XHTML Access Module

Module to enable generic document accessibility

W3C Editor's Draft 20 August 2008

This version:
http://www.w3.org/MarkUp/2008/ED-xhtml-access-20080820
Latest version:
http://www.w3.org/TR/xhtml-access
Previous version:
http://www.w3.org/TR/2008/WD-xhtml-access-20080107
Diff from previous version:
xhtml-access-diff.html
Editors:
Mark Birbeck, webBackplane mark.birbeck@webBackplane.com
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.

Copyright © 2007-2008 W3C® (MIT, ERCIM, Keio), All Rights Reserved. W3C liability, trademark and document use rules apply.

Abstract

The XHTML Access module defines an element that, when used in conjunction with other XHTML modules in XHTML Family Markup Languages, enables a more robust accessibility model than is presently possible.

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 is a Last Call Working Draft produced by the XHTML 2 Working Group. It reflects a consensus among the Working Group members, in conjunction with input from the User Agent Accessibility Guidelines Working Group and others in the community. The Last Call review period extends through 16 June 2008. The goals of the XHTML 2 Working Group are discussed in the XHTML 2 Working Group charter.

This document was produced by a group operating under the 5 February 2004 W3C Patent Policy. 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 5
2. Conformance Requirements 7
2.1. Document Conformance 7
2.2. Host Language Conformance 7
2.3. User Agent Conformance 8
3. XHTML Access Module 9
3.1. The `access` element 9
3.1.1. `activate` = (yes | no*) 9
3.1.2. `key` = `Characters` 9
3.1.3. `media` = MediaDesc 10
3.1.4. `order` = (document* | list) 11
3.1.5. `targetid` = `IDREFs` 11
3.1.6. `targetrole` = `CURIEs` 12
A. Schema Implementation 13
A.1. Access Element Module 13
B. DTD Implementation 15

B.1. Qualified Names Module 15
B.2. Element Definition Module 16
C. References 19
C.1. Normative References 19
C.2. Other References 19
D. Acknowledgments 21

1. Introduction

This section is informative.

This document contains a single module designed to be used to help make XHTML-family markup languages more effective at supporting the needs of the Accessibility Community. It has been developed in conjunction with the W3C's Web Accessibility Initiative and other interested parties. It provides a generic mechanism for defining the relationship between document components and well-known accessibility vocabularies.

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.19]].

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 XHTML Family Markup Languages. 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 its host language implementation.

2. If the host language is not in the XHTML namespace, and the host language does not incorporate this module into its own namespace, then the document MUST contain an `xmlns:` declaration for the XHTML Access namespace [XMLNAMES [p.19]]. 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 element defined in this specification MUST be included in the content model of the host language. The element defined in this specification MAY be incorporated into the namespace of the host language, or it MAY remain in the XHTML namespace. Finally, XHTML Access requires the availability of the XHTML Role Attribute Module [XHTMLROLE [p.19]].

2.3. User Agent Conformance

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

3. XHTML Access Module

This section is *normative*.

This module defines the access element.

Element	Attributes	Minimal Content Model
access	activate, key, media, targetid, targetrole	EMPTY

Implementations: XML Schema [p.13] , XML DTD [p.15]

3.1. The `access` element

The access element assigns an accessibility mapping to elements within a document. Actuating the mapping results in the element gaining focus (either the document focus or an inspection focus, as determined by the user agent), and, if set by the author and permitted by the user's settings, in one or more other events being activated.

An access element must have either a targetrole or a targetid attribute specified. If neither a targetrole nor a targetid attribute are specified, the user agent MUST NOT define a mapping nor deliver any events.

Attributes

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

The activate 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". User agents MUST provide mechanisms for overriding the author setting with user-specified settings in order to ensure that the act of moving content focus does not cause the user agent to take any further action (as per Checkpoint 9.5 of UAAG 1.0 [UAAG1 [p.19]]).

User agents MUST provide keyboard mechanisms for "activating" any event associated with the focused element (as per Checkpoint 1.2 of UAAG 1.0). User agents SHOULD make available the list of events associated with the focused element (as per Checkpoint 9.6 of UAAG 1.0).

3.1.2. `key` = `Characters`

This attribute assigns one or more key mappings to an access shortcut. The value of is attribute is one or more single characters from the document character set.

