf:Application* class affords the representation of an application instance (e.g., a Web site) that is implemented in a particular GUI technology (*saaf:Technology* instance), for which it might have a specific set of technological properties depending on its implementation (*saaf:TechProperty* instance). This application will be tested against a set of checkpoints (*saaf:Checkpoint* instances), part of a specific accessibility assessment procedures (*saaf:Assessment*). These relationships are expressed through *saaf:uses*, *saaf:assessedBy*, *saaf:composedBy*, and *saaf:verifiedThrough* properties. An application is said to verify a particular technology property through the use of *saaf:verifies*, and its conformance to a particular checkpoint is asserted through *saaf:validates*. Section 4 further discusses these concepts and relationships.

Next, we present two ontologies that provide concepts on user and device domains, directly under the umbrella generic ontology. It is worth mentioning that the generic ontology can be complemented with other domains that are independent from particular accessibility assessment procedures or software technologies. The mapping ontology presented afterwards will provide the semantic intertwining between concepts. While the users and devices ontologies provide concepts that are instances of the *saaf:Characteristic* class, they still reside at the meta level of the semantic accessibility assessment. They simply provide the ground work for the definition of a semantically correct description of users and devices, untied to any accessibility assessment procedure.

### *3.1.1 Users Ontology*

As discussed previously, validating accessibility is a process that must cope with user's disabilities, as well as with each individual's preferences. Thus it is of great importance to consider both the users' personal capabilities and impairments. Consequently, different categories of disabilities (based on the ICF categorisation) are incorporated within this ontology, such as: *visual impairments*, disorders in the functions of the eye ranging form reduced capability of sight, colour-blindness to total disability to see; *hearing impairments*, disorders in perceiving audio, ranging from problems in understanding normal conversations to complete deafness; and *specific learning impairments*, disorders manifested by significant difficulties in the acquisition and use of listening, speaking, writing, reading, reasoning, or mathematical abilities. To afford the binding of such concepts to users, this ontology provides a supportive concept, *saaf:User*, which will be used in mapping procedures.

We provide within this ontology several concepts to describe users, as instances of *saaf:UserCharacteristic*. Examples include *saaf:blind*, *saaf:colorBlind*, *saaf:totalBlind*, and *saaf:ableToSee*, all of which are under the umbrella of categorisations such as the ICF. Developers can extend this ontology with additional concepts, by defining them as *saaf:Characteristic* instances.

### *3.1.2 Devices Ontology*

Owing to the rapid development of electronic technologies, it tends to be common to access Web sites outside the traditional field of a desktop PC and a computer screen (e.g., PDAs, mobile phones, assisting devices, etc.). This has brought more specific assistive technologies to improve interactivity for users with disabilities, as well as broad personal preferences. This includes the ability of coping with diverse input/output modalities combination within interactive scenarios. Since the diversity of these technologies is high (e.g., display resolution, images colouring, multimedia process, etc.), the way accessibility is assessed for software applications must also cope with these differences.

First, and along the lines of the *Users Ontology*, this ontology introduces the meta concept *saaf:Device* which will be later on mapped to *saaf:DeviceCharacteristic* instances. This way, common cases of device and device characteristics can be defined without closing the door to extensibility and odd-case scenarios. This ontology also provides several hardware concepts as instances of *saaf:Characteristic*, such as *saaf:screen*, *saaf:smallScreen*, and *saaf:colourDepth1bit*, as well as software devices such as *saaf:screenReader*, or *saaf:pointingDevice*.

## 3.2 Mapping Ontology

The next ontology in the Semantic Accessibility Assessment Framework concerns the *meta semantics* module provided by the *Mapping Ontology*. Here, while staying at the meta level, we provide a way to connect the different concepts introduced by the *Generic Ontology*. The mapping ontology comprises of a set of lexical and notational terms to express the relationship semantics between concepts.

As briefly discussed above, *saaf:Characteristic* instances can be related either to instances of *saaf:User* or *saaf:Device*. This is the most basic semantic relationship provided by the mapping ontology, thus allowing the description of different users and devices. This way, developers and designers can describe the audiences that are being targeted by the software application they are developing and assessing the accessibility. Consequently, characteristics are aggregated into audiences (either user or device centric) that form the base context for accessibility assessment. To support this mapping, we introduce the property *saaf:hasUserCharacteristic* to map the inclusive relationship between *saaf:User* and *saaf:UserCharacteristic* instances. Likewise, the *saaf:hasDeviceCharacteristic* applies to *saaf:Device* and *saaf:DeviceCharacteristic* instances. A simple example follows:

```
saaf:hasUserCharacteristic(#user1, saaf:blind)
saaf:hasDeviceCharacteristic(#dev1, saaf:screen)
```

This mapping ontology also affords the establishment of relationships between characteristics other than for the establishment of accessibility assessment contexts. To better express the semantics of relating characteristics we introduce three new properties. The first property, *saaf:refines*, allows the definition of a hierarchy between characteristics. This way, taxonomies can be easily expressed. The second property, *saaf:requires*, allows the definition of dependency between characteristics. This property is useful, for instance, in more complex cases of accessibility assessment when a given user characteristic (e.g., total blindness) can only be tested if a specific device characteristic is available (e.g., screen reader). Lastly, the third property, *saaf:incompatible*, specifies the incompatibilities between characteristics. This way, the specification of accessibility assessment contexts (and their connection with technological domains) can be easily and robustly detected. To better illustrate the usage of these properties, we present next an example using several concepts from the Generic Ontology:

```
saaf:refines(saaf:colorBlind, saaf:blind)
saaf:refines(saaf:totalBlind, saaf:blind)
saaf:refines(saaf:smallScreen, saaf:screen)

saaf:requires(saaf:screen, saaf:ableToSee)
saaf:requires(saaf:colourDepth1bit, saaf:screen)

saaf:incompatible(saaf:totalBlind, saaf:colorBlind)
saaf:incompatible(saaf:totalBlind, saaf:screen)
```

Lastly, the domain specific meta-level defined in the Generic Ontology can also be verified in a general way, through the use of semantic verification rules. Here, a mapping can be established between the validation of checkpoints on an application through the use of particular technological properties. This rule can be expressed, in SWRL [5] (compact syntax) in the following way:

```
saaf:Application(?app) &
saaf:TechProperty(?prop) &
saaf:Checkpoint(?cp) &
saaf:assessBy(?cp, ?prop) &
saaf:confirms(?app, ?prop)
=>
saaf:validates(?cp, ?app)
```

This assertion guarantees that a checkpoint is valid in the context of a given application, if the technological property confirms its verification. Furthermore, this assertion is independent from any particular technological domain, thus applicable to any SAAF-based assessments.

# 4. INSTANTIATING THE FRAMEWORK IN WEB TECHNOLOGIES

To illustrate how SAAF can cope with real scenarios, we defined several ontologies to be framed as instances within SAAF. First, we specified two ontologies related to Web concepts: accessibility guidelines and Web technologies. Then, we have specified a set of example rules to represent the semantic assessment of accessibility in the context of Web sites and Web applications, by taking into account different particularities of users and devices characterisation.

## 4.1 Web Accessibility Ontology

This domain ontology covers the main evaluation guidelines for Web accessibility assessment devised in the Web Accessibility Initiative, such as WCAG. These guidelines are divided into checkpoints and arranged based on their impact and priority. The combination of these factors is given in levels (none, A, AA, or AAA), depending on their evaluation outcomes. For instance, to claim conformance on level AA, all the priority one and two checkpoints must be satisfied.

We have specified WCAG checkpoints according to the concepts from the DSO meta-level, as well as enrich them with specific conformance semantics through the *wcag:priority* and *wcag:conformance* properties, as exemplified as follows:

```
saaf:Assessment(wcag:wcag1)
saaf:Checkpoint(wcag:cp_2_1)
saaf:verifiedThrough(wcag:wcag1, wcag:cp_2_1)

wcag:Priority(wcag:priority_1)
wcag:Priority(wcag:priority_2)
wcag:Level(wcag:levelA)
wcag:Level(wcag:levelAA)

wcag:priority(wcag:cp_2_1, wcag:priority_1)
wcag:conformance(wcag:levelA, wcag:priority_1)
wcag:conformance(wcag:levelAA, wcag:priority_1)
wcag:conformance(wcag:levelAA, wcag:priority_2)
```

By introducing these specific conformance semantic concepts, richer semantics on accessibility assessment can be explored within the rules ontology within the Web domain.

## 4.2 Web Technology Ontology

This ontology introduces a set of concepts particularly tailored to the verification of different technological properties of the Web domain. Examples include assertions about whether a given application uses Web technologies, or confirming the adequacy of the application in the light of a specific technological particularity. Since these aspects depend on the inspection of an application, they are mere placeholders for the effective testing. Consequently, when defining a technology-oriented ontology for a given domain (such as the Web Technology Ontology described below), all concepts must have a software verification counterpart.

In the context of Web technologies, we have defined a set of concepts along the lines of the domain specific ontology meta-level presented earlier, as presented next:

```
saaf:Technology(wto:HTML)
saaf:TechProperty(wto:altNonColoredInfo)

saaf:composedBy(wto:HTML, wto:altNonColoredInfo)
saaf:assessedBy(wcag:cp_2_1, wto:altNonColoredInfo)
```

Here, we have defined a small subset representing the verification of a particular technological property (which, as explained above, has an implementation counterpart that asserts its representation within the ontology) for testing if all coloured elements have an alternative coloured information (as per Checkpoint 2.1 of WCAG). Similar constructs are applied to the entire WCAG.

### 4.3 Rules Ontology

Finally, all of the previous pieces of SAAF are assembled through rules ontologies. Here, specific rules map domain specific concepts and technologies, with generic concepts (i.e., user and device characterisations) and mapping concepts. Through this type of ontologies the accessibility assessment goes beyond the syntactic analysis of accessibility, thus effectively affording the semantic verification of accessibility according to the different concepts described earlier. For the specification of such rules we have adopted SWRL, similarly to the case of the general verification rules present on the *Mapping Ontology*. An example of such rules follows, in the context of Web technologies:

```
saaf:Application(?APP) &
saaf:confirms(?APP, wto:altNonColoredInfo)
=>
saaf:User(?user) &
saaf:uses(?APP, wto:HTML) &
saaf:hasUserCharacteristic(?user, saaf:blind)

saaf:Application(?APP) &
saaf:confirms(?APP, wto:altNonColoredInfo)
=>
saaf:Device(?dev) &
saaf:uses(?APP, wto:HTML) &
saaf:hasDeviceCharacteristic(?dev,
        saaf:colourDepth1bit)
```

These rules afford the intertwining between user and device properties and specific technology implementation tests. Through the combination with concepts from all ontologies, as well as the mapping rules presented earlier, the semantic accessibility assessment becomes reality. In this small example we bind the test for *alternative non coloured information* for blind users and devices with black and white screens.

## 5. CONCLUSIONS

This paper presented SAAF, the Semantic Accessibility Assessment Framework, as the foundation for the semantic description of Web accessibility audiences, concepts and verification rules. This framework provides the basic constructs for the creation of general accessibility verification engines that are capable of performing assessments tailored to specific user audiences and interaction devices. We have divided SAAF into four dimensions, Generic, Mapping, Domain-specific and Rules in order to afford the extension of SAAF into different accessibility assessment domains.

Ongoing work is currently being done in several fronts in the SAAF realm, including: (1) building a comprehensive set for user and device feature characterisation, (2) providing support for assorted assessment procedures, (3) improving the Mapping and Rules Ontologies to express additional semantics, and (4) develop a set of case studies to bind SAAF onto existing software for accessibility assessment.

## 6. ACKNOWLEDGEMENTS

This work is funded by the EU FP7 project ACCESSIBLE, contract no. 224145, and by FCT through the SFRH/BD/29150/2006 grant and the Multiannual Funding Programme.

## 7. REFERENCES


[1] ASK-IT FP6 Integrated Project. ASK-IT Ontological Framework: Public Deliverable D1.7.1. Technical report, 2007. http://www.ask-it.org/.

[2] B. Caldwell, W. Chisholm, J. Slatin, and G. Vanderheiden. Web Content Accessibility Guidelines 2.0. W3C Working Draft, World Wide Web Consortium (W3C), May 2007. http://www.w3.org/TR/WCAG20/.

[3] T. R. Gruber. A translation approach to portable ontology specifications. *Knowl. Acquis.*, 5(2):199–220, 1993.

[4] D. Heckmann. *Ubiquitous User Modeling*. PhD thesis, Department of Computer Science, Saarland University, November 2005.

[5] I. Horrocks, P. F. Patel-Schneider, H. Boley, B. Grosof, and M. Dean. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. Technical report, World Wide Web Consortium (W3C), 2004. http://www.w3.org/Submission/SWRL/.

[6] ISO. ISO/TS 16071:2003: Ergonomics of human-system interaction – Guidance on accessibility for human-computer interfaces. Technical report, International Organization for Standardization, 2003.

[7] I. Jacobs, J. Gunderson, and E. Hansen. User Agent Accessibility Guidelines 1.0. Technical report, World Wide Web Consortium, Dec 2002. http://www.w3.org/TR/UAAG10/.

[8] B. Kelly, D. Sloan, S. Brown, J. Seale, H. Petrie, P. Lauke, and S. Ball. Accessibility 2.0: people, policies and processes. In *W4A '07: Proceedings of the 2007 international cross-disciplinary conference on Web accessibility (W4A)*, pages 138–147, New York, NY, USA, 2007. ACM.

[9] Z. Obrenovic, J. Abascal, and D. Starcevic. Universal accessibility as a multimodal design issue. *Commun. ACM*, 50(5):83–88, 2007.

[10] L. Seeman, M. Cooper, and R. Schwerdtfeger. Roles for Accessible Rich Internet Applications (WAI-ARIA Roles) Version 1.0. W3C Working Draft, World Wide Web Consortium (W3C), Oct 2007. http://www.w3.org/TR/aria-role/.

[11] B. Shneiderman. Universal Usability. *Commun. ACM*, 43(5):84–91, 2000.

[12] B. Shneiderman. Promoting universal usability with multi-layer interface design. In *CUU '03: Proceedings of the 2003 conference on Universal usability*, pages 1–8, New York, NY, USA, 2003. ACM.

[13] D. Sloan, P. Gregor, M. Rowan, and P. Booth. Accessible accessibility. In *CUU '00: Proceedings on the 2000 conference on Universal Usability*, pages 96–101, New York, NY, USA, 2000. ACM.

[14] World Health Organisation. International Classification of Functioning, Disability and Health. Accessed July 28, 2008 from http://www.who.int/classifications/icf/site/.