Abstract

This specification defines an XHTML document type that is based upon the module framework and modules defined in XHTML Modularization [XHTMLMOD]. The purpose of this document type is to serve as the basis for future extended XHTML ‘family’ document types, and to provide a consistent, forward-looking document type cleanly separated from the deprecated, legacy functionality of HTML 4 [HTML4] that was brought forward into the XHTML 1.0 [XHTML] document types. This document type is most similar to XHTML 1.0 Strict, built using XHTML Modules. This means that many facilities available in other XHTML Family
document types (e.g., XHTML Frames) are not available in this document type. These other facilities are available through modules defined in XHTML Modularization, and document authors are free to define document types based upon XHTML 1.1 that use these facilities (see [XHTMLMOD] [p.13] for information on creating new document types).

**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 a W3C Proposed Edited Recommendation. If approved, it will supersede the previous edition of XHTML 1.1. It reflects clarifications and corrections as a result of many years of use by the community. It also includes an XML Schema implementation of the language, and integrates the lang attribute to increase compatibility with User Agents and Assistive Technologies. A version that shows the specific changes from the previous Recommendation is available in diff-marked form. A disposition of comments document is also available.

Publication as a Proposed Edited Recommendation 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.

W3C Advisory Committee Members are invited to send formal review comments on this Proposed Edited Recommendation to the W3C Team until 11 November 2010. Members of the W3C Advisory Committee will find the appropriate review form for this document by consulting their list of current WBS questionnaires.

This document has been produced by the W3C XHTML 2 Working Group as part of the HTML Activity. The goals of the XHTML 2 Working Group are discussed in the XHTML 2 Working Group charter.

This document is governed by 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.

Public discussion of HTML takes place on www-html@w3.org [archive]. To subscribe send an email to www-html-request@w3.org with the word subscribe in the subject line.

Please report errors in this document to www-html-editor@w3.org [archive].

A list of current W3C Recommendations and other technical documents can be found at http://www.w3.org/TR.
Quick Table of Contents

1. Introduction ................................................................. .5
2. Conformance Definition .................................................. .7
3. The XHTML 1.1 Document Type ......................................... .9
   A. Changes from XHTML 1.0 Strict .................................... 11
   B. References ............................................................. 13
   C. XHTML 1.1 Document Type Definition ............................ 15
   D. XHTML 1.1 XML Schema Definition ............................... 27
   E. Acknowledgements ................................................... 53

Full Table of Contents

1. Introduction ................................................................. .5
2. Conformance Definition .................................................. .7
   2.1. Document Conformance ............................................. 7
       2.1.1. Strictly Conforming Documents .............................. 7
   2.2. User Agent Conformance ........................................... 8
3. The XHTML 1.1 Document Type ......................................... .9
   A. Changes from XHTML 1.0 Strict .................................... 11
   B. References ............................................................. 13
       B.1. Normative References ........................................... 13
       B.2. Informative References ....................................... 13
   C. XHTML 1.1 Document Type Definition ............................ 15
       C.1. SGML Open Catalog Entry for XHTML 1.1 .................. 15
       C.2. XHTML 1.1 Driver ................................................ 15
       C.3. XHTML 1.1 Customizations ..................................... 21
   D. XHTML 1.1 XML Schema Definition ............................... 27
       D.1. XHTML 1.1 Schema Driver ..................................... 27
       D.2. XHTML 1.1 Schema Modules .................................. 29
       D.3. XHTML 1.1 Customizations ................................... 35
       D.4. XML Schema Ruby Implementation ............................ 48
   E. Acknowledgements ................................................... 53
1. Introduction

This section is *normative*.

With the introduction of the XHTML family of modules and document types, the W3C has helped move the Internet content-development community from the days of malformed, non-standard markup into the well formed, valid world of XML [XML [p.13]]. In XHTML 1.0, this move was moderated by a goal of providing for easy migration of existing, HTML 4 (or earlier) based content to XHTML and XML. With the advent of the XHTML modules defined in XHTML Modularization, the W3C has removed support for deprecated elements and attributes from the XHTML family. These elements and attributes were largely presentation oriented functionality that is better handled via style sheets or client-specific default behavior.

Going forward, XHTML family document types will be based upon this new, more structural functional collection. In this specification, the W3C’s HTML Working Group has defined an initial document type based solely upon modules. This document type is designed to be portable to a broad collection of client devices, and applicable to the majority of Internet content. Content developers who base their content upon the functionality expressed in this specification can be confident that it will be consistently portable across XHTML family conforming user agents.
2. Conformance Definition

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.13].

2.1. Document Conformance

This version of XHTML provides a definition of strictly conforming XHTML documents, which are restricted to elements and attributes from the XHTML namespace.

2.1.1. Strictly Conforming Documents

A strictly conforming XHTML 1.1 document is a document that requires only the facilities described as mandatory in this specification. Such a document MUST meet all the following criteria:

1. The document MUST conform to the constraints expressed in the schemas in [Appendix D - XHTML 1.1 Schema][p.27] and [Appendix C - XHTML 1.1 Document Type Definition][p.15].

2. The local part of the root element of the document MUST be `html`.

3. The start tag of the root element of the document MUST explicitly contain an `xmlns` declaration for the XHTML namespace [XMLNAMES][p.13]. The namespace URI for XHTML is defined to be `http://www.w3.org/1999/xhtml`. The start tag MAY also contain the declaration of the XML Schema Instance Namespace and an XML Schema Instance `schemaLocation` attribute [XMLSCHEMA][p.13]. Such an attribute would associate the XHTML namespace `http://www.w3.org/1999/xhtml` with the XML Schema at the URI `http://www.w3.org/MarkUp/SCHEMA/xhtml11.xsd`.

Sample root element

```html
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xsi:schemaLocation="http://www.w3.org/1999/xhtml
                       http://www.w3.org/MarkUp/SCHEMA/xhtml11.xsd"
>
```

4. There MUST be a DOCTYPE declaration in the document prior to the root element. If present, the PUBLIC identifier included in the DOCTYPE declaration MUST reference the DTD found in [Appendix A][p.15] using its Formal Public Identifier. The SYSTEM identifier MAY be modified as appropriate.
5. The start tag MAY also contain a version attribute that declares the version of XHTML in use. The version of this version of XHTML is "-//W3C//DTD XHTML 1.1//EN".

Example of an XHTML 1.1 document

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
  <head>
    <title>Virtual Library</title>
  </head>
  <body>
    <p>Moved to <a href="http://example.org/">example.org</a>.</p>
  </body>
</html>
```

Note that in this example, the XML declaration is included. An XML declaration like the one above is not required in all XML documents. XHTML document authors SHOULD use XML declarations in all their documents. XHTML document authors MUST use an XML declaration when the character encoding of the document is other than the default UTF-8 or UTF-16 and no encoding is specified by a higher-level protocol.

XHTML 1.1 documents SHOULD be labeled with the Internet Media Type "application/xhtml+xml" as defined in [RFC3236][p. 14]. For further information on using media types with XHTML, see the informative note [XHTMLMIME][p. 14].

2.2. User Agent Conformance

A conforming user agent MUST meet all user agent conformance requirements defined in [XHTMLMOD][p. 13].
3. The XHTML 1.1 Document Type

This section is normative.

The XHTML 1.1 document type is a fully functional document type with rich semantics. It is not, however, as varied in functionality as the XHTML 1.0 Transitional or Frameset document types. These document types defined many presentational components that are better handled through style sheets or other similar mechanisms. Moreover, since the XHTML 1.1 document type is based exclusively upon the facilities defined in the XHTML modules [XHTMLMOD[p.13]], it does not contain any of the deprecated functionality of XHTML 1.0 nor of HTML 4. Despite these exceptions, or perhaps because of them, the XHTML 1.1 document type is a solid basis for future document types that are targeted at varied user agent environments.

The XHTML 1.1 document type is made up of the following XHTML modules. The elements, attributes, and minimal content models associated with these modules are defined in "XHTML Modularization" [XHTMLMOD[p.13]]. The elements are listed here for information purposes, but the definitions in "XHTML Modularization" should be considered definitive. In the on-line version of this document, the module names in the list below link into the definitions of the modules within the current version of "XHTML Modularization".

**Structure Module**
- body, head, html, title

**Text Module**
- abbr, acronym, address, blockquote, br, cite, code, dfn, div, em, h1, h2, h3, h4, h5, h6, kbd, p, pre, q, samp, span, strong, var

**Hypertext Module**
- a

**List Module**
- dl, dt, dd, ol, ul, li

**Object Module**
- object, param

**Presentation Module**
- b, big, hr, i, small, sub, sup, tt

**Edit Module**
- del, ins

**Bidirectional Text Module**
- bdo

**Forms Module**
- button, fieldset, form, input, label, legend, select, optgroup, option, textarea

**Tables Module**
- caption, col, colgroup, table, tbody, td, tfoot, th, thead, tr

**Image Module**
- img
XHTML 1.1 also uses the Ruby Annotation module as defined in [RUBY][p.13]:

Ruby Annotation Module

```
  ruby, rbc, rtc, rb, rt, rp
```

This specification also adds the lang attribute to the I18N attribute collection as defined in [XHTMLMOD][p.13]. The lang attribute is defined in [HTML4][p.13]. When this attribute and the xml:lang attribute are specified on the same element, the xml:lang attribute takes precedence. When both lang and xml:lang are specified on the same element, they SHOULD have the same value.

There are no additional definitions required by this document type. An implementation of this document type as an XML Schema is defined in Appendix D[p.27], and as an XML DTD is defined in Appendix C[p.15]. If there is any discrepancy between the language as defined in this section and the implementations in the appendices, the definition in this section MUST take precedence.
A. Changes from XHTML 1.0 Strict

This appendix is informative.

This Appendix describes the differences between XHTML 1.1 and XHTML 1.0 Strict. XHTML 1.1 represents a departure from both HTML 4 and XHTML 1.0. Most significant is the removal of features that were deprecated. In general, the strategy is to define a markup language that is rich in structural functionality, but that relies upon style sheets for presentation.

The differences can be summarized as follows:

1. On the a and map elements, the name attribute has been removed in favor of the id attribute (as defined in [XHTMLMOD][p.13]).
2. The "ruby" collection of elements has been added (as defined in [RUBY][p.13]).
B. References

This appendix is normative.

B.1. Normative References

[HTML4]

See: http://www.w3.org/TR/1999/REC-html401-19991224

[RUBY]

See: http://www.w3.org/TR/2001/REC-ruby-20010531

[XHTMLMOD]

See: http://www.w3.org/TR/2010/REC-xhtml-modularization-20100729

[XML]

[Extensible Markup Language (XML) 1.0 (Fourth Edition)] W3C Recommendation, T. Bray et al., eds., 16 August 2006.
Available at: http://www.w3.org/TR/2006/REC-xml-20060816
The latest version is available at: http://www.w3.org/TR/REC-xml

[XMLNAMES]

Available at: http://www.w3.org/TR/2006/REC-xml-names-20060816

[XMLESHEMA]

Available at: http://www.w3.org/TR/2004/REC-xmlschema-1-20041028/
See also [XML Schema Part 2: Datatypes Second Edition], available at:
http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/

B.2. Informative References

[CATALOG]

[Entity Management: OASIS Technical Resolution 9401:1997 (Amendment 2 to TR 9401)]
Paul Grosso, Chair, Entity Management Subcommittee, SGML Open, 10 September 1997.
See: http://www.oasis-open.org/html/a401.htm

[RFC2119]

[Key words for use in RFCs to indicate requirement levels], RFC 2119, S. Bradner, March 1997.
Available at: http://www.ietf.org/rfc/rfc2119.txt
B.2. Informative References

Available at: http://www.ietf.org/rfc/rfc2854.txt

Available at: http://www.ietf.org/rfc/rfc3236.txt

[XHTML1] XHTML 1.0: The Extensible HyperText Markup Language (Second Edition), W3C
Recommendation, Steven Pemberton, et al., 26 January 2000, revised 1 August 2002.
See: http://www.w3.org/TR/2002/REC-xhtml1-20020801

Latest version available at: http://www.w3.org/TR/xhtml-media-types
C. XHTML 1.1 Document Type Definition

This appendix is *normative*.

C.1. SGML Open Catalog Entry for XHTML 1.1

This section contains the SGML Open Catalog-format definition [CATALOG[p.13]] of the public identifiers for XHTML 1.1.