The key attribute represents an abstraction. The use of the name "key" for this attribute is historical and does not mean that there is any association with a specific "key" on a keyboard, per se. It is up to the user agent to provide a mechanism for mapping the document character set value(s) of the attribute to the input methods available to the user agent. For instance, on

some systems a user may have to press an "alt" or "cmd" key in addition to the access key. On other systems there may be voice commands, or gestures with a mouse, an eye tracking input device, etc. Regardless of the mechanism, the result is that it appears to the `access` element processor as if the defined *key* was entered.

A user entering any of the keys defined in an access element moves focus from its current position to the next element in navigation order that has one of the referenced role or id values (see activate for information on how the element may be activated). Note that it is possible to deliver alternate events via [XMLEVENTS [p.20]]. Note also that the concept of *navigation order* is a property of the Host Language, and is not defined in this specification.

User agents MUST provide mechanisms for overriding the author setting with user-specified settings in order to ensure that the act of moving content focus does not cause the user agent to take any further action, as required by UAAG 1.0, Checkpoint 9.5. [UAAG1 [p.19]] The character assigned to a key, and its relationship to a role or id attribute SHOULD be treated as an author suggestion. User agents MAY override any key assignment (e.g., if an assignment interferes with the operation of the user interface of the user agent, if the key is not available on a device, if a key is used by the operating environment). User agents MUST also allow users to override author assigned keys with their own key assignments (see Checkpoint 11.3 of UAAG 1.0). If a user chooses to change the key binding, the resultant user-defined remapping SHOULD persist across sessions.

If no key attribute is specified, the user agent SHOULD assign a key and alert the user to the key mapping. The resultant user agent assigned key mapping SHOULD persist.

The rendering of access keys depends on the user agent. We recommend that authors include the access key character in label text or wherever the access key is to apply. If the user agent can recognize that the currently mapped access key character appears in the label text of the element to which it is mapped, then the user agent may render the character in such a way as to emphasize its role as the access key and distinguish it from other characters (e.g., by underlining it).

A conforming user agent SHOULD also provide a centralized view of the current access key assignments (see Checkpoint 11.1 and Checkpoint 11.2 of UAAG 1.0).

Authors MUST NOT assign the same key value to more than one *active* `access` element. Note that it is possible to have an `access` element be *inactive* through the use of the media attribute.

3.1.3. `media` = MediaDesc

The value of this attribute is a comma-separated list of media descriptors for which this `access` element is intended. When the value of this attribute matches the current processing media, the associated `access` element is considered *active* and processed normally; otherwise it is *inactive* and ignored. The default value for this attribute is `all`.

3.1.4. `order` = (document* | list)

The order attribute indicates how this `access` element should determine the *navigation order* for its *matching elements*. The default value, `document`, indicates that the items MUST be traversed in document order. The alternate value, `list`, indicates that the *navigation order* of *matching elements* is determined by the author-defined order of items in the list of targetid or targetrole attribute values.

For the sake of determining *navigation order*, elements in the document that match the values in the targetid or targetrole attributes are called *matching elements*, and all elements that match the same value are members of an *element group*. Note: since the `id` of an element must be unique within a valid XML document, in such documents, each *element group* based on targetid values consist of no more than one *matching element*.

For each navigation operation, the *navigation order* for both *document-order* and *list-order* is sensitive to the current *focus* element. For document-order, if no element currently has focus, the first *matching element* in the document MUST be the focus target; if an element does have focus, the next *matching element* in the document MUST be the focus target. For list-order, the focus target *navigation order* is determined as follows:

- If no matching element of this `access` element currently has *focus*, the focus target MUST be the first element in document order that matches the first element group. If there is no such element, then the focus target is the first element that matches the second element group, and so on.
- If a matching element of this `access` element already has *focus*:
  1. If there are additional matching elements of the same *element group* in document order, then focus MUST be sent to the next matching element of the same element group.
  2. Otherwise, focus MUST go to the first matching element (in document order) of the next element group.
  3. If there are no remaining elements groups, then the search MUST resume from the first element group.

The location of the next *matching element* MUST be determined each time the `access` element is triggered, since it is possible that between events the contents of the document will have changed.

3.1.5. `targetid` = `IDREFs`

The targetid attribute specifies one or more IDREFs related to target elements for the associated event (i.e., the node to which the event should be delivered).

3.1.6. targetrole = CURIEs

The targetrole attribute specifies a space separated list of CURIEs [CURIE [p.19]] that maps to an element with a role attribute with the same value.

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

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

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 *informative*.

The schema implementation of XHTML Access Module conforms to the requirements defined in [XHTMLMOD [p.19]]. It is included here as an example implementation.

