

# Legal XHTML

## A Prospective

## XHTML 2.0 Family Document Type

by [John McClure](mailto:jmcclure@hypergrove.com), jmcclure@hypergrove.com

Working Draft, April 11, 2005

Legal XHTML is an XML grammar used to markup the text of legal instruments and of optionally associated workflow documents. This grammar involves three building blocks: the Extendable Hypertext Markup Language (XHTML2); the Resource Description Framework (RDF); and the Dublin Core (DC) elements. XHTML2 and RDF are recommendations from the World Wide Web Consortium (W3C), and the DC elements are recommended by the Dublin Core Metadata Initiative.

The motivation for the Legal XHTML grammar springs foremost from recommendations made by the author as a member of the eContracts Technical Committee of LegalXML, a chapter of OASIS, concerning an XML standard for legal instruments that is aligned with widely deployed W3C recommendations such as XHTML. The eContracts Technical Committee determined however to create a custom XML grammar for legal instruments that is unhindered by the considerable flexibility provided by the XHTML 2.0 architecture and its grammar.

The Legal XHTML Document Type addresses the needs of authors of XHTML-conforming legal instruments. It is presently a *prospective* grammar because it is based upon the XHTML 2.0 Working Draft dated 22 July 2004.

This document contains

- (a) One Modular XHTML specification for several new XML elements to be used in Legal XHTML documents
- (b) Numerous data models whose vocabulary terms are to be used by XML attributes on elements within a Legal XHTML document or as the names of XML elements within associated RDF documents
- (c) Numerous samples of Legal XHTML documents and RDF documents
- (d) A specification for standardized Legal XHTML formats and for a standardized transformation between these formats
- (e) A specification for a set of standardized resources that may be referenced by Legal XHTML or associated RDF documents
- (f) Information about associating electronic signatures with Legal XHTML documents
- (g) Information about using XForms, XEvents, and XML Topic Maps with Legal XHTML documents

Copyright, 2005. Hypergrove Engineering, Inc. All rights reserved.

# Table of Contents

1. [Introduction](#)
    - 1.1 [Towards A Successful Standard](#)
    - 1.2 [Overview of the Standard](#)
    - 1.3 [History of the Standard](#)
  2. [Background Information](#)
    - 2.1 [Essential XHTML](#)
    - 2.2 [Essential Dublin Core](#)
    - 2.3 [Essential RDF](#)
  3. [Legal XHTML Design](#)
    - 3.1 [Glossary of Terms](#)
    - 3.2 [Design Principles](#)
    - 3.3 [Design Overview](#)
    - 3.4 [Instrument Administration](#)
    - 3.5 [Instrument Arbitration](#)
    - 3.6 [Instrument Captioning](#)
    - 3.7 [Instrument Endorsements](#)
    - 3.8 [Instrument Events](#)
    - 3.9 [Instrument Formatting](#)
    - 3.10 [Instrument Links](#)
    - 3.11 [Instrument Origination](#)
    - 3.12 [Instrument Scripts](#)
    - 3.13 [Instrument Validation](#)
    - 3.14 [Internationalization](#)
    - 3.15 [Versioning](#)
  4. [XHTML Guidelines](#)
    - 4.1 [Legal Module](#)
    - 4.2 [Document Module](#)
    - 4.3 [Structural Module](#)
    - 4.4 [Text Module](#)
    - 4.5 [Hypertext Module](#)
    - 4.6 [List Module](#)
    - 4.7 [Core Attributes Module](#)
    - 4.8 [Hypertext Attributes Module](#)
    - 4.9 [I18N Attributes Module](#)
    - 4.10 [Bi-directional Text Attributes Module](#)
    - 4.11 [Edit Attributes Module](#)
    - 4.12 [Embedding Attributes Module](#)
    - 4.13 [Image Map Attributes Module](#)
    - 4.14 [Metadata Attributes Module](#)
    - 4.15 [Metadata Module](#)
  4. [XHTML Guidelines, Continued](#)
    - 4.16 [Object Module](#)
    - 4.17 [Ruby Module](#)
    - 4.18 [Scripting Module](#)
    - 4.19 [Style Attribute Module](#)
    - 4.20 [Stylesheet Module](#)
    - 4.21 [Tables Module](#)
    - 4.22 [XForms Core Module](#)
    - 4.23 [XForms Controls Module](#)
    - 4.24 [XEvents Module](#)
  5. [Legal XHTML Vocabulary](#)
    - 5.1 [Legal Instrument Vocabulary](#)
    - 5.2 [Contact Vocabulary](#)
    - 5.3 [Location Vocabulary](#)
    - 5.4 [Object Vocabulary](#)
    - 5.5 [Property Vocabulary](#)
    - 5.6 [Event Vocabulary](#)
    - 5.7 [Text Attributes Vocabulary](#)
  6. [RDF Guidelines](#)
    - 6.1 [Striped RDF Syntax](#)
    - 6.2 [Origination Record](#)
    - 6.3 [Administration Record](#)
    - 6.4 [Arbitration Record](#)
    - 6.5 [Validation Record](#)
  7. [RDF Instances](#)
    - 7.1 [Legal XHTML Topic Maps](#)
    - 7.2 [Legal XHTML Calendar](#)
    - 7.3 [Legal XHTML Atlas](#)
- Appendix A. [Legal XHTML Module Definition](#)
- Appendix B. [Legal XHTML RDF Schema Definition](#)
- Appendix C. [Legal XHTML Stylesheets](#)
- C-1 [Normative Presentation Stylesheet](#)
  - C-2 [Normative Transform Stylesheet](#)
  - C-3 [Normative Instrument Validation Stylesheet](#)
  - C-4 [Normative Presentation Validation Stylesheet](#)
  - C-5 [RDF Extraction Stylesheet](#)
  - C-6 [DC Extraction Stylesheet](#)
  - C-7 [Legal XHTML Extraction Stylesheet](#)
  - C-8 [Table of Contents Stylesheet](#)

---

# 1. Introduction

Working Draft

Perhaps *one billion* legal instruments are signed each year. It is vitally important to establish technical protocols and standard practices which allow these legal and economic transactions to be conducted efficiently and imaginatively over the Internet.

This document describes an XML protocol, or grammar, that represents both the information within a legal instrument and information about a legal instrument. Legal XHTML is a grammar that encodes legal instruments which may then be offered to a party for his or her digitally-applied signature, together with associated electronic documents that are used to further process the legal instrument by its parties.

Today, the most widely used grammar on the Internet is the Hypertext Markup Language (HTML), a grammar unquestionably being supplanted by its XML-conforming cousin, XHTML. The recent release of XHTML, Version 2.0 (XHTML2), clearly envisions a near universal adoption of the Resource Description Framework (RDF) as the means for conveying structured information about the relatively unstructured, flowed content of XHTML2 documents. Of the grammars that conform to the Resource Description Framework, the Dublin Core elements have proven themselves the most enduring and most widely used.

Legal XHTML weaves these three building blocks - XHTML2, RDF, and Dublin Core - into a straightforward grammar which preserves for legal instruments the same flexibility familiar to authors who today publish non-legal HTML documents on the Internet. The result is that Legal XHTML encourages the development of sophisticated business practices regarding the content, exchange, and use of legal instruments by its parties and other interested individuals.

## 1.1 Towards A Successful Standard

A 'standard' XML grammar for legal instruments can be created in one of two ways. First, a focused, custom grammar can be crafted from scratch to represent both information within an instrument plus any information about its associated workflows. Second, a grammar can be created that builds upon *prior art*, that is, upon existing standards that have already achieved a level of acceptance and use on the Web. Certainly, key to this second approach is the correct selection of the "prior work(s) of art" upon which the grammar for legal instruments is to be based.

The Legal XHTML grammar has three foundation specifications: XHTML 2.0, the Resource Description Framework, and the Dublin Core. These prior works of art were selected, rather than crafting a completely customized grammar, because

- a higher quality grammar is likelier since prior art, already validated, is leveraged
- the grammar is completed faster since no function provided by prior art is rebuilt
- less public education is required since prior art is, by definition, already in use
- less public marketing is required to surpass some threshold of use of the new grammar
- faster and greater tool support is likelier since tools already exist for prior art

It is worthwhile noting that a grammar for legal instruments could not be built upon the XHTML 1.1 specification from the W3C; only with the improvements provided by XHTML 2.0 could a useful grammar for legal instruments be created without resorting to customization.

Legal XHTML not only adds several new elements to the XHTML element set, but also provides specific guidelines for authors of legal instruments. These guidelines -- each technically necessary -- reduce coding alternatives available to an author, while increasing the usefulness and accessibility of the information within the instrument. Guidelines such as these, which create a more valuable end-product, will likely contribute substantially towards making Legal XHTML a **successful standard** within its community of users.

## 1.2 Overview of the Standard

Legal XHTML is a standard governing the exchange of legal documents over the Web between arms-length parties. These documents are exchanged using a variety of methods, one as simple as the "Save-as" command on a browser menu. Yet this simple method actually is the least controllable, because the web-page can contain more than one legal instrument embedded within application-oriented markup. For this reason, a `<legal:instrument>` element is required.

In fact, an XHTML standard needs to accommodate application-oriented material within a `<legal:instrument>` element because the elements used to markup the application material are the same as those used to markup the legal material within the instrument, and developers cannot and should not be constrained from including markup within the instrument that aids the user interface.

Other documents, both presentation and meta-information documents, can be related to the `<legal:instrument>` element by using the

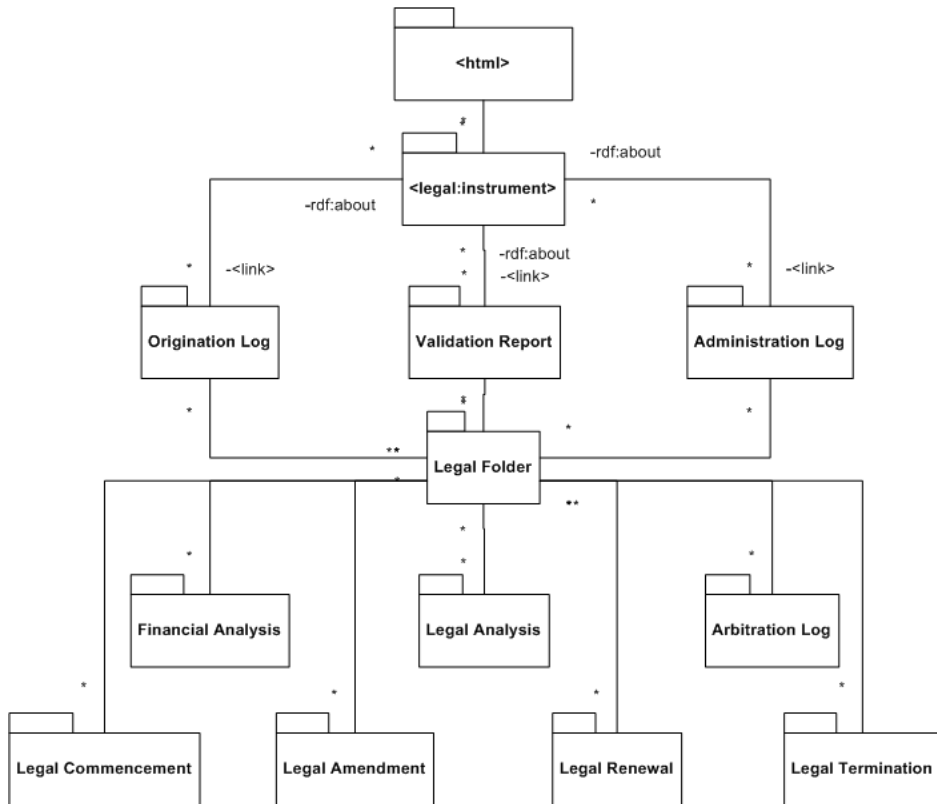
`<html:link>` element. These documents can be functional documents, such as amendments and renewals, or workflow documents pertaining to a process during the functional life of the legal instrument. In the diagram, only three documents are shown related to the Legal XHTML document, with others related indirectly; actually, all these documents may be conveniently linked within the XHTML document using the `<html:link>` element, to the extent they are known at the time the legal instrument becomes a static entity. The salient item to note is that the instrument resides in a functional **Folder** which contains all pertinent presentation and meta-information documents. Every **Folder** is normally typed to indicate what kind of information it holds. For instance, for a legal agreement between parties, the folder is representative of, and typed as, a **Contract** between the parties. For legislative documents, the folder is typed as a **Law**, while for judicial documents, the folder is typed as a **Case**. At a minimum, a **Folder** is typed as a **Legal Matter**.

Meta-information documents are encoded using the Resource Description Framework (RDF) and, therefore, are related to their source document through an *rdf:about* attribute that names the presentation document it concerns. However, it is

equally possible for an RDF document to be about another RDF document; for instance, a financial analysis can be about an entire legal matter or folder as it encompasses one or more presentation documents as easily as it can be about a specific document within the legal folder.

A **Folder** that is, a **Legal Matter**, therefore binds together a master legal instrument with its amendment documents, commencement document(s), renewal documents, arbitration documents, assignment documents, and termination document, each of which are instruments themselves. Other documents referenced by the presentation and meta-information documents may also be contained within a Legal Folder, including not only transient draft versions of a legal instrument, but also legal correspondence (requests, demands, denials) and other types of instruments involved with the legal matter.

The question of which party to the legal instrument creates the legal folder in which it is nominally stored, is determined by agreement among the parties. If multiple parties each maintains a folder, then one party's folder can certainly be *about* another party's folder.

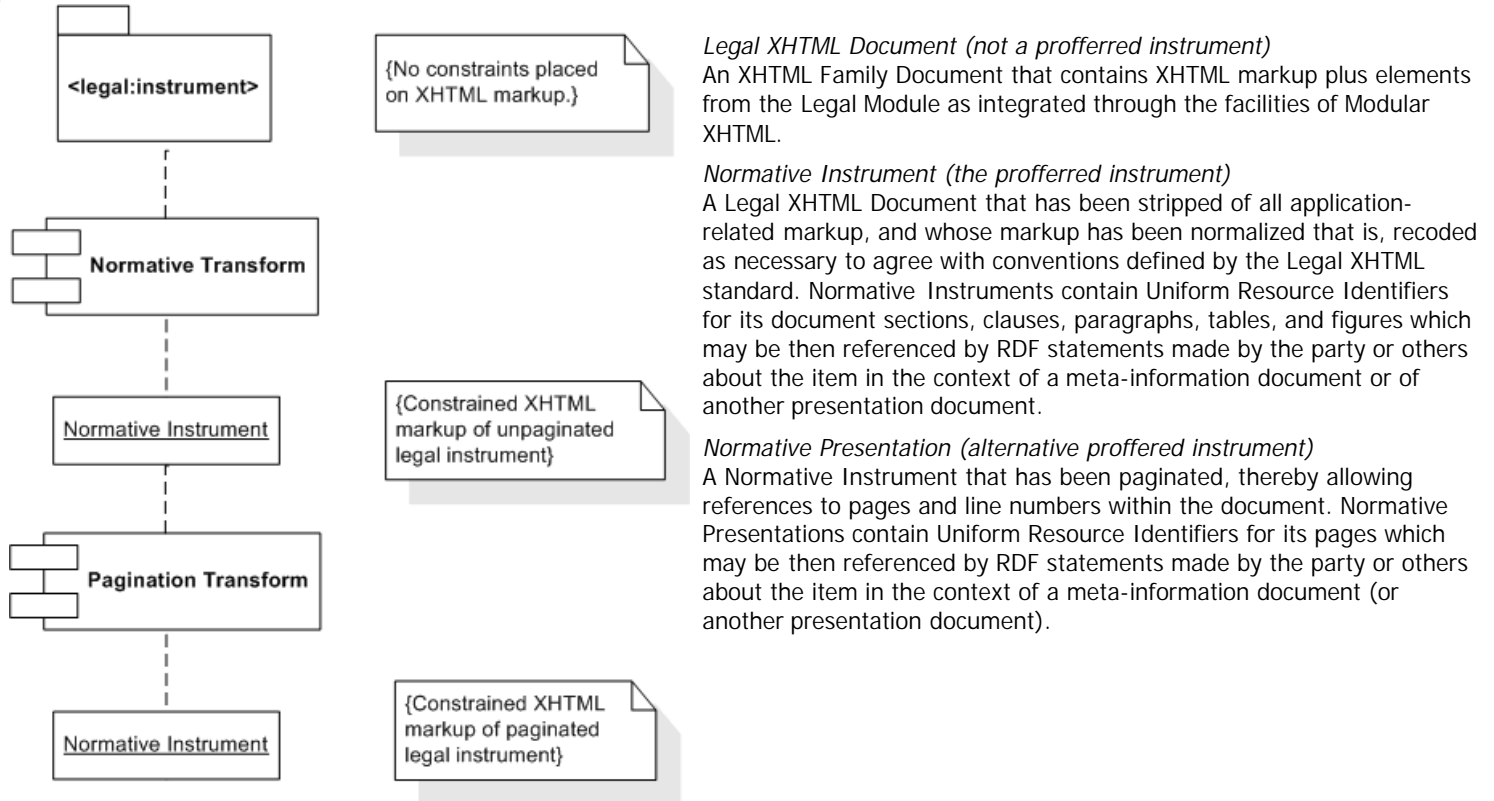


## Formats of XHTML-encoded Instruments

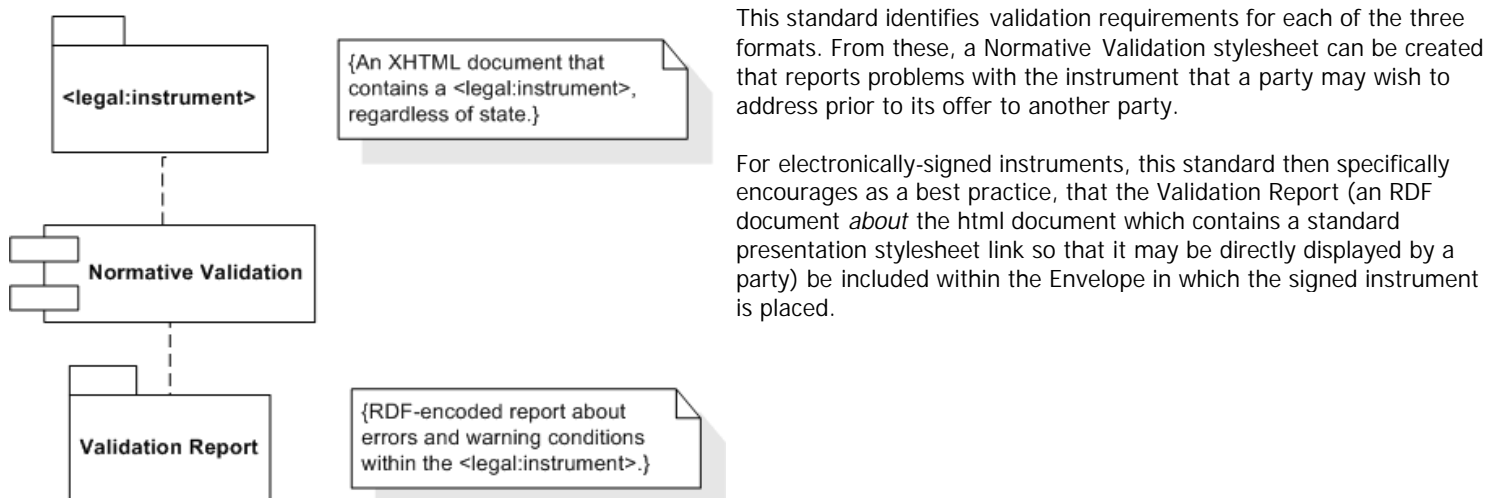
Working Draft

When a legal instrument is signed by a party, it becomes a static unchanging entity. It is for the *proffered document* that the guidelines promoted by Legal XHTML actually apply. Since Legal XHTML's primary functional objective during interchange is to enable an offeree to better determine whether the instrument is to be accepted or not, the format of the proffered instrument needs to be relatively predictable for offeree's software to operate successfully. For electronically-signed instruments, this is the same format that is placed into a digital envelope representing a legal instrument signed by a party.

This standard therefore addresses three forms for the XHTML that is used to represent a legal instrument.



Each of these formats are important because Legal XHTML implicitly identifies the requirements for transformations of the `<legal:instrument>` element(s) found within the XHTML document to a form compatible with Legal XHTML guidelines. Nevertheless, a producer of a proffered instrument has every opportunity to directly create a Normative Instrument or a Normative Presentation, without performing an explicit transformation.



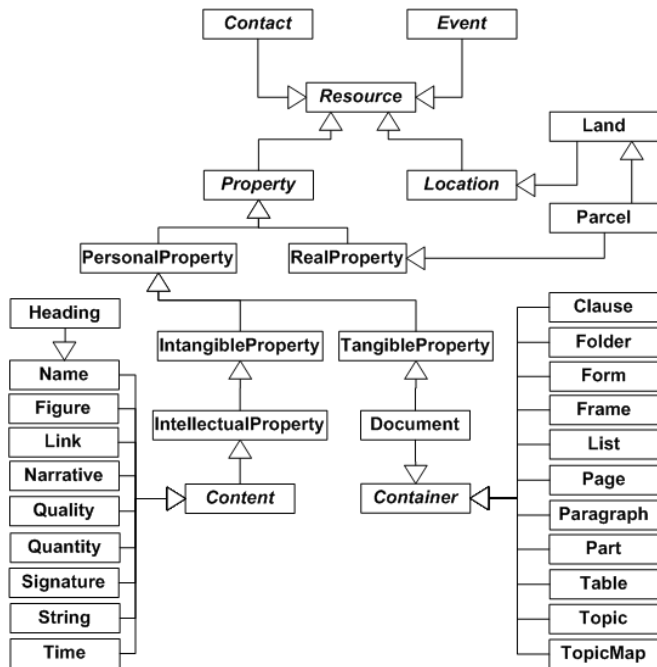


Meta-information *about* a legal instrument is encoded in either of two locations. It can be encoded in its own RDF document, or it can be encoded within an XHTML document. This powerful feature introduced by XHTML2 undergirds the Semantic Web.

Meta-information that is within the content of a document, such as its title, can be so identified through the attributes on an element containing that text. XHTML2 provides three attributes for all its elements which are used to identify the name of the property being provided by the content of the element; the RDF resource to which the property applies; and the RDF resource if any that is represented by the content of the element.

In contrast to an XML Schema which defines *XML elements and attributes*, a Resource Description Framework schema defines *information classes and their properties*. The names of these properties can be used in two contexts: as the names of XML elements within an RDF document and as the names of properties that are placed on XHTML2 or Legal XHTML elements. In further contrast to XML Schemas, the properties of classes defined in an RDF schema may be inherited by other classes, while a given class may inherit the properties of one or more classes. Lastly, a property for a class may be subtyped.

Legal XHTML standardizes three sets of RDF classes. First, Legal XHTML defines classes to represent information in a document, so that its properties can be so attached its text and image content. Second, Legal XHTML defines classes related to the information narrated by the content of the document. In this regard, a simple and intuitive approach is taken: classes are defined to represent "people, places, and things"; these are, like a document, considered to be "resources" within the Resource Description Framework. Third, Legal XHTML defines classes to represent document content about physical and virtual characteristics of resources, for concepts such as [Height](#), [Value](#), and [Date](#). Legal XHTML also establishes the [Event](#) class for actions which have the resource as its subject, such as the [Publication](#) event.



In the diagram, the basic Legal XHTML class structure contains classes for "people" (see the [Contact](#) class); "places" (see the [Location](#) class); and "things" (see the [Property](#) class). Further, the [Resource](#) class is the most basic class defined, that is, all other classes are its subclasses.

The diagram also identifies two fundamental classes for documents. The [Content](#) class represents text and image content within a document; for example the [Name](#) subclass represents text that is a name for a resource that can be found within the text of a document. Secondly, the [Container](#) class holds instances of the [Content](#) class, and which may be captioned and sub-containers for other content. It can be noted that subclasses of the [Container](#) class correlate with the XHTML2 block-style elements while XHTML2 inline-style elements correspond to subclasses of the [Content](#) class.

Legal XHTML can be said to be a property-centric model consistent with standard legal principles. All document content is designated to be [IntellectualProperty](#) while the document itself is representative of [TangibleProperty](#).

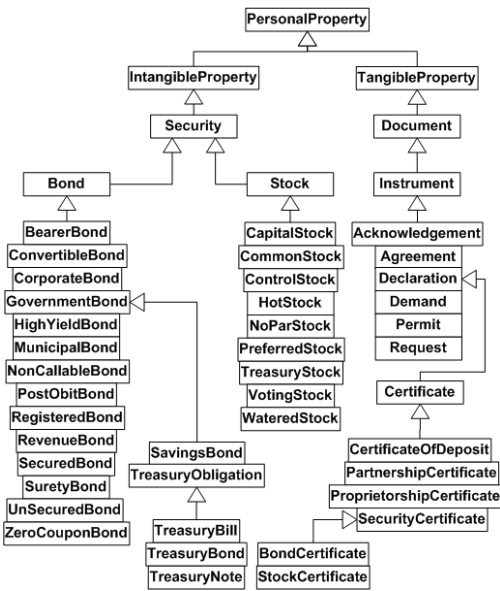
The diagram does not show the properties (or the operations) applicable to the each of the classes. In this regard, within the Resource Description Framework one defines the properties of a class, either text properties or object properties. Legal XHTML defines text properties that coincide with the [Consolidated Listing of ISO 639, ISO 4217, SI Measurement Symbols, and World Time Zones](#) and the datatypes defined by the W3C's XML Schema. For object properties, Legal XHTML generally combines the name of the attributed class with the name of the attributing class, such as [DocumentPage](#), a property applicable to instances of a [Document](#).

Legal XHTML defines a set of RDF predicates derived from the statements that an "object *has* a property" and an "object *is of a type*". Legal XHTML predicates include `<has>`, `<had>`, `<willHave>`, `<mayNotHave>`, and others. For providing attributes on incidences of types for an object, Legal XHTML establishes the `<isA>` and `<isNotA>`, among other, tags.

## Security Instruments

Working Draft

Legal XHTML preserves the distinction between the intangible property represented by an instrument, and the instrument itself. In the diagram, [Stock](#) is shown as a type of [IntangibleProperty](#), while the certificate that attests to the property is shown as a type of [Instrument](#). In the example encoding for an instance of a [StockCertificate](#) that represents 1,500 shares of common voting stock, each share of stock that is being certified as legally registered is identified by two `<rdf:type>` elements, its issuer, issuee, and date of registration.

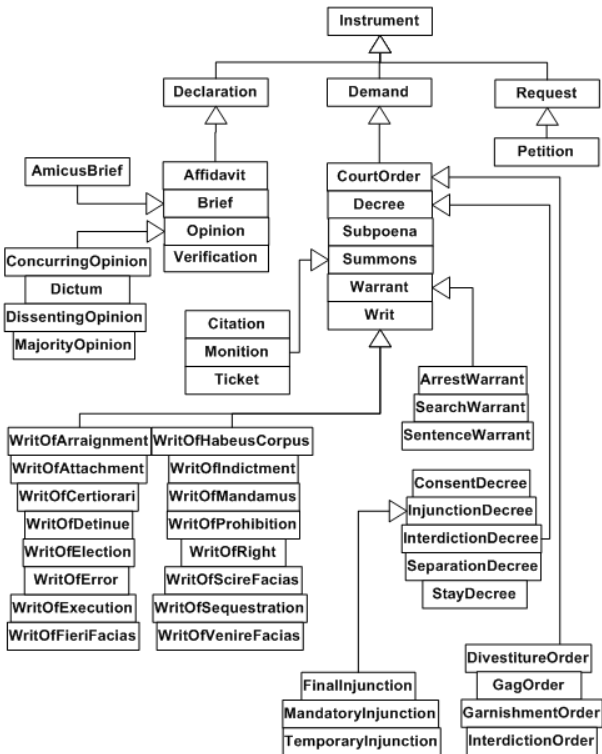


```

<!-- legal xhtml encoding -->
<legal:instrument resource='../AstockCertificate.html'>
  <p>
    <span property='InstrumentDate'>12 November 2004</span>
    This certifies that
    <span property='CertifiedOwner' resource='../issuee.rdf'>
      issuee name</span> is the registered owner of
    <span property='CertifiedShares'>1500</span>
    <span property='StockType' resource='../vocab.owl#CommonStock'>
      common shares</span> of stock...
  </p>
  ...
</legal:instrument>

<!-- Origination Record encoding -->
<StockCertificate rdf:about='../AstockCertificate.html'>
  <has>
    <InstrumentDate text='12 November 2004' />
    <CertifiedOwner rdf:about='../issuee.rdf' text='issuee name' />
    <CertifiedShares text='1500' />
    <CertifiedStock rdf:about='../share01.rdf'>
      <rdf:type rdf:resource='../vocab.owl#CommonStock' />
      <rdf:type rdf:resource='../vocab.owl#VotingStock' />
      <has>
        <PropertyCertificate rdf:about='../AstockCertificate.html' />
        <PropertyOwner rdf:about='../issuee.rdf' />
        <RegistrationDate rdf:about='../calendar.rdf#20041112' />
        <StockIssuer rdf:about='../issuer.rdf' />
      </has>
    </CertifiedStock>
  </has>
  <!-- 1,499 more CertifiedStock elements follow -->
  <StockType rdf:about='vocab.owl#CommonStock' text='common shares' />
</StockCertificate>
  
```

## Judicial Instruments

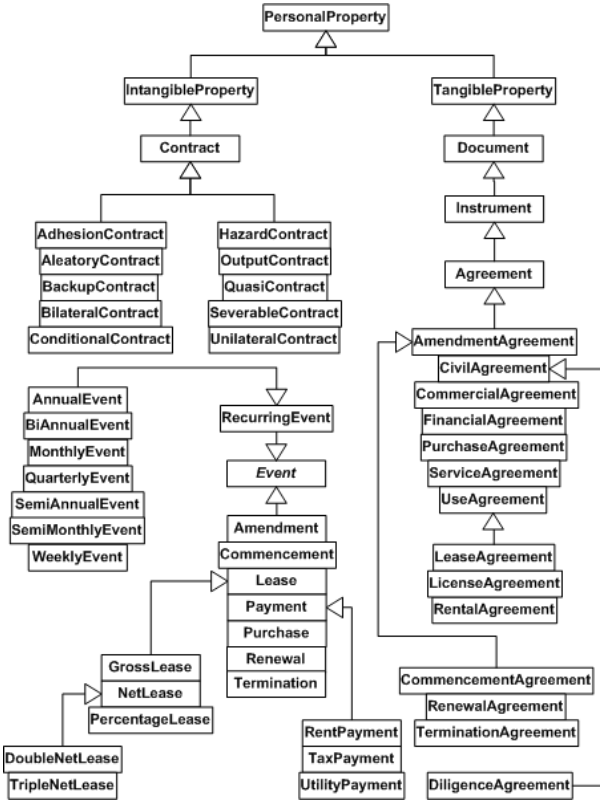


Legal XHTML provides support for instruments issued by a court or officer of the court. Because Legal XHTML creates subclasses of [Instrument](#) that are oriented towards associated workflows, judicial instruments are typed as either a [Declaration](#), [Demand](#), or [Request](#).

## Contract Instruments

Working Draft

Legal XHTML preserves the distinction between the intangible property represented by a contractual instrument, the instrument itself, and the future event(s) contemplated by a contract. In the diagram, a [Contract](#) is shown as a type of [IntangibleProperty](#), while the document itself is shown as a type of [Agreement](#). In the Origination Record example below, encoding for an instance of a [LeaseAgreement](#) covering a location renting for \$950/month, the [Agreement](#) element contains a [Contract](#) element that is typed as a [BilateralContract](#). This element contains a [ContractConsideration](#) and [ContractGoods](#) element, both of which represent an [Event](#). The [ContractGoods](#) element contains a [Lease](#) element. The [ContractConsideration](#) element contains a [Payment](#) element, which can be noted is a recurring, monthly, [RentPayment](#) consisting of two components, [BaseRent](#) and an [AdministrativeFee](#).



```

<!-- legal xhtml encoding -->
<legal:instrument resource='.../anAgreement.html'>
  <link property='OriginationRecord' href='.../anAgreement.rdf'>
  <p>
    <span property='InstrumentDate'>12 November 2004</span> This
    <span property='AgreementType' resource='.../vocab.owl#Lease'>
      lease</span> between
    <span property='InstrumentParty' resource='.../party1.rdf'>
      party #1 name</span> and
    <span property='InstrumentParty' resource='.../party2.rdf'>
      party #2 name</span> for the premises located at
    <span property='SubjectLocation' resource='.../location1.rdf'>
      123 Maryland Avenue NW, Washington DC</span> ...
  </p>
  ...
</legal:instrument>

<!-- Origination Record encoding -->
<Agreement rdf:id='.../anAgreement.rdf' rdf:about='.../anAgreement.html'>
  <rdf:type rdf:resource='.../vocab.owl#LeaseAgreement' />
  <has>
    <AgreementType rdf:about='.../vocab.owl#Lease' text='lease' />
    <EffectiveDate text='01 January 2005' />
    <ExpirationDate text='31 December 2005' />
    <InstrumentDate text='12 November 2004' />
    <LegalLessee rdf:about='.../party1.rdf' text='party1 name' />
    <LegalLessor rdf:about='.../party2.rdf' text='party2 name' />
    <SubjectLocation rdf:about='.../location1.rdf' text='...' />
    <MasterContract rdf:about='.../aContract.rdf'>
      <rdf:type rdf:resource='.../vocab.owl#BilateralContract' />
      <has>
        <MasterAgreement rdf:about='.../anAgreement.html' />
        <EffectivePeriod iso='01012005/PIY' />
        <ContractConsideration>
          <rdf:type rdf:resource='.../vocab.owl#RentPayment' />
          <rdf:type rdf:resource='.../vocab.owl#MonthlyEvent' />
          <has>
            <BaseRent usd='950.00' />
            <AdministrativeFee usd='25.00' />
            <PaymentAmount usd='975.00' />
            <PaymentAddress rdf:about='.../1500KStWDC.rdf' />
            <PrimaryPayee rdf:about='.../party1.rdf' />
          </has>
        </ContractConsideration>
        <ContractGoods rdf:about='.../anAnnualLease.rdf'>
          <rdf:type rdf:resource='.../vocab.owl#TripleNetLease' />
          <has>
            <LeaseContract rdf:about='.../aContract.rdf' />
            <LeaseDate rdf:about='.../calendar.rdf#20050101' />
            <PrimaryLandlord rdf:about='.../party1.rdf' />
            <PrimaryTenant rdf:about='.../party2.rdf' />
            <SubjectLocation rdf:about='.../location1.rdf' />
          </has>
        </ContractGoods>
      </has>
    </MasterContract>
  </has>
</Agreement>
  
```

Legal XHTML defines two other important elements that are found within a [Contract](#); each representative of an [Event](#). A [ContractObligation](#) element contains an event that one party or the other is obligated to perform during the course of the contract, excluding those involved with the subject or consideration of the contract. Secondly, the [ContractContingency](#) element contains an event that, should it occur, has additional contractual obligations that may be identified.

---

## Instrument Topic Maps

Any container of content is about one or more subjects, or topics. Legal XHTML is designed to allow an author to identify the topic(s) that a clause for instance, is about, either by referencing an entry in a set of pre-established set of topics or by providing the text for the name of a topic.

## 1.3 History

*Draft One (February 28, 2005)*

Contains (a) initial set of RDF data models (b) initial set of XHTML module specifications (c) initial background and design requirements.

### To-do

#### Narrative Material

Review/rewrite portions of Background section  
Design section review/rewrite/complete  
UML graphics about standard's parts' relationships  
RDF coding guidelines  
RDF instance requirements  
Identify issues for discussion

#### XHTML Material

Stubbed sections  
Samples for 5.8 - 5.14, 5.21.  
XForm and XEvent Module specifications and samples  
Legal Module definition in all 3 syntaxes (DTD, XMLS, RELAX)

#### RDF Material

Contact Data Models - data properties for roles.  
Location Data Models - Land, buildings, rooms, and so on.  
Property Data Models - Real, personal, and intellectual property.  
Foundation Data Models - Attribute classes (Distance, Value, Area, String, etc)  
Atomic Properties - lower case properties (eg text, ft2, m2, usd, and en)  
RDF Schema definition for all items above  
Complete RDF and XHTML samples for all data properties  
Hyperlinks to section top and to vocabulary listing  
Create instances for locations, laws, topic maps, and calendar

#### Stylesheet Material

Create normative CSS stylesheets  
Define transformation requirements

---

## 2. Background Information

**Working Draft**

## 2.1 Essential XHTML

XHTML Version 1.1, as an XML-compatible version of the HTML 4.1 language, is exceptionally *not* reliable as a data source from which information may be extracted, analyzed, and reused by XML software; this weakness of XHTML Version 1.1 seems to have been a primary issue addressed by XHTML Version 2.

This section only highlights certain strengths of XHTML2 relevant to a grammar for legal instruments; it is not an exhaustive analysis of the differences between XHTML1 and XHTML2, and some familiarity with XHTML 1.1 is assumed.

---

### 2.1.1 XHTML2 Document Model

XHTML2 bases its document model on a *grammatical paragraph* : whenever a paragraph is referenced or retrieved, all its content can be retrieved, including its captioning and any lists that are logically part of the paragraph. In this context, XHTML2 introduces a new element, the [section](#) element, and establishes a rigorous yet flexible element hierarchy:

- [section](#) elements may be nested, that is, may contain sub-sections
  - [section](#) elements may contain [p](#) elements
  - [p](#) elements may not be nested, that is, may not contain sub-paragraphs
  - [p](#) elements may not contain higher levels, that is, [div](#) or [section](#) elements
  - [div](#) elements may be nested, and may contain [section](#) or [p](#) elements
- 

### 2.1.2 XHTML2 Document Metadata

XHTML2 has good support for metadata embedded within an XHTML2 document and for external metadata descriptions of (elements located within) an XHTML2 document. In particular, XHTML2 introduces new facilities drawn from the Resource Description Framework (RDF):

- the [resource](#) attribute associates an rdf:ID to an XHTML element
- the [about](#) attribute associates an rdf:about to an XHTML element
- the [property](#) attribute names the RDF property represented by an XHTML element
- the [content](#) attribute mimics the rdf:text attribute
- the [link](#) element identifies RDF documents which are descriptive about an HTML element

XHTML2 also allows several metadata-specific elements -- the [link](#), [meta](#), and [script](#) elements -- to be used by practically all XHTML2 elements.

### 2.1.3 XHTML2 Attributes

XHTML2 defines few element-specific attributes; the great majority of the attributes it defines are part of a Common attribute set applicable to practically every XHTML2 element.

#### Global Attributes

```
<XHTML2_ELEMENTNAME
about='some_uri'      <!-- resource the property attribute applies to -->
access='some_uri'    <!-- navigational aid to href resource within target -->
cite='some_uri'      <!-- source document from which came text content -->
class='name name'    <!-- multiple presentation class names -->
content='text string' <!-- text content for property value, for empty elements -->
datatype='name'      <!-- name of datatype applicable to property value -->
datetime='yyyy/mm/dd' <!-- ISO formatted datetime for last editing action -->
dir='name'           <!-- reserved name for writing direction -->
edit='name'          <!-- reserved name for last editing action -->
href='some_uri'      <!-- resource to hyperlink to, when selected by a user -->
hreflang='code'      <!-- ISO language code pertinent to href resource -->
hreftype='name'      <!-- mimetype pertinent to href resource -->
nextfocus='an_id'   <!-- id of element to tab to -->
prevfocus='an_id'   <!-- id of element to backtab to -->
property='name'      <!-- name of property represented by element content -->
rel='name'           <!-- relation between about and resource resources -->
resource='some_uri'  <!-- resource the element defines or references -->
rev='name'           <!-- relation between instrument and resource_resource -->
src='some_uri'       <!-- resource to embed in a document -->
style='text string'  <!-- presentation-related directives for the element -->
target='some_uri'    <!-- environment to establish for src resource -->
title='text string'  <!-- a title that is often shown as a tool-tip -->
type='mime_type'     <!-- mimetype applies to src resource -->
xml:base='some_uri'  <!-- base URI for relative href URIs -->
xml:id='an_id'       <!-- unique id of object within document -->
xml:lang='code'      <!-- ISO language applicable to text content -->
>
```

### 2.1.4 XHTML2 Caption Support

XHTML2 introduces a generic element for a heading, the `h` element, while preserving the ranked heading elements, that is, the `h1` - `h6` elements. Unlike XHTML1 however any of these heading elements are to be embedded *within* the `section`, `div`, the `blockquote`, `blockquote`, the `li`, `dd`, or `object` element to which it applies.

For table captioning, XHTML2 introduces the `caption` element, while for each of the four types of lists, XHTML2 'reuses' XHTML1's `label` element for captioning the list. [Note: XHTML2 introduces a new type of list used for navigation purposes, the `ul` element.]

### 2.1.5 XHTML2 Modularization

XHTML2 packages its elements and attributes into modules that to a certain degree can be optionally selected to be included within conforming, custom document types. XHTML 1.1 elements for user interface controls (such as entry fields) and XHTML 1.1 attributes for DOM event handlers (such as onclick), are no longer packaged within XHTML2 modules. Instead, custom document types optionally may include the several modules defined by the `XForms` and `XEvents` specifications, respectively.

XHTML2 also provides an optional `Ruby` module to support notation of element content, normally used in Asian language documents.

Finally, it is noted that XHTML2 defines its modules using three schema languages: standard XML doctypes; XML Schema; and Relax NG. Legal XHTML *must* define its Legal Module using these languages.

## 2.2 Essential Dublin Core

The most widely used grammar for conveying metadata about a physical resource or about a resource on the Internet, appears to be the set of fifteen properties published by the Dublin Core Metadata Initiative (DCMI). These properties have been published as ISO 15836.

[Note: ISO 15836 publishes capitalized names for the properties while the DCMI uses lower-case names; this fact is exploited [later](#) in this document.]

Each (lower-case) Dublin Core property relates to a text string. If the string is not in the body of the legal instrument, the metadata can be transmitted by a [meta](#) element (that is required to be located within an [instrument](#) element). If the string does exist within the body of the legal instrument, such as the instrument's title, then the [property](#) attribute can merely be specified on an encapsulating [span](#) element for that string. The fifteen (lower-case) Dublin Core elements are specialized for legal instruments as follows.

### [dc:contributor](#)

A person, organization, or service responsible for making contributions to the content of the legal instrument. "Contributions" include authoring or editing one or more clauses, paragraphs, or attachments to the instrument. Contributors may not include any "Creators" of the instrument.

### [dc:coverage](#)

Reference to the jurisdiction to which the legal instrument is subject.

### [dc:creator](#)

A person, organization, or service responsible for making the content of the legal instrument. The creators for a legal instrument are its signatories, for *without* their endorsements the instrument would not legally exist. Signatories include all pertinent parties (and their agents) to the instrument.

### [dc:date](#)

The date, in standard ISO 8601 format, that is the arbitrary 'reference date' applicable to the legal instrument, if one exists. For instruments without a reference date, the date the instrument was fully executed and applicable should be used.

### [dc:description](#)

A brief account of the content of a legal instrument.

### [dc:format](#)

The device for which the legal instrument has been formatted. *URI resource definitions are forthcoming.*

### [dc:identifier](#)

The Uniform Resource Identifier (URI) for the legal instrument is recorded in this element.

### [dc:language](#)

An ISO-639 code indicating the principal language used by the legal instrument.

### [dc:publisher](#)

A person, organization, or service responsible for making the legal instrument available. A publisher for a legal instrument is involved with its formatting, storage, and publication.

### [dc:relation](#)

Reference to a resource to which the legal instrument is related. Formal citations (that is, the value of their [dc:bibliographicCitation](#) metadata) should be the value of this element. Examples of related resources include every entry in a Table of Authorities; and each associated but unattached legal or technical.

### [dc:rights](#)

Text statement concerning rights held over a form used to create the legal instrument.

### [dc:source](#)

Reference to a resource from which the legal instrument is derived. If the legal instrument is created from a legal form of some type, then its formal citation is recorded.

### [dc:subject](#)

An identifier of the subject matter for the legal instrument. For a real estate instrument, this element contains the address of the premises. For a court document, the citation to a case is transmitted by this element. For a Commencement Memo or Termination Agreement, a citation to the subject legal instrument is placed here.

### [dc:title](#)

The formal title for the legal instrument, as it may exist within the content of the instrument. This title should be the same as that found in a Table of References or Table of Authorities entry for a citation to the legal instrument.

### [dc:type](#)

One or more keywords identifying the type of or category for legal instrument. *Resource list is forthcoming.*

The DCMI has also published an extended version of its element set that has thirty-six additional elements. These are used as follows.

### [dc:abstract](#)

A summary or short legal analysis of the content of a legal instrument.

**dc:accessRights**

URI and name of person or organization who can access the legal instrument. This should not include anyone specified as a [contributor](#), [creator](#), or [publisher](#) for the legal instrument. *Resource list (for example, to specify "Public") is forthcoming.*

**dc:alternative**

Any form of the title used as a substitute or alternative to the formal title of the legal instrument. Most common is to provide a translated title, specifying a [language](#) attribute.

**dc:audience**

A class of entity for whom the resource is intended or useful. *A controlled vocabulary is forthcoming.*

**dc:available**

Date that the legal instrument during which the instrument is available for retrieval. ISO 861 format should be used.

**dc:bibliographicCitation**

The formal citation to be used for the legal instrument. Its text should be the same as that found in a Table of Authorities entry for a citation to the legal instrument.

**dc:conformsTo**

URI and name of the Legal XHTML namespace.

**dc:created**

Date that the legal instrument was created, not the date the legal instrument was signed by all parties. ISO 861 format should be used.

**dc:dateAccepted**

Date that the legal instrument was signed by all parties. ISO 861 format should be used.

**dc:dateCopyrighted**

For form-based legal instruments, the date that the copyright on the form was effective. ISO 861 format should be used.

**dc:dateSubmitted**

Date that the legal instrument was submitted for signatures by accepting parties. For contractual agreements, this is the date that the contract was offered by one party to another for acceptance. ISO 861 format should be used.

**dc:educationLevel**

No guideline.

**dc:extent**

The number of pages in the legal instrument.

**dc:hasFormat**

URI and name of a presentation form of the legal instrument, for example, a PDF or SVG version.

**dc:hasPart**

URI and name of an attachment to the legal instrument.

**dc:hasVersion**

URI and name of a translated version of the legal instrument. The [language](#) attribute must be specified.

**dc:isFormatOf**

URI and name of an unpaginated version of the legal instrument.

**dc:isPartOf**

URI and name of a legal resource with which the legal instrument is integrally associated. Examples of a 'legal resource' include a judicial or arbitration case, or a legal relationship such as a legal contract. This value is carried in XHTML as a [up](#) type of [link](#) element.

**dc:isReferencedBy**

URI and name of another legal instrument or of a legal resource (for example, a court case) that references this legal instrument.

**dc:isReplacedBy**

URI and name of another legal instrument that supplants, displaces, or supersedes the legal instrument, that is, the next version of the legal instrument. This value is carried in XHTML as a [next](#) type of [link](#) element which is related to the [isPartOfProperty](#) attribute property.

**dc:isRequiredBy**

For legal instruments that are amended instruments, this property records the URI and name of the amending instrument.

**dc:issued**

Date that the legal instrument was issued to its audience, was filed with a public registrar, or was legally served to its parties. ISO 861 format should be used.

**dc:isVersionOf**

URI and name of the original, un-translated version of the legal instrument. The [language](#) attribute must be specified.

**dc:license**

URI and name of a copyright statement associated with the legal form used as a basis for the legal instrument.

**dc:mediator**

URI and name of an arbitrator, mediator, or court to whom are directed grievances concerning the legal instrument.

**dc:medium**

Entry from a controlled vocabulary concerning the "material or physical carrier" applicable to the legal instrument, that is, its paper stock. *This list is forthcoming.*

**dc:modified**

Date that the legal instrument was last modified. ISO 861 format should be used.

**dc:provenance**

A statement about the changes occurring to the content of the legal instrument when it was last modified.

**dc:references**

URI and name of a legal instrument, web resource, or physical resource that is referenced directly by the legal instrument.

**dc:replaces**

URI and name of another legal instrument that supplants, displaces, or supersedes the legal instrument, that is, the amended version of the legal instrument, or a document that is an amendment or renewal. This value is carried in XHTML as a [prev](#) type of [link](#) element which is related to the [isPartOfProperty](#) attribute property.

**dc:requires**

URI and name of an associated but unattached legal instrument or other document or web resource that is required by the legal instrument to support its function, delivery, or coherence of content. For legal instruments that are amendments, this property records the URI and name of the amended instrument.

**dc:rightsHolder**

URI and name of the owner of the copyright to a legal form used as a basis for the legal instrument.

**dc:spatial**

No guideline.

**dc:tableOfContents**

URI and name of each titled section or division within the legal instrument.

**dc:temporal**

No guideline.

**dc:valid**

Date on which, or period during which, the legal instrument is effective. ISO 861 format should be used.

```

<legal:instrument resource='../R12345.html'>
  ...
  <!-- Core set of DC properties -->
  <meta property="DC.description" content="A brief description" xml:lang='EN' />
  <meta property="DC.rights" content="form-relatedstatement" xml:lang='EN' />
  <meta property="DC.title" content="instrument title" xml:lang='EN' />

  <meta property="DC.date" content="2005-12-31"/>
  <meta property="DC.identifier" content="../../../R12345.html"/>
  <meta property="DC.language" content="EN"/>

  <meta property="DC.contributor" content="contributor name" resource='contributor_uri'/>
  <meta property="DC.creator" content="signatory name" resource='signatory_uri'/>
  <meta property="DC.coverage" content="jurisdiction name" resource='jurisdiction_uri'/>
  <meta property="DC.publisher" content="publisher name" resource='publisher_uri'/>
  <meta property="DC.relation" content="citation title" resource='cited_resource_uri'/>
  <meta property="DC.source" content="form title" resource='form_uri'/>
  <meta property="DC.subject" content="property address" resource='property_uri'/>

  <meta property="DC.format" content="Page" resource='../vocab.owl#Page'/>
  <meta property="DC.type" content="Agreement" resource='../vocab.owl#Agreement'/>

  <!-- Extension set of DC properties -->

  <meta property="DC.abstract" content="legal analysis" xml:lang='EN' />
  <meta property="DC.alternate" content="translated title" xml:lang='SP' />
  <meta property="DC.bibliographicCitation" content="citation form for instrument title"/>
  <meta property="DC.extent" content="12 pages"/>
  <meta property="DC.provenance" content="why or what changes made since last version"/>

  <meta property="DC.available" content="2006-01-01/P9Y"/>
  <meta property="DC.created" content="2005-11-21"/>
  <meta property="DC.dateAccepted" content="2005-12-31"/>
  <meta property="DC.dateCopyrighted" content="2002"/>
  <meta property="DC.dateSubmitted" content="2005-12-21"/>
  <meta property="DC.issued" content="2006-01-22"/>
  <meta property="DC.modified" content="2005-12-20"/>
  <meta property="DC.valid" content="2006-01-01/P2Y"/>

  <meta property="DC.accessRights" content="contact name" resource='contact_uri'/>
  <meta property="DC.hasFormat" content="instrument title" resource='pdf_or_svg_uri'/>
  <meta property="DC.hasPart" content="attachment title" resource='attachment_uri'/>
  <meta property="DC.hasVersion" content="instrument title" resource='translation' lang='SP' />
  <meta property="DC.isFormatOf" content="normative title" resource='unpaginated_uri'/>
  <meta property="DC.isPartOf" content="contract or case name" resource='case_uri'/>
  <meta property="DC.isReferencedBy" content="instrument title" resource='referencing_uri'/>
  <meta property="DC.isReplacedBy" content="next version title" resource='next_folder_doc_uri'/>
  <meta property="DC.isRequiredBy" content="amendment title" resource='amendment_uri'/>
  <meta property="DC.isVersionOf" content="instrument title" resource='original_uri' lang='EN' />
  <meta property="DC.license" content="license title" resource='license_uri' lang='EN' />
  <meta property="DC.mediator" content="mediator name" resource='mediator_uri'/>
  <meta property="DC.reference" content="resource title" resource='resource_uri'/>
  <meta property="DC.replaces" content="previous title" resource='next_folder_doc_uri'/>
  <meta property="DC.requires" content="master title" resource='amended_instrument_uri'/>
  <meta property="DC.rightsHolder" content="contact name" resource='contact_uri'/>
  <meta property="DC.tableOfContents" content="component title" resource='component_uri'/>

  <meta property="DC.audience" content="industry name" resource='../vocab.owl#AnIndustryName'/>
  <meta property="DC.medium" content="Legal Paper" resource='../vocab.owl#LegalSizePage'/>

  <meta property="DC.conformsTo" content="--//LEGALXHTML//DTD XHTML LEGAL 1.0 //EN"
    resource="http://www.legalxhtml.org/xmlns/legalxhtml-1"/>

  <meta property="DC.educationLevel" content="" resource='' /> <!-- No guideline -->
  <meta property="DC.spatial" content="" resource='' /> <!-- No guideline -->
  <meta property="DC.temporal" content="" resource='' /> <!-- No guideline -->
  ...
</legal:instrument >

```

## 2.3 Essential RDF

The Resource Description Framework (RDF) is an XML dialect and set of conventions used to encode and to interpret metadata related to a physical resource or a resource on the Internet. RDF is used in either of two modes:

- to make statements defining a resource

In this case, the RDF statements are encapsulated in a [rdf:Description](#) element bearing an [rdf:ID](#) attribute. There should be only one definition of a resource by a given publisher.

- to make statements about a resource

In this case, the RDF statements are encapsulated in a [rdf:Description](#) element bearing an [rdf:about](#) attribute. There can be many statements about a resource by a given publisher, while each of these statements may have its own unique [rdf:ID](#) attribute in addition to the [rdf:about](#) attribute.

Metadata refers to named text strings that represent a value of a property of the resource being described. Every resource may have many properties. A property is merely a characteristic of a resource, as the "height of a person" is a property for any "Person" but is not a property for any "Corporation". RDF thus establishes a simple model for "statements" made for or about a resource: **a resource has many properties**, such as "a person has a height".

RDF represents each statement made for or about a resource using one or two XML elements, depending on whether the property is a literal text string property or a resource property.

### RDF Statements

```
<rdf:Description rdf:about='.../R12345.html'>
  <!-- RDF statement of a text property for the resource -->
  <dc:description>A brief description</dc:description>
  <!-- RDF statement of a resource property for the resource -->
  <has>
    <Contributor rdf:about='contributor_uri' />
  </has>
  <!-- RDF statement about a resource property for the resource -->
  <has>
    <Contributor rdf:about='contributor2 uri'>
      <dc:title>Name for contributor number 2</dc:title>
    </Contributor>
  </has>
  <!-- Multiple resource properties (formal version) -->
  <has>
    <rdf:Bag>
      <rdf:li>
        <Creator rdf:about='creator1 uri'>
          <dc:title>Name for author #1</dc:title>
        </Creator>
      </rdf:li>
      <rdf:li>
        <Creator rdf:about='creator2 uri'>
          <dc:title>Name for author #2</dc:title>
        </Creator>
      </rdf:li>
    </rdf:Bag>
  </has>
  <!-- Multiple resource properties (short version) -->
  <has>
    <Publisher rdf:about='publisher1 uri'>
      <dc:title>Name of publisher #1</dc:title>
    </Publisher>
    <Publisher rdf:about='publisher2 uri'>
      <dc:title>Name of publisher #2</dc:title>
    </Publisher>
  </has>
</rdf:Description>
```

RDF provides a comprehensive method for associating *typing* information to a resource. Using a [resource](#) attribute on a [rdf:type](#) element, an entry in an RDF Schema definition is referenced. One or more types can be associated with a resource, and their order of evaluation is unimportant.

### RDF Typing

```
<rdf:Description rdf:about='.../R12345.html'>
  <rdf:type rdf:resource='.../vocab.owl#Instrument' />
  <rdf:type rdf:resource='.../vocab.owl#Amendment' />
  <!-- RDF statement about a resource property for the resource -->
  <has>
    <Contributor rdf:about='contributor_uri' />
    <rdf:type rdf:resource='.../vocab.owl#Person' />
    <dc:title>Name for the contributor</dc:title>
```

```

    </Contributor>
  </has>
  <!-- Text properties can be represented as an attribute -->
  <has>
    <Contributor rdf:about='contributor_uri' dc:title="Name for the contributor">
      <rdf:type rdf:resource='.../vocab.owl#Person' />
    </Contributor>
  </has>
</rdf:Description>
<!-- This is equivalent to the previous RDF statement -->
<!-- Note that the rdf:Description element is not strictly necessary! -->
<Instrument rdf:about='.../R12345.html'>
  <rdf:type rdf:resource='.../vocab.owl#Amendment' />
  ...
</Instrument>
<!-- This is ALSO equivalent to the previous RDF statement -->
<Amendment rdf:about='.../R12345.html'>
  <rdf:type rdf:resource='.../vocab.owl#Instrument' />
  ...
</Instrument>

```

As can be seen it is trivial to combine elements from multiple XML namespaces in a single RDF description. In the example above, one's RDF Schema would define (1) an "Instrument" class of resource (2) an "Amendment" class of resource (3) a "Contributor" class of resource and (4) a "has" predicate. Definitions for the "dc:description" and "dc:title" properties could be provided by an RDF Schema created for just Dublin Core properties.

#### RDF Skeleton

```

<!-- three XML namespaces are used in this document -->
<rdf:RDF xmlns='legal_xhtml_schema_uri'
  xmlns:dc='dublincore_schema_uri'
  xmlns:rdf='rdf_uri'>
  <Instrument rdf:about='.../R12345.html'>
    <rdf:type rdf:resource='.../vocab.owl#Amendment' />
    ...
  </Instrument>
</rdf:RDF>

```

---

### 3. Legal XHTML Design

**Working Draft**

## 3.1 Glossary

### *Administration Record*

A Resource Description Framework document containing metadata about a Legal XHTML Document, applicable from its original execution until its termination as an effective legal instrument.

### *Document Division*

A part of an instrument that usually implies a page-break when printed, including attachments such as subordinate or related instruments and appendices. Also included are individual pages such as a Cover Page, a Table of Contents page, and a Blank Page. Technically, a Document Division is represented by a [div](#) element or, in the case of an instrument's text body, the [instrument](#) element

### *Identifier Template*

An element structure that specifies the XML content to generate for a caption identifier when a Normative Instrument or Normative Presentation is created.

### *Legal Instrument*

A document that is characterized by material that establishes legal facts; and indications of assent of pertinent parties to their veracity; both as determined by a court of law.

### *Legal XHTML Document*

An XML document instance conforming to the Legal XHTML Document Type.

### *Legal XHTML Document Type*

A document type definition that extends the W3C XHTML 2.0 Document Type through the addition of the Legal XHTML Module. Technically, Legal XHTML is an XHTML Host Language Conforming Document Type.

### *Legal XHTML Module*

A definition of XML elements and attributes specific to legal instruments that are not defined by the XHTML 2.0 specification.

### *Arbitration Record*

A Resource Description Framework document containing metadata about a Legal XHTML Document, reporting a perceived grievance of a party to the legal instrument.

### *Origination Record*

A Resource Description Framework document containing metadata about a Legal XHTML Document, applicable from its original drafting or assembly, through to its signing and placement in a Signed Envelope.

### *Normative Instrument*

An unpaginated Legal XHTML document whose structure and content conforms to certain guidelines established in this document. Normative Instruments can be placed within a Signed Envelope.

### *Normative Presentation*

A paginated Normative Instrument whose presentation varies insubstantially between devices designated for a certain media type. For instance, a Normative Instrument formatted for a Legal Page Media device would have presentation fidelity across devices supporting paper with legal page dimensions. Normative Presentations can be placed within a Signed Envelope.

### *Normative Transform*

A process that creates a Normative Instrument from a Legal XHTML document or some other digital representation.

### *Presentation Transform*

A process that creates a Normative Presentation for a Normative Instrument.

### *Signed Envelope*

The data and element structure standardized by the [Digital Signatures](#) standard.

### *Validation Record*

A Resource Description Framework document containing validation messages describing the content of a Legal XHTML Document, its signature envelope(s), and about metadata associated with the legal instrument.

## 3.2 Design Principles

Working Draft

### Accessibility

Normative Legal XHTML instruments *must* adhere to the design principles within [W3C Web Content Accessibility Guidelines 2.0](#) principally to ensure that individuals with disabilities can equally access these documents. "When these principles are employed, they also make Web content accessible to a variety of Web-enabled devices, such as phones, handheld devices, kiosks, network appliances. By making content accessible to a variety of devices, that content will also be accessible to people in a variety of situations."

### Assumed Standards

Legal XHTML is based on standards expected to predominate among technicians. The great mass of Internet documents are today encoded in HTML and XHTML1.1, and are expected to be encoded in XHTML2 as it is popularized. It is a lowest-denomination choice compared to either Word XML, Office XML, or one's custom markup language, a language that arguably merely would duplicate much Modular XHTML and XHTML2 functionality. Each alternative approach likely will create transforms to and from XHTML2, the W3C standard.

In order to support legal forms and for updated DOM support, the XForms and XEvents standards are followed by Legal XHTML. For metadata representation, the W3C Digital Signatures, Dublin Core, RDF, and RDF Schema standards are used. For conformance to Modular XHTML requirements, the XML Schema, XML Namespaces, and XML Base standards are used. The forthcoming XML ID standard is also assumed. Legal XHTML also adheres to OASIS' [Universal Business Language \(UBL\) Naming and Design Rules v1.0](#).

### Burden

The originator of a legal instrument normally is the creator of the Normative Instrument. A legal contract for example when offered by one party to another is then naturally represented in the form of a Normative Instrument because an offerer will have already signed the document, involving creation of a Normative Instrument as the offer is being prepared.

### Modularity

Technically Legal XHTML conforms to Modular XHTML. Functionally it packages the data within its scope into two document types, one for the flowed text document itself, and one for the metadata that describes the state of the document during a certain period of time.

### Predictability

A predictable element structure is essential when extracting information from an XML document.

### Redundancy

At times standards can specify multiple ways to record a certain piece of information, such as the title of a document. Legal XHTML *shall not* require any one markup conventions over another, but rather all conventions will be simultaneously supported.

### Scope

Legal XHTML is standardizing the XML protocol used to represent the legal document that is exchanged between its parties and others, and the XML protocol for metadata concerned with the instrument as gathered during its drafting, execution, administration, and arbitration by each of the parties.

Technically, restrictions shall not be placed on XML markup external to the legal instrument. A legal instrument can be located anywhere within a larger XHTML file. Accordingly, restrictions on the content model for XHTML files containing legal instruments can only exist at the point that the instrument is placed into a Signed Envelope.

### Stability

The essential nature of a legal instrument is that, once signed, its representation to a viewer *shall not* change. Because of this, Normative Instruments cannot contain executable objects or scripts that are to be evaluated at the time of a viewing of the Normative Instrument. Further, it cannot contain elements such as [address](#) whose content can naturally change over time.

### Singularity

No function overloading onto names.

### Visibility

Normative Legal XHTML instruments *must* contain only visible text and image content and invisible metadata XML attributes. This minimizes or eliminates the chances of legal repudiation of the protocol on the grounds that its electronic content was not predictably visible, that is, no difference may exist between an electronic and a human appraisal of what is "in" the instrument, and what is not.

### 3.3 Design Overview

Legal instruments differ from most other documents on the Internet today by the relatively strict rules that their content and presentation have historically had to follow. Both content and presentation are crucial sets of information: legal documents often refer to content not only by specific clause captions, but also by specific pages within a part of the document, and sometimes even specific lines of text on a page. It would be incomplete to create a protocol for a legal instrument that excludes either its presentation structure or its clause structure.

In short, the Legal XHTML protocol requires that normative legal instruments *must* be paginated for the specific device used to create the "official" version of the instrument. For purely on-line instruments whose printed counterpart will not be legally filed, this can be a single page. This is the [Normative Presentation](#) version of the legal instrument, and its XML can be created directly by an author but more usually is created by an XSL-Transform stylesheet operating upon either a [Normative Instrument](#) version, or a [Legal XHTML document](#) version, of the instrument.

The choice between the two versions depends on the user's application for creating the instrument. One class of applications places the instrument within a larger XHTML-based user interface document. This results in the need to uniquely identify an instrument out of that larger document, should the document be "saved" for further processing by a user, or should the document's URI be retrieved directly by processing software.

### 3.4 Instrument Administration

Basic information required during origination, administration, or arbitration *must* include a chronology of relevant *events* occurring (and not occurring) during the life of a legal instrument, and the *documents* that were exchanged among the parties pursuant to the instrument. This information can be leveraged as the foundation for a contract administration application, for example.

A legal instrument can wholly define a **legal relationship** between the instrument parties, or it can be one of a number of documents that define the legal relationship. Descriptions of the status of the parties' legal relationship is referenced or defined by the metadata records (the Origination, Administration, and Arbitration Records) associated with the instrument.

### 3.5 Instrument Arbitration

**Working Draft**

### 3.6 Instrument Captioning

Automated content generation is provided by [CSS2](#), however [visibility rules](#) dictate that all non-metadata content must be visible in all [Normative Instruments](#). Legal XHTML *must* adopt or build upon the CSS functionality for content generation and *must* have a stylesheet for generating captions and a table of contents from a Legal XHTML document.

The content of a legal instrument often refers to text and image objects that are located elsewhere within the legal instrument. Each of these items have a caption. Per [singularity rules](#), Legal XHTML strives to have one element type defined for each type of captionable item in a document.

- Document pages; document and page divisions; and document figures; all are captioned objects with an optional title and an optional sequence identifier.
- Clauses are captioned objects with an optional title and a required sequence identifier.
- Paragraphs are captioned objects with an optional title and no sequence identifier.
- List items are captioned objects with no title and a required sequence identifier.
- Phrases are objects with no title or sequence identifier.

### 3.7 Instrument Endorsements

Endorsers of a legal instrument include its parties if any and their legal agents, and legal agents of a court. Functionally endorsements are affixed to a legal instrument, and are not an intrinsic part of its flowed text, meaning this standard defers the actual representation of a signature to relevant structures provided by the [Digital Signature](#) standard.

As each endorser signs the instrument, a new signed digital envelope resource is created which contains at least the signed envelope originally received by the requestor plus the [Validation Record](#) retrieved by the requestor. Whenever this newly signed resource is later retrieved by a requestor, an updated [Validation Record](#) is required to re-validate the multiple levels of signed envelopes.

The [Validation Record](#) makes RDF statements about the proper values for the [endorsed](#) attribute on each of the [endorsement](#) elements within the instrument. The validation process will report whether the instrument has a complete set of endorsements or not, that is, whether all [endorsed](#) attributes in the instrument have a value of **endorsed** .

### 3.8 Instrument Events

A Normative Instrument is expected to contain links to its relevant [Origination](#), [Administration](#), and [Validation](#) Records at the moment the legal instrument is first signed and placed into a signed envelope. From the normative instrument, one may then retrieve these metadata records, any of which may contain information about events and event triggers pertinent to the (current status of the) instrument and events; and any of which may contain pointers to sequential versions of the metadata, or versions of the metadata written by different authors.

An **event trigger** is a description of the required, optional, and inapplicable counter-events the legal instrument anticipates to be performed given the occurrence of a one-time, recurring, or non-recurring legal, natural, economic, or document event. Legal XHTML defines a starter set of each of these categories of events. An **event** represents an instance of an occurrence of an event type and it may or may not reference an event trigger as its source.

An Event is contained within a Resource Description Framework document, and is not related to the DOM-related [XEvents Module](#) included in the Legal XHTML document type.

### 3.9 Instrument Formatting

Pagination is a necessary feature for Legal XHTML in order to accommodate references to specific pages of an 'official' instrument and to lines of text on a page within the instrument.

Unpaginated instruments contain a structured hierarchy of document divisions, sections, and subsections. Paginated instruments contain a series of pages from which an unpaginated instrument can be mechanically reconstructed.

For instance, a Certificate of Service is a topic that can be associated with a paragraph of text following one or more paragraphs devoted to party signatures. Pagination of the instrument may place the Certificate of Service on a separate page from a page allocated just to party signatures.

Standardized formatting stylesheets (CSS stylesheets) are necessary for Normative Legal XHTML Presentations so that its default formatting matches default formatting applied to bit-mapped presentation protocols, such as Scalable Vector Graphics. SVG (and PDF) documents for instance do not re-flow content outside the boundaries of the presentation device for which it is formatted, a rule to be followed for Normative Presentations.

### 3.10 Instrument Links

Legal instruments contain not only links to their metadata records, but also citations to other documents; to authoritative clauses or other components within another instrument; and to their own clauses and components.

Links identified within the flowed text of an instrument do not normally contain markup of the text displayed for the link. Rather, structured metadata records provide the opportunity for making further statements about the linked resource by virtue of identifying, for instance, all the Dublin Core data that is applicable to the linked resource.

### 3.11 Instrument Origination

During drafting of a legal instrument, huge volumes of pertinent information not contained within the boundaries of the printed instrument, are exchanged between its parties, pertinent to interpretation of the instrument by parties, arbitrators, and the legal profession.

Standardizing and exchanging this information is a primary goal of Legal XHTML and can include user interface components such as navigation maps to document components, and legal descriptions of each; and comments and replies occurring within a workgroup.

The [Origination Record](#) also can include a structured description of the contents of each clause, and historical documentation about the creation and editing of the clause. Any clause can contain items of information related to one or more topics. The [Origination Record](#) identifies the topics applicable to each component and, to the extent possible, structured data applicable to the topic(s), that is, a 'topical form' that is expressed using XForms and XEvents technologies within the dictionary.

Legal XHTML provides XSL-Transform stylesheets which generate lists of actual and potential topics within an instrument. For instance, if the legal jurisdiction is marked-up within the text of the legal instrument (by setting the [property](#) attribute on a [span](#) element to [GoverningJurisdiction](#) and that property can only be associated with a [Jurisdiction Clause](#), then that document component is reported associated with that topic. In this manner, many-to-many relationships between instrument components (sections, paragraphs, and tables) and instrument topics can be robustly documented.

### 3.12 Instrument Scripts

Executable scripts are necessary for generation of content or of XML markup for a legal instrument. These scripts may be defined only within a Legal XHTML document. Within a Normative Instrument, the element containing the script output (generated text or generated markup) may contain a reference to the script to accommodate re-evaluations of the generated text or markup. That is, the [src](#) attribute on a [script](#) element identifies the source script for the element.

Executable scripts are often necessary for selection of an element for inclusion in a legal instrument. These scripts may be defined only within a Legal XHTML document. Within a Normative Instrument, the element containing the script output (generated text or generated markup) may contain a reference to the script to accommodate re-evaluations of the element's status in the instrument. That is, the [src](#) attribute on a [script](#) element identifies the selection script for the element.

### 3.13 Instrument Validation

Constraints of Modular XHTML and XHTML2 prevent the specification of "Normative Instrument" content models or document type whose business rules for child XHTML elements of an [instrument](#) element varies from those applicable to XHTML elements that are siblings of that element. Because a focused set of different business rules do apply to Normative Instruments (and to Normative Presentations), a complete validation of the instrument cannot be achieved using a Modular XHTML schema or document type: validation therefore requires an XSL-Transform stylesheet that reports errors in the markup for a Normative Instrument (or Normative Presentation).

### 3.14 Internationalization

### 3.15 Versioning

**Working Draft**

---

## References

### ASTER

For information about ASTER, an "Audio System For Technical Readings", consult [T. V. Raman's home page](#).

### CSS1

[CSS, level 1 Recommendation](#), B. Bos, H. Wium Lie, eds., 17 December 1996, revised 11 January 1999. This CSS1 Recommendation is <http://www.w3.org/TR/1999/REC-CSS1-19990111>. [The latest version of CSS1](#) is available at <http://www.w3.org/TR/REC-CSS1>.

### CSS2

[CSS, level 2 Recommendation](#), B. Bos, H. Wium Lie, C. Lilley, and I. Jacobs, eds., 12 May 1998. This CSS2 Recommendation is <http://www.w3.org/TR/1998/REC-CSS2-19980512/>. [The latest version of CSS2](#) is available at <http://www.w3.org/TR/REC-CSS2>.

### HTML4

Dave Raggett, Arnaud Le Hors, Ian Jacobs, Eds., [HTML 4.01 Specification](#), W3C Recommendation. (See <http://www.w3.org/TR/html401/>.)

### IBMJAVA

[IBM Guidelines for Writing Accessible Applications Using 100% Pure Java](#) are available from IBM Special Needs Systems.

### JAVAACCESS

Information about [Java Accessibility and Usability](#) is available from the Trace R and D Center.

### MACROMEDIA

[Flash OBJECT and EMBED Tag Syntax](#) from Macromedia.

### MATHML

["Mathematical Markup Language"](#), P. Ion and R. Miner, eds., 7 April 1998, revised 7 July 1999. This MagroupML 1.0 Recommendation is <http://www.w3.org/1999/07/REC-MagroupML-19990707>. [The latest version of MagroupML 1.0](#) is available at <http://www.w3.org/TR/REC-MagroupML>.

### TRACE

The [Trace Research and Development Center](#). Consult this site for a variety of information about accessibility, including a [scrolling Java applet that may be frozen by the user](#).

### WAI-ER

The [WAI Evaluation and Repair Working Group](#)

### WCAG10-CSS-TECHNIQUES

["CSS Techniques for Web Content Accessibility Guidelines 1.0"](#), W. Chisholm, G. Vanderheiden, and I. Jacobs, eds. [The latest version of this document](#) is available at <http://www.w3.org/TR/WCAG10-CSS-TECHS/>.

### WCAG20

["Web Content Accessibility Guidelines 2.0"](#), B. Caldwell, W. Chisholm, J. White, and G. Vanderheiden, eds.

### XHTML1

["XHTML 1.0 The Extensible HyperText Markup Language \(Second Edition\)"](#) Steven Pemberton, et al., 26 January 2000, revised 1 August 2002. This XHTML1 Recommendation is <http://www.w3.org/TR/2002/REC-xhtml1-20020801/>. [The latest version of XHTML1](#) is <http://www.w3.org/TR/xhtml1/>.

### ISOConsol

["Consolidated Listing of ISO 639, ISO 4217, SI Measurement Symbols, and World Time Zones"](#) John McClure, et al., 2002. This document is located at <http://www.hypergrove.com/Publications/Symbols.html>.