XHTML RDFa Modules

Modules to support RDF annotation of elements

W3C Editor's Draft 17 April 2007

This version:
http://www.w3.org/MarkUp/2007/ED-xhtml-rdfa-20070417
Latest version:
http://www.w3.org/TR/xhtml-rdfa
Previous Editor's Draft:
http://www.w3.org/MarkUp/2007/ED-xhtml-rdfa-20070410
Diff from previous Editor's Draft:
xhtml-rdfa-diff.html
Editors:
Mark Birbeck, x-port.net Ltd.
Shane McCarron, Applied Testing and Technology, Inc.

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 W3C® (MIT, ERCIM, Keio), All Rights Reserved. W3C liability, trademark and document use rules apply.

Abstract

The XHTML RDFa (RDF using attributes) modules define a collection of elements and attributes that enhance a document authors ability to annotate the relationships of content within and among documents. These modules can be integrated into any markup language based upon XHTML Modularization [XHTMLMOD [p.55]].

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.55]] and is therefore considered nearly complete.

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 5
2. Conformance Requirements 7
2.1. Document Conformance 7
2.2. Host Language Conformance 7
2.3. User Agent Conformance 7
3. XHTML Hypertext Attributes Module 9
3.1. Hypertext Attribute Collection 9
3.1.1. The href attribute 9
3.1.2. The hreflang attribute 9
3.1.3. The hrefmedia attribute 10
3.1.4. The hreftype attribute 10
4. XHTML Metainformation Attributes Module 11
4.1. Metadata Attribute Collection 11
4.1.1. The about attribute 11
4.1.2. The content attribute 11
4.1.3. The datatype attribute 11
4.1.4. The property attribute 12
4.1.5. The rel attribute 13
4.1.6. The rev attribute 16
4.2. Metadata Attributes and RDF 17

4.3. Metadata as Content 17
4.4. Mapping Lexical Content 19
5. XHTML Metainformation Module 21
5.1. The link element 21
5.1.1. Forward and reverse links 22
5.1.2. Links and search engines 22
5.2. The meta element 24
5.2.1. meta and search engines 25
5.3. Literals and Resources 25
5.4. Document Properties 26
5.4.1. Literals 26
5.4.2. Resources 27
5.4.3. Making Use of External Lists of Properties 28
5.5. Properties of Other Resources 28
5.5.1. Resources Within the Containing Document 29
5.5.2. External Resources 29
A. Schema Implementation 31
A.1. Hypertext Attributes Module 31
A.2. XHTML Metainformation Attributes Module 31
A.3. XHTML Metainformation Module 32
B. DTD Implementation 35
B.1. Qualified Names Module 35
B.2. XHTML HyperAttributes Module 37
B.3. XHTML Metainformation Attributes Module 38
B.4. XHTML Metainformation Module 41
C. DTD Markup Language Example 43
C.1. XHTML+RDFa Content Model Module 43
C.2. XHTML+RDFa Driver Module 48
D. References 55
D.1. Normative References 55
D.2. Other References 55
E. Acknowledgments 57

1. Introduction

This section is informative.

This document contains three modules 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 RDF and semantic web communities to make it easier to annotate XHTML documents so that RDF information can be automatically extracted.

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.55]].

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

2.1. Document Conformance

XHTML RDFa is not a stand-alone document type. It is intended to be integrated into other host languages such as XHTML. A conforming XHTML RDFa 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.55]]. The namespace for XHTML RDFa 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 RDFa 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 RDFa requires the availability 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. XHTML Hypertext Attributes Module

This section is *normative*.

The Hypertext Attributes Module defines the Hypertext attribute collection. This collection allows an element to be the start point of a hypertext link to a remote resource. When this module is selected, this collection is added to the Common attribute collection as defined in [XHTMLMOD [p.55]].

encoding

Do we need to bring the encoding attribute over too?

3.1. Hypertext Attribute Collection

3.1.1. The href attribute

This attribute specifies a URI to associate with the specified resource. *Note that when used in conjuction with elements such as* `a` *in XHTML, this attribute has additional semantics.*

Example

```
<link href="top.html" hreflang="en" hreftype="text/html" rel="contents"/>
```

3.1.2. The hreflang attribute

This attribute specifies the primary language of the resource designated by href [p.??] . It takes a value of type LanguageCodes. At its most general, it is a comma-separated list of language ranges with optional accept parameters, as defined in section 14.4 of [RFC2616 [p.??]] as the field value of the Accept-Language request header.

In its simplest case, this is just a language code, such as "nl", but it may also contain variant specifications such as "en-gb".

The user agent must use this list as the field value of the `accept-language` request header when requesting the resource using HTTP.

If this attribute is not present, the user agent must use its default value of the accept-language request header.

Example

```
<p>
   <a href="http://www.w3.org/2003/06/semantictour-pressrelease"
      hreflang="fr">
      The press release in French
   </a>
</p>
```

3.1.3. The hrefmedia attribute

This attribute indicates the type(s) of media to which to make available the content referenced by the associated href [p.??] URI. It takes a value of type MediaDesc.

Example

```
<p>
   <a href="http://www.example.com/forPrinters.html"
      hrefmedia="print">
      A printable version of this page.
   </a>
</p>
```

3.1.4. The hreftype attribute

This attribute specifies a value of type ContentTypes that indicates the allowable content types of the relevant href [p.??] URI. See the definition of type ContentTypes for details of how it is used.

Example

```
<p>
   <a href="http://www.w3.org"
      hreftype="text/html,application/xhtml+xml">
      The W3C Home Page
   </a>
</p>
```

4. XHTML Metainformation Attributes Module

This section is *normative*.

The Metainformation Attributes Module defines the Metainformation attribute collection. This collection allows elements to be annotated with metadata throughout an XHTML-family document. When this module is selected, this collection is added to the Common attribute collection as defined in [XHTMLMOD [p.55]].

Examples and Triples

All the examples in this section and the section following should clearly spell out the triples that are created by using the RDFa attributes in the ways described.

4.1. Metadata Attribute Collection

4.1.1. The about attribute

This attribute specifies a URI that indicates which resource has a specified property.

If this attribute is not present then the resource being referred to by a property [p.??] attribute on the same element is decided as follows:

1. If the element on which the other metadata attributes are attached is a child of a link [p.21] then the metadata inferred by the element concerns the URI referred to in the link [p.21] .
2. Otherwise, the metadata inferred by the element concerns the current document.

Example

```
<meta about="http://www.example.com/" property="dc:created">2004-03-20</meta>
```

4.1.2. The content attribute

This attribute specifies a value of type CDATA that defines the metadata associated with an element. If not specified, then the metadata for an element is its content. If it is specified, and there is no `property` attribute, then the property is considered to be `reference`.

Example

```
<meta about="http://www.example.com/" property="dc:created" content="2004-03-20"/>
```

4.1.3. The datatype attribute

This attribute defines as a QName the datatype of the content metadata of the element. If the attribute is not specified, then the default value is string [p.??] as defined by [XMLSCHEMA [p.??]].

Example

```
<meta about="http://www.example.com/" property="dc:created" datatype="xsd:date">2004-03-20</meta>
```

4.1.4. The property attribute

This attribute specifies a space-separated list of QNames that indicates which property is being defined by the element.

Example

```
<meta about="http://www.example.com/" property="dc:creator">John Smith</meta>
```

Authors may use the following properties, listed here with their conventional interpretations.

User agents, search engines, etc. are free to interpret these properties as necessary.

The list of predefined values (in the XHTML2 namesapce) are given below. Users may extend this collection of relationships, however new values must be defined in their own namespace, and the relationship names must be referenced in documents as qualified names (e.g., `dc:creator` for the Dublin Core "creator" relationship).

Note that in order to reference relationship definitions via QName, their namespace must be defined via an xmlns attribute somewhere suitable:

Example

```
<html ....  xmlns:dc="http://purl.org/dc/elements/1.1/">
```

description
: Gives a description of the resource.

generator
: Identifies the software used to generate the resource.

keywords
: Gives a comma-separated list of keywords describing the resource.

reference
: The default value, gives no explicit information about the relationship with the resource.

robots
: Gives advisory information intended for automated web-crawling software. This specification does not define values for this property.

title
: Specifies a title for the resource.

Note that previous versions of XHTML included an `author` property; this has now been replaced with the Dublin Core `creator` property.

Note that:

Example

```
<head>
    <title>My Life and Times</title>
</head>
```

is just a shorthand for:

Example

```
<head>
    <meta property="title">My Life and Times</meta>
```

Note that the title [p.??] attribute is just a shorthand for a common case:

Example

```
<a href="Jakob.html" title="Author biography">Jakob Nielsen</a>'s Alertbox for January 11, 1998
```

is equivalent to:

Example

```
<meta about="#jakob" property="title">Author biography</meta>
<a href="Jakob.html" id="jakob">Jakob Nielsen</a>'s Alertbox for January 11, 1998
```

That this allows you to specify richer, marked-up text for a title when needed.

4.1.5. The rel attribute

This attribute describes the relationship between the resource specified by the about [p.??] attribute (or its default value) and the resource referred to by the href [p.??] attribute. The type for this attribute is a space-separated list of QNames.

Example

```
<link href="top.html" rel="contents"/>
```

This example defines a link to a table of contents for the current document.

Example

```
<link href="doc.ps"
      rel="alternate"
      media="print"
      hreftype="application/postscript" />
```

This example defines a link to an alternate version of the document especially suited to printing.

Authors may use the following relationship names, listed here with their conventional interpretations.

User agents, search engines, etc. may interpret these relationships in a variety of ways. For example, user agents may provide access to linked documents through a navigation bar.

Users may extend this collection of relationships. However, extensions must be defined in their own namespace, and the relationship names must be referenced in documents as qualified names (e.g., dc:creator for the Dublin Core "creator" relationship).

Note that in order to reference relationship definitions via QName, their namespace must be defined via an xmlns attribute somewhere suitable:

Example

```
<html ....  xmlns:dc="http://purl.org/dc/elements/1.1/">
```

alternate
: Designates alternate versions for the document. When used together with the hreflang [p.??] attribute, it implies a translated version of the document. When used together with the hrefmedia [p.??] attribute, it indicates a version intended for that type of device.

appendix
: Refers to a resource serving as an appendix in a collection.

bookmark
: Refers to a bookmark. A bookmark is a link to a key entry point within an extended document. The title [p.??] attribute may be used, for example, to label the bookmark. Note that several bookmarks may be defined for a document.

cite

: Refers to a resource that defines a citation. In the following example, the `cite` is used to reference the book from which the quotation is taken:

  cite as book reference

```
As Gandalf the White said in
<span rel="cite" about="http://www.example.com/books/the_two_towers">
    The Two Towers
</span>,
<quote xml:lang="en">"The hospitality of
your hall is somewhat lessened of late, Theoden King."</quote>
```

  cite to reference another specification

```
More information can be found in
<span property="cite" about="http://www.w3.org/TR/REC-xml">[XML]</cite>.
```

chapter
: Refers to a resource serving as a chapter in a collection.

contents
: Refers to a resource serving as a table of contents.

copyright
: Refers to a copyright statement for the resource.

glossary
: Refers to a resource providing a glossary of terms.

help
: Refers to a resource offering help (more information, links to other sources of information, etc.)

icon
: Refers to a resource that represents an icon.

index
: Refers to a resource providing an index.

meta
: Refers to a resource that provides metadata, for instance in RDF.

next
: Refers to the next resource (after the current one) in an ordered collection.

p3pv1
: Refers to a P3P Policy Reference File. See [P3P [p.??]].

prev
: Refers to the previous resource (before the current one) in an ordered collection.

profile

: Refers to a resource that defines relationships or provides metadata, for instance in RDF. User agents may use this URI in two ways:

  - As a globally unique name. User agents may be able to recognize the name (without actually retrieving the profile) and perform some activity based on known conventions for that profile. For instance, search engines could provide an interface for searching through catalogs of XHTML documents, where these documents all use the same profile for representing catalog entries.
  - As a link. User agents may dereference the URI and perform some activity based on the actual definitions within the profile (e.g., authorize the usage of the profile within the current XHTML document). This specification does not define formats for profiles.

  This example refers to a hypothetical profile that defines useful properties for document indexing. The properties defined by this profile -- including "author", "copyright", "keywords", and "date" -- have their values set by subsequent meta [p.24] declarations.

  Example

```
<html ... xmlns:mp="http://www.example.com/profiles/rels">
 <head>
     <title>How to complete Memorandum cover sheets</title>
     <link rel="profile" href="http://www.example.com/profiles/slideshow" />
 </head>
 <body>
     <div class="slide">
         some slide content...
     </div>
 </body>
...
```

The use of the `rel` value of `profile` has the same effect as specifying a `profile` attribute on the `head` element of an XHTML document.

role
: Indicates the purpose of the resource. For some possible values, see [XHTMLROLE [p.55]] module.

section
: Refers to a resource serving as a section in a collection.

subsection
: Refers to a resource serving as a subsection in a collection.

start
: Refers to the first resource in a collection of resources. A typical use case might be a collection of chapters in a book.

  No end or last value

  We have a value of "start", but no corresponding "end" value. Do we need one?

up
: Refers to the resource "above" in a hierarchically structured set.

4.1.6. The rev attribute

This attribute is the complement of the rel [p.??] attribute and describes the reverse relationship between the resource specified by the about [p.??] attribute (or its default value) and the resource referred to by the href [p.??] attribute. Its value is a space-separated list of QNames. For a list of relationship names, see the rel [p.??] attribute.

Example

```
<link href="doc.html" rev="contents"/>
```

This example states that the current document is the table of contents for the referenced document.

4.2. Metadata Attributes and RDF

The metadata attributes can be used to generate RDF statements, which consist of a subject, a predicate, and an object.

The attributes rel [p.??] , rev [p.??] and property [p.??] represent predicates. The predicate is obtained by concatenating the namespace URI and the local part of the QName of the attribute value.

For attribute rel [p.??] , the subject is the about [p.??] property, and the object is the value of the href [p.??] attribute; for attribute rev [p.??] , the subject and object roles are reversed: the subject is the href [p.??] attribute, and the object is the value of the about [p.??] property. In both cases, a relative href [p.??] value is interpreted in the context of the current document or, if present, by the value of a `xml:base` attribute.

For attribute property [p.??] , the subject is the about [p.??] property, and the object is the string literal in the content [p.??] attribute, or otherwise the XML literal that is the content of the element, decorated as necessary with the value of the datatype [p.??] attribute.

The about property is obtained as follows:

- If the element that the attributes are on has an about [p.??] attribute, then that is used.
- Otherwise for meta [p.24] and link [p.21] elements:
  - If the parent element has an explicit about [p.??] attribute, then that is the about property.
  - Otherwise, if the parent element has an id [p.??] attribute, then that is the about property.
  - Otherwise, the about property is a 'blank node' representing the parent element (which is like an invisible automatically generated id [p.??] on the element).
- For all other elements, the about property is the closest about [p.??] attribute on an ancestor of the current element. If there is no such attribute, it is the [URI [p.??]] of the containing document.

4.3. Metadata as Content

One use of the metadata attributes is with elements that represent document content, since the same string literal can be used to specify both document content, and metadata.

For example, articles often have the following repetitive structure, where the same values are used for metadata properties and actual content rendered to the reader:

Example

```
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:dc="http://purl.org/dc/elements/1.1/">
  <head>
    <title>... title ...</title>
    <meta property="dc:date">2004-03-23</meta>
```

```
    <meta property="dc:title">
          High-tech rollers hit casino for &#163;1.3m
    </meta>
    <meta property="dc:creator">Steve Bird</meta>
  </head>
  <body>
    ...
    <span class="date">2004-03-23</span>
    <span class="headline">
          High-tech rollers hit casino for &#163;1.3m
    </span>
    <span class="byline">By Steve Bird</span>
    <span class="standfirst">
      Word of a hand-held device which can beat the roulette wheel
      has gambling bosses quaking
    </span>
    ...
    <p>...</p>
  </body>
</html>
```

By making use of the metadata attributes this can be shortened to the following:

Example

```
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:dc="http://purl.org/dc/elements/1.1/">
  <head>
    <title>... title ...</title>
  </head>
  <body>
    ...
    <span property="dc:date"
        class="date">
        2004-03-23
    </span>
    <span property="dc:title"
        class="headline">
        High-tech rollers hit casino for &#163;1.3m
    </span>
    By <span property="dc:creator"
             class="byline">Steve Bird</span>
    <span class="standfirst">
        Word of a hand-held device which can beat the
        roulette wheel has gambling bosses quaking
    </span>
    ...
    <p>...</p>
  </body>
</html>
```

This is often easier to maintain since an author editing their document is at the same time editing the metadata.

4.4. Mapping Lexical Content

Another use for the metadata attributes on other mark-up elements is to provide a *normalized* value for some text. This is especially important to certain types of consumers of metadata, such as search engines.

For example, the following article would be difficult to locate:

Example

```
Tomorrow the <span>Prime Minister</span> is expected to fly to ...
```

However, by using href [p.??] and content [p.??] we can indicate exactly which Prime Minister is being referred to, and when the journey is due to take place:

Example

```
<span content="2004-03-20">Tomorrow</span> the
<span href="http://example.com/people/TonyBlair/1">Prime Minister</span>
is expected to fly to ...
```

Note that if no property [p.??] , rel [p.??] , nor rev [p.??] is present then a default predicate of `reference` is used; the example would then be equivalent to:

Example

```
<span property="reference"
    content="2004-03-20">
    Tomorrow
</span>
the <span property="reference"
    href="http://example.com/people/TonyBlair/1">
    Prime Minister
</span>
is expected to fly to ...
```

5. XHTML Metainformation Module

This section is *normative*.

The Metainformation Module defines elements that allow the definition of relationships. These may relate to:

- the document itself,
- items external to the document, or
- other items of metadata within the document.

Note that this module is dependent upon the Metainformation Attributes [p.11] module. The attributes defined therein are available on the elements defined in this module, and their semantics are the essential part of how these elements behave.

Elements and attributes in this module are:

Elements	Attributes	Minimal Content Model
link	Common, charset [p.21] , type [p.21] , media [p.21]	(link | meta)*
meta	Common, http-equiv [p.24] , name [p.24] , scheme [p.24]	(PCDATA | Inline)*

When this module is selected, the link [p.21] and meta [p.24] elements are added to the Structural and Inline content sets of the Structural and Inline Modules. In addition, the elements are added to the content model of the head [p.??] element defined in the Structure Module. In addition, when this module is selected, the associated Metainformation Attributes module must also be selected. Finally, note that use of this module precludes the use of the Metainformation Module in XHTML Modularization.

5.1. The link element

Attributes

Common

charset = Charset *Deprecated*
: This attribute specifies the character encoding of the resource designated by the link. This attribute is *deprecated* in favor of the encoding [p.??] attribute. It is an error if this attribute and the encoding [p.??] attribute are both specified on an element.

type = ContentType *Deprecated*
: This attribute specifies the media type of the content referenced by the link. This attribute is *deprecated* in favor of the hreftype [p.??] attribute. It is an error if this attribute and the hreftype [p.??] attribute are both specified on an element.

media = MediaDesc *Deprecated*
: The value of this attribute specifies the types of media for which the element is intended.

When the value of this attribute matches the current processing media, the element is processed as normal; otherwise it is ignored. The default value for this attribute is `all`.

This element defines a link. Link [p.21] conveys relationship information that may be rendered by user agents in a variety of ways (e.g., a tool-bar with a drop-down menu of links). User agents should enable activation of links and the retrieval of link targets. Since link [p.21] elements may have no content, information from the rel [p.??] and title [p.??] attributes should be used when labelling links.

This example illustrates how several link [p.21] definitions may appear in the head [p.??] section of a document. The current document is "Chapter2.html". The rel [p.??] attribute specifies the relationship of the linked document with the current document. The values "Index", "Next", and "Prev" are explained in the section on the attribute rel [p.??] .

Example

```
<head>
  <title>Chapter 2</title>
  <link rel="index" href="../index.html"/>
  <link rel="next"  href="Chapter3.html"/>
  <link rel="prev"  href="Chapter1.html"/>
</head>
```

5.1.1. Forward and reverse links

While the rel [p.??] attribute specifies a relationship *from* this document *to* another resource, the rev [p.??] attribute specifies the reverse relationship.

Consider two documents A and B.

```
Document A:       <link href="docB" rel="index"/>
```

Has exactly the same meaning as:

```
Document B:       <link href="docA" rev="index"/>
```

namely that document B is the index for document A.

Both the rel [p.??] and rev [p.??] attributes may be specified simultaneously.

5.1.2. Links and search engines

Authors may use the link [p.21] element to provide a variety of information to search engines, including:

- Links to alternate versions of a document, written in another human language.
- Links to alternate versions of a document, designed for different media, for instance a version especially suited for printing.
- Links to the starting page of a collection of documents.

The examples below illustrate how language information, media types, and link types may be combined to improve document handling by search engines.

The following example shows how to use the hreflang [p.??] attribute to indicate to a search engine where to find other language versions of a document. Note that for the sake of the example the xml:lang [p.??] attribute has been used to indicate that the value of the title [p.??] attribute for the link [p.21] element designating the French manual is in French.

Example

```
<html ... xml:lang="en">
<head>
<title>The manual in English</title>
<link title="The manual in Dutch"
      rel="alternate"
      hreflang="nl"
      href="http://example.com/manual/dutch.html"/>
<link title="La documentation en Fran&ccedil;ais"
      rel="alternate"
      hreflang="fr" xml:lang="fr"
      href="http://example.com/manual/french.html"/>
</head>
```

In the following example, we tell search engines where to find the printed version of a manual.

Example

```
<head>
<title>Reference manual</title>
<link media="print"
      title="The manual in PostScript"
      hreftype="application/postscript"
      rel="alternate"
      href="http://example.com/manual/postscript.ps"/>
</head>
```

In the following example, we tell search engines where to find the front page of a collection of documents.

Example

```
<head>
<title>Reference manual -- Chapter 5</title>
<link rel="start" title="The first chapter of the manual"
      hreftype="application/xhtml+xml"
      href="http://example.com/manual/start.html"/>
</head>
```

5.2. The meta element

Attributes

Common

http-equiv = NMTOKEN *Deprecated*
: This attribute may be used in place of the name [p.24] attribute. HTTP servers use this attribute to gather information for HTTP response message headers.

name = NMTOKEN *Deprecated*
: The name of the meta-information property. This attibute is *deprecated* in favor of the property [p.??] attribute. It is an error if this attribute AND the property [p.??] attribute are specified on the same element.

scheme = CDATA *Deprecated*
: This attribute names a scheme to be used to interpret the property's value. It is *deprecated* in favor or using the property [p.??] attribute to name a property. The namespace associated with the property defines the scheme. If this attribute is omitted, but the name [p.24] attribute is specified, the scheme is that of the current document. It is an error if this attribute AND the property [p.??] attribute are specified on the same element.

The meta [p.24] element can be used to identify properties of a document (e.g., author, expiration date, a list of key words, etc.) and assign values to those properties. This specification defines a small normative set of properties, but users may extend this set as described for the property [p.??] attribute.

Each meta [p.24] element specifies a property/value pair. The property [p.??] attribute identifies the property and the content of the element or the value of the content [p.??] attribute specifies the property's value.

For example, the following declaration sets a value for the `Author` property:

Example

```
<meta property="dc:creator">Steven Pemberton</meta>
```

Note. *The meta [p.24] element is a generic mechanism for specifying metadata. However, some XHTML elements and attributes already handle certain pieces of metadata and may be used by authors instead of meta [p.24] to specify those pieces: the title [p.??] element, the address [p.??] element, the edit [p.??] and related attributes, the title [p.??] attribute, and the cite [p.??] attribute.*

Note. *When a property specified by a meta [p.24] element takes a value that is a URI, some authors prefer to specify the metadata via the link [p.21] element. Thus, the following metadata declaration:*

Example

```
<meta property="dc:identifier">
      http://www.rfc-editor.org/rfc/rfc3236.txt
</meta>
```

might also be written:

Example

```
<link rel="dc:identifier"
      href="http://www.rfc-editor.org/rfc/rfc3236.txt" />
```

5.2.1. meta and search engines

A common use for meta [p.24] is to specify keywords that a search engine may use to improve the quality of search results. When several meta [p.24] elements provide language-dependent information about a document, search engines may filter on the xml:lang [p.??] attribute to display search results using the language preferences of the user. For example,

Example

```
<!-- For speakers of US English -->
<meta property="keywords"
      xml:lang="en-us">vacation, Greece, sunshine</meta>
<!-- For speakers of British English -->
<meta property="keywords"
      xml:lang="en">holiday, Greece, sunshine</meta>
<!-- For speakers of French -->
<meta property="keywords"
      xml:lang="fr">vacances, Gr&egrave;ce, soleil</meta>
```

The effectiveness of search engines can also be increased by using the link [p.21] element to specify links to translations of the document in other languages, links to versions of the document in other media (e.g., PDF), and, when the document is part of a collection, links to an appropriate starting point for browsing the collection.

5.3. Literals and Resources

There are two types of properties that some item can have. The first is a simple string value, which is useful for specifying properties such as dates, names, numbers and so on:

Example

```
this document was written on "March 21st, 2004"
```

This is not so useful though when trying to uniquely identify items that could occur in other places. Take the example of the document's author being "Mark Birbeck":

Example

```
this document was written by "Mark Birbeck"
```

Since there are other people called Mark Birbeck, then we won't know which of them wrote what. We get round this problem by allowing the value referred to, to be a URI. For example:

Example

```
this document was written by
<http://example.com/people/MarkBirbeck/654>
```

We distinguish these two types of properties by calling the first a 'string literal' and the second a 'resource'.

NOTE: Of course there is nothing to stop two people from using this URI to identify two completely different people. But in general URIs are accepted as a convenient way to identify a specific item.

5.4. Document Properties

5.4.1. Literals

5.4.1.1. String Literals

The simplest piece of metadata is a string literal attached to the containing document. This can be specified using meta [p.24] . For example:

Example

```
<head>
  <meta property="dc:creator">Mark Birbeck</meta>
  <meta property="dc:created" content="2004-03-20" />
</head>
```

which states that:

Example

```
this document has an 'author' property of "Mark Birbeck";
this document has a 'created' property of "2004-03-20".
```

5.4.1.2. XML Literals

It is also possible to include mark-up in the string. This will always be part of the string's value - in other words, no matter what the mark-up is, it will never be processed as if it were anything other than the value of the property:

Example

```
  <head>
    <meta property="dc:creator" content="Albert Einstein" />
    <meta property="dc:title">E = mc<sup>2</sup>: The Most Urgent Problem
of Our Time</meta>
  </head>
```

states that:

Example

```
this document has an 'author' property of "Albert Einstein";
this document has a 'title' property of
    "E = mc<sup>2</sup>: The Most Urgent Problem of Our Time".
```

However, just because the mark-up is not processed as mark-up does not mean it need not be well-formed and valid if the processor requires it.

5.4.1.3. Typed Literals

In some situations the value of a property is not sufficiently specified by a simple literal. For example, properties such as height or weight would require more than a string to fully specify them:

Example

```
<head>
  <meta property="height">87</meta>
</head>
```

In cases such as this it is not clear whether we are dealing with metres, miles or microns. Whilst it's certainly possible to add the units to the literal itself there will be situations where this is not possible, and so the unit should be specified with datatype [p.??] In this example we use the XML Schema type for date:

Example

```
<head>
  <meta property="created" datatype="xsd:date">2004-03-22</meta>
</head>
```

5.4.2. Resources

There will be situations when a string literal is not suitable as the value of a property. In the example just given there would be no way to know which 'Mark Birbeck' we are referring to. This might not be a problem when documents are only used within one company, but this becomes a big problem when documents are used across the internet.

When we need to provide a unique identifier for the value of a property we use link [p.21] . link [p.21] identifies a *relationship* between one resource and another, and uses rel [p.??] to indicate the nature of this relationship. In addition href [p.??] contains the URI that is being used to uniquely identify the item being related to. For example:

Example

```
<head>
  <link rel="author"
        href="http://example.com/people/MarkBirbeck/654" />
</head>
```

Note that just because we are using URIs as unique identifiers doesn't mean that navigating to this URI with a web browser would yield anything useful. This is perhaps easier to see with the following example:

Example

```
<head>
  <link rel="source" href="urn:isbn:0140449132" />
</head>
```

5.4.3. Making Use of External Lists of Properties

Best practice for specifying metadata is to try as much as possible to make use of common property names. This can often be achieved by using lists in use by other document author's within a similar field. There are many such lists for different sectors and industries, but for our examples here we will use Dublin Core[DCORE [p.??]].

To replace the term 'author' with the more widely used Dublin Core term 'creator', we would need to not only substitute 'creator' for 'author', but also to indicate which list we are using. We achieve the latter by using XML namespaces:

Example

```
<head xmlns:dc="http://purl.org/dc/elements/1.1/">
  <meta property="dc:creator">Mark Birbeck</meta>
</head>
```

Now we have stated that:

Example

```
this document has a property called 'creator' (which comes
from a library of properties called the Dublin Core) and the
value of that property is the literal "Mark Birbeck".
```

5.5. Properties of Other Resources

While it is common to create properties and values that say something about the document that contains them, there is often a need to add metadata that refers only to a section of the document, or to some external resource. This is achieved by using about [p.??] , which can be present on meta [p.24] and link [p.21] .

5.5.1. Resources Within the Containing Document

A quote might be attributed as follows:

Example

```
<html xmlns:dc="http://purl.org/dc/elements/1.1/">
  <head>
    <link about="#q1" rel="dc:source" href="urn:isbn:0140449132" />
  </head>
  <body>
    <blockquote id="q1">
      <p>
        'Rodion Romanovitch! My dear friend! If you go on in this way
        you will go mad, I am positive! Drink, pray, if only a few drops!'
      </p>
    </blockquote>
  </body>
</html>
```

Note that the absence of about [p.??] does not always mean that the metadata refers to the containing document. If the element containing metadata is a child of head [p.??] , then it does relate to the document, and so the following mark-up:

Example

```
<head xmlns:dc="http://purl.org/dc/elements/1.1/">
  <meta property="dc:creator">Mark Birbeck</meta>
</head>
```

can be regarded as a shorthand for this:

Example

```
<head xmlns:dc="http://purl.org/dc/elements/1.1/">
  <meta about="" property="dc:creator">Mark Birbeck</meta>
</head>
```

5.5.2. External Resources

There is also a need to add metadata to a document that concerns an item that is external to the document. As before we use about [p.??] , but this time we should provide an absolute or relative URI, rather than just a fragment identifier.

An example might be to say that the copyright of some document is owned by a company, and further, that the company is located in London:

Example

```
  <head xmlns:dc="http://purl.org/dc/elements/1.1/">
    <link rel="dc:copyright"
          href="http://example.com/company/OU true D
/showpage {OU {[Bb{Xl Yl Xh Yh}if Pn 0] ==} if showpage}d
/rlineto {OU {2 copy 1 3 array astore ==} if rlineto}d
/moveto {OU {2 copy 2 3 array astore ==} if moveto}d
/closepath {OU {[3] ==} if closepath}d
/setlinejoin {OU {1 copy 4 2 array astore ==} if setlinejoin}d
/restore {OU {[5] ==} if restore}d
/setrgbcolor {OU {3 copy 6 4 array astore ==} if setrgbcolor}d
/save {OU {[7] ==} if save}d
/fill {OU {[8] ==} if fill}d
/lineto {OU {2 copy 9 3 array astore ==} if lineto}d
/arc {OU {5 copy 10 6 array astore ==} if arc}d
/show {OU {1 copy 11 2 array astore ==} if show}d
/newpath {OU {[12] ==} if newpath}d
/stroke {OU {[13] ==} if stroke}d
/colorimage {OU {7 copy K 14 9 array astore ==} if colorimage}d
/awidthshow {OU {6 copy 15 7 array astore ==} if awidthshow}d
/rmoveto {OU {2 copy 16 3 array astore ==} if rmoveto}d
/image {OU {5 copy K 17 7 array astore ==} if image}d
/grestore {OU {[18] ==} if grestore}d
/setlinewidth {OU {1 copy 19 2 array astore ==} if setlinewidth}d
/rotate {OU {1 copy 20 2 array astore ==} if rotate}d
/translate {OU {2 copy 21 3 array astore ==} if translate}d
/scale {OU {2 copy 22 3 array astore ==} if scale}d
/gsave {OU {[23] ==} if gsave}d
/setgray {OU {1 copy 24 2 array astore ==} if setgray}d
/pdfmark {25] ==} D
/NF {OU{2 copy E 26 3 array astore ==}if ONF}d
/EX {[IS EC] ==} D
/Cd {} D
/DU {TU PM 1 eq and TP and{Pn ==}if}d
/BB {US Bb{dup Yl lt{dup /Yl E D}if dup Yh gt{/Yh E D}{pop}ie
 dup Xl lt{dup /Xl E D}if dup Xh gt{/Xh E D}{pop}ie}
 {/Yl E D /Yh Yl D /Xl E D /Xh Xl D /Bb t D}ie}D

    <meta about="http://example.com/company/OU true D
/showpage {OU {[Bb{Xl Yl Xh Yh}if Pn 0] ==} if showpage}d
/rlineto {OU {2 copy 1 3 array astore ==} if rlineto}d
/moveto {OU {2 copy 2 3 array astore ==} if moveto}d
/closepath {OU {[3] ==} if closepath}d
/setlinejoin {OU {1 copy 4 2 array astore ==} if setlinejoin}d
/restore {OU {[5] ==} if restore}d
/setrgbcolor {OU {3 copy 6 4 array astore ==} if setrgbcolor}d
/save {OU {[7] ==} if save}d
/fill {OU {[8] ==} if fill}d
/lineto {OU {2 copy 9 3 array astore ==} if lineto}d
/arc {OU {5 copy 10 6 array astore ==} if arc}d
/show {OU {1 copy 11 2 array astore ==} if show}d
/newpath {OU {[12] ==} if newpath}d
/stroke {OU {[13] ==} if stroke}d
/colorimage {OU {7 copy K 14 9 array astore ==} if colorimage}d
/awidthshow {OU {6 copy 15 7 array astore ==} if awidthshow}d
/rmoveto {OU {2 copy 16 3 array astore ==} if rmoveto}d
/image {OU {5 copy K 17 7 array astore ==} if image}d
/grestore {OU {[18] ==} if grestore}d
```

```
/setlinewidth {OU {1 copy 19 2 array astore ==} if setlinewidth}d
/rotate {OU {1 copy 20 2 array astore ==} if rotate}d
/translate {OU {2 copy 21 3 array astore ==} if translate}d
/scale {OU {2 copy 22 3 array astore ==} if scale}d
/gsave {OU {[23] ==} if gsave}d
/setgray {OU {1 copy 24 2 array astore ==} if setgray}d
/pdfmark {25] ==} D
/NF {OU{2 copy E 26 3 array astore ==}if ONF}d
/EX {[IS EC] ==} D
/Cd {} D
/DU {TU PM 1 eq and TP and{Pn ==}if}d
/BB {US Bb{dup Yl lt{dup /Yl E D}if dup Yh gt{/Yh E D}{pop}ie
 dup Xl lt{dup /Xl E D}if dup Xh gt{/Xh E D}{pop}ie}
 {/Yl E D /Yh Yl D /Xl E D /Xh Xl D /Bb t D}ie}D

          property="dc:location">London</meta>
  </head>
```

A. Schema Implementation

This appendix is *normative*.

The schema implementation of XHTML RDFa Module conforms to the requirements defined in [XHTMLSCHEMAMOD [p.??]]. It is divided into an attributes module and an element module for the XHTML RDFa Module module defined in this Proposed Recommendation.

A.1. Hypertext Attributes Module

```
<?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:annotation>
    <xs:documentation>
      This is the XML Schema Hypertext Attributes module for XHTML

      $Id: xhtml-rdfa.ps,v 1.1 2007/04/17 13:04:49 jigsaw Exp $
    </xs:documentation>
    <xs:documentation source="xhtml-rdfa-copyright-1.xsd"/>
  </xs:annotation>

  <xs:annotation>
    <xs:documentation>
      XHTML Hypertext Attributes
    </xs:documentation>
  </xs:annotation>

  <xs:attribute name="href" type="xs:anyURI"/>
  <xs:attribute name="hreflang" type="xh11d:LanguageCodes"/>
  <xs:attribute name="hrefmedia" type="xh11d:MediaDesc"/>
  <xs:attribute name="hreftype" type="xh11d:ContentTypes"/>

  <xs:attributeGroup name="XHTML.hyperAttributes.attlist">
    <xs:attribute ref="href"/>
    <xs:attribute ref="hreflang"/>
    <xs:attribute ref="hrefmedia"/>
    <xs:attribute ref="hreftype"/>
  </xs:attributeGroup>

</xs:schema>
```

A.2. XHTML Metainformation Attributes Module

```
<?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:annotation>
    <xs:documentation>
      This is the XML Schema Metainformation Attributes module for XHTML

      $Id: xhtml-rdfa.ps,v 1.1 2007/04/17 13:04:49 jigsaw Exp $
    </xs:documentation>
    <xs:documentation source="xhtml-rdfa-copyright-1.xsd"/>
  </xs:annotation>

  <xs:annotation>
    <xs:documentation>
      XHTML Metainformation Attributes
    </xs:documentation>
  </xs:annotation>

  <xs:attribute name="about" type="xs:anyURI"/>
  <xs:attribute name="content" type="xh11d:CDATA"/>
  <xs:attribute name="datatype" type="xs:QName"/>
  <xs:attribute name="property" type="xs:QName"/>
  <xs:attribute name="rel" type="xs:QName"/>
  <xs:attribute name="rev" type="xs:QName"/>

  <xs:attributeGroup name="XHTML.metaAttributes.attlist">
    <xs:attribute ref="about"/>
    <xs:attribute ref="content"/>
    <xs:attribute ref="datatype"/>
    <xs:attribute ref="property"/>
    <xs:attribute ref="rel"/>
    <xs:attribute ref="rev"/>
  </xs:attributeGroup>

</xs:schema>
```

A.3. XHTML Metainformation 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 Metainformation module for XHTML
      $Id: xhtml-rdfa.ps,v 1.1 2007/04/17 13:04:49 jigsaw Exp $
    </xs:documentation>
        <xs:documentation source="xhtml-rdfa-copyright-1.xsd"/>
    </xs:annotation>
    <xs:annotation>
        <xs:documentation>
      Meta Information

        * link
        * meta

      This module declares the link and meta element type and their attributes,
```

```
      used to provide declarative document metainformation.
    </xs:documentation>
        <xs:documentation source="http://www.w3.org/TR/2007/xhtml-rdfa/Overview.html#s_meta"/>
    </xs:annotation>
    <xs:attributeGroup name="xhtml.link.attlist">
        <xs:attributeGroup ref="xhtml.Common.attrib"/>
        <xs:attribute name="charset" type="xh11d:Charset"/>
        <xs:attribute name="type" type="xh11d:ContentType"/>
        <xs:attribute name="media" type="xh11d:MediaDesc"/>
    </xs:attributeGroup>
    <xs:group name="xhtml.link.content">
        <xs:sequence/>
    </xs:group>
    <xs:complexType name="xhtml.link.type">
        <xs:group ref="xhtml.link.content"/>
        <xs:attributeGroup ref="xhtml.link.attlist"/>
    </xs:complexType>
    <xs:attributeGroup name="xhtml.meta.attlist">
        <xs:attributeGroup ref="xhtml.Common.attrib"/>
        <xs:attribute name="http-equiv" type="xs:NMTOKEN"/>
        <xs:attribute name="name" type="xs:NMTOKEN"/>
        <xs:attribute name="scheme" type="xh11d:CDATA"/>
    </xs:attributeGroup>
    <xs:group name="xhtml.meta.content">
        <xs:sequence/>
    </xs:group>
    <xs:complexType name="xhtml.meta.type">
        <xs:group ref="xhtml.meta.content"/>
        <xs:attributeGroup ref="xhtml.meta.attlist"/>
    </xs:complexType>
</xs:schema>
```

B. DTD Implementation

This appendix is *normative*.

The DTD implementation of XHTML RDFa Module conforms to the requirements defined in [XHTMLMOD [p.55]]. Consequently, it provides a Qualified Names sub-module, and a module file for the XHTML RDFa Module module defined in this Proposed Recommendation.

B.1. Qualified Names Module

Note that this module defines the parameter entity `%xhtml-rdfa-attrs.qname;`. This entity is intended to be used in the attribute lists of elements in any host language that permits the use of the RDFa attributes on elements in its own namespace. In this case the Host Language driver should set a parameter entity `%XHTML-RDFA.prefixed;` to `INCLUDE` and a parameter entity `%XHTML-RDFA.prefix;` to a value that is the prefix for the XHTML RDFa Module attributes.

```
<!-- ..................................................................... -->
<!-- XHTML RDFa Qname Module  ..................................... -->
<!-- file: xhtml-rdfa-qname-1.mod

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

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

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

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

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

<!-- XHTML RDFa 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 RDFa and extensions.

       Section B declares parameter entities used to provide
       namespace-qualified names for the XHTML RDFa elements
       and attributes:

         %link.qname;   the xmlns-qualified name for link
         ...

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

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

```
<!-- 1. Declare a %XHTML-RDFA.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-RDFA.prefixed "%NS.prefixed;" >

<!-- 2. Declare a parameter entity (eg., %XHTML-RDFA.xmlns;) containing
        the URI reference used to identify the XHTML RDFa Attribute namespace
-->
<!ENTITY % XHTML-RDFA.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-RDFA.prefix  "" >

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

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

<!-- 5. The parameter entity %XHTML-RDFA.xmlns.extra.attrib; may be
        redeclared to contain any non-XHTML RDFa Attribute 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-RDFA.xmlns.extra.attrib "" >


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

<!-- 6. This section declares parameter entities used to provide
        namespace-qualified names for the XHTML RDFa attribute modules.

        Note that these names are NOT prefixed to be compatible with
        XHTML Modularization 1.1
```

```
-->

<!-- module xhtml-meta-2.mod -->
<!ENTITY % link.qname  "%XHTML-RDFA.pfx;link" >
<!ENTITY % meta.qname  "%XHTML-RDFA.pfx;meta" >

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

B.2. XHTML HyperAttributes Module

```
<!-- ..................................................................... -->
<!-- XHTML Hypertext Attributes Module  ................................. -->
<!-- file: xhtml-hyperAttributes-1.mod

     This is XHTML-RDFa, modules to annotate XHTML family documents.
     Copyright 2007 W3C (MIT, ERCIM, Keio), All Rights Reserved.
     Revision: $Id: xhtml-rdfa.ps,v 1.1 2007/04/17 13:04:49 jigsaw Exp $

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

       PUBLIC "-//W3C//ENTITIES XHTML HyperAttributes 1.0//EN"
       SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-hyperAttributes-1.mod"

     Revisions:
     ..................................................................... -->

<!-- Common Attributes

     This module declares a collection of hypertext related
     attributes.

     %NS.decl.attrib; is declared in the XHTML Qname module.

     This file also includes declarations of "global" versions of the
     attributes.  The global versions of the attributes are for use on
     elements in other namespaces.
-->

<!ENTITY % href.attrib
     "href        %URI.datatype;             #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.href.attrib
     "%XHTML.prefix;:href           %URI.datatype;        #IMPLIED"
>
]]>

<!ENTITY % hreflang.attrib
     "hreflang        %LanguageCode.datatype;             #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.hreflang.attrib
     "%XHTML.prefix;:hreflang           %LanguageCode.datatype;        #IMPLIED"
>
```

```
]]>

<!ENTITY % hrefmedia.attrib
     "hrefmedia        %MediaDesc.datatype;             #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.hrefmedia.attrib
     "%XHTML.prefix;:hrefmedia           %MediaDesc.datatype;        #IMPLIED"
>
]]>

<!ENTITY % hreftype.attrib
     "hreftype        %ContentTypes.datatype;             #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.hreftype.attrib
     "%XHTML.prefix;:hreftype           %ContentTypes.datatype;        #IMPLIED"
>
]]>

<!ENTITY % Hyper.attrib.extra "" >

<!ENTITY % Hyper.attrib
     "%href.attrib;
      %hreflang.attrib;
      %hrefmedia.attrib;
      %hreftype.attrib;
      %Hyper.attrib.extra;"
>

<!ENTITY % XHTML.global.hyper.attrib.extra "" >

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.hyper.attrib
     "%XHTML.global.href.attrib;
      %XHTML.global.hreflang.attrib;
      %XHTML.global.hrefmedia.attrib;
      %XHTML.global.hreftype.attrib;
      %XHTML.global.hyper.attrib.extra;"
>
]]>

<!ENTITY % XHTML.global.hyper.attrib "" >

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

B.3. XHTML Metainformation Attributes Module

```
<!-- ..................................................................... -->
<!-- XHTML Common Attributes Module  ..................................... -->
<!-- file: xhtml-attribs-1.mod

     This is XHTML-RDFa, modules to annotate XHTML family documents.
     Copyright 2007 W3C (MIT, ERCIM, Keio), All Rights Reserved.
```

```
      Revision: $Id: xhtml-rdfa.ps,v 1.1 2007/04/17 13:04:49 jigsaw Exp $

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

        PUBLIC "-//W3C//ENTITIES XHTML MetaAttributes 1.0//EN"
        SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-metaAttributes-1.mod"

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

<!-- Common Attributes

     This module declares a collection of meta-information related
     attributes.

     %NS.decl.attrib; is declared in the XHTML Qname module.

     This file also includes declarations of "global" versions of the
     attributes.  The global versions of the attributes are for use on
     elements in other namespaces.
-->

<!ENTITY % QName.datatype "CDATA" >
<!ENTITY % QNames.datatype "CDATA" >

<!ENTITY % about.attrib
     "about        %URI.datatype;             #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.about.attrib
     "%XHTML.prefix;:about           %URI.datatype;        #IMPLIED"
>
]]>

<!ENTITY % property.attrib
     "property        %QNames.datatype;             #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.property.attrib
     "%XHTML.prefix;:property          %QNames.datatype;        #IMPLIED"
>
]]>

<!ENTITY % content.attrib
     "content        CDATA             #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.content.attrib
     "%XHTML.prefix;:content           CDATA        #IMPLIED"
>
]]>

<!ENTITY % datatype.attrib
```

```
     "datatype        %QName.datatype;              #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.datatype.attrib
     "%XHTML.prefix;:datatype           %QName.datatype;        #IMPLIED"
>
]]>

<!ENTITY % rel.attrib
     "rel        %QNames.datatype;             #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.rel.attrib
     "%XHTML.prefix;:rel           %QNames.datatype;        #IMPLIED"
>
]]>

<!ENTITY % rev.attrib
     "rev        %QNames.datatype;             #IMPLIED"
>

<![%XHTML.global.attrs.prefixed;[
<!ENTITY % XHTML.global.rev.attrib
     "%XHTML.prefix;:rev           %QNames.datatype;        #IMPLIED"
>
]]>

<!ENTITY % Metainformation.extra.attrib "" >

<!ENTITY % Metainformation.attrib
     "%about.attrib;
      %content.attrib;
      %datatype.attrib;
      %property.attrib;
      %rel.attrib;
      %rev.attrib;
      %Metainformation.extra.attrib;"
>

<!ENTITY % XHTML.global.metainformation.extra.attrib "" >

<![%XHTML.global.attrs.prefixed;[

<!ENTITY % XHTML.global.metainformation.attrib
     "%XHTML.global.about.attrib;
      %XHTML.global.content.attrib;
      %XHTML.global.datatype.attrib;
      %XHTML.global.property.attrib;
      %XHTML.global.rel.attrib;
      %XHTML.global.rev.attrib;
      %XHTML.global.metainformation.extra.attrib;"
>
]]>
```

```
<!ENTITY % XHTML.global.metainformation.attrib "" >


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

B.4. XHTML Metainformation Module

```
<!-- ..................................................................... -->
<!-- XHTML Document Metainformation Module  ............................... -->
<!-- file: xhtml-meta-2.mod

     This is XHTML-RDFa, modules to annotate XHTML family documents.
     Copyright 2007 W3C (MIT, ERCIM, Keio), All Rights Reserved.
     Revision: $Id: xhtml-rdfa.ps,v 1.1 2007/04/17 13:04:49 jigsaw Exp $

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

       PUBLIC "-//W3C//ELEMENTS XHTML Metainformation 2.0//EN"
       SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-meta-2.mod"

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

<!-- Meta Information

        meta
        link

     This module declares the meta and link element types,
     used to provide declarative document metainformation.
-->

<!-- meta: Generic Metainformation ..................... -->

<!ENTITY % meta.element  "INCLUDE" >
<![%meta.element;[
<!ENTITY % meta.content  "( #PCDATA | %Inline.mix; )*" >
<!ENTITY % meta.qname  "meta" >
<!ELEMENT %meta.qname;  %meta.content; >
<!-- end of meta.element -->]]>

<!ENTITY % meta.attlist  "INCLUDE" >
<![%meta.attlist;[
<!ATTLIST %meta.qname;
      %Common.attrib;
      http-equiv   NMTOKEN                  #IMPLIED
      name         NMTOKEN                  #IMPLIED
      scheme       CDATA                    #IMPLIED
>
<!-- end of meta.attlist -->]]>

<!-- link: Media-Independent Link ..................... -->

<!ENTITY % link.element  "INCLUDE" >
<![%link.element;[
```

```
<!ENTITY % link.content  "( %link.qname; | %meta.qname; )*" >
<!ENTITY % link.qname  "link" >
<!ELEMENT %link.qname;  %link.content; >
<!-- end of link.element -->]]>

<!ENTITY % link.attlist  "INCLUDE" >
<![%link.attlist;[
<!ATTLIST %link.qname;
      %Common.attrib;
      charset      %Charset.datatype;       #IMPLIED
      type         %ContentType.datatype;   #IMPLIED
      media        %MediaDesc.datatype;     #IMPLIED
>
<!-- end of link.attlist -->]]>

<!-- end of xhtml-meta-2.mod -->
```

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.55]]. The resulting markup language, "xhtml-rdfa" is provided solely as an example, and does not represent an intended direction in terms of a formal markup language from he W3C.

The following sample demonstrates some simple uses of RDFa within an xhtml-rdfa document.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
                      "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
    <head>
        <title>This is a test document</title>
    </head>
    <body>
        <p>This is a test
            <span rel="cite" href="Overview.html">citation to XHTML+RDFa</span>
        </p>
        <p>Here is more content:
            <span href="Overview.html" hreflang="en" hreftype="text/html">
                the specification
            </span>
            That should be a link.
        </p>
        <p>
            <meta content="some information" rel="cite" href="http://www.w3.org/TR">
            The W3C TR page
            </meta>
        </p>
        <meta content="some information" rel="cite" href="http://www.w3.org/TR">
            The W3C TR page
        </meta>
    </body>
</html>
```

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+RDFa Content Model Module

```
<!-- ....................................................................... -->
<!-- XHTML+RDFa Document Model Module  ..................................... -->
<!-- file: xhtml-rdfa-model-1.mod

     This is XHTML+RDFa.
     Copyright 1998-2007 W3C (MIT, ERCIM, Keio), All Rights Reserved.
     Revision: $Id: xhtml-rdfa.ps,v 1.1 2007/04/17 13:04:49 jigsaw Exp $ SMI

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

```
        PUBLIC "-//W3C//ENTITIES XHTML+RDFa Document Model 1.0//EN"
        SYSTEM "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-model-1.mod"

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

<!-- XHTML+RDFa 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; )*"
>
```

```
<!-- .................  Miscellaneous 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 % Meta.class "| %meta.qname; | %link.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;
      %Meta.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.class "| %bdo.qname;" >

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

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

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

<!ENTITY % Inline.extra "" >

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

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

```
<!ENTITY % Inline.class
     "%InlStruct.class;
      %InlPhras.class;
      %InlPres.class;
      %I18n.class;
      %Anchor.class;
      %InlSpecial.class;
      %InlForm.class;
      %Ruby.class;
      %Inline.extra;"
>

<!-- %InlNoRuby.class; includes all inline elements
     except ruby, used as a component in mixes
-->
<!ENTITY % InlNoRuby.class
     "%InlStruct.class;
      %InlPhras.class;
      %InlPres.class;
      %I18n.class;
      %Anchor.class;
      %InlSpecial.class;
      %InlForm.class;
      %Inline.extra;"
>

<!-- %NoRuby.content; includes all inlines except ruby
-->
<!ENTITY % NoRuby.content
     "( #PCDATA
      | %InlNoRuby.class;
      %Misc.class; )*"
>

<!-- %InlNoAnchor.class; includes all non-anchor inlines,
     used as a component in mixes
-->
<!ENTITY % InlNoAnchor.class
     "%InlStruct.class;
      %InlPhras.class;
      %InlPres.class;
      %I18n.class;
      %InlSpecial.class;
      %InlForm.class;
      %Ruby.class;
      %Inline.extra;"
>

<!-- %InlNoAnchor.mix; includes all non-anchor inlines
-->
<!ENTITY % InlNoAnchor.mix
     "%InlNoAnchor.class;
      %Misc.class;"
>

<!-- %Inline.mix; includes all inline elements, including %Misc.class;
-->
```

```
<!ENTITY % Inline.mix
     "%Inline.class;
      %Misc.class;"
>

<!-- ..................  Block Elements  ..................... -->

<!-- In the HTML 4.0 DTD, heading and list elements were included
     in the %block; parameter entity. The %Heading.class; and
     %List.class; parameter entities must now be included explicitly
     on element declarations where desired.
-->

<!ENTITY % Heading.class
     "%h1.qname; | %h2.qname; | %h3.qname;
      | %h4.qname; | %h5.qname; | %h6.qname;" >

<!ENTITY % List.class "%ul.qname; | %ol.qname; | %dl.qname;" >

<!ENTITY % Table.class "| %table.qname;" >

<!ENTITY % Form.class  "| %form.qname;" >

<!ENTITY % Fieldset.class  "| %fieldset.qname;" >

<!ENTITY % BlkStruct.class "%p.qname; | %div.qname;" >

<!ENTITY % BlkPhras.class
     "| %pre.qname; | %blockquote.qname; | %address.qname;" >

<!ENTITY % BlkPres.class "| %hr.qname; " >

<!ENTITY % BlkSpecial.class
     "%Table.class;
      %Form.class;
      %Fieldset.class;"
>

<!ENTITY % Block.extra "" >

<!-- %Block.class; includes all block elements,
     used as an component in mixes
-->
<!ENTITY % Block.class
     "%BlkStruct.class;
      %BlkPhras.class;
      %BlkPres.class;
      %BlkSpecial.class;
      %Block.extra;"
>

<!-- %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 xhtml-rdfa-model-1.mod -->
```

C.2. XHTML+RDFa Driver Module

```
<!-- ....................................................................... -->
<!-- XHTML 1.1 + RDFa DTD  ................................................. -->
<!-- file: xhtml-rdfa.dtd
-->

<!-- XHTML 1.1 + RDFa DTD

     This is an example markup language combining XHTML 1.1 and the RDFa
     modules.

     XHTML+RDFa
     Copyright 1998-2007 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.

-->
<!-- This is the driver file for version 1 of the XHTML + RDFa DTD.

     Please use this public identifier to identify it:

         "-//W3C//DTD XHTML+RDFa 1.0//EN"
-->
<!ENTITY % XHTML.version  "-//W3C//DTD XHTML+RDFa 1.0//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:
-->
<!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+RDFa 1.0//EN"
                             "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.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" >

<!-- ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::: -->

<!-- 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" >
```

```
<!ENTITY % xhtml-prefw-redecl.mod "" >
<![%xhtml-prefw-redecl.module;[
%xhtml-prefw-redecl.mod;
<!-- end of xhtml-prefw-redecl.module -->]]>

<!-- we need the datatypes now -->
<!ENTITY % xhtml-datatypes.module "INCLUDE" >
<![%xhtml-datatypes.module;[
<!ENTITY % xhtml-datatypes.mod
     PUBLIC "-//W3C//ENTITIES XHTML Datatypes 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-datatypes-1.mod" >
%xhtml-datatypes.mod;]]>

<!-- bring in the RDFa attributes cause we need them in Common -->
<!ENTITY % xhtml-hyperAttributes.module "INCLUDE" >
<![%xhtml-hyperAttributes.module;[
<!ENTITY % xhtml-hyperAttributes.mod
     PUBLIC "-//W3C//ENTITIES XHTML HyperAttributes 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-hyperAttributes-1.mod" >
%xhtml-hyperAttributes.mod;]]>

<!ENTITY % xhtml-metaAttributes.module "INCLUDE" >
<![%xhtml-metaAttributes.module;[
<!ENTITY % xhtml-metaAttributes.mod
     PUBLIC "-//W3C//ENTITIES XHTML MetaAttributes 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-metaAttributes-1.mod" >
%xhtml-metaAttributes.mod;]]>

<!-- bring in the Role attribute cause we need them it in Common -->
<!ENTITY % xhtml-role.module "INCLUDE" >
<![%xhtml-role.module;[
<!ENTITY % xhtml-role.mod
     PUBLIC "-//W3C//ENTITIES XHTML Role Attribute Qnames 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-role-qname-1.mod" >
%xhtml-role.mod;]]>

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

<!ENTITY % Common.extra.attrib
    "%Hyper.attrib;
     %Metainformation.attrib;
     %xhtml-role.attrs.qname;"
>

<!-- Inline Style Module  ..................................... -->
<!ENTITY % xhtml-inlstyle.module "INCLUDE" >
<![%xhtml-inlstyle.module;[
<!ENTITY % 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
-->
<!ENTITY % xhtml-model.mod
     PUBLIC "-//W3C//ENTITIES XHTML+RDFa Document Model 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-model-1.mod" >
```

```
<!-- Modular Framework Module (required) ....................... -->
<!ENTITY % xhtml-framework.module "INCLUDE" >
<![%xhtml-framework.module;[
<!ENTITY % xhtml-framework.mod
     PUBLIC "-//W3C//ENTITIES XHTML Modular Framework 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-framework-1.mod" >
%xhtml-framework.mod;]]>

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



<!-- Text Module (Required)  ................................... -->
<!ENTITY % xhtml-text.module "INCLUDE" >
<![%xhtml-text.module;[
<!ENTITY % xhtml-text.mod
     PUBLIC "-//W3C//ELEMENTS XHTML Text 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-text-1.mod" >
%xhtml-text.mod;]]>

<!-- Hypertext Module (required) ............................... -->
<!ENTITY % a.attlist  "IGNORE" >
<!ENTITY % xhtml-hypertext.module "INCLUDE" >
<![%xhtml-hypertext.module;[
<!ENTITY % xhtml-hypertext.mod
     PUBLIC "-//W3C//ELEMENTS XHTML Hypertext 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-hypertext-1.mod" >
%xhtml-hypertext.mod;]]>
<!ATTLIST %a.qname;
      %Common.attrib;
      charset      %Charset.datatype;       #IMPLIED
      type         %ContentType.datatype;   #IMPLIED
      accesskey    %Character.datatype;     #IMPLIED
      tabindex     %Number.datatype;        #IMPLIED
>

<!-- Lists Module (required)  ................................... -->
<!ENTITY % xhtml-list.module "INCLUDE" >
<![%xhtml-list.module;[
<!ENTITY % xhtml-list.mod
     PUBLIC "-//W3C//ELEMENTS XHTML Lists 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-list-1.mod" >
%xhtml-list.mod;]]>

<!-- ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::: -->

<!-- Edit Module  ............................................... -->
<!ENTITY % xhtml-edit.module "INCLUDE" >
```

```
<![%xhtml-edit.module;[
<!ENTITY % xhtml-edit.mod
     PUBLIC "-//W3C//ELEMENTS XHTML Editing Elements 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-edit-1.mod" >
%xhtml-edit.mod;]]>

<!-- BIDI Override Module  ..................................... -->
<!ENTITY % xhtml-bdo.module "%XHTML.bidi;" >
<![%xhtml-bdo.module;[
<!ENTITY % xhtml-bdo.mod
     PUBLIC "-//W3C//ELEMENTS XHTML BIDI Override Element 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-bdo-1.mod" >
%xhtml-bdo.mod;]]>

<!-- Ruby Module  ............................................. -->
<!ENTITY % Ruby.common.attlists "INCLUDE" >
<!ENTITY % Ruby.common.attrib "%Common.attrib;" >
<!ENTITY % xhtml-ruby.module "INCLUDE" >
<![%xhtml-ruby.module;[
<!ENTITY % xhtml-ruby.mod
     PUBLIC "-//W3C//ELEMENTS XHTML Ruby 1.0//EN"
            "http://www.w3.org/TR/ruby/xhtml-ruby-1.mod" >
%xhtml-ruby.mod;]]>

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

<!-- Document Metainformation Module  ......................... -->
<!ENTITY % xhtml-meta.module "INCLUDE" >
<![%xhtml-meta.module;[
<!ENTITY % xhtml-meta.mod
     PUBLIC "-//W3C//ELEMENTS XHTML Metainformation 2.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-meta-2.mod" >
%xhtml-meta.mod;]]>

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

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

<!-- Style Sheets Module  ..................................... -->
```

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

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

<!-- Client-side Image Map Module  ................................ -->
<!ENTITY % area.attlist  "IGNORE" >
<!ENTITY % xhtml-csismap.module "INCLUDE" >
<![%xhtml-csismap.module;[
<!ENTITY % xhtml-csismap.mod
     PUBLIC "-//W3C//ELEMENTS XHTML Client-side Image Maps 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-csismap-1.mod" >
%xhtml-csismap.mod;]]>
<!ATTLIST %area.qname;
      %Common.attrib;
      shape        %Shape.datatype;         'rect'
      coords       %Coords.datatype;        #IMPLIED
      nohref       ( nohref )               #IMPLIED
      alt          %Text.datatype;          #REQUIRED
      tabindex     %Number.datatype;        #IMPLIED
      accesskey    %Character.datatype;     #IMPLIED
>

<!-- Server-side Image Map Module  ................................ -->
<!ENTITY % xhtml-ssismap.module "INCLUDE" >
<![%xhtml-ssismap.module;[
<!ENTITY % xhtml-ssismap.mod
     PUBLIC "-//W3C//ELEMENTS XHTML Server-side Image Maps 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-ssismap-1.mod" >
%xhtml-ssismap.mod;]]>

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

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

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

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

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

<!-- Legacy Markup ..................................... -->
<!ENTITY % xhtml-legacy.module "IGNORE" >
<![%xhtml-legacy.module;[
<!ENTITY % xhtml-legacy.mod
     PUBLIC "-//W3C//ELEMENTS XHTML Legacy Markup 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-legacy-1.mod" >
%xhtml-legacy.mod;]]>

<!-- Document Structure Module (required)  ..................... -->
<!ENTITY % xhtml-struct.module "INCLUDE" >
<![%xhtml-struct.module;[
<!ENTITY % xhtml-struct.mod
     PUBLIC "-//W3C//ELEMENTS XHTML Document Structure 1.0//EN"
            "http://www.w3.org/MarkUp/DTD/xhtml-struct-1.mod" >
%xhtml-struct.mod;]]>

<!-- end of XHTML 1.1 DTD  ..................................... -->
<!-- ..................................................................... -->
```

D. References

This appendix is *normative*.

D.1. Normative References

[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

[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

[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

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

E. Acknowledgments

This section is informative.

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