```
-- .......................................................................... --
-- File catalog ............................................................ --
-- XHTML 1.1 Catalog Data File
Revision: @(#)xhtml11.cat 1.9 2001/04/04 SMI
See "Entity Management", SGML Open Technical Resolution 9401 for detailed information on supplying and using catalog data. This document is available from OASIS at URL:
--
-- .......................................................................... --
-- SGML declaration associated with XHTML .......................... --
OVERRIDE YES
SGMLDECL "xml1.dcl"
-- :::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::: --
-- XHTML 1.1 DTD modular driver file ..................................... --
PUBLIC "-//W3C//DTD XHTML 1.1//EN"                             "xhtml11.dtd"
-- XHTML 1.1 framework modules ......................................... --
PUBLIC "-//W3C//ENTITIES XHTML 1.1 Document Model 1.0//EN"    "xhtml11-model-1.mod"
-- End of catalog data ..................................................... --
-- .......................................................................... --
```

C.2. XHTML 1.1 Driver

This section contains the driver for the XHTML 1.1 document type implementation as an XML DTD. It relies upon XHTML module implementations defined in [XHTMLMOD[p.13]] and in [RUBY[p.13]].
<!-- .......................................................... -->
<!-- XHTML 1.1 DTD ............................................. -->
<!-- file: xhtml11.dtd -->
-->
<!-- XHTML 1.1 DTD -->

This is XHTML, a reformulation of HTML as a modular XML application.

The Extensible HyperText Markup Language (XHTML)
Copyright 1998-2008 World Wide Web Consortium
(Massachusetts Institute of Technology, European Research Consortium
for Informatics and Mathematics, Keio University).
All Rights Reserved.

Permission to use, copy, modify and distribute the XHTML DTD and its
accompanying documentation for any purpose and without fee is hereby
granted in perpetuity, provided that the above copyright notice and
this paragraph appear in all copies. The copyright holders make no
representation about the suitability of the DTD for any purpose.

It is provided "as is" without expressed or implied warranty.

Author: Murray M. Altheim <altheim@eng.sun.com>
-->

<!-- This is the driver file for version 1.1 of the XHTML DTD. -->

Please use this public identifier to identify it:

"-//W3C//DTD XHTML 1.1//EN"
-->

<!ENTITY % XHTML.version "-//W3C//DTD XHTML 1.1//EN" >

<!-- Use this URI to identify the default namespace: -->

"http://www.w3.org/1999/xhtml"

See the Qualified Names module for information
on the use of namespace prefixes in the DTD.

Note that XHTML namespace elements are not prefixed by default,
but the XHTML namespace prefix is defined as "xhtml" so that
other markup languages can extend this one and use the XHTML
prefixed global attributes if required.

-->

<!ENTITY % NS.prefixed "IGNORE" >
<!ENTITY % XHTML.prefix "xhtml" >

<!-- Be sure to include prefixed global attributes - we don’t need
them, but languages that extend XHTML 1.1 might. -->

<!ENTITY % XHTML.global.attrs.prefixed "INCLUDE" >

<!-- Reserved for use with the XLink namespace: -->

- 16 -
<!ENTITY % XLINK.xmlns "" >
<!ENTITY % XLINK.xmlns.attrib "" >

<!-- For example, if you are using XHTML 1.1 directly, use the public
identifier in the DOCTYPE declaration, with the namespace declaration
on the document element to identify the default namespace:

<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
  "http://www.w3.org/MarkUp/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
  xml:lang="en">
  ...
</html>

Revisions:
  (none)
-->

<!-- reserved for future use with document profiles -->
<!ENTITY % XHTML.profile "" >

<!-- ensure XHTML Notations are disabled -->
<!ENTITY % xhtml-notations.module "IGNORE" >

<!-- Bidirectional Text features
This feature-test entity is used to declare elements
and attributes used for bidirectional text support.
-->
<!ENTITY % XHTML.bidi "INCLUDE" >

<?doc type="doctype" role="title" { XHTML 1.1 } ?>

<!-- ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::: -->
<!-- Pre-Framework Redeclaration placeholder .................... -->
<!-- this serves as a location to insert markup declarations
into the DTD prior to the framework declarations.
-->
<!ENTITY % xhtml-prefw-redecl.module "IGNORE" >
<![%xhtml-prefw-redecl.module;[
  %xhtml-prefw-redecl.mod;
  <!-- end of xhtml-prefw-redecl.module -->]]>

<!ENTITY % xhtml-events.module "INCLUDE" >

<!-- Inline Style Module ........................................ -->
<!ENTITY % xhtml-inlstyle.module "INCLUDE" >
<![%xhtml-inlstyle.module;[
  %xhtml-inlstyle.mod
  PUBLIC "-//W3C//ELEMENTS XHTML Inline Style 1.0//EN"
  "http://www.w3.org/MarkUp/DTD/xhtml-inlstyle-1.mod" >
  %xhtml-inlstyle.mod;]]>

<!-- declare Document Model module instantiated in framework
-->
C.3. XHTML 1.1 Customizations

An XHTML Family Document Type (such as XHTML 1.1) must define the content model that it uses. This is done through a separate content model module that is instantiated by the XHTML Modular Framework. The content model module and the XHTML 1.1 Driver (above) work together to customize the module implementations to the document type’s specific requirements. The content model module for XHTML 1.1 is defined below:

<!DOCTYPE PUBLIC "-//W3C//ELEMENTS XHTML Embedded Object 1.0//EN" "http://www.w3.org/MarkUp/DTD/xhtml-object-1.mod">
<!DOCTYPE PUBLIC "-//W3C//ELEMENTS XHTML Tables 1.0//EN" "http://www.w3.org/MarkUp/DTD/xhtml-table-1.mod">
<!DOCTYPE PUBLIC "-//W3C//ELEMENTS XHTML Forms 1.0//EN" "http://www.w3.org/MarkUp/DTD/xhtml-form-1.mod">
<!DOCTYPE PUBLIC "-//W3C//ELEMENTS XHTML Legacy Markup 1.0//EN" "http://www.w3.org/MarkUp/DTD/xhtml-legacy-1.mod">
<!DOCTYPE PUBLIC "-//W3C//ELEMENTS XHTML Document Structure 1.0//EN" "http://www.w3.org/MarkUp/DTD/xhtml-struct-1.mod">

C.3. XHTML 1.1 Customizations

This is XHTML 1.1, a reformulation of HTML as a modular XML application. Copyright 1998-2008 W3C (MIT, ERCIM, Keio), All Rights Reserved. Revision: $Id: xhtml11-model-1.mod,v 1.18 2009/06/24 17:24:55 ahby Exp $
This DTD module is identified by the PUBLIC and SYSTEM identifiers:

PUBLIC "-//W3C//ENTITIES XHTML 1.1 Document Model 1.0//EN"
SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml11-model-1.mod"

Revisions:
(None)
.................................................................................. -->

<!-- XHTML 1.1 Document Model

This module describes the groupings of elements that make up
common content models for XHTML elements.

XHTML has three basic content models:

%Inline.mix;  character-level elements
%Block.mix;   block-like elements, eg., paragraphs and lists
%Flow.mix;    any block or inline elements

Any parameter entities declared in this module may be used
to create element content models, but the above three are
considered 'global' (insofar as that term applies here).

The reserved word '#PCDATA' (indicating a text string) is now
included explicitly with each element declaration that is
declared as mixed content, as XML requires that this token
occur first in a content model specification.
-->  

<!-- Extending the Model

While in some cases this module may need to be rewritten to
accommodate changes to the document model, minor extensions
may be accomplished by redeclaring any of the three *.extra;
parameter entities to contain extension element types as follows:

%Misc.extra;  whose parent may be any block or
             inline element.

%Inline.extra;  whose parent may be any inline element.

%Block.extra;  whose parent may be any block element.

If used, these parameter entities must be an OR-separated
list beginning with an OR separator ('|'), eg., "| a | b | c"

All block and inline *.class parameter entities not part
of the *struct.class classes begin with | " to allow for
exclusion from mixes.
-->  

<!-- ............. Optional Elements in head ................. -->

<!ENTITY % HeadOpts.mix
"( %script.qname; | %style.qname; | %meta.qname;
   | %link.qname; | %object.qname; )*"
Misellaneous Elements

-- ins and del are used to denote editing changes

--
<!ENTITY % Edit.class "| %ins.qname; | %del.qname;" >

-- script and noscript are used to contain scripts
and alternative content

--
<!ENTITY % Script.class "| %script.qname; | %noscript.qname;" >

<!ENTITY % Misc.extra "" >

These elements are neither block nor inline, and can
essentially be used anywhere in the document body.

--
<!ENTITY % Misc.class "%Edit.class;
 %Script.class;
 %Misc.extra;" >

Inline Elements

--
<!ENTITY % InlStruct.class "%br.qname; | %span.qname;" >

<!ENTITY % InlPhras.class "| %em.qname; | %strong.qname; | %dfn.qname; | %code.qname;
 %samp.qname; | %kbd.qname; | %var.qname; | %cite.qname;
 %abbr.qname; | %acronym.qname; | %q.qname;" >

<!ENTITY % InlPres.class "| %tt.qname; | %i.qname; | %b.qname; | %big.qname;
 %small.qname; | %sub.qname; | %sup.qname;" >

<!ENTITY % I18n "| %bdo.qname;" >

<!ENTITY % Anchor "| %a.qname;" >

<!ENTITY % InlSpecial "| %img.qname; | %map.qname;
 %object.qname;" >

<!ENTITY % InlForm "| %input.qname; | %select.qname; | %textarea.qname;
 %label.qname; | %button.qname;" >

<!ENTITY % Inline.extra "" >

<!ENTITY % Ruby "| %ruby.qname;" >

-- %Inline.class; includes all inline elements,
used as a component in mixes

--
<!ENTITY % Inline.class "%InlStruct.class;" >
<!ENTITY % Block.class "%BlkStruct.class; %BlkPhras.class; %BlkPres.class; %BlkSpecial.class; %Block.extra;" >

<!ENTITY % Block.mix; includes all block elements plus %Misc.class; -->
<!ENTITY % Block.mix "%Heading.class; | %List.class; | %Block.class; %Misc.class;" >
<!-- .................... All Content Elements .................... -->

<!-- %Flow.mix; includes all text content, block and inline -->
<!ENTITY % Flow.mix "%Heading.class;
  | %List.class;
  | %Block.class;
  | %Inline.class;
  %Misc.class;"
>

<!-- end of xhtml11-model-1.mod -->
D. XHTML 1.1 XML Schema Definition

This appendix is normative.

D.1. XHTML 1.1 Schema Driver

This section contains the driver for the XHTML 1.1 document type implementation as an XML Schema. It relies upon XHTML module implementations defined in [XHTMLMOD] and in [RUBY].

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
   xmlns:xs="http://www.w3.org/2001/XMLSchema"
   targetNamespace="http://www.w3.org/1999/xhtml"
   xmlns:xh11d="http://www.w3.org/1999/xhtml/datatypes/"
   xmlns="http://www.w3.org/1999/xhtml"
   elementFormDefault="qualified">
   <xs:annotation>
     <xs:documentation>
       This is the XML Schema driver for XHTML 1.1.
       Please use this namespace for XHTML elements:
       "http://www.w3.org/1999/xhtml"
     </xs:documentation>
   </xs:annotation>
   ...
</xs:schema>
```

This is the Schema Driver file for XHTML1.1 Document Type

This schema + imports external schemas (xml.xsd)

$Id: xhtml11.xsd,v 1.7 2009/02/03 15:14:49 ahby Exp $
XHTML1.1 Document Type includes the following Modules

XHTML Core modules (Required for XHTML Family Conformance)
+ text
+ hypertext
+ lists
+ structure

Other XHTML modules
+ Edit
+ Bdo
+ Presentational
+ Link
+ Meta
+ Base
+ Scripting
+ Style
+ Image
+ Applet
+ Object
+ Param (Applet/Object modules require Param Module)
+ Tables
+ Forms
+ Client side image maps
+ Server side image maps
+ Ruby

This import brings in the XML namespace attributes
The XML attributes are used by various modules.

Document Model module for the XHTML1.1 Document Type.
This schema file defines all named models used by XHTML Modularization Framework for XHTML1.1 Document Type

Schema that includes all modules (and redefinitions) for XHTML1.1 Document Type.
D.2. XHTML 1.1 Schema Modules

XHTML Family implementations using XML Schema are required to provide their own schema module that imports the required modules from XHTML Modularization.

```xml
<?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/"
  xmlns="http://www.w3.org/1999/xhtml"
  targetNamespace="http://www.w3.org/MarkUp/SCHEMA/xhtml-html-1.xsd"
  schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-html-1.xsd" />

<xs:annotation>
  <xs:documentation>
  This schema includes all modules for XHTML1.1 Document Type.
  $Id: xhtml11-modules-1.xsd,v 1.10 2009/02/03 15:14:49 ahby Exp $
  </xs:documentation>
</xs:annotation>
```

D.2. XHTML 1.1 Schema Modules

XHTML Family implementations using XML Schema are required to provide their own schema module that imports the required modules from XHTML Modularization.

```
<xs:include schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-framework-1.xsd">
  <xs:annotation>
    <xs:documentation>
      Schema Framework Component Modules:
      + notations
      + datatypes
      + common attributes
      + character entities
    </xs:documentation>
  </xs:annotation>
</xs:include>
```

```
<xsl:include schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-text-1.xsd">
<xsl:annotation>
  <xs:documentation>
  Text module
  The Text module includes declarations for all core text container elements and their attributes.
  + block phrasal
  + block structural
  + inline phrasal
  + inline structural
  Elements defined here:
  * address, blockquote, pre, h1, h2, h3, h4, h5, h6
  * div, p
  * abbr, acronym, cite, code, dfn, em, kbd, q, samp, strong, var
```
Bidirectional element module

Elements defined here:
* bdo

Presentational module

Elements defined here:
* hr, b, big, i, small, sub, sup, tt

Meta module

Elements defined here:
* meta

Base module

Elements defined here:
* base
Scripting module

Elements defined here:
  * script, noscript

Style module

Elements defined here:
  * style

Image module

Elements defined here:
  * img
Target Module - Area Attribute Additions

Server-side image maps module

Attributes defined here:
- ismap on img

Object module

Elements defined here:
- object

Param module

Elements defined here:
- param

Tables module

Elements defined here:
- table, caption, thead, tfoot, tbody, colgroup, col, tr, th, td

Forms module

Elements defined here:
- form, label, input, select, optgroup, option,
- textarea, fieldset, legend, button

Changes to XHTML Form Attlist

Original Form Attributes (declared in Forms Module)

XHTML Events Module - Attribute additions
D.2. XHTML 1.1 Schema Modules

XHTML™ 1.1 - Module-based XHTML - Second Edition

</xs:attributeGroup>
<xs:attributeGroup ref="shtml.form.target.attlist">
  <xs:documentation>
  HTML Target Module - Attribute additions
  </xs:documentation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:annotation>

Changes to XHTML Form Input Element
</xs:annotation>
</xs:attributeGroup>
<xs:attributeGroup ref="shtml.input.attlist">
  <xs:documentation>
  Original Input Attributes (in Forms Module)
  </xs:documentation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup ref="shtml.csim.attlist">
  <xs:documentation>
  Redefinition by Client Side Image Map Module
  </xs:documentation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup ref="shtml.input.events.attlist">
  <xs:documentation>
  Redefinition by Event Attribute Module
  </xs:documentation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup ref="shtml.label.attlist">
  <xs:documentation>
  Original Label Attributes (in Forms Module)
  </xs:documentation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup ref="shtml.select.attlist">
  <xs:documentation>
  Original Select Attributes (in Forms Module)
  </xs:documentation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup ref="shtml.textarea.attlist">
  <xs:documentation>
  Original TextArea Attributes (in Forms Module)
  </xs:documentation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup ref="shtml.textarea.events.attlist">
  <xs:documentation>
  Redefinition by Event Attribute Module
  </xs:documentation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup ref="shtml.button.attlist">
  <xs:documentation>
  </xs:documentation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup name="xhtml.label.attlist">
  <xs:annotation>
  Original Label Attributes (in Forms Module)
  </xs:annotation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup name="xhtml.select.attlist">
  <xs:annotation>
  Original Select Attributes (in Forms Module)
  </xs:annotation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup name="xhtml.textarea.attlist">
  <xs:annotation>
  Original TextArea Attributes (in Forms Module)
  </xs:annotation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup name="xhtml.button.attlist">
  <xs:annotation>
  </xs:annotation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup name="xhtml.label.events.attlist">
  <xs:annotation>
  Redefinition by Event Attribute Module
  </xs:annotation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup name="xhtml.select.events.attlist">
  <xs:annotation>
  Redefinition by Event Attribute Module
  </xs:annotation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup name="xhtml.textarea.events.attlist">
  <xs:annotation>
  Redefinition by Event Attribute Module
  </xs:annotation>
</xs:attributeGroup>
</xs:attributeGroup>
</xs:attributeGroup>
<xs:attributeGroup name="xhtml.button.events.attlist">
  <xs:annotation>
  </xs:annotation>
</xs:attributeGroup>
</xs:attributeGroup>
D.3. XHTML 1.1 Customizations

An XHTML Family Document Type (such as XHTML 1.1) must define the content model that it uses. This is done through a separate content model module that is instantiated by the XHTML Modular Framework. The content model module and the XHTML 1.1 Driver (above) work together to customize the module implementations to the document type’s specific requirements. The content model module for XHTML 1.1 is defined below:

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema

    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:xh11d="http://www.w3.org/1999/xhtml/datatypes/"
    elementFormDefault="qualified">

    <xs:import
        namespace="http://www.w3.org/1999/xhtml/datatypes/"
        schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-datatypes-1.xsd"/>

    <xs:annotation>
        <xs:documentation>
            This is the XML Schema module of common content models for XHTML11
        </xs:documentation>
    </xs:annotation>

    $Id: xhtml11-model-1.xsd,v 1.9 2009/02/03 15:14:49 ahby Exp$
    <xs:documentation>
        <xs:documentation source="xhtml-copyright-1.xsd"/>
    </xs:documentation>
</xs:schema>
This module describes the groupings of elements/attributes that make up common content models for XHTML elements. XHTML has following basic content models:

- xhtml.Inline.mix; character-level elements
- xhtml.Block.mix; block-like elements, e.g., paragraphs and lists
- xhtml.Flow.mix; any block or inline elements
- xhtml.HeadOpts.mix; Head Elements
- xhtml.InlinePre.mix; Special class for pre content model
- xhtml.InlineNoAnchor.mix; Content model for Anchor

Any groups declared in this module may be used to create element content models, but the above are considered 'global' (insofar as that term applies here). XHTML has the following Attribute Groups:

- xhtml.Core.extra.attrib
- xhtml.I18n.extra.attrib
- xhtml.Common.extra

The above attribute Groups are considered Global.
<xs:annotation>
  <xs:documentation> Extend Core Attributes </xs:documentation>
</xs:annotation>

<xs:attributeGroup
  name="xhtml.Core.extra.attrib">
  <xs:annotation>
    <xs:documentation> Extend Core Attributes </xs:documentation>
  </xs:annotation>
</xs:attributeGroup>

<xs:attributeGroup
  name="xhtml.Global.core.extra.attrib">
  <xs:annotation>
    <xs:documentation> Extended Global Core Attributes </xs:documentation>
  </xs:annotation>
</xs:attributeGroup>

<xs:attributeGroup
  name="xhtml.Global.I18n.extra.attrib">
  <xs:annotation>
    <xs:documentation> Extended Global I18n attributes </xs:documentation>
  </xs:annotation>
</xs:attributeGroup>

<xs:attributeGroup
  name="xhtml.Global.Common.extra">
  <xs:annotation>
    <xs:documentation> Extended Global Common Attributes </xs:documentation>
  </xs:annotation>
</xs:attributeGroup>

<xs:group
  name="xhtml.Head.extra">
  <xs:sequence/>
</xs:group>

<xs:group
  name="xhtml.HeadOpts.mix">
  <xs:choice>
    <xs:element
      name="script"
      type="xhtml.script.type"/>
    <xs:element
      name="style"
      type="xhtml.style.type"/>
    <xs:element
      name="meta"
      type="xhtml.meta.type"/>
    <xs:element
      name="link"
      type="xhtml.link.type"/>
    <xs:element
      name="object"
      type="xhtml.object.type"/>
    <xs:group
      ref="xhtml.Head.extra"/>
  </xs:choice>
</xs:group>

<xs:group
  name="xhtml.head.content">
  <xs:sequence>
    <xs:group
      name="xhtml.Headopts.mix">
      <xs:choice>
        <xs:element
          name="script"
          type="xhtml.script.type"/>
        <xs:element
          name="style"
          type="xhtml.style.type"/>
        <xs:element
          name="meta"
          type="xhtml.meta.type"/>
        <xs:element
          name="link"
          type="xhtml.link.type"/>
        <xs:element
          name="object"
          type="xhtml.object.type"/>
        <xs:group
          ref="xhtml.Head.extra"/>
      </xs:choice>
    </xs:group>
  </xs:sequence>
</xs:group>
ref="xhtml.HeadOpts.mix"
minOccurs="0"
maxOccurs="unbounded"/>
<xs:choice>
<xs:sequence>
<xs:element
name="title"
minOccurs="1"
maxOccurs="1"
type="xhtml.title.type"/>
<xs:group
ref="xhtml.HeadOpts.mix"
minOccurs="0"
maxOccurs="unbounded"/>
<xs:sequence
minOccurs="0">
<xs:element
name="base"
type="xhtml.base.type"/>
<xs:group
ref="xhtml.HeadOpts.mix"
minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:sequence>
</xs:element
name="base"
type="xhtml.base.type"
minOccurs="1"
maxOccurs="1"/>
<xs:group
ref="xhtml.HeadOpts.mix"
minOccurs="0"
maxOccurs="unbounded"/>
<xs:element
name="title"
minOccurs="1"
maxOccurs="1"
type="xhtml.title.type"/>
<xs:group
ref="xhtml.HeadOpts.mix"
minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:choice>
</xs:group>
<!--
ins and del are used to denote editing changes
-->
<xs:group
name="xhtml.Edit.class">
<xs:choice>
<xs:element
name="ins"
type="xhtml.edit.type"/>
<xs:element name="del"
    type="xhtml.edit.type"/>
</xs:choice>
</xs:group>
<!--
script and noscript are used to contain scripts
and alternative content
-->  
<xs:group
    name="xhtml.Script.class">
<xs:choice>
    <xs:element
        name="script"
        type="xhtml.script.type"/>
    <xs:element
        name="noscript"
        type="xhtml.noscript.type"/>
</xs:choice>
</xs:group>
<xs:group
    name="xhtml.Misc.extra">
<xs:sequence/>
</xs:group>
<!--
These elements are neither block nor inline, and can
essentially be used anywhere in the document body.
-->  
<xs:group
    name="xhtml.Misc.class">
<xs:choice>
    <xs:group
        ref="xhtml.Edit.class"/>
    <xs:group
        ref="xhtml.Script.class"/>
    <xs:group
        ref="xhtml.Misc.extra"/>
</xs:choice>
</xs:group>
<!-- Inline Elements -->
<xs:group
    name="xhtml.InlStruct.class">
<xs:choice>
    <xs:element
        name="br"
        type="xhtml.br.type"/>
    <xs:element
        name="span"
        type="xhtml.span.type"/>
</xs:choice>
</xs:group>
<xs:group
    name="xhtml.InlPhras.class">
<xs:choice>
    <xs:element
        name="em"
        type="xhtml.em.type"/>
</xs:choice>
</xs:group>
<xs:element name="strong"
type="xhtml.strong.type"/>
<xs:element name="dfn"
type="xhtml.dfn.type"/>
<xs:element name="code"
type="xhtml.code.type"/>
<xs:element name="samp"
type="xhtml.samp.type"/>
<xs:element name="kbd"
type="xhtml.kbd.type"/>
<xs:element name="var"
type="xhtml.var.type"/>
<xs:element name="cite"
type="xhtml.cite.type"/>
<xs:element name="abbr"
type="xhtml.abbr.type"/>
<xs:element name="acronym"
type="xhtml.acronym.type"/>
<xs:element name="q"
type="xhtml.q.type"/>
</xs:choice>
</xs:group>
<xs:group name="xhtml.InlPres.class">
<xs:choice>
<xs:element name="tt"
type="xhtml.InlPres.type"/>
<xs:element name="i"
type="xhtml.InlPres.type"/>
<xs:element name="b"
type="xhtml.InlPres.type"/>
<xs:element name="big"
type="xhtml.InlPres.type"/>
<xs:element name="small"
type="xhtml.InlPres.type"/>
<xs:element name="sub"
type="xhtml.InlPres.type"/>
<xs:element name="sup"
type="xhtml.InlPres.type"/>
</xs:choice>
</xs:group>
<xs:sequence>
  <xs:element
    name="ruby"
    type="xhtml.ruby.type"/>
</xs:sequence>
</xs:group>
</xs:group>
<!--
Inline.class includes all inline elements,
used as a component in mixes
-->
<xs:group name="xhtml.InlinePre.mix">
  <xs:choice>
    <xs:group ref="xhtml.InlStruct.class"/>
    <xs:group ref="xhtml.InlPhras.class"/>
    <xs:element name="tt" type="xhtml.InlPres.type"/>
    <xs:element name="i" type="xhtml.InlPres.type"/>
    <xs:element name="b" type="xhtml.InlPres.type"/>
    <xs:group ref="xhtml.I18n.class"/>
    <xs:group ref="xhtml.Anchor.class"/>
    <xs:group ref="xhtml.Misc.class"/>
    <xs:element name="map" type="xhtml.map.type"/>
    <xs:group ref="xhtml.Inline.extra"/>
  </xs:choice>
</xs:group>

<!--
InlNoAnchor.class includes all non-anchor inlines, used as a component in mixes
-->
<!-- InlNoAnchor.mix includes all non-anchor inlines -->
<xs:group name="xhtml.InlNoAnchor.mix">
  <xs:choice>
    <xs:group ref="xhtml.InlNoAnchor.class"/>
    <xs:group ref="xhtml.Misc.class"/>
  </xs:choice>
</xs:group>

<!-- Inline.mix includes all inline elements, including Misc.class -->
<xs:group name="xhtml.Inline.mix">
  <xs:choice>
    <xs:group ref="xhtml.Inline.class"/>
    <xs:group ref="xhtml.Misc.class"/>
  </xs:choice>
</xs:group>

<!-- InlNoRuby.mix includes all of inline.mix elements except ruby -->
<xs:group name="xhtml.InlNoRuby.mix">
  <xs:choice>
    <xs:group ref="xhtml.InlNoRuby.class"/>
    <xs:group ref="xhtml.Misc.class"/>
  </xs:choice>
</xs:group>

<!-- In the HTML 4 DTD, heading and list elements were included in the block group. The Heading.class and List.class groups must now be included explicitly on element declarations where desired. -->
<xs:group name="xhtml.Heading.class">
  <xs:choice>
    <xs:element name="h1" type="xhtml.h1.type"/>
    <xs:element name="h2" type="xhtml.h2.type"/>
    <xs:element name="h3" type="xhtml.h3.type"/>
  </xs:choice>
</xs:group>
<xs:element name="pre" type="xhtml.pre.type"/>
<xs:element name="blockquote" type="xhtml.blockquote.type"/>
<xs:element name="address" type="xhtml.address.type"/>
</xs:choice>
</xs:group>
<xs:group name="xhtml.BlkPres.class">
<xs:sequence>
<xs:element name="hr" type="xhtml.hr.type"/>
</xs:sequence>
</xs:group>
<xs:group name="xhtml.BlkSpecial.class">
<xs:choice>
<xs:group ref="xhtml.Table.class"/>
<xs:group ref="xhtml.Form.class"/>
<xs:group ref="xhtml.Fieldset.class"/>
</xs:choice>
</xs:group>
<xs:group name="xhtml.Block.extra">
<xs:sequence/>
</xs:group>
<!--
Block.class includes all block elements, used as an component in mixes
-->
</xs:choice>
</xs:group>
<!--
Block.mix includes all block elements plus %Misc.class;
-->
<xs:group
    name="xhtml.Block.mix">
    <xs:choice>
        <xs:group
            ref="xhtml.Heading.class"/>
        <xs:group
            ref="xhtml.List.class"/>
        <xs:group
            ref="xhtml.Block.class"/>
        <xs:group
            ref="xhtml.Misc.class"/>
    </xs:choice>
</xs:group>
<!--
All Content Elements
Flow.mix includes all text content, block and inline
Note that the "any" element included here allows us to add data from any other namespace, a necessity for compound document creation.
Note however that it is not possible to add to any head level element without further modification. To add RDF metadata to the head of a document, modify the structure module.
-->
<xs:group
    name="xhtml.Flow.mix">
    <xs:choice>
        <xs:group
            ref="xhtml.Heading.class"/>
        <xs:group
            ref="xhtml.List.class"/>
        <xs:group
            ref="xhtml.Block.class"/>
        <xs:group
            ref="xhtml.Inline.class"/>
        <xs:group
            ref="xhtml.Misc.class"/>
    </xs:choice>
</xs:group>
<!--
BlkNoForm.mix includes all non-form block elements, plus Misc.class
-->
<xs:group
    name="xhtml.BlkNoForm.mix">
    <xs:choice>
        <xs:group
            ref="xhtml.Heading.class"/>
        <xs:group
            ref="xhtml.List.class"/>
        <xs:group
            ref="xhtml.BlkStruct.class"/>
D.4. XML Schema Ruby Implementation

The RUBY specification does not currently define an XHTML Module using XML Schema. One is defined here:

```xml
<?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 Ruby module for XHTML
            $Id: xhtml-ruby-1.xsd,v 1.3 2009/01/06 15:38:11 ahby Exp $
        </xs:documentation>
        <xs:documentation source="xhtml-copyright-1.xsd"/>
    </xs:annotation>

    <xs:annotation>
        <xs:documentation>
            "Ruby" are short runs of text alongside the base text, typically used in East Asian documents to indicate pronunciation or to provide a short annotation. The full specification for Ruby is here:

            http://www.w3.org/TR/2001/REC-ruby-20010531/

            This module defines "Ruby" or "complex Ruby" as described in the specification:

            http://www.w3.org/TR/2001/REC-ruby-20010531/#complex

            Simple or Basic Ruby are defined in a separate module.

            This module declares the elements and their attributes used to support complex ruby annotation markup. Elements defined here
        </xs:documentation>
    </xs:annotation>
</xs:schema>
```
This module expects the document model to define the following content models
+ InlNoRuby.mix
</xs:documentation>
<xs:documentation
  source="http://www.w3.org/TR/2001/REC-ruby-20010531/">
</xs:annotation>

<xs:group name="xhtml.ruby.content.simple">
  <xs:sequence>
    <xs:element name="rb" type="xhtml.rb.type"/>
    <xs:choice>
      <xs:element name="rt" type="xhtml.rt.type"/>
      <xs:sequence>
        <xs:element name="rp" type="xhtml.rp.type"/>
        <xs:element name="rt" type="xhtml.rt.type"/>
        <xs:element name="rp" type="xhtml.rp.type"/>
      </xs:sequence>
    </xs:choice>
  </xs:sequence>
</xs:group>

<xs:group name="xhtml.ruby.content.complex">
  <xs:sequence>
    <xs:element name="rbc" type="xhtml.rbc.type"/>
    <xs:element name="rtc" type="xhtml.rtc.type" maxOccurs="2"/>
  </xs:sequence>
</xs:group>

<!--
add to this group any common attributes for all Ruby elements
-->
<xs:attributeGroup name="xhtml.ruby.common.attrib"/>

<xs:group name="xhtml.ruby.content">
  <xs:choice>
    <xs:group ref="xhtml.ruby.content.simple"/>
    <xs:group ref="xhtml.ruby.content.complex"/>
  </xs:choice>
</xs:group>

<complexType name="xhtml.ruby.type">
  <xs:group ref="xhtml.ruby.content"/>
  <xs:attributeGroup ref="xhtml.ruby.common.attrib"/>
</complexType>

<!--
rbc (ruby base component) element
-->
<xs:attributeGroup name="xhtml.rbc.attlist">
  <xs:attributeGroup ref="xhtml.ruby.common.attrib"/>
</xs:attributeGroup>

<xs:group name="xhtml.rbc.content">
  <xs:sequence>
<xs:element name="rb" type="xhtml.rb.type" />
</xs:sequence>
</xs:group>

<xs:complexType name="xhtml.rbc.type">
  <xs:group ref="xhtml.rbc.content"/>
  <xs:attributeGroup ref="xhtml.rbc.attlist"/>
</xs:complexType>

<!--
rtc (ruby text component) element
-->
<xs:attributeGroup name="xhtml.rtc.attlist">
  <xs:attributeGroup ref="xhtml.ruby.common.attrib"/>
</xs:attributeGroup>

<xs:group name="xhtml.rtc.content">
  <xs:sequence>
    <xs:element name="rt" type="xhtml.rt.type" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:group>

<xs:complexType name="xhtml.rtc.type">
  <xs:group ref="xhtml.rt.content"/>
  <xs:attributeGroup ref="xhtml.rtc.attlist"/>
</xs:complexType>

<!--
rb (ruby base) element
-->
<xs:attributeGroup name="xhtml.rb.attlist">
  <xs:attributeGroup ref="xhtml.ruby.common.attrib"/>
</xs:attributeGroup>

<xs:group name="xhtml.rb.content">
  <xs:sequence>
    <xs:group ref="xhtml.InlNoRuby.mix" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:group>

<xs:complexType name="xhtml.rb.type" mixed="true">
  <xs:group ref="xhtml.rb.content"/>
  <xs:attributeGroup ref="xhtml.rb.attlist"/>
</xs:complexType>

<!--
rt (ruby text) element
-->
<xs:attributeGroup name="xhtml.rt.attlist">
  <xs:attributeGroup ref="xhtml.ruby.common.attrib"/>
  <xs:attribute name="rbspan" type="xhtml11d:Number" default="1"/>
</xs:attributeGroup>

<xs:group name="xhtml.rt.content">
  <xs:sequence>
    <xs:group ref="xhtml.InlNoRuby.mix" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:group>

<xs:complexType name="xhtml.rt.type">
  <xs:group ref="xhtml.rt.content"/>
  <xs:attributeGroup ref="xhtml.rt.attlist"/>
</xs:complexType>
<xs:group>

<xs:complexType name="xhtml.rt.type" mixed="true">
  <xs:group ref="xhtml.rt.content"/>
  <xs:attributeGroup ref="xhtml.rt.attlist"/>
</xs:complexType>

<!-- rp (ruby parenthesis) element -->
<xs:attributeGroup name="xhtml.rp.attlist">
  <xs:attributeGroup ref="xhtml.ruby.common.attrib"/>
</xs:attributeGroup>

<xs:group name="xhtml.rp.content">
  <xs:sequence/>
</xs:group>

<xs:complexType name="xhtml.rp.type" mixed="true">
  <xs:group ref="xhtml.rp.content"/>
  <xs:attributeGroup ref="xhtml.rp.attlist"/>
</xs:complexType>

</xs:schema>
E. Acknowledgements

This appendix is *informative*.

This specification was prepared by the W3C HTML Working Group. The members at the time of publication of the first edition were:

- Steven Pemberton, CWI (HTML Working Group Chair)
- Murray Altheim, Sun Microsystems
- Daniel Austin, Mozquito Technologies
- Jonny Axelsson, Opera Software
- Mark Baker, Sun Microsystems
- Tantek Çelik, Microsoft
- Doug Dominiak, Openwave Systems
- Herman Elenbaas, Philips Electronics
- Beth Epperson, Netscape/AOL
- Masayasu Ishikawa, W3C (HTML Activity Lead)
- Shin’ichi Matsui, Panasonic
- Shane McCarron, Applied Testing and Technology
- Ann Navarro, WebGeek, Inc.
- Peter Stark, Ericsson
- Michel Suignard, Microsoft
- Jeremy Wadsworth, Quark Inc.
- Malte Wedel, Mozquito Technologies
- Ted Wugofski, Openwave Systems

The second edition of this specification was prepared by the W3C XHTML2 Working Group. The members at the time of publication of the second edition were:

- Roland Merrick, IBM (XHTML 2 Working Group Co-Chair)
- Steven Pemberton, CWI (XHTML 2 Working Group Co-Chair)
- Mark Birbeck, webBackplane (Invited Expert)
- Susan Borgrink, Progeny Systems
- Christina Bottomley, Society for Technical Communication (STC)
- Alessio Cartocci, International Webmasters Association / HTML Writers Guild (IWA-HWG)
- Alexander Graf, University of Innsbruck
- Markus Gylling, [DAISY Consortium](https://www.daisy.org)
- Tina Holmboe, Greytower Technologies (Invited Expert)
- John Kugelman, Progeny Systems
- Luca Mascaro, International Webmasters Association / HTML Writers Guild (IWA-HWG)
- Shane McCarron, Applied Testing and Technology, Inc. (Invited Expert)
- Michael Rawling, IVIS Group Limited
- Gregory Rosmaita, Invited Expert
- Sebastian Schnitzenbaumer, Dreamlab Technologies AG
• Richard Schwerdtfeger, IBM
• Elias Torres, IBM
• Masataka Yakura, Mitsue-Links Co., Ltd.
• Toshihiko Yamakami, ACCESS Co., Ltd.