# Legal Instrument Markup
# Using XHTML 2.0

**by John McClure, Hypergrove Engineering**
**Working Draft, July 26 2004**

### Background

The Structure Subcommittee was organized in February 2004 by the LegalXML eContracts TC to examine XHTML 2.0 as an implementing technology for the representation of the document structure common to legal contracts and other legal instruments. Initial discussions focused on the fundamental architecture of legal clauses, resulting in the unanimous recommendation that the eContracts TC adopt the "grammatical paragraph" model, where all ~~text~~ block-level content of any "clause" is to be located within <p> elements only.

Our subsequent work focused on additions to, changes of, and deletions from, the elements defined by the XHTML 2.0 language. Our functional requirements were established by the TC relating to first, the structure of an XHTML file representing a contract and second, to more general objectives of interested stakeholders. Specifically, the Subcommittee did not examine the use of XHTML 2.0 to meet TC requirements for "semantic" markup, that is, XML elements or attributes identifying the functional purpose of text within a digital contract, e.g., the contract's title, parties, dates, terms, or any metadata about the production, acceptance, or management of the contract. The scope was delineated as the "structure" of the contract.

Note: Our discussions are now based on the XHTML 2.0 Working Draft of July 22, 2004.

**Colophon.** This XHTML file is encoded using the eContracts XML proposed in this document. It is displayed using a CSS-2 stylesheet, and is tested only under Mozilla. Items in green denote additions. Items in gray are struck, and content in blue are emphasized.

**Summary**

The primary objective of this document is to demonstrate the adequacy of the XHTML 2.0 elements for meeting practically all TC functional requirements for markup of a contract's "structure" aspect. Contract structure consists of both the 'language' of the contract, and of its attached material.

> *This document recommends as few modifications as possible to the XHTML 2.0 set of elements, a guideline driven by the view that minimizing departures from the letter or intent of the XHTML 2.0 specification materially enhances the prospect of acceptance of the eContracts schema by the marketplace.*

Overall, this document makes the following recommendations.

1. Add seven new elements to the XHTML 2.0 set of elements.

1.1 <instrument> - a container to specifically identify legal material within a larger XHTML document. The <instrument> element contains one optional <recitals> element and one required <body> element which contain the primary content of an instrument, and it contains multiple optional <signature> elements.

1.2 <nr> - a container for the numeric portion of a caption. This element is then included in the content models used for divisions of a document, clauses, tables, images, and lists within the text of a document.

1.3 <list> and <item> - a container and item element to represent a list compatible with the grammatical paragraph model, rather than significantly redefining the content models for the XHTML ordered list element.

Because these elements would necessarily have a "namespace prefix" when used in an XHTML 2.0 file, it would be appropriate to request the W3C to consider adding the <nr> and <list> elements to the XHTML 2.0 dialect. This prefix is desirable only for the <legal:instrument>, <legal:recitals>, <legal:body>, and <legal:signature> elements. Note: the <nr> element was defined in an earlier XHTML 2.0 WD.

2. Add one new optional attribute, @name, to the <section> <p>, <item>, and <hx> elements (the clause, paragraph, list-item, and heading elements) to identify the heading for and membership of a group of clauses, paragraphs, or list-items, as implied by their heading. This technique is recommended rather the alternative, creating one or more "group" elements, because stylesheet-based auto-numbering cannot be performed correctly in the presence of "group" elements.

3. Constrain the content model of certain XHTML 2.0 structural elements to conform to the grammatical paragraph model adopted by the TC. A potential consequence of this recommendation is that the TC should request the W3C to establish a new XHTML conformance level scheme.

4. Use the <div> element appropriately as indicated by the XHTML 2.0 WD, that is, as a container for material that is structural with respect to the legal content of the instrument. This includes headers, footers, footnotes, left and right margin contents, ~~captioned tables and figures,~~ plus all material appended and prepended to the legal content of the instrument.

5. Adopt metadata conventions established by the XHTML 2.0 WD specification. Most particularly, on all XHTML elements, use the @property attribute to identify the purpose of certain text strings within the document; and the @resource and @about attributes on all XHTML elements to identify the data objects being referenced by the text of the contract.

6. Eliminate certain antiquated elements, for example, <code>, from the XHTML 2.0 set of elements.

7. Adopt the XHTML standards for all other document constructs such as citations, definitions, tables of contents and, most importantly, elements necessary for form contracts. This will ensure that the default presentation of the markup is handled consistently by XHTML 2.0 User Agents.

### Example Markup

```
<legal:instrument type='legal:Agreement' rdf:ID='http://www.dot.com/R123.xhtml'
                  xmlns:legal="http://www.legalxml.org">
    <h>Skeleton Contract Template</h>
    <h>
        <l>THIS AGREEMENT dated 4 April 2004</l>
        <l> BETWEEN </l>
        <l>Fiona First LLC ("First Party")</l>
        <l> AND </l>
        <l>Simon Second LLC ("Second Party")</l>
    </h>
    <legal:recitals>
        <h>BACKGROUND</h>
        <section>
            <p>Simon would like to ...</p>
        </section>
        <section>
            <p>... according to the terms and conditions of this agreement.</p>
        </section>
    </legal:recitals>
    <legal:body>
          <h>IT IS HEREBY AGREED:</h>
          <section>
              <nr>1.</nr>
              <h>First clause</h>
              <p>Clause Body. Clause Body. Clause Body. ... </p>
          </section>
    </legal:body>
    <legal:signature>
        <l property='legal:Signer1Line'/>
        <l property='legal:Signer1Name'>Put 1st party name here.</l>
        <l property='legal:Signer1Date'>Put 1st party signature date here</l>
        <legal:instrument type='legal:Notarization'>
            <h>Notary Public Certificate</h>
            <legal:body>
              <p>My commission expires on December 31, 2010.</p>
            </legal:body>
            <legal:signature>
              <l property='legal:SignerLine'/>
              <l property='legal:SignerName'>Place notary name here</l>
              <l property='legal:SignerLocation>Notary jurisdiction here</l>
            </legal:signature>
        </legal:instrument>
    </legal:signature>
    <div>
        <h>Schedules</h>
        <div>
            <nr>Schedule 1 - </nr>
            <h>Key Personnel</h>
            <table/>
        </div>
        <div>
            <nr>Schedule 2 - </nr>
            <h>Proforma NDA</h>
            <legal:instrument type='legal:Agreement'>
                <h>Non disclosure Agreement</h>
                <legal:body></legal:body>
                <legal:signature></legal:signature>
            </legal:instrument>
        </div>
    </div>
</legal:instrument>
```

## 1. Clauses

*Background*

The term "clause" represents a block of text that is numerically captioned, at a minumum, within the language of a legal instrument. No consensus exists in the international legal community about the name for this text block.

The XHTML 2.0 language introduces the <section> element as a new block-level element, with no designation about its functional use. The <section> element is a recursive element, meaning that a <section> may contain <section> elements. The other block-level, flowing-text recursive element provided by XHTML is the <div> element.

*Issues Examined*

1.1 The <section> content model breaks the "grammatical paragraph" model requirement by allowing #PCDATA, ~~<div>,~~ inline elements, and <table> elements as components.

1.2 Groups of clauses, indicated by a preceding heading, must be represented in a contract structure. The XHTML 2.0 mechanism for grouping elements, the @name attribute, was evaluated for application to groups of clauses.

1.2.1 A contract contains both operative and inoperative clauses, the latter normally indicated by strikeout applied through its text. XHTML support for identifying such "deleted" material is the @edit attribute, whose value would be "deleted".

*Recommendations*

1.3 Establish the <section> element as the markup element for a numbered clause. Allow multiple <section> elements within an <instrument> element.

1.4 Identify inoperative clauses by the presence of the @edit attribute, with a value of "deleted". Require CSS-based user agents to render inoperative clauses by a default styling of {text-decoration:strikethrough}.

1.5 Require that all clauses be numerically captioned, and optionally titled, to (a) functionally distinguish clauses from uncaptioned grammatical paragraph content (b) enable the creation of a table of contents and (c) generate standardized reference strings to one or more clauses.

1.6 Remove #PCDATA, inline elements, and the <table> ~~<div>, and list~~ elements from the <section> element content model.

1.7 Identify membership of a clause in a specific clause group by using the @name attribute on the <section> element. The value of this attribute is assigned by an author or by the application. Its value should should correlate to the value of a @name attribute on an associated <h*x*> element located within the <section> element's parent <section> element. This attribute can be used by applications to provide group-specific styling and for efficient clause repositioning during contract editing. necessary

## 2. Captions

*Background*

Captions are used to identify specific content within a legal instrument, each consisting of a number and a title, depending on the item captioned. All clauses have, at a minimum, a numeric caption. Tables and images are optionally captioned with a number and or a title. All other structural blocks of text in the document, whether appended or prepended to the primary content of the instrument, are captioned, at a minimum, with a title.

The numeric part of a caption may itself consist of a text or image prefix to a structured number reflecting at least the sequence of its captioned material, and optionally its depth within the clause tree. The numeric part may optionally be suffixed by text, often a "period", however images should be allowed as well.

The XHTML 2.0 language introduces the <h> element as an addition to the <h1> to <h6> elements provided by XHTML 1.1; in this document, these are referred to collectively as "the <h*x*> elements". XHTML 2.0 also introduces the <caption> element as a component of the content models for the <table> and <object> elements.

*Issues Examined*

2.1 No element exists in XHTML 2.0 to identify the numeric part of a caption. Numeric captions are an established requirement for clauses, tables, images, and appended and prepended material.

2.2 Auto-numbering and limited text generation is a feature of CSS-2, a styling standard inconsistently implemented across tools. Our ability to meet our presentation fidelity requirements must not be predicated on complete implementation of CSS-2 by User Agents.

2.3 Captions for grouped paragraphs and grouped clauses are a mandatory requirement. XHTML 1.0 provided a grouping mechanism, the @name attribute for its radio buttons, that could be used here.

*Recommendations*

2.4 Designate a new in-line element -- <nr> -- as the container for numeric caption text strings.

2.5 Require numeric captioning for each <section> element and text captioning for each <div> element. Captioning for the <legal:recitals>, <legal:body>, and <legal:signature> elements would be optional.

2.6 Use the term "should not" when describing stylesheet-based generation of any content in the document until broad industry support of such capabilities is assured. The objective is to eliminate the uncertainty of maintaining presentation fidelity for those parties equipped with software tools unable to generate presentation text or caption numbers.

2.7 Allow <section> and <div> elements to contain a single <nr> element and multiple <h*x*> elements.

2.8 Add the <nr> element as an optional component of the content model for the <caption> and <h*x*> element. When used in this way by an author, an <nr> element cannot also have a sibling <h*x*> element. This accommodates multiple layout and styling requirements an author may have.

2.9 Provide a @name attribute on the <h*x*> element, to accommodate clause and paragraph group captioning requirements.

## 3. Tables

*Background*

XHTML 2.0 introduces the <caption> element which can be used on table and object elements. Captioning for the <table> element is now supported .

*Issues Examined*

3.1 The <table> element is allowed within various XHTML 2.0 elements that are outside the scope of a <p> element, breaking the "grammatical paragraph" model requirement.

3.2 The <table> element supports styling attributes that jeopardize the fidelity of a contract's representation. Styling is already provided through mechanisms external to the <table> element and its child elements.

3.3 The CALS/OASIS table markup model is not supported by XHTML 2.0; this is an important standard from our organization.

3.5 Table bodies, rable rows, and table cells cannot be captioned by XHTML 2.0 elements. Legal instruments reference material located in a table by row-group, by row, and by cell. Without captioning, standardized reference strings cannot be generated.

*Recommendations*

3.6 Remove the <table> element from the content models for ~~<body>,~~ <section> ~~and <div>~~ element, to meet the "grammatical paragraph" requirement.

~~3.7 Add the <nr> and <h*x*> elements as optional components of the <table> element's content model, to support table captioning.~~

3.8 Remove the @rules attribute from the <table> element, to promote presentation fidelity.

3.9 Use standard XML Namespace conventions, to support inclusion of CALS/OASIS table markup in an XHTML file. Identify or create XSLT stylesheets to transform a CALS table to XHTML 2.0 markup.

3.10 Add the <nr> and <h*x*> elements as optional components of the <tbody>, <tr>, and <td> content models, to provide captioning for table row groups, table rows, and table cells.

## 4. Objects

*Background*

The WD says: "The Object Module provides elements for general-purpose object inclusion; this includes images and other media, as well as executable content. .. When this module is used, it adds the object element to the Inline content set of the Inline Text module."

XHTML 2.0 introduces the <caption> element that can be used on table and object elements. Captioning for the <object> element is now supported .

*Issues Examined*

4.1 The <object> element is allowed within various XHTML 2.0 elements that are outside the scope of a <p> element, breaking the "grammatical paragraph" model requirement.

4.2 The <object> element does not support captioning. Legal instruments contain captioned images, refer to captioned images, and index captioned images.

4.3 The <object> element allows executable content, i.e., a reference to an executable program. Processing by this program may jeopardize the fidelity of a contract presented some time following its acceptance by parties.

*Recommendations*

4.4 Remove the <object> element from the content models for <body>, <section>, and <div> elements, to meet the "grammatical paragraph" requirement.

~~4.5 Add the <nr> and <hx> elements as optional components of the <object> element's content model, to support image captioning.~~

4.6 Use the term "should not" to describe the use of executable programs anywhere within an <instrument> element, to ~~ensure~~promote presentation integrity of the contract following its acceptance by parties.

## 5. **Appended Material**

*Background*

Section 8.4 of the WD says:

> *"The div element, in conjunction with the id and class attributes, offers a generic mechanism for adding extra structure to documents. This element defines no presentational idioms on the content. Thus, authors may use this element in conjunction with stylesheets , the xml:lang attribute, etc., to tailor XHTML to their own needs and tastes."*

*Issues Examined*

5.1 No direct support for material appended or prepended to the primary content of a legal instrument exists in XHTML 2.0. Numerous legal instruments require multiple front- matter sections including, for example, cover pages, contents table pages, and summary pages; and multiple back-matter sections including, for example, subordinate legal instruments, generic documents, and index pages.

5.2 Multiple structured documents must be able to be represented within the material appended or prepended to the main portion of a legal instrument. XHTML 2.0 does provide the <link> element for identifying documents appended to a document, however these are normally assumed to be external to the contract and, therefore, not immediately presumed to have been presented to a party at the time of contract acceptance. This is an important presentation security requirement.

5.3 Three types of strucure-oriented text blocks have been identified (uncaptioned, text-captioned, and numeric-captioned). An XHTML element is asigned to each of these roles (<p>, <div>, and <section>, respectively). All <div> elements require a text-caption. ~~XHTML 2.0 <div> content model does not require captioning of any kind. The <section> element requires numeric captioning; the <p> element allows no captioning; and all prepended and appended material requires caption titles. These titles are used for generation of reference strings relating to the material.~~

5.3.1 Footnotes, footer lines, banners and other page areas are important requirements for exchange of a legal instrument, possibly more so than of a contract. The XHTML 2.0 <div> element content model defined by the W3C already addresses such a requirement for identifying specific structural components peculiar to a given type of a document (please see definition above). While this content can be generated by an XSL-T stylesheet, such practice should not be advised to authors exchanging the instrument with other parties because consuming applications cannot be required to perform the transformation prior to its processing of the instrument. [The second reason is concerned with the resulting dependency on 'programming instructions', a non-standardized XML feature.] [The third reason concerns the technical definition of a "legal instrument" that is exchanged between parties. That is, it is the presentation of the legal instrument (the result of the transform) that is packaged in a digital signature envelope, not the input file and transformative stylesheet -- only in this way (packaging the XHTML generated by the XSL-T stylesheet) may something close to presentation fidelity for subsequent viewings be best assured.]

*Recommendations*

5.4 Use the <div> element as the container for any material appended or prepended to the primary content of the legal instrument. The <div> element is recursive, meeting the need for embedding one or more documents (parts) within appended or prepended material.

5.5 Add the <nr> and <h*x*> elements to the <div> element content model, to support captions and table of contents generation.

~~5.6 Remove the <div> element from the content model of the <body> and <section> elements; Retain the <div> element within a table cell element content model.~~

5.7 Add the <instrument> element as an optional component of the <div> content model, to support appended subordinate legal instruments.

5.8 Add the <div> element as an optional component of the content models for several elements, to support page area content. The following @property keywords would be recognized:

For <instrument>

| | |
|---|---|
| layout:RunningBanner | contains content for a repeating page banner area |
| layout:RunningFooter | contains content for a repeating page footer area |
| layout:RunningLeftMargin | contains content for the left margin of the presentation for the document |
| layout:RunningRightMargin | contains content for the right margin of the presentation for the document |

For <section>

| | |
|---|---|
| layout:Footnote | contains content for a footnote cited by the parent clause element |
| layout:RunningBanner | contains content for a repeating page banner area to begin on the page that the section element content appears on. |
| layout:RunningFooter | contains content for a repeating page footer area to begin on the page that the section element content appears on. |
| layout:RunningLeftMargin | contains content for the left margin of the presentation for the document to begin on the page that the section element content appears on. |
| layout:RunningRightMargin | contains content for the right margin of the presentation for the document to begin on the page that the section element content appears on. |

For <paragraph>

| | |
|---|---|
| layout:Footnote | contains content for a footnote cited by the parent paragraph element |

For <div>

| | |
|---|---|
| layout:Footnote | contains content for a footnote cited by the parent element |
| layout:RunningBanner | contains content for a repeating page banner area to begin on the page that the <div> element content appears on. |
| layout:RunningFooter | contains content for a repeating page footer area to begin on the page that the <div> element content appears on. |
| layout:RunningLeftMargin | contains content for the left margin of the presentation for the document to begin on the page that the <div> element content appears on. |
| layout:RunningRightMargin | contains content for the right margin of the presentation for the document to begin on the page that the <div> element content appears on. |

## 6. **Lists**

*Background*

The XHTML 2.0 specification defines four types of lists: ordered list, unordered list, navigation list, and definition list. XHTML 2.0 defines a new attribute, @value, that "specifies the value to be used when determining the value of the enumerator on a list item within an ol". The <nl> (navigation list) element is a container for a set of links, introduced by XHTML 2.0.

[Note. The Subcommittee earlier reached consensus on several items that are not supported below. (1) that three of the four XHTML 2.0 lists were redundant; (2) that modifying the XHTML ordered list's list item element was the best design approach; (3) that list item elements contain grammatical paragraphs and further block content, and are not mixed content elements.]

*Issues Examined*

6.1 The <nr> element is a container for numeric captions, such as normally formatted by XHTML User Agents for items of any <ol> element. If the <nr> element is added to the content model for a <ol> element, there is concern that the default formatting of the numeric captioning of each item in an <ol> will interfere with the contents of an <nr> element for that <li> element.

6.2 XMTML 2.0 lists have a common child element, the <li> element. The <li> element may contain <p>, <table> and other block level elements, breaking the grammatical paragraph model.

6.3 XHTML 2.0 provides no support for inline-lists, whose items have a simpler content model than the <li> element excluding for example, block-level content.

*Recommendations*

6.4 Include the standard XHTML unordered, ordered, definition, and navigation list elements in the content model for a <div> element. Navigation lists are menus that can be displayed during presentation of a legal instrument, either in a Table of Contents, but also the Banner, Margin, or Footer; capturing the elements of a presentation meets general stakeholder requirements. Ordered and unordered lists are found in legacy material that may be attached to the instrument (all encapsulated within a <div>).

6.5 Establish two new elements, <list> and <item>. The content model for the <item> element includes only in-line elements which would include <nr>. The presentation effect of "paragraphs" and "headings" as they occur within a list item can be achieved by applying styling to in-line elements.

6.6 For ease of authoring, allow the <item> element to have mixed content.

6.7 Include the <dl> (definition list) element in the content model for grammatical paragraphs.

## 7. Instrument Structure

*Background*

The working definition for a "legal instrument" is a document that has three significant parts: recitals of the facts of the instruments, body of the instrument, and signatures of the parties. An instrument may have either pertinent or relatively non-pertinent information located either prior to the recitals, or following the party signatures.

*Issues Examined*

7.1 XHTML 2.0 provides no direct support for these concepts. All are mandatory TC requirements.

7.2 A legal instrument may be located within the body of a larger XHTML file. An element is required to identify the legal instrument so that validation activities specific to a legal instrument may be performed.

*Recommendations*

7.3 Use the <instrument> element to identify all legal information. Provide one optional <recitals> element, one required <body> element, and one optional <signature< element within the <instrument> element content model.

7.4 In the <instrument> content model, surround the three legal elements with multiple optional <div> elements, to represent ancillary information to the legal recitals, body, and signatures.

7.5 Provide identical content models for the <recitals> and <body> elements. Each of these elements should allow multiple <p>, <section>, and optional <h*x*> elements. These should require either a paragraph or a clause must be present.

7.6 Allow only <l> (line) elements within the <signature> element. Use the @property attribute to identify the following specific items found in a signature area of a legal instrument, so as to accommodate instrument acceptance scenarios geared by the need for a party to, for instance, tab to a particular position in the presentation of the legal instrument.

| | |
|---|---|
| legal:SignerLine | Identifies an area used for a handwritten signature of a party |
| legal:Signer1Line | Identifies an area used for a handwritten signature of a First Party |
| legal:Signer2Line | Identifies an area used for a handwritten signature of a Second Party |
| legal:Signer3Line | Identifies an area used for a handwritten signature of a Third Party |
| legal:SignerName | Identifies a string that is the name of a signer who is a party |
| legal:Signer1Name | Identifies a string that is the name of a signer who is a First Party |
| legal:Signer2Name | Identifies a string that is the name of a signer who is a Second Party |
| legal:Signer3Name | Identifies a string that is the name of a signer who is a Third Party |
| legal:SignerDate | Identifies a string that is the date/time of an individual signature |
| legal:Signer1Date | Identifies a string that is the date/time of an individual signature |
| legal:Signer2Date | Identifies a string that is the date/time of an individual signature |
| legal:Signer3Date | Identifies a string that is the date/time of an individual signature |
| legal:SignerLocation | Identifies a string that is any part of an individual signer's location |
| legal:Signer1Location | Identifies a string that is any part of an individual signer's location |
| legal:Signer2Location | Identifies a string that is any part of an individual signer's location |
| legal:Signer3Location | Identifies a string that is any part of an individual signer's location |
| legal:SignerTitle | Identifies a string that provides a title for a signer |
| legal:Signer1Title | Identifies a string that provides a title for a signer |
| legal:Signer2Title | Identifies a string that provides a title for a signer |
| legal:Signer3Title | Identifies a string that provides a title for a signer |
| legal:SignerRole | Identifies a string that identifies the role for a signer (.e.g, Tenant). |
| legal:Signer1Role | Identifies a string that identifies the role for a signer |
| legal:Signer2Role | Identifies a string that identifies the role for a signer |
| legal:Signer3Role | Identifies a string that identifies the role for a signer |

Please note that specifically no effort is made to establish a structured element for these information items. No effort is made to accommodate multiple persons fulfilling the role of first, second, or third parties.

7.7 Promote the use of the @resource attribute on the < legal:signature > element. This attribute contains the URI for a (publisher's) structured description of a party or notary, or location. This information effectively eliminates parsing the content of those signature area elements in order to, for instance correlate parties across legal instruments.

7.8 Establish a @type attribute for the <instrument> element that includes these values.

| | |
|---|---|
| legal:Agreement | Identifies the instrument as a legal agreement between parties. |
| legal:Consent | Identifies the instrument as a legal consent by a party. |
| legal:Declaration | Identifies the instrument as a legal declaration by a party. |
| legal:Demand | Identifies the instrument as a legal demand of one party to another. |
| legal:Notice | Identifies the instrument as a legal notice of one party to another. |
| legal:Receipt | Identifies the instrument as a legal certificate of receipt. |
| legal:Request | Identifies the instrument as a legal request of one party to another. |
| legal:Service | Identifies the instrument as a legal service of an instrument to its parties |
| legal:Notarization | Identifies the instrument as a notarization of a signature |

7.9 Locate notarization certificates within the content model for <legal:signature> elements.

7.10 The author should provide an @rdf:ID to the <instrument> element in order for other applications to easily provide metadata descriptions of the contents of the legal instrument. The use of the XML "id" attribute for any element within the <instrument> element should be discouraged in favor of Universal Resource Identifiers.

## 8. **Form Contracts**

*Background*

XHTML 2.0 has eliminated all form elements found in XHTML 1.1, instead adopting XForms' elements. Elements eliminated, here called 'widgets', include form, input, radio, checkbox, select, and button.

*Issues Examined*

8.1 Form contracts and other legal instruments may locate 'widgets' anywhere within the recitals, body, or signature areas of the instrument. Support for form contracts is a mandatory TC requirement.

*Recommendations*

8.2 Allow all XForms elements to appear within an <instrument> element whereever text or images may appear.

## 9. **Antiquated XHTML Elements**

*Background*

*Issues Examined*

9.1 Certain elements have been identified whose presence in the XHTML 2.0 schema is thought to be confusing and not useful to authors of legal instruments.

*Recommendations*

9.2 Eliminate the <address>, <blockcode>, <code>, <samp>, <kbd>, <pre>, and <var> elements from the content model for an <instrument> element.

9.3 Designate a functional and or legal difference between the <em> and <strong> elements.

---

## 10. **Quotations**

*Background*

XHTML 2.0 provides the <blockquote> element for block-level quotations, and provides the <quote> element for in-line quotations.

*Issues Examined*

10.1 Representing quotations is a mandatory TC requirement.

*Recommendations*

10.2 Use both elements as described by the XHTML 2.0 specification.

---

## 11. **Citations**

*Background*

Citations with or without a hyperlink to cited material, can be specified in a number of ways using XHTML 2.0. Today, the <a> (anchor) element is most commonly used, however the <cite> element has also been available. XHTML 2.0 allows hyperlinking to be accomplished with any element. The XHTML WD says:

> *"The Hypertext Module provides (the HTML anchor element, <a>) that traditionally has been used in HTML to define hypertext links to other resources. Although all elements may now play the role of a hyperlink (using the href attribute) or hyperlink anchor (using the id attribute), this element remains for explicit markup of links, though is otherwise identical in semantics to the span element."*

*Issues Examined*

11.1 Citations to individual and grouped structural components of a legal instrument are a mandatory TC requirement.

*Recommendations*

11.2 Require all citations to use the <cite> element, reserving the use of both the <a> (anchor) element and hyperlink attributes (such as @href) both on the <cite> element and on other elements for the author's own presentation uses.

11.3 Provide a set of values for the @property attribute to be used on hyperlink elements so as to provide for standardized reference strings and a rich User Experience. These values should reflect the components found in a contract structure.

| | |
|---|---|
| legal:DocumentReference | Hyperlinks to a document that is not a legal instrument |
| legal:InstrumentReference | Hyperlinks to a related, external instrument |
| legal:SubInstrumentReference | Hyperlinks to subordinate instrument |
| legal:RecitalsReference | Hyperlinks to the recitals area of the instrument |
| legal:BodyReference | Hyperlinks to the legal body area of the instrument |
| legal:SignatureReference | Hyperlinks to a signature area of the instrument |
| legal:TermReference | Hyperlinks to a definition of a term within a legal instrument |
| legal:BlockReference | Hyperlinks to a captioned <div> within the instrument |
| legal:SectionReference | Hyperlinks to a specific clause |
| legal:SectionGroupReference | Hyperlinks to a group of clauses |
| legal:TableReference | Hyperlinks to a table |
| legal:RowGroupReference | Hyperlinks to a group of rows in a table |
| legal:RowReference | Hyperlinks to a specific row in a table |
| legal:CellReference | Hyperlinks to a cell in a table |
| legal:FigureReference | Hyperlinks to a figure object |
| legal:FrontMatterReference | Hyperlinks to material prepended to the instrument |
| legal:AttachmentReference | Hyperlinks to material appended to the instrument |

11.4 Use the following examples in the published specification:

```
<!-- citation to an external document -->
<cite cite='http://www.w3.org/TR/REC-xml" property='legal:DocumentReference'>[XML]</cite>

<!-- citation to a clause -->
<cite cite='#Section.1.2.3" property='legal:SectionReference'>Section 1.2.3</cite>

<!-- citation to a group of clauses -->
<cite cite='#Section.1.2.3" property='legal:SectionGroupReference'>Section 1.2.3</cite>
through <cite cite='#Section.1.2.7" property='legal:SectionGroupReference'> 1.2.7</cite>

<!-- citation to a defined term -->
<cite cite='#party1" property='legal:TermReference'>first party</cite>
```

## 12. **Document Terms**

*Background*

XHTML 2.0 provides the <defn> (definition) element which "contains the defining instance of the enclosed term".

*Issues Examined*

12.1 Identifying terms defined by a legal instrument is a mandatory TC requirement.

12.1 Identifying the use by a legal instrument of terms defined within itself is a mandatory TC requirement.

*Recommendations*

12.2 Use the <defn> element for identifying terms defined by a legal document.

12.2 Use the <cite> element for identifying the use of terms defined by a legal document.

## 13. **Text Entities**

*Background*

*Issues Examined*

13.1 Accommodating so-called "boilerplate language" of a legal instrument is a mandatory TC requirement.

*Recommendations*

13.2 Text entities representing boilerplate language should be promoted.

13.3 The use of IDREF to indicate an information "delegate" has generally been replaced by standard URI attributes which are functionally named (see the @cite attribute on the <cite> element). IDREFs should be avoided by the specification.

## 14. **Validation**

*Background*

**Syntax Validation**

**Properties Validation**

**Presentation Validation**

*Issues Examined*

14.1

*Recommendations*

14.2

## 15. **CSS Styling**

*Background*

*Issues Examined*

15.1

*Recommendations*

15.2 The stylesheet used to format LegalXML reports and specifications should be promoted as a standardized stylesheet available to application authors.

## 15. **XSL Transforms**

*Background*

Like executable objects, transforms either embedded within or linked to the file in which the <instrument> element is located, pose a serious presentation security risk.

*Issues Examined*

15.1

*Recommendations*

15.2

## 17. **DOCTYPEs**

*Background*

*Issues Examined*

17.1

*Recommendations*

17.2 Publish one doctype that is used for authoring the content of a legal instrument, and a separate doctype conforming to Modular XHTML requirements for the XHTML 2.0 language.

17.3 Publish all schemas conventional DTD syntax; XML Schema syntax; RELAX-NG syntax; and when appropriate, RDF Schema syntax.

## 18. **Links**

*Background*

XHTML 2.0 provides the <link> element within the <head> element to identify (a) resources necessary to the presentation of the instrument; and (b) resources describing the contents of the resource (profiles).

*Issues Examined*

18.1 Links to document profiles is the standard mechanism promoted by XHTML 2.0 for linking a document to one or more "descriptive metadata" about the instrument. Although the XHTML 2.0 WD makes no committment to a particular schema for a document profile, the Resource Description Framework is nonetheless particularly suggested.

*Recommendations*

18.2 Create an RDF Schema which standardizes descriptive information about the production, acceptance, and management of the legal instrument.

## 19. **Scripts**

*Background*

Like executable objects, scripts either embedded within or linked to the file in which the <instrument> element is located, pose a serious presentation security risk.

*Issues Examined*

19.1

*Recommendations*

19.2 Authors should not embed scripts or link to scripts that create or modify the language of the instrument.

## 20. **Document Metadata**

*Background*

*Issues Examined*

20.1 The W3C is actively encouraging the use of the Dublin Core metadata items for documents of the type being standardized by the TC.

20.2

*Recommendations*

20.3 Encourage and promote the use of the Dublin Core property-set, by specifying that legal instruments may use any of the properties defined by the Dublin Core Initiative, such as:

```
<meta name="http-equiv"    content="Content-Type: text/html; charset=ISO-8859-1" />
<meta name='dc:creator'    content='John McClure'/>
<meta name='dc:coverage'   content='International'/>
<meta name='dc:date'       content='July 25, 2004'/>
<meta name='dc:description' content='This minority report contains  ...'/>
<meta name='dc:format'     content='text/xml'/>
<meta name='dc:identifier' content='25July04_MinorityReport'/>
<meta name='dc:language'   content='en'/>
<meta name='dc:publisher'  content='OASIS/LegalXML'/>
<meta name='dc:relation'   content='uri of SC Report '/>
<meta name='dc:rights'     content='See OASIS Intellectual Property statement.'/>
<meta name='dc:source'     content=''/>
<meta name='dc:subject'    content='XML;Standard;Recommendation'/>
<meta name='dc:title'      content='Legal Instrument Markup using XHTML 2.0'/>
<meta name='dc:type'       content='XML Standards Recommendation'/>
```