A.1. Access Element Module

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
 xmlns:xs="http://www.w3.org/2001/XMLSchema"
 elementFormDefault="qualified"
 xmlns:xh11d="http://www.w3.org/1999/xhtml/datatypes/"
>
    <xs:import namespace="http://www.w3.org/1999/xhtml/datatypes/"
               schemaLocation="xhtml-datatypes-1.xsd" />

    <xs:annotation>
        <xs:documentation>
      This is the XML Schema module for XHTML Access
      $Id: xhtml-access.ps,v 1.2 2008/08/20 16:03:51 smccarro Exp $
    </xs:documentation>
        <xs:documentation source="xhtml-copyright-1.xsd"/>
        <xs:documentation source="http://www.w3.org/TR/xhtml-role#A_role"/>
    </xs:annotation>
    <xs:attributeGroup name="xhtml.access.attlist">
        <xs:attributeGroup ref="xhtml.Common.attrib"/>
        <xs:attribute name="activate" default="no">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="yes"/>
                    <xs:enumeration value="no"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="key" type="xh11d:Character"/>
        <xs:attribute name="media" type="xh11d:MediaDesc"/>
        <xs:attribute name="order" default="document">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="document"/>
                    <xs:enumeration value="list"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="targetid">
            <xs:simpleType>
                <xs:list itemType="xs:IDREF"/>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="targetrole" type="xh11d:CURIEs"/>
    </xs:attributeGroup>
    <xs:group name="xhtml.access.content">
```

```
        <xs:sequence/>
    </xs:group>
    <xs:complexType name="xhtml.access.type">
        <xs:group ref="xhtml.access.content"/>
        <xs:attributeGroup ref="xhtml.access.attlist"/>
    </xs:complexType>
</xs:schema>
```

B. DTD Implementation

This appendix is *normative*.

The DTD implementation of XHTML Access Module conforms to the requirements defined in [XHTMLMOD [p.19]]. Consequently, it provides a Qualified Names sub-module, and a module file for the XHTML Access Module defined in this specification.

B.1. Qualified Names Module

```
<!-- ....................................................................... -->
<!-- XHTML Access Qname Module  ............................................ -->
<!-- file: xhtml-access-qname-1.mod

     This is XHTML Access - the Access Attribute Module for XHTML.

     Copyright 2007-2008 W3C (MIT, ERCIM, Keio), All Rights Reserved.

     This DTD module is identified by the PUBLIC and SYSTEM identifiers:

       PUBLIC "-//W3C//ENTITIES XHTML Access Attribute Qnames 1.0//EN"
       SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-access-qname-1.mod"

     Revisions:
     (none)
     ....................................................................... -->

<!-- XHTML Access Attribute Qname (Qualified Name) Module

     This module is contained in two parts, labeled Section 'A' and 'B':

       Section A declares parameter entities to support namespace-
       qualified names, namespace declarations, and name prefixing
       for XHTML Access and extensions.

       Section B declares parameter entities used to provide
       namespace-qualified names for the XHTML access element:

         %XHTML.access.qname;   the xmlns-qualified name for access
         ...

     XHTML Access extensions would create a module similar to this one.
-->

<!-- Section A: XHTML Access Attribute XML Namespace Framework :::::::::::::: -->

<!-- 1. Declare a %XHTML.prefixed; conditional section keyword, used
        to activate namespace prefixing. The default value should
        inherit '%NS.prefixed;' from the DTD driver, so that unless
        overridden, the default behavior follows the overall DTD
        prefixing scheme.
-->
<!ENTITY % NS.prefixed "IGNORE" >
<!ENTITY % XHTML.prefixed "%NS.prefixed;" >
```

```
<!-- 2. Declare a parameter entity (eg., %XHTML.xmlns;) containing
        the URI reference used to identify the XHTML Access Attribute namespace
-->
<!ENTITY % XHTML.xmlns  "http://www.w3.org/1999/xhtml" >

<!-- 3. Declare parameter entities (eg., %XML.prefix;) containing
        the default namespace prefix string(s) to use when prefixing
        is enabled. This may be overridden in the DTD driver or the
        internal subset of an document instance. If no default prefix
        is desired, this may be declared as an empty string.

     NOTE: As specified in [XMLNAMES], the namespace prefix serves
     as a proxy for the URI reference, and is not in itself significant.
-->
<!ENTITY % XHTML.prefix  "" >

<!-- 4. Declare parameter entities (eg., %XHTML.pfx;) containing the
        colonized prefix(es) (eg., '%XHTML.prefix;:') used when
        prefixing is active, an empty string when it is not.
-->
<![%XHTML.prefixed;[
<!ENTITY % XHTML.pfx  "%XHTML.prefix;:" >
]]>
<!ENTITY % XHTML.pfx  "" >

<!-- declare qualified name extensions here ............ -->
<!ENTITY % xhtml-access-qname-extra.mod "" >
%xhtml-access-qname-extra.mod;

<!-- 5. The parameter entity %XHTML.xmlns.extra.attrib; may be
        redeclared to contain any non-XHTML Access namespace
        declaration attributes for namespaces embedded in XML. The default
        is an empty string.  XLink should be included here if used
        in the DTD.
-->
<!ENTITY % XHTML.xmlns.extra.attrib "" >


<!-- Section B: XML Qualified Names ::::::::::::::::::::::::::::: -->

<!-- 6. This section declares parameter entities used to provide
        namespace-qualified names for the XHTML Access element.
-->

<!ENTITY % XHTML.access.qname  "%XHTML.pfx;access" >

<!-- end of xhtml-access-qname-1.mod -->
```

B.2. Element Definition Module

```
<!-- .................................................................... -->
<!-- XHTML Access Module ................................................. -->
<!-- file: xhtml-access-1.mod

     This is XHTML Access - the Access Module for XHTML.
```

```
      Copyright 2007-2008 W3C (MIT, ERCIM, Keio), All Rights Reserved.

      This DTD module is identified by the PUBLIC and SYSTEM identifiers:

        PUBLIC "-//W3C//ELEMENTS XHTML Access Element 1.0//EN"
        SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-access-1.mod"

      Revisions:
      (none)
      ..................................................................... -->


<!ENTITY % Character.datatype "CDATA" >
<!ENTITY % CURIEs.datatype "CDATA" >
<!ENTITY % IDREFs.datatype "CDATA" >
<!ENTITY % MediaDesc.datatype "CDATA" >

<!ENTITY % XHTML.access.element  "INCLUDE" >
<![%XHTML.access.element;[
<!ENTITY % XHTML.access.content  "EMPTY" >
<!ENTITY % XHTML.access.qname  "access" >
<!ELEMENT %XHTML.access.qname;  %XHTML.access.content; >
<!-- end of XHTML.access.element -->]]>

<!ENTITY % XHTML.access.attlist  "INCLUDE" >
<![%XHTML.access.attlist;[
<!ATTLIST %XHTML.access.qname;
      %Common.attrib;
      activate     ( yes | no )              #IMPLIED
      order        ( document | list )       #IMPLIED
      key          %Character.datatype;      #IMPLIED
      media        %MediaDesc.datatype;      #IMPLIED
      targetid     %IDREFs.datatype;         #IMPLIED
      targetrole   %CURIEs.datatype;         #IMPLIED
>
<!-- end of XHTML.access.attlist -->]]>

<!-- end of xhtml-access-1.mod -->
```

C. References

This appendix is *normative*.

C.1. Normative References

[CURIE]
"*CURIE Syntax 1.0*", W3C Working Draft, M. Birbeck, S. McCarron, *ed.*, 2 April 2008.
Available at: http://www.w3.org/TR/2008/WD-curie-20080402/
The latest version is available at: http://www.w3.org/TR/curie

[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

[IRI]
"*Internationalized Resource Identifiers (IRI)*", RFC 3987, M.Duerst, M. Suignard January 2005.
Available at: http://www.ietf.org/rfc/rfc3987.txt

[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.*, 7 April 2008.
Available at: http://www.w3.org/TR/2008/WD-xhtml-role-20080407/
The latest version is available at: http://www.w3.org/TR/xhtml-role

C.2. Other References

[UAAG1]
"*User Agent Accessibility Guidelines 1.0*". Ian Jacobs *et al.*, 17 December 2002.
Available at: http://www.w3.org/TR/2002/REC-UAAG10-20021217
The latest version is available at: http://www.w3.org/TR/UAAG10

[XHTML2]
"*XHTML™ 2.0*". J. Axelsson *et al.*, 26 July 2006.
Available at: http://www.w3.org/TR/2006/WD-xhtml2-20060726

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

D. Acknowledgments

This section is informative.

At the time of publication, the participants in the W3C XHTML 2 Working Group were: