XHTML Access Module

Module to enable generic document accessibility

W3C Editor's Draft 24 October 2007

This version:
http://www.w3.org/MarkUp/2007/ED-xhtml-access-20071024
Latest version:
http://www.w3.org/TR/xhtml-access
Previous Editor's Draft:
http://www.w3.org/MarkUp/2007/ED-xhtml-access-20070905
Diff from previous Editor's Draft:
xhtml-access-diff.html
Editors:
Mark Birbeck, x-port.net Ltd.
Shane McCarron, Applied Testing and Technology, Inc.
Steven Pemberton, CWI/W3C®
T. V. Raman, Google, Inc.
Richard Schwerdtfeger, IBM Corporation

This document is also available in these non-normative formats: PostScript version, PDF version, ZIP archive, and Gzip'd TAR archive.

The English version of this specification is the only normative version. Non-normative translations may also be available.

Abstract

The XHTML Access module defines an element that, when used in conjunction with other XHTML modules, enables a more robust accessibility model than that found in traditional HTML.

Status of This Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at http://www.w3.org/TR/.

This document is an internal editors draft for development purposes. However, its content is based upon mature materials from [XHTML2 [p.21]] and is therefore considered relatively mature.

This document has been produced by the W3C HTML Working Group (*Members only*) as part of the HTML Activity. The goals of the HTML Working Group are discussed in the HTML Working Group charter.

This document was produced by a group operating under the 24 January 2002 CPP as amended by the W3C Patent Policy Transition Procedure. W3C maintains a public list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains Essential Claim(s) must disclose the information in accordance with section 6 of the W3C Patent Policy.

Publication as a Working Draft does not imply endorsement by the W3C Membership. This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this document as other than work in progress.

Please report errors in this specification to www-html-editor@w3.org (archive). It is inappropriate to send discussion email to this address. Public discussion may take place on www-html@w3.org (archive).

Table of Contents

1. Introduction

This section is informative.

This document contains a single module designed to be used to help extend the scope of XHTML-family markup languages into new environments. It has been developed in conjunction with the W3C's Web Accessibility Initiative and other interested parties. It is designed to provide a generic mechanism for defining the relationship between document components and well-known accessibility taxonomies.

2. Conformance Requirements

This section is *normative*.

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119 [p.21]].

Note that all examples in this document are informative, and are not meant to be interpreted as normative requirements.

2.1. Document Conformance

XHTML Access is not a stand-alone document type. It is intended to be integrated into other host languages such as XHTML. A conforming XHTML Access document is a document that requires only the facilities described as mandatory in this specification and the facilities described as mandatory in its host language. Such a document must meet all the following criteria:

1. The document MUST conform to the constraints expressed in Appendix A - Schema Implementation or Appendix B - DTD Implementation, combined with the constraints expressed in its host language implementation.

2. If the host language is not in the XHTML namespace, and the host language does not incorporate these modules into its own namespace, then the document MUST contain an `xmlns` declaration for the XHTML RDFa namespace [XMLNAMES [p.21]]. The namespace for XHTML Access Module is defined to be `http://www.w3.org/1999/xhtml`. An example start tag of a root element might look like:

   Example

   ```
   <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" >
   ```

2.2. Host Language Conformance

When XHTML Access is included in a host language, all of the facilities required in this specification MUST be included in the host language. In addition, the elements and attributes defined in this specification MUST be included in the content model of the host language. Finally, XHTML Access requires the availability of the XHTML Role Attribute Module [XHTMLROLE [p.21]] and of the Core Attribute Collection as defined in XHTML Modularization.

2.3. User Agent Conformance

A conforming user agent MUST support all of the features required in this specification.

3. Terms and Definitions

This section is *normative*.

3.1. Compact URIs

In order to allow for the scoped expression of `targetrole` values, this specification uses a superset of of QNames that allows the contraction of all URIs (QNames have a syntactic restriction on the sorts of URI that can be contracted).

These Compact URIs are called CURIEs here.

3.1.1. CURIE Syntax Definition

Note that this syntax definition will ultimately be defined in an external document [CURIE [p.??]].

A basic CURIE is comprised of two components, a *prefix* and a *reference*. The prefix is separated from the reference by a colon (`:`).

```
curie       :=   [ [ prefix ] ':' ] reference

prefix      :=   NCName

reference   :=   irelative-ref (as defined in [IRI])
```

The prefix value MUST be defined using the 'xmlns:' syntax specified in [XMLNAMES [p.21]].

If the prefix is omitted from a CURIE, the default value of http://www.w3.org/1999/xhtml MUST be used.

A CURIE is a representation of a full IRI. This IRI is obtained by concatenating the IRI associated with the `prefix` with the `reference`. The result MUST be a syntactically valid IRI [IRI [p.??]].

The CURIE prefix '_' is reserved. For this reason, prefix declarations using '_' SHOULD be avoided by authors.

Host languages MAY define additional constraints on these syntax rules when CURIES are used in the context of those host languages. Host languages MUST NOT relax the CURIE syntax constraints defined in this specification.

4. XHTML Access Module

This section is *normative*.

This module defines the access [p.??] element.

Element	Attributes	Minimal Content Model
access [p.??]	Common, activate [p.??] , key [p.??] , targetid [p.??] , targetrole [p.??]	EMPTY

Implementations: XML DTD [p.??]

4.1. The `access` element

The access [p.??] element assigns an accessibility mapping to elements within a document. Actuating the mapping results in the element gaining focus or, optionally, in some other event being delivered.

Attributes

4.1.1. `activate` = (yes | no*)

The activate [p.??] attribute indicates whether a target element should be activated or not once it obtains focus. The default value for this attribute is "no", indicating that the element will not be "activated".

4.1.2. `key` = Character

This attribute assigns a key mapping to an access shortcut. An access key is a single character from the document character set. **Note:** Authors should consider the input method of the expected reader when specifying an accesskey.

Triggering an access key defined in an access element changes focus to the next element in navigation order from the current focus that has one of the the referenced role or id values. Note that it is possible to deliver alternate events via [XMLEVENTS [p.21]]. It is also possible to have the target element activated through the use of the activate [p.??] attribute. Also, it is possible to associate additional event handlers with target which might then perform additional actions once focus is changed.

If neither a `targetrole` nor a `targetid` attribute are specified, the user agent MUST NOT define a mapping nor deliver any events.

An `access` element must have either a `targetrole` or a `targetid` attribute specified.

The invocation of access keys depends on the implementation. For instance, on some systems one may have to press the "alt" key in addition to the access key. On other systems, one generally has to press the "cmd" key in addition to the access key.

The rendering of access keys depends on the user agent. We recommend that authors include the access key in label text or wherever the access key is to apply. User agents should render the value of an access key in such a way as to emphasize its role and to distinguish it from other characters (e.g., by underlining it).

The character assigned to a key, and its relationship to a role [p.??] or id [p.??] attribute, are a suggestion of the author. User agents may provide mechanisms for overriding, disabling, or re-assigning keys. In such user agents, user-specified assignments must take precendence. If no key attribute is specified, the user agent SHOULD assign a key.

We talk about the id attribute, but we might also need to accomodate xml:id. How can we do that consistently?

4.1.3. `targetid` = IDREF

The targetid [p.??] attribute specifies an IDREF of the target element for the associated event (i.e., the node to which the event should be delivered).

4.1.4. `targetrole` = `basic-curie`

The targetrole [p.??] attribute specifies space separated list if `basic-curie` that maps to an element with a role [p.??] attribute with the same value.

If no `key` attribute is specified, the user agent may assign an appropriate device mapping.

If a targetid [p.??] and a targetrole [p.??] are both specified for an element, the targetid [p.??] attribute value must take precedence.

Access element that focuses into a field

```
<access key="s"
        title="Social Security Number"
        targetrole="ss:number" />
```

Accessing a table of contents

```
<access key="c"
        title="Table of Contents"
        targetrole="toc" />
```

Access that moves to the main content

```
<access key="m"
        title="Main content"
        targetrole="main" />
```

Access element that goes to a specific element

```
<access key="u"
        title="Username"
        targetid="username" />
```

Access element with no specific key mapping

```
<access title="Navigation bar"
        targetrole="navigation" />
```

A. Schema Implementation

This appendix is *normative*.

The schema implementation of XHTML Access Module conforms to the requirements defined in [XHTMLSCHEMAMOD [p.??]].

A.1. Access Element Module

```
Module SCHEMA/xhtml-access-1.xsd not found!
```

B. DTD Implementation

This appendix is *normative*.

The DTD implementation of XHTML Access Module conforms to the requirements defined in [XHTMLMOD [p.21]]. Consequently, it provides a Qualified Names sub-module, and a module file for the XHTML Access Module module defined in this specification.

B.1. Qualified Names Module

```
Module DTD/xhtml-access-qname-1.mod not found!
```

B.2. XHTML Access Module

```
Module DTD/xhtml-hyperAttributes-1.mod not found!
```

C. DTD Markup Language Example

This appendix is *informative*.

This appendix includes an example of a markup language created using the modules in this specification, coupled with other modules from [XHTMLMOD [p.21]]. The resulting markup language, "xhtml-access" is provided solely as an example, and does not represent an intended direction in terms of a formal markup language from the W3C.

The following sample demonstrates some simple uses of the access element within an xhtml-access document.

```
Module DTD/examples/sample.xml not found!
```

The actual markup language is created by combining the basics of XHTML 1.1 and the modules in this specification. This is done by using a content model module, and then a driver module:

C.1. XHTML+Access Content Model Module

```
Module DTD/examples/xhtml-access-model-1.mod not found!
```

C.2. XHTML+Access Driver Module

```
Module DTD/examples/xhtml-access-1.dtd not found!
```

D. References

This appendix is *normative*.

D.1. Normative References

[DOM2EVENTS]
"*Document Object Model (DOM) Level 2 Events Specification*", W3C Recommendation, T. Pixley, *ed.*, 13 November 2000.
Available at: http://www.w3.org/TR/DOM-Level-2-Events/
The latest version is available at: http://www.w3.org/TR/DOM-Level-2-Events

[RFC2119]
"*Key words for use in RFCs to indicate requirement levels*", RFC 2119, S. Bradner, March 1997.
Available at: http://www.rfc-editor.org/rfc/rfc2119.txt

[XHTMLMOD]
"*Modularization of XHTML™ 1.1*", W3C Working Draft, D. Austin *et al.*, *eds.*, 5 July 2006.
Available at: http://www.w3.org/TR/2006/WD-xhtml-modularization-20060705
The latest version is available at: http://www.w3.org/TR/xhtml-modularization

[XMLNAMES]
"*Namespaces in XML*", W3C Recommendation, T. Bray *et al.*, *eds.*, 14 January 1999.
Available at: http://www.w3.org/TR/1999/REC-xml-names-19990114
The latest version is available at: http://www.w3.org/TR/REC-xml-names

[XHTMLROLE]
"*XHTML Role Attribute Module*", W3C Working Draft, M. Birbeck *et al.*, *eds.*, 13 November 2006.
Available at: http://www.w3.org/TR/2006/WD-xhtml-role-20061113/
The latest version is available at: http://www.w3.org/TR/xhtml-role

D.2. Other References

[XHTML2]
"*XHTML™ 2.0*". J. Axelsson *et al.*, 27 May 2005.
Available at: http://www.w3.org/TR/2005/WD-xhtml2-20050527
The latest version is available at: http://www.w3.org/TR/xhtml2

[XMLEVENTS]
"*XML Events*", W3C Recommendation, S. McCarron *et al.*, *eds.*, 14 October 2003.
Available at: http://www.w3.org/TR/2003/REC-xml-events-20031014

E. Acknowledgments

This section is informative.

At the time of publication, the participants in the W3C HTML Working Group were: