



# XMPP

## XEP-0071: XHTML-IM

Peter Saint-Andre

<mailto:stpeter@jabber.org>

<xmpp:stpeter@jabber.org>

<https://stpeter.im/>

2008-09-03

Version 1.4

Status	Type	Short Name
Draft	Standards Track	xhtml-im

This specification defines an XHTML 1.0 Integration Set for use in exchanging instant messages that contain lightweight text markup. The protocol enables an XMPP entity to format a message using a small range of commonly-used HTML elements, attributes, and style properties that are suitable for use in instant messaging. The protocol also excludes HTML constructs that may introduce malware and other vulnerabilities (such as scripts and objects) for improved security.

# Legal

## Copyright

This XMPP Extension Protocol is copyright © 1999 - 2012 by the [XMPP Standards Foundation](#) (XSF).

## Permissions

Permission is hereby granted, free of charge, to any person obtaining a copy of this specification (the "Specification"), to make use of the Specification without restriction, including without limitation the rights to implement the Specification in a software program, deploy the Specification in a network service, and copy, modify, merge, publish, translate, distribute, sublicense, or sell copies of the Specification, and to permit persons to whom the Specification is furnished to do so, subject to the condition that the foregoing copyright notice and this permission notice shall be included in all copies or substantial portions of the Specification. Unless separate permission is granted, modified works that are redistributed shall not contain misleading information regarding the authors, title, number, or publisher of the Specification, and shall not claim endorsement of the modified works by the authors, any organization or project to which the authors belong, or the XMPP Standards Foundation.

## Warranty

## NOTE WELL: This Specification is provided on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. ##

## Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall the XMPP Standards Foundation or any author of this Specification be liable for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising from, out of, or in connection with the Specification or the implementation, deployment, or other use of the Specification (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if the XMPP Standards Foundation or such author has been advised of the possibility of such damages.

## Conformance

This XMPP Extension Protocol has been contributed in full conformance with the XSF's Intellectual Property Rights Policy (a copy of which can be found at <http://xmpp.org/about-xmpp/xsf/xsf-ipr-policy/> or obtained by writing to XMPP Standards Foundation, 1899 Wynkoop Street, Suite 600, Denver, CO 80202 USA).

## Contents

1	Introduction	1
2	Choice of Technology	1
3	Requirements	2
4	Concepts and Approach	3
5	Wrapper Element	4
6	XHTML-IM Integration Set	5
6.1	Structure Module Definition	6
6.2	Text Module Definition	6
6.3	Hypertext Module Definition	7
6.4	List Module Definition	7
6.5	Image Module Definition	8
6.6	Style Attribute Module Definition	8
7	Recommended Profile	8
7.1	Structure Module Profile	8
7.2	Text Module Profile	9
7.3	Hypertext Module Profile	9
7.4	List Module Profile	9
7.5	Image Module Profile	9
7.6	Style Attribute Module Profile	10
7.6.1	Recommended Style Properties	10
7.7	Recommended Attributes	11
7.7.1	Common Attributes	11
7.7.2	Specialized Attributes	12
7.7.3	Other Attributes	12
7.8	Summary of Recommendations	13
8	Business Rules	13
9	Examples	15
10	Discovering Support for XHTML-IM	20
10.1	Explicit Discovery	20
10.2	Implicit Discovery	21
11	Security Considerations	21
11.1	Malicious Objects	21
11.2	Phishing	21
11.3	Presence Leaks	22

12	W3C Considerations	22
12.1	Document Type Name . . . . .	22
12.2	User Agent Conformance . . . . .	23
12.3	XHTML Modularization . . . . .	24
12.4	W3C Review . . . . .	24
12.5	XHTML Versioning . . . . .	24
13	IANA Considerations	24
14	XMPP Registrar Considerations	24
14.1	Protocol Namespaces . . . . .	24
15	XML Schemas	25
15.1	XHTML-IM Wrapper . . . . .	25
15.2	XHTML-IM Schema Driver . . . . .	26
15.3	XHTML-IM Content Model . . . . .	27
16	Acknowledgements	32

## 1 Introduction

This document defines methods for exchanging instant messages that contain lightweight text markup. In the context of this document, "lightweight text markup" is to be understood as a combination of minimal structural elements and presentational styles that can easily be rendered on a wide variety of devices without requiring a full rich-text rendering engine such as a web browser. Examples of lightweight text markup include basic text blocks (e.g., paragraphs and blockquotes), structural elements (e.g., emphasis and strength), lists, hyperlinks, image references, and font styles (e.g., sizes and colors).

Note: This specification essentially defines a recommended set of XHTML elements and attributes for use in instant messaging by applying a series of "filters" to the wealth of XHTML features; however, developers of Jabber/XMPP clients mainly need to pay attention to the [Summary of Recommendations](#) rather than the complexities of how the recommendations are derived.

## 2 Choice of Technology

In the past, there have existed several incompatible methods within the Jabber community for exchanging instant messages that contain lightweight text markup. The most notable such methods have included derivatives of [XHTML 1.0](#)<sup>1</sup> as well as of [Rich Text Format \(RTF\)](#)<sup>2</sup>. Although it is sometimes easier for client developers to implement RTF support (this is especially true on certain Microsoft Windows operating systems), there are several reasons (consistent with the [XMPP Design Guidelines](#)<sup>3</sup>) for the [XMPP Standards Foundation \(XSF\)](#)<sup>4</sup> to avoid the use of RTF in developing a protocol for lightweight text markup. Specifically:

1. RTF is not a structured vocabulary derived from SGML (as is [HTML 4.0](#)<sup>5</sup>) or, more relevantly, from XML (as is XHTML 1.0).
2. RTF is under the control of the Microsoft Corporation and thus is not an open standard maintained by a recognized standards development organization; therefore the XSF is unable to contribute to or influence its development if necessary, and any protocol the XSF developed using RTF would introduce unwanted dependencies.

Conversely, there are several reasons to prefer XHTML for lightweight text markup:

---

<sup>1</sup>XHTML 1.0 <<http://www.w3.org/TR/xhtml1>>.

<sup>2</sup>Rich Text Format (RTF) Version 1.5 Specification <<http://msdn.microsoft.com/library/en-us/dnrtfspec/html/rtfspec.asp>>.

<sup>3</sup>XEP-0134: XMPP Design Guidelines <<http://xmpp.org/extensions/xep-0134.html>>.

<sup>4</sup>The XMPP Standards Foundation (XSF) is an independent, non-profit membership organization that develops open extensions to the IETF's Extensible Messaging and Presence Protocol (XMPP). For further information, see <<http://xmpp.org/xsf/>>.

<sup>5</sup>HTML 4.0 <<http://www.w3.org/TR/REC-html40>>.

1. XHTML is a structured format that is defined as an application of [XML 1.0](#)<sup>6</sup>, making it especially appropriate for sending over Jabber/XMPP, which is at root a technology for streaming XML (see [XMPP Core](#)<sup>7</sup>).
2. XHTML is an open standard developed by the [World Wide Web Consortium \(W3C\)](#)<sup>8</sup>, a recognized standards development organization.

Therefore, this document defines support for lightweight text markup in the form of an XMPP extension that encapsulates content defined by an XHTML 1.0 Integration Set that we label "XHTML-IM". The remainder of this document discusses lightweight text markup in terms of XHTML 1.0 only and does not further consider RTF or other technologies.

### 3 Requirements

HTML was originally designed for authoring and presenting structured documents on the World Wide Web, and was subsequently extended to handle more advanced functionality such as image maps and interactive forms. However, the requirements for publishing documents (or developing transactional websites) for presentation by dedicated XHTML user agents on traditional computers or small-screen devices are fundamentally different from the requirements for lightweight text markup of instant messages; for this reason, only a reduced set of XHTML features is needed for XHTML-IM. In particular:

1. IM clients are not XHTML clients: their primary purpose is not to read pre-existing XHTML documents, but to read and generate relatively large numbers of fairly small instant messages.
2. The underlying context for XHTML content in Jabber/XMPP instant messaging is provided not by a full XHTML document, but by an XML stream, and specifically by a message stanza within that stream. Thus the `<head/>` element and all its children are unnecessary. Only the `<body/>` element and some of its children are appropriate for use in instant messaging.
3. The XHTML content that is read by one's IM client is normally generated on the fly by one's conversation partner (or, to be precise, by his or her IM client). Thus there is an inherent limit to the sophistication of the XHTML markup involved. Even in normal XHTML documents, fairly basic structural and rendering elements such as definition lists, abbreviations, addresses, and computer input handling (e.g., `<kbd/>` and `<var/>`)

---

<sup>6</sup>Extensible Markup Language (XML) 1.0 (Fourth Edition) <http://www.w3.org/TR/REC-xml/>.

<sup>7</sup>RFC 6120: Extensible Messaging and Presence Protocol (XMPP): Core <http://tools.ietf.org/html/rfc6120>.

<sup>8</sup>The World Wide Web Consortium defines data formats and markup languages (such as HTML and XML) for use over the Internet. For further information, see <http://www.w3.org/>.

are relatively rare. There is little or no foreseeable need for such elements within the context of instant messaging.

4. The foregoing is doubly true of more advanced markup such as tables, frames, and forms (however, there exists an XMPP extension that provides an instant messaging equivalent of the latter, as defined in [Data Forms](#)<sup>9</sup>).
5. Although ad-hoc styles are useful for messaging (by means of the 'style' attribute), full support for [Cascading Style Sheets](#)<sup>10</sup> (defined by the <style/> element or a standalone .css file, and implemented via the 'class' attribute) would be overkill since many CSS1 properties (e.g., box, classification, and text properties) were developed especially for sophisticated page layout.
6. Background images, audio, animated text, layers, applets, scripts, and other multimedia content types are unnecessary, especially given the existence of XMPP extensions such as [SI File Transfer](#)<sup>11</sup>, [Jingle](#)<sup>12</sup>, and [Jingle File Transfer](#)<sup>13</sup>.
7. Content transformations such as those defined by [XSL Transformations](#)<sup>14</sup> must not be necessary in order for an instant messaging application to present lightweight text markup to an end user.

As explained below, some of these requirements are addressed by the definition of the XHTML-IM Integration Set itself, while others are addressed by a recommended "profile" for that Integration Set in the context of instant messaging applications.

## 4 Concepts and Approach

This document defines an adaptation of XHTML 1.0 (specifically, an XHTML 1.0 Integration Set) that makes it possible to provide lightweight text markup of instant messages (mainly for Jabber/XMPP instant messages, although the Integration Set defined herein could be used by other protocols). This pattern is familiar from email, wherein the HTML-formatted version of the message supplements but does not supersede the text-only version of the message.<sup>15</sup>

---

<sup>9</sup>XEP-0004: Data Forms <<http://xmpp.org/extensions/xep-0004.html>>.

<sup>10</sup>Cascading Style Sheets, level 1 <<http://www.w3.org/TR/REC-CSS1>>.

<sup>11</sup>XEP-0096: SI File Transfer <<http://xmpp.org/extensions/xep-0096.html>>.

<sup>12</sup>XEP-0166: Jingle <<http://xmpp.org/extensions/xep-0166.html>>.

<sup>13</sup>XEP-0234: Jingle File Transfer <<http://xmpp.org/extensions/xep-0234.html>>.

<sup>14</sup>XSL Transformations <<http://www.w3.org/TR/xslt/>>.

<sup>15</sup>The XHTML is merely an alternative version of the message body or bodies, and the semantic meaning is to be derived from the textual message body or bodies rather than the XHTML version.

In Jabber/XMPP communications, the meaning (as opposed to markup) of the message MUST always be represented as best as possible in the normal `<body/>` child element or elements of the `<message/>` stanza qualified by the 'jabber:client' (or 'jabber:server') namespace. Lightweight text markup is then provided within an `<html/>` element qualified by the 'http://jabber.org/protocol/xhtml-im' namespace.<sup>16</sup> However, this `<html/>` element is used solely as a "wrapper" for the XHTML content itself, which content is encapsulated via one or more `<body/>` elements qualified by the 'http://www.w3.org/1999/xhtml' namespace, along with appropriate child elements thereof.

The following example illustrates this approach.

Listing 1: A simple example

```
<message>
  <body>hi!</body>
  <html xmlns='http://jabber.org/protocol/xhtml-im'>
    <body xmlns='http://www.w3.org/1999/xhtml'>
      <p style='font-weight:bold'>hi!</p>
    </body>
  </html>
</message>
```

Technically speaking, there are three aspects to the approach taken herein:

1. Definition of the `<html/>` "wrapper" element, which functions as an XMPP extension within XMPP `<message/>` stanzas.
2. Definition of the XHTML-IM Integration Set itself in terms of supported XHTML 1.0 modules, using the concepts defined in [Modularization of XHTML](#)<sup>17</sup>.
3. A recommended "profile" regarding the specific XHTML 1.0 elements and attributes to be supported from each XHTML 1.0 module.

These three aspects are defined in the three document sections that follow.

## 5 Wrapper Element

The root element for including XHTML content within XMPP stanzas is `<html/>`. This element is qualified by the 'http://jabber.org/protocol/xhtml-im' namespace. From the perspective of XMPP, the wrapper element functions as an XMPP extension element; from the perspective of XHTML, it functions as a wrapper for XHTML 1.0 content qualified by the 'http://www.w3.org/1999/xhtml' namespace. Such XHTML content MUST be contained in

---

<sup>16</sup>It might have been better to use an element name other than `<html/>` for the wrapper element; however, changing it would not be backwards-compatible with the older protocol and existing implementations.

<sup>17</sup>Modularization of XHTML <http://www.w3.org/TR/2004/WD-xhtml-modularization-20040218/>.



one or more `<body/>` elements qualified by the `'http://www.w3.org/1999/xhtml'` namespace and MUST conform to the XHTML-IM Integration Set defined in the following section. If more than one `<body/>` element is included in the `<html/>` wrapper element, each `<body/>` element MUST possess an `'xml:lang'` attribute with a distinct value, where the value of that attribute MUST adhere to the rules defined in RFC 5646<sup>18</sup>. A formal definition of the `<html/>` element is provided in the [XHTML-IM Wrapper Schema](#).

The XHTML `<body/>` element is not to be confused with the XMPP `<body/>` element, which is a child of a `<message/>` stanza and is qualified by the `'jabber:client'` or `'jabber:server'` namespace as described in XMPP IM<sup>19</sup>. The `<html/>` wrapper element is intended for inclusion only as a direct child element of the XMPP `<message/>` stanza and only in order to specify a marked-up version of the message `<body/>` element or elements, but MAY be included elsewhere in accordance with the "extended namespace" rules defined in the XMPP IM specification.

Until and unless (1) additional integration sets are defined and (2) mechanisms are specified for discovering or negotiating which integration sets are supported, the XHTML markup contained within the `<html/>` wrapper element:

1. MUST NOT include elements and attributes that are not part of the XHTML-IM Integration Set defined in Section 6 of this document; any such elements and attributes MUST be ignored if received.
2. SHOULD NOT include elements and attributes that are not part of the recommended profile of the XHTML-IM Integration Set defined in Section 7 of this document; any such elements and attributes SHOULD be ignored if received.

Note: In the foregoing restrictions, the meaning of "ignore" is defined by the conformance requirements of Modularization of XHTML, as summarized in the [User Agent Conformance](#) section of this document).

## 6 XHTML-IM Integration Set

This section defines an XHTML 1.0 Integration Set for use in the context of instant messaging. Given its intended usage, we label it "XHTML-IM".

Modularization of XHTML provides the ability to formally define subsets of XHTML 1.0 via the concept of "modularization" (which may be familiar from [XHTML Basic](#)<sup>20</sup>). Many of the defined modules are not necessary or useful in the context of instant messaging, and in the context of Jabber/XMPP instant messaging specifically some modules have been superseded by well-defined XMPP extensions. This document specifies that XHTML-IM shall be based on the following XHTML 1.0 modules:

---

<sup>18</sup>RFC 5646: Tags for Identifying Languages <http://tools.ietf.org/html/rfc5646>.

<sup>19</sup>RFC 6121: Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence <http://tools.ietf.org/html/rfc6121>.

<sup>20</sup>XHTML Basic <http://www.w3.org/TR/xhtml-basic>.

- Core Modules
  - Structure Module
  - Text Module
  - Hypertext Module
  - List Module
- Image Module
- Style Attribute Module

Modularization of XHTML defines many additional modules, such as Table Modules, Form Modules, Object Modules, and Frame Modules. None of these modules is part of the XHTML-IM Integration Set. If support for such modules is desired, it **MUST** be defined in a separate and distinct integration set.

### 6.1 Structure Module Definition

The Structure Module is defined as including the following elements and attributes: <sup>21</sup>

Element	Attributes
<body/>	class, id, title; style
<head/>	profile
<html/>	version
<title/>	

### 6.2 Text Module Definition

The Text Module is defined as including the following elements and attributes:

Element	Attributes
<abbr/>	class, id, title; style
<acronym/>	class, id, title; style
<address/>	class, id, title; style
<blockquote/>	class, id, title; style; cite
 	class, id, title; style

---

<sup>21</sup>The 'style' attribute is specified herein where appropriate because the Style Attribute Module is included in the definition of the XHTML-IM Integration Set, whereas the event-related attributes (e.g., 'onclick') are not specified because the Implicit Events Module is not included.

Element	Attributes
<cite/>	class, id, title; style
<code/>	class, id, title; style
<dfn/>	class, id, title; style
<div/>	class, id, title; style
<em/>	class, id, title; style
<h1/>	class, id, title; style
<h2/>	class, id, title; style
<h3/>	class, id, title; style
<h4/>	class, id, title; style
<h5/>	class, id, title; style
<h6/>	class, id, title; style
<kbd/>	class, id, title; style
<p/>	class, id, title; style
<pre/>	class, id, title; style
<q/>	class, id, title; style; cite
<samp/>	class, id, title; style
<span/>	class, id, title; style
<strong/>	class, id, title; style
<var/>	class, id, title; style

### 6.3 Hypertext Module Definition

The Hypertext Module is defined as including the <a/> element only:

Element	Attributes
<a/>	class, id, title; style; accesskey, charset, href, hreflang, rel, rev, tabindex, type

### 6.4 List Module Definition

The List Module is defined as including the following elements and attributes:

Element	Attributes
<dl/>	class, id, title; style
<dt/>	class, id, title; style
<dd/>	class, id, title; style
<ol/>	class, id, title; style

Element	Attributes
<ul/>	class, id, title; style
<li/>	class, id, title; style

## 6.5 Image Module Definition

The Image Module is defined as including the <img/> element only:

Element	Attributes
<img/>	class, id, title; style; alt, height, longdesc, src, width

## 6.6 Style Attribute Module Definition

The Style Attribute Module is defined as including the style attribute only, as included in the preceding definition tables.

# 7 Recommended Profile

Even within the restricted set of modules specified as defining the XHTML-IM Integration Set (see preceding section), some elements and attributes are inappropriate or unnecessary for the purpose of instant messaging; although such elements and attributes MAY be included in accordance with the XHTML-IM Integration Set, further recommended restrictions regarding which elements and attributes to include in XHTML content are specified below.

## 7.1 Structure Module Profile

The intent of the protocol defined herein is to support lightweight text markup of XMPP message bodies only. Therefore the <head/>, <html/>, and <title/> elements SHOULD NOT be generated by a compliant implementation, and SHOULD be ignored if received (where the meaning of "ignore" is defined by the conformance requirements of Modularization of XHTML, as summarized in the [User Agent Conformance](#) section of this document). However, the <body/> element is REQUIRED, since it is the root element for all XHTML-IM content.

## 7.2 Text Module Profile

Not all of the Text Module elements are appropriate in the context of instant messaging, since the XHTML content that one views is generated by one's conversation partner in what is often a rapid-fire conversation thread. Only the following elements are RECOMMENDED in XHTML-IM:

- `<blockquote/>`
- `<br/>`
- `<cite/>`
- `<em/>`
- `<p/>`
- `<span/>`
- `<strong/>`

The other Text Module elements MAY be generated by a compliant implementation but MAY be ignored if received (where the meaning of "ignore" is defined by the conformance requirements of Modularization of XHTML, as summarized in the [User Agent Conformance](#) section of this document).

## 7.3 Hypertext Module Profile

The only recommended attributes of the `<a/>` element are specified in the [Recommended Attributes](#) section of this document.

## 7.4 List Module Profile

Because it is unlikely that an instant messaging user would generate a definition list, only ordered and unordered lists are RECOMMENDED. Definition lists SHOULD NOT be generated by a compliant implementation, and SHOULD be ignored if received (where the meaning of "ignore" is defined by the conformance requirements of Modularization of XHTML, as summarized in the [User Agent Conformance](#) section of this document).

## 7.5 Image Module Profile

The only recommended attributes of the `<img/>` element are specified in the [Recommended Attributes](#) section of this document. In addition, for security reasons or because of display constraints, a compliant client MAY choose to display 'alt' text only, not the image itself (for

details, see the [Malicious Objects](#) section of this document).

## 7.6 Style Attribute Module Profile

This module **MUST** be supported in XHTML-IM if possible; although clients written for certain platforms (e.g., console clients, mobile phones, and handheld computers) or for certain classes of users (e.g., text-to-speech clients) might not be able to support all of the recommended styles directly, they **SHOULD** attempt to emulate or translate the defined style properties into text or other presentation styles that are appropriate for the platform or user base in question.

A full list of recommended style properties is provided below.

### 7.6.1 Recommended Style Properties

CSS1 defines 42 "atomic" style properties (which are categorized into font, color and background, text, box, and classification properties) as well as 11 "shorthand" properties ("font", "background", "margin", "padding", "border-width", "border-top", "border-right", "border-bottom", "border-left", "border", and "list-style"). Many of these properties are not appropriate for use in text-based instant messaging, for one or more of the following reasons:

1. The property applies to or depends on the inclusion of images other than those handled by the XHTML Image Module (e.g., the "background-image", "background-repeat", "background-attachment", "background-position", and "list-style-image" properties).
2. The property is intended for advanced document layout (e.g., the "line-height" property and most of the box properties, with the exception of "margin-left", which is useful for indenting text, and "margin-right", which can be useful when dealing with images).
3. The property is unnecessary since it can be emulated via user input or recommended XHTML structural elements (e.g., the "text-transform" property can be emulated by the user's keystrokes or use of the caps lock key)
4. The property is otherwise unlikely to ever be used in the context of rapid-fire conversations (e.g., the "font-variant", "word-spacing", "letter-spacing", and "list-style-position" properties).
5. The property is a shorthand property but some of the properties it includes are not appropriate for instant messaging applications according to the foregoing considerations (in fact this applies to all of the shorthand properties).

Unfortunately, CSS1 does not include mechanisms for defining profiles thereof (as does XHTML 1.0 in the form of XHTML Modularization). While there exist reduced sets of CSS2, these introduce more complexity than is desirable in the context of XHTML-IM. Therefore we

simply provide a list of recommended CSS1 style properties.  
XHTML-IM stipulates that only the following style properties are RECOMMENDED:

- background-color
- color
- font-family
- font-size
- font-style
- font-weight
- margin-left
- margin-right
- text-align
- text-decoration

Although a compliant implementation MAY generate or process other style properties defined in CSS1, such behavior is NOT RECOMMENDED by this document.

## 7.7 Recommended Attributes

### 7.7.1 Common Attributes

Section 5.1 of Modularization of XHTML describes several "common" attribute collections: a "Core" collection ('class', 'id', 'title'), an "I18N" collection ('xml:lang', not shown below since it is implied in XML), an "Events" collection (not included in the XHTML-IM Integration Set because the Intrinsic Events Module is not selected), and a "Style" collection ('style'). The following table summarizes the recommended profile of these common attributes within the XHTML 1.0 content itself:

Attribute	Usage	Reason
class	NOT RECOMMENDED	External stylesheets (which 'class' would typically reference) are not recommended.
id	NOT RECOMMENDED	Internal links and message fragments are not recommended in IM content, nor are external stylesheets (which also make use of the 'id' attribute).

Attribute	Usage	Reason
title	NOT RECOMMENDED	Granting of titles to elements in IM content seems unnecessary.
style	REQUIRED	The 'style' attribute is required since it is the vehicle for presentational styles.
xml:lang	NOT RECOMMENDED	Differentiation of language identification SHOULD occur at the level of the <body/> element only.

### 7.7.2 Specialized Attributes

Beyond the "common" attributes, certain elements within the modules selected for the XHTML-IM Integration Set are allowed to possess other attributes, such as eight attributes for the <a/> element and five attributes for the <img/> element. The recommended profile for such attributes is provided in the following table:

Element Scope	Attribute	Usage
<a/>	accesskey	NOT RECOMMENDED
<a/>	charset	NOT RECOMMENDED
<a/>	href	REQUIRED
<a/>	hreflang	NOT RECOMMENDED
<a/>	rel	NOT RECOMMENDED
<a/>	rev	NOT RECOMMENDED
<a/>	tabindex	NOT RECOMMENDED
<a/>	type	RECOMMENDED
<cite/>	uri	NOT RECOMMENDED
<img/>	alt	REQUIRED
<img/>	height	RECOMMENDED
<img/>	longdesc	NOT RECOMMENDED
<img/>	src	REQUIRED
<img/>	width	RECOMMENDED

### 7.7.3 Other Attributes

Other XHTML 1.0 attributes SHOULD NOT be generated by a compliant implementation, and SHOULD be ignored if received (where the meaning of "ignore" is defined by the conformance requirements of Modularization of XHTML, as summarized in the [User Agent Conformance](#) section of this document).



## 7.8 Summary of Recommendations

The following table summarizes the elements and attributes that are recommended within the XHTML-IM Integration Set.

Element	Attributes
<a/>	href, style, type
<blockquote/>	style
<body/>	style, xml:lang When contained within the <html xmlns='http://jabber.org/protocol/xhtml-im'> element, a <body/> element is qualified by the 'http://www.w3.org/1999/xhtml' namespace; naturally, this is a namespace declaration rather than an attribute per se, and therefore is not mentioned in the attribute enumeration.
 	-none-
<cite/>	style
<em/>	-none-
<img/>	alt, height, src, style, width
<li/>	style
<ol/>	style
<p/>	style
<span/>	style
<strong/>	-none-
<ul/>	style

Any other elements and attributes defined in the XHTML 1.0 modules that are included in the XHTML-IM Integration Set SHOULD NOT be generated by a compliant implementation, and SHOULD be ignored if received (where the meaning of "ignore" is defined by the conformance requirements of Modularization of XHTML, as summarized in the [User Agent Conformance](#) section of this document).

## 8 Business Rules

The following rules apply to the generation and processing of XHTML content by Jabber clients or other XMPP entities.

1. XHTML-IM content is designed to provide a formatted version of the XML character data provided in the <body/> of an XMPP <message/> stanza; if such content is included in an XMPP message, the <html/> element MUST be a direct child of the <message/> stanza and the XHTML-IM content MUST be understood as a formatted version of the message

- body. XHTML-IM content MAY be included within XMPP <iq/> stanzas (or children thereof), but any such usage is undefined. In order to preserve bandwidth, XHTML-IM content SHOULD NOT be included within XMPP <presence/> stanzas; however, if it is so included, the <html/> element MUST be a direct child of the <presence/> stanza and the XHTML-IM content MUST be understood as a formatted version of the XML character data provided in the <status/> element.
2. The sending client MUST ensure that, if XHTML content is sent, its meaning is the same as that of the plaintext version, and that the two versions differ only in markup rather than meaning.
  3. XHTML-IM is a reduced set of XHTML 1.0 and thus also of XML 1.0. Therefore all opening tags MUST be completed by inclusion of an appropriate closing tag.
  4. XMPP Core specifies that an XMPP <message/> MAY contain more than one <body/> child as long as each <body/> possesses an 'xml:lang' attribute with a distinct value. In order to ensure correct internationalization, if an XMPP <message/> stanza contains more than one <body/> child and is also sent as XHTML-IM, the <html/> element SHOULD also contain more than one <body/> child, with one such element for each <body/> child of the <message/> stanza (distinguished by an appropriate 'xml:lang' attribute).
  5. Section 11.1 of XMPP Core stipulates that character entities other than the five general entities defined in Section 4.6 of the XML specification (i.e., &lt;, &gt;, &amp;, &apos;, and &quot;) MUST NOT be sent over an XML stream. Therefore implementations of XHTML-IM MUST NOT include predefined XHTML 1.0 entities such as &nbsp; -- instead, implementations MUST use the equivalent character references as specified in Section 4.1 of the XML specification (even in non-obvious places such as URIs that are included in the 'href' attribute).
  6. For elements and attributes qualified by the 'http://www.w3.org/1999/xhtml' namespace, user agent conformance is guided by the requirements defined in Modularization of XHTML; for details, refer to the [User Agent Conformance](#) section of this document.
  7. The use of structural elements is NOT RECOMMENDED where presentational styles are desired, which is why very few structural elements are specified herein. Implementations SHOULD use appropriate 'style' attributes (e.g., <span style='font-weight: bold'>this is bold</span> and <p style='margin-left: 5%'>this is indented</p>) rather than XHTML structural elements (e.g., <strong/> and <blockquote/>) wherever possible.

8. Nesting of block structural elements (<p/>) and list elements (<dl/>, <ol/>, <ul/>) is NOT RECOMMENDED, except within <div/> elements.
9. It is RECOMMENDED for implementations to replace line breaks with the <br/> element and to replace significant whitespace with the appropriate number of non-breaking spaces (via the NO-BREAK SPACE character or its equivalent), where "significant whitespace" means whitespace that makes some material difference (e.g., one or more spaces at the beginning of a line or more than one space anywhere else within a line), not "normal" whitespace separating words or punctuation.
10. When rendering XHTML-IM content, a user agent SHOULD NOT render as a hyperlink text that is not structured via the <a/> element from the Hypertext Module; therefore if the sender wishes text to be linked, the sending user agent MUST represent the text using the <a/> element and appropriate attributes.

## 9 Examples

The following examples provide an insight into the inclusion of XHTML content in XMPP <message/> stanzas but are by no means exhaustive or definitive.

(Note: The examples might not render correctly in all web browsers, since not all web browsers comply fully with the XHTML 1.0 and CSS1 standards. Markup in the examples may include line breaks for readability. Example renderings are shown with a colored background to set them off from the rest of the text.)

Listing 2: Emphasis, font colors, strength

```
<message>
  <body>Wow, I&apos;m green with envy!</body>
  <html xmlns='http://jabber.org/protocol/xhtml-im'>
    <body xmlns='http://www.w3.org/1999/xhtml'>
      <p style='font-size:large'>
        <em>Wow</em>, I&apos;m <span style='color:green'>green</span>
        with <strong>envy</strong>!
      </p>
    </body>
  </html>
</message>
```

This could be rendered as follows:

Wow, I'm green with envy!

Listing 3: Blockquote, cite

```

<message>
  <body>As Emerson said in his essay Self-Reliance:

    &quot;A foolish consistency is the hobgoblin of little minds.&
    quot;
  </body>
  <html xmlns='http://jabber.org/protocol/xhtml-im'>
    <body xmlns='http://www.w3.org/1999/xhtml'>
      <p>As Emerson said in his essay <cite>Self-Reliance</cite>:</p>
      <blockquote>
        &quot;A foolish consistency is the hobgoblin of little minds.&
        quot;
      </blockquote>
    </body>
  </html>
</message>

```

This could be rendered as follows:

As Emerson said in his essay Self-Reliance:

”A foolish consistency is the hobgoblin of little minds.”

Listing 4: An image and a hyperlink

```

<message>
  <body>Hey, are you licensed to Jabber?

  http://www.xmpp.org/images/psa-license.jpg
</body>
  <html xmlns='http://jabber.org/protocol/xhtml-im'>
    <body xmlns='http://www.w3.org/1999/xhtml'>
      <p>Hey, are you licensed to <a href='http://www.jabber.org/'>
        Jabber</a>?</p>
      <p><img src='http://www.xmpp.org/images/psa-license.jpg'
        alt='A_License_to_Jabber'
        height='261'
        width='537' /></p>
    </body>
  </html>
</message>

```

This could be rendered as follows:

Hey, are you licensed to [Jabber](http://www.jabber.org/)?

Note the large size of the image. Including the 'height' and 'width' attributes is therefore quite friendly, since it gives the receiving application hints as to whether the image is too large to fit into the current interface (naturally, these are hints only and cannot necessarily be relied upon in determining the size of the image).

Rendering the 'alt' value rather than the image would yield something like the following:

Hey, are you licensed to [Jabber](http://www.jabber.org/)?

IMG: "A License to Jabber"

Listing 5: Two lists

```
<message>
  <body>Here&apos;s my .plan for today:
    1. Add the following examples to XEP-0071:
      - ordered and unordered lists
      - more styles (e.g., indentation)
    2. Kick back and relax
  </body>
  <html xmlns='http://jabber.org/protocol/xhtml-im'>
    <body xmlns='http://www.w3.org/1999/xhtml'>
      <p>Here&apos;s my .plan for today:</p>
      <ol>
        <li>Add the following examples to XEP-0071:
          <ul>
            <li>ordered and unordered lists</li>
            <li>more styles (e.g., indentation)</li>
          </ul>
        </li>
        <li>Kick back and relax</li>
      </ol>
    </body>
  </html>
</message>
```

This could be rendered as follows:

Here's my .plan for today:

1. Add the following examples to XEP-0071:
  - ordered and unordered lists
  - more styles (e.g., indentation)
2. Kick back and relax

Listing 6: Quoted text

```
<message>
  <body>
You wrote:

  I think we have consensus on the following:

  1. Remove <div/>
  2. Nesting is not recommended
  3. Don't_preserve_whitespace
```

```

Yes, no, maybe?

That seems fine to me.
</body>
<html xmlns='http://jabber.org/protocol/xhtml-im'>
  <body xmlns='http://www.w3.org/1999/xhtml'>
    <p>You wrote: </p>
    <blockquote>
      <p>I think we have consensus on the following: </p>
      <ol>
        <li>Remove &lt;div/></li>
        <li>Nesting is not recommended</li>
        <li>Don't preserve whitespace</li>
      </ol>
      <p>Yes, no, maybe?</p>
    </blockquote>
    <p>That seems fine to me.</p>
  </body>
</html>
</message>

```

Although quoting received messages is relatively uncommon in IM, it does happen. This could be rendered as follows:

You wrote:

I think we have consensus on the following:

1. Remove <div/>
2. Nesting is not recommended
3. Don't preserve whitespace

Yes, no, maybe?

That seems fine to me.

#### Listing 7: Multiple bodies

```

<message>
  <body xml:lang='en-US'>awesome!</body>
  <body xml:lang='de-DE'>ausgezeichnet!</body>
  <html xmlns='http://jabber.org/protocol/xhtml-im'>
    <body xml:lang='en-US' xmlns='http://www.w3.org/1999/xhtml'>
      <p><strong>awesome!</strong></p>
    </body>
    <body xml:lang='de-DE' xmlns='http://www.w3.org/1999/xhtml'>
      <p><strong>ausgezeichnet!</strong></p>
    </body>
  </html>
</message>

```

```
</html>
</message>
```

How multiple bodies would best be rendered will depend on the user agent and relevant application. For example, a specialized Jabber client that is used in foreign language instruction might show two languages side by side, whereas a dedicated IM client might show content only in a human user's preferred language as captured in the client configuration.

Listing 8: Unrecognized Elements and Attributes

```
<message>
  <body>
    The XHTML user agent conformance requirements say to ignore
    elements and attributes you don't understand, to wit:

    .....4. If a user agent encounters an element it does
    .....not recognize, it must continue to process the
    .....children of that element. If the content is text,
    .....the text must be presented to the user.

    .....5. If a user agent encounters an attribute it does
    .....not recognize, it must ignore the entire attribute
    .....specification (i.e., the attribute and its value).
  </body>
  <html xmlns='http://jabber.org/protocol/xhtml-im'>
    <body xmlns='http://www.w3.org/1999/xhtml'>
      <p>The <acronym>XHTML</acronym> user agent conformance
      requirements say to ignore elements and attributes
      you don't understand, to wit:</p>
      <ol type='1' start='4'>
        <li><p>
          If a user agent encounters an element it does
          not recognize, it must continue to process the
          children of that element. If the content is text,
          the text must be presented to the user.
        </p></li>
        <li><p>
          If a user agent encounters an attribute it does
          not recognize, it must ignore the entire attribute
          specification (i.e., the attribute and its value).
        </p></li>
      </ol>
    </body>
  </html>
</message>
```

Let us assume that the recipient's user agent recognizes neither the `<acronym/>` element (which is discouraged in XHTML-IM) nor the 'type' and 'start' attributes of the `<ol/>` element (which, after all, were deprecated in HTML 4.0), and that it does not render nested elements

(e.g., the `<p/>` elements within the `<li/>` elements); in this case, it could render the content as follows (note that the element value is shown as text and the attribute value is not rendered): The XHTML user agent conformance requirements say to ignore elements and attributes you don't understand, to wit:

1. If a user agent encounters an element it does not recognize, it must continue to process the children of that element. If the content is text, the text must be presented to the user.
2. If a user agent encounters an attribute it does not recognize, it must ignore the entire attribute specification (i.e., the attribute and its value).

## 10 Discovering Support for XHTML-IM

This section describes methods for discovering whether a Jabber client or other XMPP entity supports the protocol defined herein.

### 10.1 Explicit Discovery

The primary means of discovering support for XHTML-IM is [Service Discovery](#)<sup>22</sup> (or the dynamic profile of service discovery defined in [Entity Capabilities](#)<sup>23</sup>).

Listing 9: Seeking to Discover XHTML-IM Support

```
<iq type='get'
  from='juliet@shakespeare.lit/balcony'
  to='romeo@shakespeare.lit/orchard'
  id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#info'/>
</iq>
```

If the queried entity supports XHTML-IM, it MUST return a `<feature/>` element with a 'var' attribute set to a value of "http://jabber.org/protocol/xhtml-im" in the IQ result.

Listing 10: Contact Returns Disco Info Results

```
<iq type='result'
  from='romeo@shakespeare.lit/orchard'
  to='juliet@shakespeare.lit/balcony'
  id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
    ...
```

<sup>22</sup>XEP-0030: Service Discovery <http://xmpp.org/extensions/xep-0030.html>.

<sup>23</sup>XEP-0115: Entity Capabilities <http://xmpp.org/extensions/xep-0115.html>.



```
<feature var='http://jabber.org/protocol/xhtml-im' />
...
</query>
</iq>
```

Naturally, support can also be discovered via the dynamic, presence-based profile of service discovery defined in [Entity Capabilities](#) <sup>24</sup>.

## 10.2 Implicit Discovery

A Jabber user's client MAY send XML <message/> stanzas containing XHTML-IM extensions without first discovering if the conversation partner's client supports XHTML-IM. If the user's client sends a message that includes XHTML-IM markup and the conversation partner's client replies to that message but does not include XHTML-IM markup, the user's client SHOULD NOT continue sending XHTML-IM markup.

# 11 Security Considerations

## 11.1 Malicious Objects

The exclusion of scripts, applets, binary objects, and other potentially executable code from XHTML-IM reduces the risk of exposure to harmful or malicious objects caused by inclusion of XHTML content. To further reduce the risk of such exposure, an implementation MAY choose to:

- Not make hyperlinks clickable
- Not fetch or present images but instead show only the 'alt' text.

In addition, an implementation MUST make it possible for a user to prevent the automatic fetching and presentation of images (rather than leave it up to the implementation).

## 11.2 Phishing

To reduce the risk of phishing attacks <sup>25</sup>, an implementation MAY choose to:

- Display the value of the XHTML 'href' attribute instead of the XML character data of the <a/> element.

---

<sup>24</sup>XEP-0115: Entity Capabilities <<http://xmpp.org/extensions/xep-0115.html>>.

<sup>25</sup>Phishing has been defined by the Financial Services Technology Consortium Counter-Phishing Initiative as "a broadly launched social engineering attack in which an electronic identity is misrepresented in an attempt to trick individuals into revealing personal credentials that can be used fraudulently against them".

- Display the value of XHTML 'href' attribute in addition to the XML character data of the <a/> element if the two values do not match.

### 11.3 Presence Leaks

The network availability of the receiver may be revealed if the receiver's client automatically loads images or the receiver clicks a link included in a message. Therefore an implementation MAY choose to:

- Not fetch images offered by senders that are not authorized to view the receiver's presence.
- Warn the receiver before allowing the user to visit a URI provided by the sender.

## 12 W3C Considerations

The usage of XHTML 1.0 defined herein meets the requirements for XHTML 1.0 Integration Set document type conformance as defined in Section 3 ("Conformance Definition") of Modularization of XHTML.

### 12.1 Document Type Name

The Formal Public Identifier (FPI) for the XHTML-IM document type definition is:

`-//JSF//DTD Instant Messaging with XHTML//EN`

The fields of this FPI are as follows:

1. The leading field is "-//", which indicates that this is a privately-defined resource.
2. The second field is "JSF" (an abbreviation for Jabber Software Foundation, the former name for the [XMPP Standards Foundation \(XSF\)](http://xmpp.org/xsf/) <sup>26</sup>), which identifies the organization that maintains the named item.
3. The third field contains two constructs:
  - a) The public text class is "DTD", which adheres to ISO 8879 Clause 10.2.2.1.
  - b) The public text description is "Instant Messaging with XHTML", which contains but does not begin with the string "XHTML" (as recommended for an XHTML 1.0 Integration Set).

---

<sup>26</sup>The XMPP Standards Foundation (XSF) is an independent, non-profit membership organization that develops open extensions to the IETF's Extensible Messaging and Presence Protocol (XMPP). For further information, see <<http://xmpp.org/xsf/>>.

4. The fourth field is "EN", which identifies the language (English) in which the item is defined.

## 12.2 User Agent Conformance

A user agent that implements this specification MUST conform to Section 3.5 ("XHTML Family User Agent Conformance") of Modularization of XHTML. Many of the requirements defined therein are already met by Jabber clients simply because they already include XML parsers. However, "ignore" has a special meaning in XHTML modularization (different from its meaning in XMPP). Specifically, criteria 4 through 6 of Section 3.5 of Modularization of XHTML state:

1. W3C TEXT: If a user agent encounters an element it does not recognize, it must continue to process the children of that element. If the content is text, the text must be presented to the user.  
XSF COMMENT: This behavior is different from that defined by XMPP Core, and in the context of XHTML-IM implementations applies only to XML elements qualified by the 'http://www.w3.org/1999/xhtml' namespace as defined herein. This criterion MUST be applied to all XHTML 1.0 elements except those explicitly included in XHTML-IM as described in the [XHTML-IM Integration Set](#) and [Recommended Profile](#) sections of this document. Therefore, an XHTML-IM implementation MUST process all XHTML 1.0 child elements of the XHTML-IM <html/> element even if such child elements are not included in the XHTML 1.0 Integration Set defined herein, and MUST present to the recipient the XML character data contained in such child elements.
2. W3C TEXT: If a user agent encounters an attribute it does not recognize, it must ignore the entire attribute specification (i.e., the attribute and its value).  
XSF COMMENT: This criterion MUST be applied to all XHTML 1.0 attributes except those explicitly included in XHTML-IM as described in the [XHTML-IM Integration Set](#) and [Recommended Profile](#) sections of this document. Therefore, an XHTML-IM implementation MUST ignore all attributes of elements qualified by the 'http://www.w3.org/1999/xhtml' namespace if such attributes are not explicitly included in the XHTML 1.0 Integration Set defined herein.
3. W3C TEXT: If a user agent encounters an attribute value it doesn't recognize, it must use the default attribute value.  
XSF COMMENT: Since not one of the attributes included in XHTML-IM has a default value defined for it in XHTML 1.0, in practice this criterion does not apply to XHTML-IM implementations.

### 12.3 XHTML Modularization

For information regarding XHTML modularization in XML schema for the XHTML 1.0 Integration Set defined in this specification, refer to the [Schema Driver](#) section of this document.

### 12.4 W3C Review

The XHTML 1.0 Integration Set defined herein has been reviewed informally by an editor of the XHTML Modularization in XML Schema specification but has not undergone formal review by the W3C. Before the XHTML-IM specification proceeds to a status of Final within the standards process of the XMPP Standards Foundation, the [XMPP Council](#) <sup>27</sup> is encouraged to pursue a formal review through communication with the Hypertext Coordination Group within the W3C.

### 12.5 XHTML Versioning

The W3C is actively working on [XHTML 2.0](#) <sup>28</sup> and may produce additional versions of XHTML in the future. This specification addresses XHTML 1.0 only, but it may be superseded or supplemented in the future by a XMPP Extension Protocol specification that defines methods for encapsulating XHTML 2.0 content in XMPP.

## 13 IANA Considerations

This document requires no interaction with the [Internet Assigned Numbers Authority \(IANA\)](#) <sup>29</sup>.

## 14 XMPP Registrar Considerations

### 14.1 Protocol Namespaces

The [XMPP Registrar](#) <sup>30</sup> includes 'http://jabber.org/protocol/xhtml-im' in its registry of protocol namespaces.

---

<sup>27</sup>The XMPP Council is a technical steering committee, authorized by the XSF Board of Directors and elected by XSF members, that approves of new XMPP Extensions Protocols and oversees the XSF's standards process. For further information, see <<http://xmpp.org/council/>>.

<sup>28</sup>XHTML 2.0 <<http://www.w3.org/TR/xhtml2>>.

<sup>29</sup>The Internet Assigned Numbers Authority (IANA) is the central coordinator for the assignment of unique parameter values for Internet protocols, such as port numbers and URI schemes. For further information, see <<http://www.iana.org/>>.

<sup>30</sup>The XMPP Registrar maintains a list of reserved protocol namespaces as well as registries of parameters used in the context of XMPP extension protocols approved by the XMPP Standards Foundation. For further information, see <<http://xmpp.org/registrar/>>.

## 15 XML Schemas

### 15.1 XHTML-IM Wrapper

The following schema defines the XMPP extension element that serves as a wrapper for XHTML content.

```
<?xml version='1.0' encoding='UTF-8'?>

<xs:schema
  xmlns:xs='http://www.w3.org/2001/XMLSchema'
  xmlns:xhtml='http://www.w3.org/1999/xhtml'
  targetNamespace='http://jabber.org/protocol/xhtml-im'
  xmlns='http://jabber.org/protocol/xhtml-im'
  elementFormDefault='qualified'>

  <xs:import namespace='http://www.w3.org/1999/xhtml'
    schemaLocation='http://www.w3.org/2002/08/xhtml/xhtml1-strict.xsd' />

  <xs:annotation>
    <xs:documentation>

      This schema defines the <html/> element qualified by
      the 'http://jabber.org/protocol/xhtml-im' namespace.
      The only allowable child is a <body/> element qualified
      by the 'http://www.w3.org/1999/xhtml' namespace. Refer
      to the XHTML-IM schema driver for the definition of the
      XHTML 1.0 Integration Set.

      Full documentation of this Integration Set is contained in
      "XEP-0071: _XHTML-IM", a specification published by the
      XMPP Standards Foundation.

      http://www.xmpp.org/extensions/xep-0071.html

    </xs:documentation>
    <xs:documentation source="http://www.xmpp.org/extensions/xep-0071.html"/>
  </xs:annotation>

  <xs:element name='html'>
    <xs:complexType>
      <xs:sequence>
        <xs:element ref='xhtml:body'
          minOccurs='0'

```

```

maxOccurs='unbounded' />
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

## 15.2 XHTML-IM Schema Driver

The following schema defines the modularization schema driver for XHTML-IM.

```

<?xml version='1.0' encoding='UTF-8'?>

<xs:schema
  xmlns:xs='http://www.w3.org/2001/XMLSchema'
  targetNamespace='http://www.w3.org/1999/xhtml'
  xmlns='http://www.w3.org/1999/xhtml'
  elementFormDefault='qualified'>

  <xs:annotation>
    <xs:documentation>

      This is the XML Schema driver for XHTML-IM, an XHTML 1.0
      Integration Set for use in exchanging marked-up instant
      messages between entities that conform to the Extensible
      Messaging and Presence Protocol (XMPP). This Integration
      Set includes a subset of the modules defined for XHTML 1.0
      but does not redefine any existing modules, nor does it
      define any new modules. Specifically, it includes the
      following modules only:

        - Structure
        - Text
        - Hypertext
        - List
        - Image
        - Style Attribute

      The Formal Public Identifier (FPI) for this Integration
      Set is:

        -//JSF//DTD Instant Messaging with XHTML//EN

      Full documentation of this Integration Set is contained in
      "XEP-0071:_XHTML-IM", a specification published by the
      XMPP Standards Foundation, which is available at the
      following URL:
    
```

```

        http://www.xmpp.org/extensions/xep-0071.html

    </xs:documentation>
    <xs:documentation source="http://www.xmpp.org/extensions/xep-0071.
        html"/>
</xs:annotation>

<xs:include
    schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-blkphras
        -1.xsd"/>
<xs:include
    schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-blkstruct
        -1.xsd"/>
<xs:include
    schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-hypertext
        -1.xsd"/>
<xs:include
    schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-image-1.
        xsd"/>
<xs:include
    schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-inlphras
        -1.xsd"/>
<xs:include
    schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-inlstruct
        -1.xsd"/>
<xs:include
    schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-inlstyle
        -1.xsd"/>
<xs:include
    schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-list-1.xsd
        "/>
<xs:include
    schemaLocation="http://www.w3.org/MarkUp/SCHEMA/xhtml-struct-1.
        xsd"/>

</xs:schema>

```

### 15.3 XHTML-IM Content Model

The following schema defines the content model for XHTML-IM.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    targetNamespace="http://www.w3.org/1999/xhtml"
    xmlns="http://www.w3.org/1999/xhtml">

    <xs:annotation>
        <xs:documentation>

```

This is the XML Schema module of named XHTML 1.0 content models for XHTML-IM, an XHTML 1.0 Integration Set for use in exchanging marked-up instant messages between entities that conform to the Extensible Messaging and Presence Protocol (XMPP). This Integration Set includes a subset of the modules defined for XHTML 1.0 but does not redefine any existing modules, nor does it define any new modules. Specifically, it includes the following modules only:

- Structure
- Text
- Hypertext
- List
- Image
- Style Attribute

Therefore XHTML-IM uses the following content models:

Block.mix;	Block-like elements, e.g., paragraphs
Flow.mix;	Any block or inline elements
Inline.mix;	Character-level elements
InlineNoAnchor.class;	Anchor element
InlinePre.mix;	Pre element

XHTML-IM also uses the following Attribute Groups:

```
Core.extra.attrib
I18n.extra.attrib
Common.extra
```

Full documentation of this Integration Set is contained in "XEP-0071: XHTML-IM", a specification published by the XMPP Standards Foundation.

<http://www.xmpp.org/extensions/xep-0071.html>

```
</xs:documentation>
<xs:documentation source="http://www.xmpp.org/extensions/xep-0071.
  html"/>
</xs:annotation>

<!-- BEGIN ATTRIBUTE GROUPS -->

<xs:attributeGroup name="Core.extra.attrib"/>
<xs:attributeGroup name="I18n.extra.attrib"/>
<xs:attributeGroup name="Common.extra">
  <xs:attributeGroup ref="style.attrib"/>
</xs:attributeGroup>
```



```
<!-- END ATTRIBUTE GROUPS -->

<!-- BEGIN HYPERTEXT MODULE "PRIMITIVES" -->

<xs:group name="Anchor.class">
  <xs:sequence>
    <xs:element ref="a"/>
  </xs:sequence>
</xs:group>

<!-- END HYPERTEXT MODULE "PRIMITIVES" -->

<!-- BEGIN IMAGE MODULE "PRIMITIVES" -->

<xs:group name="Image.class">
  <xs:choice>
    <xs:element ref="img"/>
  </xs:choice>
</xs:group>

<!-- END IMAGE MODULE "PRIMITIVES" -->

<!-- BEGIN LIST MODULE "PRIMITIVES" -->

<xs:group name="List.class">
  <xs:choice>
    <xs:element ref="ul"/>
    <xs:element ref="ol"/>
    <xs:element ref="dl"/>
  </xs:choice>
</xs:group>

<!-- END LIST MODULE "PRIMITIVES" -->

<!-- BEGIN TEXT MODULE "PRIMITIVES" -->

<xs:group name="BlkPhras.class">
  <xs:choice>
    <xs:element ref="address"/>
    <xs:element ref="blockquote"/>
    <xs:element ref="pre"/>
  </xs:choice>
</xs:group>

<xs:group name="BlkStruct.class">
  <xs:choice>
    <xs:element ref="div"/>
    <xs:element ref="p"/>
  </xs:choice>
</xs:group>
```

```

    </xs:choice>
</xs:group>

<xs:group name="Heading.class">
  <xs:choice>
    <xs:element ref="h1"/>
    <xs:element ref="h2"/>
    <xs:element ref="h3"/>
    <xs:element ref="h4"/>
    <xs:element ref="h5"/>
    <xs:element ref="h6"/>
  </xs:choice>
</xs:group>

<xs:group name="InlPhras.class">
  <xs:choice>
    <xs:element ref="abbr"/>
    <xs:element ref="acronym"/>
    <xs:element ref="cite"/>
    <xs:element ref="code"/>
    <xs:element ref="dfn"/>
    <xs:element ref="em"/>
    <xs:element ref="kbd"/>
    <xs:element ref="q"/>
    <xs:element ref="samp"/>
    <xs:element ref="strong"/>
    <xs:element ref="var"/>
  </xs:choice>
</xs:group>

<xs:group name="InlStruct.class">
  <xs:choice>
    <xs:element ref="br"/>
    <xs:element ref="span"/>
  </xs:choice>
</xs:group>

<!-- END TEXT MODULE "PRIMITIVES" -->

<!-- BEGIN BLOCK COMBINATIONS -->

<xs:group name="Block.class">
  <xs:choice>
    <xs:group ref="BlkPhras.class"/>
    <xs:group ref="BlkStruct.class"/>
  </xs:choice>
</xs:group>

<!-- END BLOCK COMBINATIONS -->

```

```
<!-- BEGIN INLINE COMBINATIONS -->

<!-- Any inline content -->
<xs:group name="Inline.class">
  <xs:choice>
    <xs:group ref="Anchor.class"/>
    <xs:group ref="Image.class"/>
    <xs:group ref="InlPhras.class"/>
    <xs:group ref="InlStruct.class"/>
  </xs:choice>
</xs:group>

<!-- Inline content contained in a hyperlink -->
<xs:group name="InlNoAnchor.class">
  <xs:choice>
    <xs:group ref="Image.class"/>
    <xs:group ref="InlStruct.class"/>
    <xs:group ref="InlPhras.class"/>
  </xs:choice>
</xs:group>

<!-- END INLINE COMBINATIONS -->

<!-- BEGIN TOP-LEVEL MIXES -->

<xs:group name="Block.mix">
  <xs:choice>
    <xs:group ref="Block.class"/>
    <xs:group ref="Heading.class"/>
    <xs:group ref="List.class"/>
  </xs:choice>
</xs:group>

<xs:group name="Flow.mix">
  <xs:choice>
    <xs:group ref="Block.class"/>
    <xs:group ref="Heading.class"/>
    <xs:group ref="Inline.class"/>
    <xs:group ref="List.class"/>
  </xs:choice>
</xs:group>

<xs:group name="Inline.mix">
  <xs:choice>
    <xs:group ref="Inline.class"/>
  </xs:choice>
</xs:group>
```

```
<xs:group name="InlNoAnchor.mix">
  <xs:choice>
    <xs:group ref="InlNoAnchor.class"/>
  </xs:choice>
</xs:group>

<xs:group name="InlinePre.mix">
  <xs:choice>
    <xs:group ref="Anchor.class"/>
    <xs:group ref="InlPhras.class"/>
    <xs:group ref="InlStruct.class"/>
  </xs:choice>
</xs:group>

<!-- END TOP-LEVEL MIXES -->

</xs:schema>
```

## 16 Acknowledgements

This specification formalizes and extends earlier work by Jeremie Miller and Julian Misig on XHTML formatting of Jabber messages. Many thanks to Shane McCarron for his assistance regarding XHTML modularization and conformance issues. Thanks also to contributors on the discussion list of the XSF's [Standards SIG](#)<sup>31</sup> for their feedback and suggestions.

---

<sup>31</sup>The Standards SIG is a standing Special Interest Group devoted to development of XMPP Extension Protocols. The discussion list of the Standards SIG is the primary venue for discussion of XMPP protocol extensions, as well as for announcements by the XMPP Extensions Editor and XMPP Registrar. To subscribe to the list or view the list archives, visit <http://mail.jabber.org/mailman/listinfo/standards/>.