Network Working Group
Internet Draft
<draft-ietf-html-i18n-01.txt>
Expires 30 March 1996

F. Yergeau
G. Nicol
G. Adams
M. Duerst

25 September 1995

Internationalization of the Hypertext Markup Language

Status of this Memo

Abstract

The Hypertext Markup Language (HTML) is a simple markup language used to create hypertext documents that are platform independent. Initially, the application of HTML on the World Wide Web was seriously restricted by its reliance on the ISO-8859-1 coded character set, which is appropriate only for Western European languages. Despite this restriction, HTML has been widely used with other languages, using other coded character sets or character encodings, through various ad hoc extensions to the language.

This document is meant to address the issue of the internationalization of HTML by extending the specification of HTML 2.0 and giving

additional recommendations for proper internationalisation support. A foremost consideration is to make sure that HTML remains a valid application of SGML, while enabling its use in all languages of the world.

The "text/html; version=2.1" Internet Media Type [RFC1590] and MIME Content Type [RFC1521] is defined by this specification, taken together with the HTML 2.0 specification [HTML-2].

Table of contents

1. Introduction

The Hypertext Markup Language (HTML) is a simple markup language used to create hypertext documents that are platform independent. Initially, the application of HTML on the World Wide Web was seriously restricted by its reliance on the ISO-8859-1 coded character set, which is appropriate only for Western European languages. Despite this restriction, HTML has been widely used with other languages, using other coded character sets or character encodings, through various ad hoc extensions to the language [TAKADA].

This document is meant to address the issue of the

internationalization of HTML by extending the specification of HTML 2.0 and giving additional recommendations for proper internationalisation support. It is in good part based on a paper by one of the authors on multilingualism on the WWW [NICOL]. A foremost consideration is to make sure that HTML remains a valid application of SGML, while enabling its use in all languages of the world.

The specific issues addressed are the SGML document character set to be used for HTML, the proper treatment of the charset parameter associated with the "text/html" content type and the specification of language tags and additional entities.

1.1 Scope

HTML has been in use by the World-Wide Web (WWW) global information initiative since 1990. This specification extends the capabilities of HTML 2.0 (RFC xxx), primarily by removing the restriction to the ISO-8859-1 coded character set [ISO-8859-1]. Together with the HTML 2.0 specification, it defines a new version of HTML to be known as "HTML 2.1".

HTML is an application of ISO Standard 8879:1986, Information Processing Text and Office Systems -- Standard Generalized Markup Language (SGML) [ISO-8879]. The HTML Document Type Definition (DTD) is a formal definition of the HTML syntax in terms of SGML. This specification amends the DTD of HTML 2.0 in order to make it applicable to documents encompassing a character repertoire much larger than that of ISO-8859-1, while still remaining SGML conformant.

Together with the HTML 2.0, specification, this specification also defines HTML as an Internet Media Type [RFC1590] and MIME Content Type [RFC1521] called "text/html", or "text/html; version=2.1". As such, it defines the semantics of the HTML syntax and how that syntax should be interpreted by user agents.

1.2 Conformance

This specification governs the syntax of HTML documents and aspects of the behavior of HTML user agents.

1.2.1 Documents

A document is a conforming HTML document if:

* It is a conforming SGML document, and it conforms to the HTML DTD (see 7.1, "HTML DTD").

* It conforms to the application conventions in this specification. For example, the value of the HREF attribute of the <A> element must conform to the URI syntax.

1.2.2. User agents

An HTML user agent conforms to this specification if:

* It parses the characters of an HTML document into data characters and markup according to SGML [ISO-8879].

  NOTE -- In the interest of robustness and extensibility, there are a number of widely deployed conventions for handling non-conforming documents. See section 4.2.1 of the HTML 2.0 specification [HTML-2], "Undeclared Markup Error Handling" for details.

* It supports at least the ISO-8859-1 character encoding scheme and processes each character in the ISO Latin Alphabet No. 1 as specified in section 6.1 of [HTML-2].

  To ensure interoperability and proper support for at least ISO-8859-1 in an environment where character encoding schemes other than ISO-8859-1 are present, user agents must correctly interpret the charset parameter accompanying an HTML document received from the network.

  Furthermore, conforming user-agents are required to at least parse correctly numeric character references within the range of the Basic Multilingual Plane (BMP) of ISO 10646-1 [ISO-10646].

  NOTE -- To support non-western writing systems, HTML user agents are encouraged to support `UNICODE-1-1' or similar character encoding schemes and as much of the character repertoire of [ISO-10646] as is practical.

* It behaves identically for documents whose parsed token sequences are identical.

  For example, comments and the whitespace in tags disappear during tokenization, and hence they do not influence the behavior of conforming user agents.

* It allows the user to traverse (or at least attempt to traverse, resources permitting) all hyperlinks from <A> elements in an HTML document.

An HTML user agent is a level 2 user agent if, additionally:

* It allows the user to express all form field values specified in an HTML document and to (attempt to) submit the values as requests to information services.

2. The document character set

2.1. Reference processing model

This overview explains the reference processing model used for HTML 2.1, and in particular the SGML concept of a document character set. An actual implementation may widely differ in its internal workings from the model given below, but should behave as described to an outside observer.

Because there are various widely differing encodings of text, SGML does not directly address the question of how characters are encoded e.g. in a file. SGML views the characters as a single set (called a "character repertoire"), and a "code set" that assigns an integer number (known as "character number") to each character in the repertoire. The document character set declaration defines what each of the character numbers represents [GOLD90, p. 451]. In most cases, an SGML DTD and all documents that refer to it have a single document character set, and all markup and data characters are part of this set.

HTML, as an application of SGML, does not directly address the question of how characters are encoded as octets in external representations such as files. This is deferred to mechanisms external to HTML, such as the HTTP protocol, or MIME for electronic mail.

For the HTTP protocol [HTTP], the way characters are encoded is defined by the "charset" parameter[1] of the "Content-Type" field of the header of an HTTP response. For example, to indicate that the transmitted document is encoded in the "JIS" encoding of Japanese [RFC1468], the header will contain the following line:

Content-Type: text/html; charset=ISO-2022-JP

---

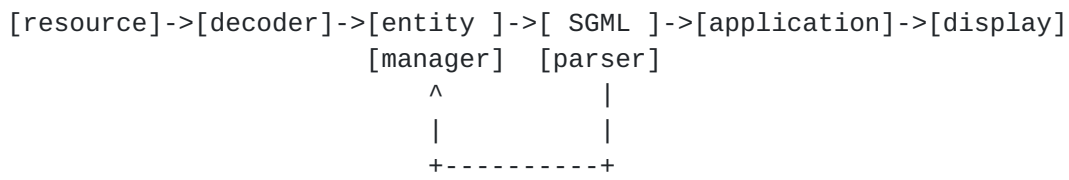
[1] The term "charset" in MIME is used to designate a character encoding, rather than a coded character set as the term may suggest. A character encoding is a mapping (possibly many-to-one) of a sequence of octets to a sequence of characters taken from one or more character repertoires. A coded character set is a mapping between individual bit patterns and individual characters from a single character repertoire.

The default charset parameter in case of the HTTP protocol is ISO-8859-1 (the so-called "Latin-1" for Western European characters). The HTTP protocol also defines a mechanism for the client to specify the character encodings it can accept. Clients and servers are strongly requested to use these mechanisms to assure correct transmission and interpretation of any document. Provisions that can be taken to help correct interpretation, even in cases where a server or client do not yet use these mechanisms, are described in section 6.

Similarly, if HTML documents are transferred by electronic mail, the character encoding is defined by the "charset" parameter of the "Content-Type" MIME header line [RFC1521].

In the case any other way of transferring and storing HTML documents are defined or become popular, it is advised that similar provisions should be made to clearly identify the character encoding used and/or to use a single/default encoding capable of representing the widest range of characters used in an international context.

Whatever the external character encoding actually be, the reference processing model translates it to a representation of the document character set specified in Section 2.2 before processing specific to SGML/HTML. The reference processing model can be depicted as follows:



The decoder is responsible for decoding the external representation of the resource to a representation using the document character set. The entity manager, the parser, and the application deal only with characters of the document character set. A display-oriented part of the application or the display machinery itself may again convert characters represented in the document character set to some other representation more suitable for their purpose. In any case, the entity manager, the parser, and the application, as far as character semantics are concerned, are using the HTML 2.1 document character set only.

An actual implementation may choose, or not, to translate the document into some encoding of the document character set as described above; the behaviour described by this reference processing model can be achieved otherwise. This subject is well out of the scope of this specification, however, and the reader is invited to consult the SGML standard [ISO-8879] or a SGML handbook [BRYAN88] [GOLD90] [VANH90]

[SQ91] for further information.

The most important consequence of this reference processing model is that numeric character references are always resolved to the same characters, whatever the external encoding actually used. For an example, see Section 2.2.

2.2. The HTML 2.1 document character set

The document character set, in the SGML sense, of HTML 2.1 is the Basic Multilingual Plane of ISO 10646:1993 [ISO-10646], also known as UCS-2. This is code-by-code identical with the Unicode standard [UNICODE]. The adoption of this document character set implies a change in the SGML declaration specified in the HTML 2.0 specification (section 9.5 of [HTML-2]). The change amounts to removing the two BASESET specifications and their accompanying DESCSET declarations, replacing them with the following declaration:

```
BASESET "ISO Registration Number 176//CHARSET
         ISO/IEC 10646-1:1993 UCS-2 with implementation level 3
         //ESC 2/5 2/15 4/5"
DESCSET  0   9     UNUSED
         9   2     9
         11  2     UNUSED
         13  1     13
         14  18    UNUSED
         32  95    32
         127 1     UNUSED
         128 32    UNUSED
         160 65374 160
```

Making UCS-2 the document character set does not create non-conformance of any expression, construct or document that is conforming to HTML 2.0. It does make conforming certain constructs that are not admissible in HTML 2.0. One consequence is that data characters outside the repertoire of ISO-8859-1, but within that of UCS-2 become valid SGML characters. Another is that the upper limit of the range of numeric character references is extended from 255 to 65533[2] ; thus, И is a valid reference to a "CYRILLIC CAPITAL LETTER I". [ERCS] is a good source of information on Unicode and SGML, although its scope and technical content differ greatly from this

---

[2] 65533 (FFFD hexadecimal) is the last valid character in UCS-2. 65534 (FFFE hexadecimal) is unassigned and reserved as the byte-swapped version of ZERO WIDTH NON-BREAKING SPACE for byte-sex detection purposes. **65535 (FFFF hexadecimal) is unassigned.**

specification.

ISO 10646-1:1993 is the most encompassing character set currently existing, and there is no other character set that could take its place as the document character set for HTML 2.1. Also, it is expected that with future extensions of ISO 10646, this specification may also be extended. If nevertheless for a specific application there is a need to use characters outside this standard, this should be done by avoiding any conflicts with present or future versions of ISO 10646, i.e. by assigning these characters to a private zone. Also, it should be borne in mind that such a use will be highly unportable; in many cases, it may be better to use inline bitmaps.

2.3. Undisplayable characters

With the document character set being the full ISO 10646 BMP, the possibility that a character cannot be displayed due to lack of appropriate resources (fonts) cannot be avoided. Because there are many different things that can be done in such a case, this document does not recommend any specific behaviour. Depending on the implementation, this may also be handled by the underlaying display system and not the application itself. The following considerations, however, may be of help:

- A clearly visible, but unobtrusive behaviour should be preferred. Some documents may contain many characters that cannot be rendendered, and so showing an alert for each of them is not the right thing to do.

- In case a numeric representation of the missing character is given, its hexadecimal (not decimal) form is to be preferred, because this form is used in character set standards [ERCS].

3. Language tags

Language tags can be used to control rendering of a marked up document in various ways: character disambiguation, in cases where the character encoding is not sufficient to resolve to a specific glyph; quotation marks; hyphenation; ligatures; spacing; voice synthesis; etc. Independently of rendering issues, language markup is useful as content markup for purposes such as classification and searching.

The language attribute, LANG, takes as its value a language tag that identifies a natural language spoken, written, or otherwise conveyed by human beings for communication of information to other human beings. Computer languages are explicitly excluded.

The syntax and registry of HTML language tags is the same as that

defined by RFC 1766 [RFC1766]. In summary, a language tag is composed of one or more parts: A primary language tag and a possibly empty series of subtags:

```
language-tag  = primary-tag *( "-" subtag )
primary-tag   = 1*8ALPHA
subtag        = 1*8ALPHA
```

Whitespace is not allowed within the tag and all tags are case-insensitive. The namespace of language tags is administered by the IANA. Example tags include:

```
en, en-US, en-cockney, i-cherokee, x-pig-latin
```

Two-letter primary-tags are reserved for ISO 639 language abbreviations [ISO-639], and three-letter primary-tags for the language abbreviations of the "Ethnologue" [ETHNO] (the latter is in addition to the requirements of RFC 1766). Any two-letter initial subtag is an ISO 3166 country code [ISO-3166].

In the context of HTML, a language tag is not to be interpreted as a single token, as per RFC 1766, but as a hierarchy. For example, a user agent that adjusts rendering according to language should consider that it has a match when a language tag in a style sheet entry matches the initial portion of the language tag of an element. An exact match should be preferred. This interpretation allows an element marked up as, for instance, "en-US" to trigger styles corresponding to, in order of preference, US-English ("en-US") or 'plain' or 'international' English ("en").

> NOTE -- using the language tag as a hierarchy does not imply that all languages with a common prefix will be understood by those fluent in one or more of those languages; it simply allows the user to request this commonality when it is true for that user.

Since any text can logically be assigned a language, almost all HTML elements admit the LANG attribute. The DTD reflects this. It is also intended that any new element introduced in later versions of HTML will admit the LANG attribute, unless there is a good reason not to do so.

The rendering of elements is meant to be controlled (in part) by the LANG attribute. Specific user preferences set within the browser should override the value of the LANG attribute, which in turn overrides the value specified by the LANG attribute of any enclosing element. If none of these are set, a suitable default, perhaps controlled by the user's locale, should be used to control rendering.

4. Additional entities, attributes and elements

4.1. Full Latin-1 entity set

According to the suggestion of section 14 of [HTML-2], the set of Latin-1 entities is extended to cover the whole right part of ISO-8859-1. The names of the entities are taken from the appendices of [SGML]. A list is provided in section 7.3.1 of this specification.

4.2. Markup for language-dependent presentation

For the correct presentation of text from certain languages (irrespective of formatting issues), some support in the form of additional entities and elements is needed. In particular, bidirectional text (BIDI for short) requires markup in special circumstances where ambiguities as to the directionnality of some characters have to be resolved. Plain text may contain this markup in the form of special-purpose characters; in HTML, these are replaced by SGML markup to be described below.

This markup affects the ability to render BIDI text in a semantically legible fashion. That is, without this special BIDI markup, cases arise which would prevent *any* rendering whatsoever that reflected the basic meaning of the text. It is for this reason that these special characters were added to Unicode (and, thence, to ISO/IEC 10646). If it were possible to do reliable layout and rendering of bidirectionnal text without them, they definitely would not have been included in Unicode.

First, a set of named character entities is added that allows partial support of the Unicode bidirectional algorithm [UNICODE], plus some help with languages requiring contextual analysis for rendering:

```
<!ENTITY zwnj CDATA "&#8204;"--=zero width non-joiner-->
<!ENTITY zwj  CDATA "&#8205;"--=zero width joiner-->
<!ENTITY lrm  CDATA "&#8206;"--=left-to-right mark-->
<!ENTITY rlm  CDATA "&#8207;"--=right-to-left mark-->
```

The first two, zwnj and zwj, are used to force or block joining behavior in contexts which joining would occur but should not or would not occur but should. For example, ARABIC LETTER HEH is used to abbreviate "Hijri" (the Islamic calendrical system); however, the isolated form of HEH looks like the digit five as employed in Arabic script (actually based on Indic digits). In order to prevent one from reading HEH as a final digit five in a year, the initial form of HEH is used. However, there is no following context (i.e., a joining letter) to which the HEH can join. Therefore, the ZWJ is used to

provide that context. In Farsi texts, there are cases where a letter that normally would join a subsequent letter in a cursive connection does not. Here the ZWNJ is used.

The other two, lrm and rlm, are used to disambiguate directionality of directionally neutral characters, e.g., if you have a double quote sitting between an Arabic and a Latin letter, then which direction does the quote resolve to? These characters are like zero width spaces which have a directional property (but no word/line break property).

Next, an attribute called DIR is introduced, restricted to the values LTR and RTL and admitted by most elements. On block-type elements, the DIR attribute indicates the base directionnality of the text in the block; if omitted it is inherited from the parent element. On inline elements, it makes the element start a new embedding level; if omitted the inline element does not start a new embedding level. Embedding is used to handle nested directional runs; a common need for the embedding characters is to handle text that has been pasted from one bidi context to another, and the possibility of multiply embedded pastings. Following is an example of a case where embedding is needed, showing its effect:

> Given the following latin (upper case) and arabic (lower case) letters in backing store with the specified embeddings (LRE is shorthand for <SPAN DIR=LTR>, RLE for <SPAN DIR=RTL> and PDF for </SPAN>):
>
> LRE A B RLE a b LRE C D PDF c d PDF E F PDF
>
> One gets the following rendering (with [] showing the directional transitions):
>
> [A B [d c [C D] b a] E F]
>
> On the other hand, without these characters, e.g., with
>
> A B a b C D c d E F
>
> and a base level of LTR one gets the following rendering:
>
> [A B [b a] C D [d c] E F]
>
> Notice that b,a is on the left and d,c on the right unlike the above case where the embedding levels are used. Without the embedding characters one has at most two levels: a base directional level and a single counterflow directional level.

A directionnal override feature is needed to deal with unusual pieces of text in which directionality cannot be resolved from context in an unambiguous fashion. For example, in part numbers, formulas, telephone numbers, and other similar pieces of text, it is difficult or impossible to derive the directionality of numbers, punctuation, and other neutrals from their context. To this effect, a new element called BDO (BIDI override) is introduced, which requires the DIR attribute to specify whether the override is left-to-right or right-to-left.

A few other additional elements are important to have for proper language-dependent rendering. First, a generic container is needed to carry the LANG and BIDI attributes in cases where no other element is appropriate; the SPAN element is introduced for that purpose.

Short quotations, and in particular the quotation marks surrounding them, are typically rendered differently in different languages and on platforms with different graphic capabilities: "a quotation in English", `another, slightly better one', ,,a quotation in German", << a quotation in French >>. The <Q> element is introduced for that purpose.

Many languages, including English, require superscripts for proper rendering: "the XXth century" should have "th" in superscript. The <SUP> element, and its sibling <SUB>, are introduced to allow proper markup of such text. <SUP> and <SUB> contents are restricted to PCDATA to avoid nesting problems.

Finally, in many languages text justification is much more important than it is in Western languages, and justifies markup. The ALIGN attribute, admitting values of LEFT, RIGHT, CENTER and JUSTIFY, is added to a selection of elements where it makes sense (block-like).

5. Forms

5.1. DTD additions

It is natural to expect input in any language in forms, as they provide one of the only ways of obtaining user input. While this is primarily a UI issue, there are some things that should be specified at the HTML level to guide behavior and promote interoperability.

To ensure interoperability, it is necessary for the user agent (and

the user) to have an indication of the character set(s) that the server providing a form will be able to handle upon submission of the filled-in form. Such an indication is provided by the ACCEPT-CHARSET attribute of the FORM element, modeled on the HTTP Accept-Charset header (see [HTTP]), which contains a space and/or comma delimited list of character sets acceptable to the server. A user agent may want to somehow advise the user of the contents of this attribute, or to restrict his possibility to enter unacceptable characters.

> NOTE -- The list of character sets is to be interpreted as an EXCLUSIVE-OR list; the server announces that it is ready to accept any ONE of these character encoding schemes for each part of a multipart entity.

5.2. Form submission

The HTML 2.0 form submission mechanism, based on the "application/x-www-form-urlencoded" media type, is hopelessly broken with regard to internationalization. In fact, since URLs are restricted to ASCII characters, the mechanism is broken even for ISO-8859-1 text. Section 2.2 of [RFC1738] specifies that octets may be encoded using the "%HH" notation, but text submitted from a form is composed of characters, not octets. Lacking a specification of a character encoding scheme, the "%HH" notation has no meaning.

A partial solution to this sorry state of affairs is to specify a default character encoding scheme to be assumed when the GET method of form submission is used. Specifying UCS-2 would break all existing forms, so the only sensible way is to designate ISO-8859-1. That is, the encoded URL sent to submit a form by the GET method is to be interpreted as a sequence of single-octet characters encoded according to ISO-8859-1, and further encoded according to the scheme of [RFC1738] (the "%HH" notation). This is clearly insufficient, so the GET method of form submission is deprecated and should not be used in future documents, despite the language of section XX of [HTML-2].

A better solution is to add a MIME charset parameter to the Content-Type header sent along with a POST method form submission, with the understanding that the URL encoding of [RFC1738] is applied on top of the specified character encoding, as a kind of implicit Content-Transfer-Encoding. The default ISO-8859-1 is to be implied in the absence of a charset parameter.

The best solution is to use the "multipart/form-data" media type described in [FILE-UPLOAD] with the POST method of form submission. This mechanism encapsulates the value part of each name-value pair in a body-part of a multipart MIME body that is sent as the HTTP entity;

each body part can be labeled with an appropriate Content-Type, including if necessary a charset parameter that specifies the character encoding scheme. The changes to the DTD necessary to support this method of form submission have been incorporated in the DTD included in this specification.

How the user agent determines the encoding of the text entered by the user is outside the scope of this specification.

6. Miscellaneous

Proper interpretation of a text document requires that the character encoding scheme be known. Current HTTP servers, however, do not generally include an appropriate charset parameter with the Content-Type header, even when the encoding scheme is different from the default ISO-8859-1. This is bad behaviour, and as such strongly discouraged, but some preventive measures can be taken to minimize the detrimental effects.

In the case where a document is accessed from a hyperlink in an origin HTML document, a CHARSET attribute is added to the attribute list of elements with link semantics (A and LINK), specifically by adding it to the linkExtraAttributes entity. The value of that attribute is to be considered a hint to the User Agent as to the character encoding scheme used by the ressource pointed to by the hyperlink; it should be the appropriate value of the MIME charset parameter for that ressource.

In any document, it may be wise to include an indication of the encoding scheme like the following, as early as possible within the HEAD of the document:

```
<META HTTP-EQUIV="Content-Type"
 CONTENT="text/html; charset=ISO-2022-JP">
```

This is not foolproof, but will work if the encoding scheme is such that ASCII characters stand for themselves at least until the META element is parsed.

For definiteness, the "charset" parameter received from the source of the document should be considered the most authoritative, followed in order of preference by the contents of a META element such as the above, and finally the CHARSET parameter of the anchor that was followed (if any).

When HTML text is transmitted directly in UCS-2 (charset=UNICODE-1-1), the question of byte order arises: does the high-order byte of each two-byte character come first or second? For

definiteness, this specification recommends that UCS-2 be transmitted in big-endian byte order (high order byte first), which corresponds both to the established network byte order for two-byte quantities and to the Unicode recommendation for serialized text data. Furthermore, to maximize chances of proper interpretation, it is recommended that documents transmitted as UCS-2 always begin with a ZERO-WIDTH NON-BREAKING SPACE character (hexadecimal FEFF) which, when byte-reversed becomes number FFFE, a character guaranteed to be never assigned. Thus, a user-agent receiving an FFFE as the first octets of a text would know that bytes have to be reversed for the remainder of the text.

7. HTML Public Text

7.1. HTML DTD

```
<!--    html-2.1.dtd

        Document Type Definition for the HyperText Markup Language,
        version 2.1 (HTML DTD)

        Last revised: 95/09/25

        Authors: Daniel W. Connolly <connolly@w3.org>
                 Francois Yergeau <yergeau@alis.com>
-->

<!ENTITY % HTML.Version
        "-//IETF//DTD HTML 2.1//EN"

        -- Typical usage:

            <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.1//EN">
            <html>
            ...
            </html>
        --
        >


<!--============ Feature Test Entities ========================-->

<!ENTITY % HTML.Recommended "IGNORE"
        -- Certain features of the language are necessary for
           compatibility with widespread usage, but they may
           compromise the structural integrity of a document.
           This feature test entity enables a more prescriptive
           document type definition that eliminates
```

```
            those features.
         -->

   <![ %HTML.Recommended [
         <!ENTITY % HTML.Deprecated "IGNORE">
   ]]>

   <!ENTITY % HTML.Deprecated "INCLUDE"
         -- Certain features of the language are necessary for
            compatibility with earlier versions of the specification,
            but they tend to be used and implemented inconsistently,
            and their use is deprecated. This feature test entity
            enables a document type definition that eliminates
            these features.
         -->

   <!ENTITY % HTML.Highlighting "INCLUDE"
         -- Use this feature test entity to validate that a
            document uses no highlighting tags, which may be
            ignored on minimal implementations.
         -->

   <!ENTITY % HTML.Forms "INCLUDE"
         -- Use this feature test entity to validate that a document
            contains no forms, which may not be supported in minimal
            implementations
         -->

   <!--============== Imported Names ==============================-->

   <!ENTITY % Content-Type "CDATA"
         -- meaning an internet media type
            (aka MIME content type, as per RFC1521)
         -->

   <!ENTITY % HTTP-Method "GET | POST"
         -- as per HTTP specification, in progress
         -->

   <!--========== DTD "Macros" =====================-->

   <!ENTITY % heading "H1|H2|H3|H4|H5|H6">

   <!ENTITY % list " UL | OL | DIR | MENU " >

   <!ENTITY % attrs -- common attributes for elements --
          "LANG  NAME      #IMPLIED  -- RFC 1766 language tag --
           DIR  (ltr|rtl) #IMPLIED  -- text directionnality --">
```

```
   <!ENTITY % just -- an attribute for text justification --
            "ALIGN  (left|right|center|justify)  #IMPLIED">

   <!--======= Character mnemonic entities =================-->

   <!ENTITY % ISOlat1 PUBLIC
     "ISO 8879-1986//ENTITIES Added Latin 1//EN//HTML">
   %ISOlat1;

   <!--Entities for markup significant characters -->
   <!ENTITY amp CDATA "&#38;"     -- ampersand          -->
   <!ENTITY gt CDATA "&#62;"      -- greater than       -->
   <!ENTITY lt CDATA "&#60;"      -- less than          -->
   <!ENTITY quot CDATA "&#34;"    -- double quote       -->

   <!--Entities for language-dependent presentation (BIDI and contextual
analysis) -->
   <!ENTITY zwnj CDATA "&#8204;"-- zero width non-joiner-->
   <!ENTITY zwj  CDATA "&#8205;"-- zero width joiner-->
   <!ENTITY lrm  CDATA "&#8206;"-- left-to-right mark-->
   <!ENTITY rlm  CDATA "&#8207;"-- right-to-left mark-->

   <!--========= SGML Document Access (SDA) Parameter Entities =====-->

   <!-- HTML 2.0 contains SGML Document Access (SDA) fixed attributes
   in support of easy transformation to the International Committee
   for Accessible Document Design (ICADD) DTD
            "-//EC-USA-CDA/ICADD//DTD ICADD22//EN".
   ICADD applications are designed to support usable access to
   structured information by print-impaired individuals through
   Braille, large print and voice synthesis.  For more information on
   SDA & ICADD:
           - ISO 12083:1993, Annex A.8, Facilities for Braille,
             large print and computer voice
           - ICADD ListServ
             <ICADD%ASUACAD.BITNET@ARIZVM1.ccit.arizona.edu>
           - Usenet news group bit.listserv.easi
           - Recording for the Blind, +1 800 221 4792
   -->

   <!ENTITY % SDAFORM  "SDAFORM  CDATA  #FIXED"
             -- one to one mapping        -->
   <!ENTITY % SDARULE  "SDARULE  CDATA  #FIXED"
             -- context-sensitive mapping -->
   <!ENTITY % SDAPREF  "SDAPREF  CDATA  #FIXED"
             -- generated text prefix     -->
   <!ENTITY % SDASUFF  "SDASUFF  CDATA  #FIXED"
             -- generated text suffix     -->
```

```
<!ENTITY % SDASUSP  "SDASUSP  NAME   #FIXED"
```

```
            -- suspend transform process -->

   <!--========== Text Markup ====================-->

   <![ %HTML.Highlighting [

   <!ENTITY % font " TT | B | I ">

   <!ENTITY % phrase "EM | STRONG | CODE | SAMP | KBD | VAR | CITE">

   <!ENTITY % text "#PCDATA|A|IMG|BR|%phrase|%font|SPAN|Q|BDO|SUP|SUB">

   <!ELEMENT (%font;|%phrase) - - (%text)*>
   <!ATTLIST ( TT | CODE | SAMP | KBD | VAR )
           %attrs;
           %SDAFORM; "Lit"
           >
   <!ATTLIST ( B | STRONG )
           %attrs;
           %SDAFORM; "B"
           >
   <!ATTLIST ( I | EM | CITE )
           %attrs;
           %SDAFORM; "It"
           >

   <!-- <TT>       Typewriter text                        -->
   <!-- <B>        Bold text                              -->
   <!-- <I>        Italic text                            -->
   <!-- <EM>       Emphasized phrase                      -->
   <!-- <STRONG>   Strong emphasis                        -->
   <!-- <CODE>     Source code phrase                     -->
   <!-- <SAMP>     Sample text or characters              -->
   <!-- <KBD>      Keyboard phrase, e.g. user input       -->
   <!-- <VAR>      Variable phrase or substituable        -->
   <!-- <CITE>     Name or title of cited work            -->

   <!ENTITY % pre.content "#PCDATA|A|HR|BR|%font|%phrase|SPAN|BDO">

   ]]>

   <!ENTITY % text "#PCDATA|A|IMG|BR|SPAN|Q|BDO|SUP|SUB">

   <!-- Should the BDO element have an SDAFORM attr.?  Which? -->
   <!ELEMENT BDO - - (%text)+>
   <!ATTLIST BDO
           LANG   NAME      #IMPLIED
           DIR    (ltr|rtl) #REQUIRED
```

```
          >

   <!-- <BDO>     Control bidirectionnal text             -->

   <!ELEMENT BR    - O EMPTY>
   <!ATTLIST BR
          %SDAPREF; "&#RE;"
          >

   <!-- <BR>       Line break      -->

   <!-- Should the SPAN element have an SDAFORM attr.?  Which? -->
   <!ELEMENT SPAN - - (%text)*>
   <!ATTLIST SPAN
          %attrs;
   >

   <!-- <SPAN>     Generic container        -->

   <!ELEMENT Q - - (%text)*>
   <!ATTLIST Q
          %attrs;
          %SDAFORM; "It"  -- to be verified --
          >

   <!-- <Q>        Short quotation                          -->

   <!ELEMENT (SUP|SUB) - - (#PCDATA)>
   <!ATTLIST (SUP|SUB)
          %attrs;
          >

   <!-- <SUP>      Superscript              -->
   <!-- <SUB>      Subscript                -->

   <!--========= Link Markup ======================-->

   <!ENTITY % linkType "NAME">

   <!ENTITY % linkExtraAttributes
          "REL %linkType #IMPLIED
          REV %linkType #IMPLIED
          URN CDATA #IMPLIED
          TITLE CDATA #IMPLIED
          METHODS NAMES #IMPLIED
          CHARSET NAME #IMPLIED
          ">
```

```
   <![ %HTML.Recommended [
           <!ENTITY % A.content   "(%text)*"
           -- <H1><a name="xxx">Heading</a></H1>
                    is preferred to
              <a name="xxx"><H1>Heading</H1></a>
           -->
   ]]>

   <!ENTITY % A.content   "(%heading|%text)*">

   <!ELEMENT A     - - %A.content -(A)>
   <!ATTLIST A
           %attrs;
           HREF CDATA #IMPLIED
           NAME CDATA #IMPLIED
           %linkExtraAttributes;
           %SDAPREF; "<Anchor: #AttList>"
           >
   <!-- <A>                Anchor; source/destination of link      -->
   <!-- <A NAME="...">     Name of this anchor                     -->
   <!-- <A HREF="...">     Address of link destination             -->
   <!-- <A URN="...">      Permanent address of destination        -->
   <!-- <A REL=...>        Relationship to destination             -->
   <!-- <A REV=...>        Relationship of destination to this     -->
   <!-- <A TITLE="...">    Title of destination (advisory)         -->
   <!-- <A METHODS="...">  Operations on destination (advisory)    -->
   <!-- <A CHARSET="...">  Charset of destination (advisory)       -->


   <!--========== Images ==========================-->

   <!ELEMENT IMG    - O EMPTY>
   <!ATTLIST IMG
           %attrs;
           SRC CDATA  #REQUIRED
           ALT CDATA #IMPLIED
           ALIGN (top|middle|bottom) #IMPLIED
           ISMAP (ISMAP) #IMPLIED
           %SDAPREF; "<Fig><?SDATrans Img: #AttList>#AttVal(Alt)</Fig>"
           >

   <!-- <IMG>              Image; icon, glyph or illustration      -->
   <!-- <IMG SRC="...">    Address of image object                 -->
   <!-- <IMG ALT="...">    Textual alternative                     -->
   <!-- <IMG ALIGN=...>    Position relative to text               -->
   <!-- <IMG ISMAP>        Each pixel can be a link                -->

   <!--========== Paragraphs======================-->
```

```
   <!ELEMENT P     - O (%text)*>
   <!ATTLIST P
           %attrs;
           %just;
           %SDAFORM; "Para"
           >

   <!-- <P>        Paragraph       -->


   <!--========== Headings, Titles, Sections ===============-->

   <!ELEMENT HR    - O EMPTY>
   <!ATTLIST HR
           %attrs;
           %just;
           %SDAPREF; "&#RE;&#RE;"
           >

   <!-- <HR>       Horizontal rule -->

   <!ELEMENT ( %heading )  - -  (%text;)*>
   <!ATTLIST H1
           %attrs;
           %just;
           %SDAFORM; "H1"
           >
   <!ATTLIST H2
           %attrs;
           %just;
           %SDAFORM; "H2"
           >
   <!ATTLIST H3
           %attrs;
           %just;
           %SDAFORM; "H3"
           >
   <!ATTLIST H4
           %attrs;
           %just;
           %SDAFORM; "H4"
           >
   <!ATTLIST H5
           %attrs;
           %just;
           %SDAFORM; "H5"
           >
   <!ATTLIST H6
```

```
        %attrs;
        %just;
        %SDAFORM; "H6"
        >

<!-- <H1>       Heading, level 1 -->
<!-- <H2>       Heading, level 2 -->
<!-- <H3>       Heading, level 3 -->
<!-- <H4>       Heading, level 4 -->
<!-- <H5>       Heading, level 5 -->
<!-- <H6>       Heading, level 6 -->


<!--========== Text Flows ======================-->

<![ %HTML.Forms [
        <!ENTITY % block.forms "BLOCKQUOTE | FORM | ISINDEX">
]]>

<!ENTITY % block.forms "BLOCKQUOTE">

<![ %HTML.Deprecated [
        <!ENTITY % preformatted "PRE | XMP | LISTING">
]]>

<!ENTITY % preformatted "PRE">

<!ENTITY % block "P | %list | DL
        | %preformatted
        | %block.forms">

<!ENTITY % flow "(%text|%block)*">

<!ENTITY % pre.content "#PCDATA | A | HR | BR | SPAN | BDO">
<!ELEMENT PRE - - (%pre.content)*>
<!ATTLIST PRE
        %attrs;
        WIDTH NUMBER #implied
        %SDAFORM; "Lit"
        >

<!-- <PRE>              Preformatted text               -->
<!-- <PRE WIDTH=...>    Maximum characters per line     -->

<![ %HTML.Deprecated [

<!ENTITY % literal "CDATA"
        -- historical, non-conforming parsing mode where
```

```
            the only markup signal is the end tag
            in full
          -->

   <!ELEMENT (XMP|LISTING) - -  %literal>
   <!ATTLIST XMP
           %attrs;
           %SDAFORM; "Lit"
           %SDAPREF; "Example:&#RE;"
           >
   <!ATTLIST LISTING
           %attrs;
           %SDAFORM; "Lit"
           %SDAPREF; "Listing:&#RE;"
           >

   <!-- <XMP>              Example section         -->
   <!-- <LISTING>          Computer listing        -->

   <!ELEMENT PLAINTEXT - O %literal>
   <!-- <PLAINTEXT>        Plain text passage      -->

   <!ATTLIST PLAINTEXT
           %attrs;
           %SDAFORM; "Lit"
           >
   ]]>


   <!--========== Lists ==================-->

   <!ELEMENT DL    - -  (DT | DD)+>
   <!ATTLIST DL
           %attrs;
           COMPACT (COMPACT) #IMPLIED
           %SDAFORM; "List"
           %SDAPREF; "Definition List:"
           >

   <!ELEMENT DT    - O (%text)*>
   <!ATTLIST DT
           %attrs;
           %SDAFORM; "Term"
           >

   <!ELEMENT DD    - O %flow>
   <!ATTLIST DD
           %attrs;
```

```
          %SDAFORM; "LItem"
          >

   <!-- <DL>               Definition list, or glossary    -->
   <!-- <DL COMPACT>       Compact style list              -->
   <!-- <DT>               Term in definition list         -->
   <!-- <DD>               Definition of term              -->

   <!ELEMENT (OL|UL) - -  (LI)+>
   <!ATTLIST OL
          %attrs;
          %just;
          COMPACT (COMPACT) #IMPLIED
          %SDAFORM; "List"
          >
   <!ATTLIST UL
          %attrs;
          %just;
          COMPACT (COMPACT) #IMPLIED
          %SDAFORM; "List"
          >
   <!-- <UL>               Unordered list                  -->
   <!-- <UL COMPACT>       Compact list style              -->
   <!-- <OL>               Ordered, or numbered list       -->
   <!-- <OL COMPACT>       Compact list style              -->


   <!ELEMENT (DIR|MENU) - -  (LI)+ -(%block)>
   <!ATTLIST DIR
          %attrs;
          %just;
          COMPACT (COMPACT) #IMPLIED
          %SDAFORM; "List"
          %SDAPREF; "<LHead>Directory</LHead>"
          >
   <!ATTLIST MENU
          %attrs;
          %just;
          COMPACT (COMPACT) #IMPLIED
          %SDAFORM; "List"
          %SDAPREF; "<LHead>Menu</LHead>"
          >

   <!-- <DIR>              Directory list                  -->
   <!-- <DIR COMPACT>      Compact list style              -->
   <!-- <MENU>             Menu list                       -->
   <!-- <MENU COMPACT>     Compact list style              -->
```

```
   <!ELEMENT LI    - O %flow>
   <!ATTLIST LI
           %attrs;
           %just;
           %SDAFORM; "LItem"
           >

   <!-- <LI>               List item                       -->

   <!--========== Document Body ====================-->

   <![ %HTML.Recommended [
           <!ENTITY % body.content "(%heading|%block|HR|ADDRESS|IMG)*"
           -- <h1>Heading</h1>
              <p>Text ...
                   is preferred to
              <h1>Heading</h1>
              Text ...
           -->
   ]]>

   <!ENTITY % body.content "(%heading | %text | %block |
                                  HR | ADDRESS)*">

   <!ELEMENT BODY O O  %body.content>
   <!ATTLIST BODY
           %attrs;
           >

   <!-- <BODY>     Document body   -->

   <!ELEMENT BLOCKQUOTE - - %body.content>
   <!ATTLIST BLOCKQUOTE
           %attrs;
           %just;
           %SDAFORM; "BQ"
           >

   <!-- <BLOCKQUOTE>       Quoted passage  -->

   <!ELEMENT ADDRESS - - (%text|P)*>
   <!ATTLIST  ADDRESS
           %attrs;
           %just;
           %SDAFORM; "Lit"
           %SDAPREF; "Address:&#RE;"
           >
```

```
   <!-- <ADDRESS>  Address, signature, or byline   -->


   <!--======= Forms ====================-->

   <![ %HTML.Forms [

   <!ELEMENT FORM - - %body.content -(FORM) +(INPUT|SELECT|TEXTAREA)>
   <!ATTLIST FORM
           %attrs;
           ACTION CDATA #IMPLIED
           METHOD (%HTTP-Method) GET
           ENCTYPE %Content-Type; "application/x-www-form-urlencoded"
           ACCEPT-CHARSET CDATA #IMPLIED
           %SDAPREF; "<Para>Form:</Para>"
           %SDASUFF; "<Para>Form End.</Para>"
           >

   <!-- <FORM>                     Fill-out or data-entry form     -->
   <!-- <FORM ACTION="...">        Address for completed form      -->
   <!-- <FORM METHOD=...>          Method of submitting form       -->
   <!-- <FORM ENCTYPE="...">       Representation of form data     -->

   <!ENTITY % InputType "(TEXT | PASSWORD | CHECKBOX |
                          RADIO | SUBMIT | RESET |
                          IMAGE | HIDDEN | FILE )">
   <!ELEMENT INPUT - O EMPTY>
   <!ATTLIST INPUT
           %attrs;
           TYPE %InputType TEXT
           NAME CDATA #IMPLIED
           VALUE CDATA #IMPLIED
           SRC CDATA #IMPLIED
           CHECKED (CHECKED) #IMPLIED
           SIZE CDATA #IMPLIED
           MAXLENGTH NUMBER #IMPLIED
           ALIGN (top|middle|bottom) #IMPLIED
           ACCEPT CDATA #IMPLIED --list of content types --
           %SDAPREF; "Input: "
           >

   <!-- <INPUT>                    Form input datum                -->
   <!-- <INPUT TYPE=...>           Type of input interaction       -->
   <!-- <INPUT NAME=...>           Name of form datum              -->
   <!-- <INPUT VALUE="...">        Default/initial/selected value  -->
   <!-- <INPUT SRC="...">          Address of image                -->
   <!-- <INPUT CHECKED>            Initial state is "on"           -->
   <!-- <INPUT SIZE=...>           Field size hint                 -->
```

```
   <!-- <INPUT MAXLENGTH=...>      Data length maximum             -->
   <!-- <INPUT ALIGN=...>          Image alignment                 -->

   <!ELEMENT SELECT - - (OPTION+) -(INPUT|SELECT|TEXTAREA)>
   <!ATTLIST SELECT
           %attrs;
           NAME CDATA #REQUIRED
           SIZE NUMBER #IMPLIED
           MULTIPLE (MULTIPLE) #IMPLIED
           %SDAFORM; "List"
           %SDAPREF;
           "<LHead>Select #AttVal(Multiple)</LHead>"
           >

   <!-- <SELECT>                   Selection of option(s)          -->
   <!-- <SELECT NAME=...>          Name of form datum              -->
   <!-- <SELECT SIZE=...>          Options displayed at a time     -->
   <!-- <SELECT MULTIPLE>          Multiple selections allowed     -->

   <!ELEMENT OPTION - O (#PCDATA)*>
   <!ATTLIST OPTION
           %attrs;
           SELECTED (SELECTED) #IMPLIED
           VALUE CDATA #IMPLIED
           %SDAFORM; "LItem"
           %SDAPREF;
           "Option: #AttVal(Value) #AttVal(Selected)"
           >

   <!-- <OPTION>                   A selection option              -->
   <!-- <OPTION SELECTED>          Initial state                   -->
   <!-- <OPTION VALUE="...">       Form datum value for this option-->

   <!ELEMENT TEXTAREA - - (#PCDATA)* -(INPUT|SELECT|TEXTAREA)>
   <!ATTLIST TEXTAREA
           %attrs;
           NAME CDATA #REQUIRED
           ROWS NUMBER #REQUIRED
           COLS NUMBER #REQUIRED
           %SDAFORM; "Para"
           %SDAPREF; "Input Text -- #AttVal(Name): "
           >

   <!-- <TEXTAREA>            An area for text input -->
   <!-- <TEXTAREA NAME=...>   Name of form datum     -->
   <!-- <TEXTAREA ROWS=...>   Height of area         -->
   <!-- <TEXTAREA COLS=...>   Width of area          -->
```

```
   ]]>


   <!--======= Document Head ======================-->

   <![ %HTML.Recommended [
           <!ENTITY % head.extra "">
   ]]>
   <!ENTITY % head.extra "& NEXTID?">

   <!ENTITY % head.content "TITLE & ISINDEX? & BASE? %head.extra">

   <!ELEMENT HEAD O O  (%head.content) +(META|LINK)>
   <!ATTLIST HEAD
           %attrs;           >

   <!-- <HEAD>     Document head   -->

   <!ELEMENT TITLE - -  (#PCDATA)* -(META|LINK)>
   <!ATTLIST TITLE
           %attrs;
           %SDAFORM; "Ti"    >

   <!-- <TITLE>    Title of document -->

   <!ELEMENT LINK - O EMPTY>
   <!ATTLIST LINK
           %attrs;
           HREF CDATA #REQUIRED
           %linkExtraAttributes;
           %SDAPREF; "Linked to : #AttVal (TITLE) (URN) (HREF)>"    >

   <!-- <LINK>             Link from this document                -->
   <!-- <LINK HREF="..."> Address of link destination            -->
   <!-- <LINK URN="..."> Lasting name of destination             -->
   <!-- <LINK REL=...>     Relationship to destination           -->
   <!-- <LINK REV=...>     Relationship of destination to this   -->
   <!-- <LINK TITLE="..."> Title of destination (advisory)       -->
   <!-- <LINK CHARSET="..."> Charset of destination (advisory)   -->
   <!-- <LINK METHODS="..."> Operations allowed (advisory)       -->

   <!ELEMENT ISINDEX - O EMPTY>
   <!ATTLIST ISINDEX
           %attrs;
           %SDAPREF;
      "<Para>[Document is indexed/searchable.]</Para>">

   <!-- <ISINDEX>          Document is a searchable index         -->
```

```
   <!ELEMENT BASE - O EMPTY>
   <!ATTLIST BASE
           HREF CDATA #REQUIRED     >

   <!-- <BASE>             Base context document                  -->
   <!-- <BASE HREF="..."> Address for this document              -->

   <!ELEMENT NEXTID - O EMPTY>
   <!ATTLIST NEXTID
           N CDATA #REQUIRED     >

   <!-- <NEXTID>           Next ID to use for link name           -->
   <!-- <NEXTID N=...>     Next ID to use for link name           -->

   <!ELEMENT META - O EMPTY>
   <!ATTLIST META
           HTTP-EQUIV  NAME    #IMPLIED
           NAME        NAME    #IMPLIED
           CONTENT     CDATA   #REQUIRED
           >

   <!-- <META>                    Generic Meta-information        -->
   <!-- <META HTTP-EQUIV=...>     HTTP response header name       -->
   <!-- <META NAME=...>           Meta-information name           -->
   <!-- <META CONTENT="...">      Associated information          -->

   <!--======= Document Structure =================-->

   <![ %HTML.Deprecated [
           <!ENTITY % html.content "HEAD, BODY, PLAINTEXT?">
   ]]>
   <!ENTITY % html.content "HEAD, BODY">

   <!ELEMENT HTML O O  (%html.content)>
   <!ENTITY % version.attr "VERSION CDATA #FIXED '%HTML.Version;'">

   <!ATTLIST HTML
           %attrs;
           %version.attr;
           %SDAFORM; "Book"
           >

   <!-- <HTML>                    HTML Document   -->
```

7.2. SGML Declaration for HTML

```
   <!SGML  "ISO 8879:1986"
```

```
   --
           SGML Declaration for HyperText Markup Language version 2.x
           (HTML 2.x).

   --

   CHARSET
            BASESET  "ISO Registration Number 176//CHARSET
                      ISO/IEC 10646-1:1993 UCS-2 with
                      implementation level 3//ESC 2/5 2/15 4/5"
            DESCSET  0   9     UNUSED
                     9   2     9
                     11  2     UNUSED
                     13  1     13
                     14  18    UNUSED
                     32  95    32
                     127 1     UNUSED
                     128 32    UNUSED
                     160 65376 160


   CAPACITY        SGMLREF
                   TOTALCAP        150000
                   GRPCAP          150000
                   ENTCAP          150000

   SCOPE    DOCUMENT
   SYNTAX
            SHUNCHAR CONTROLS 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
                     17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 127
            BASESET  "ISO 646:1983//CHARSET
                      International Reference Version
                      (IRV)//ESC 2/5 4/0"
            DESCSET  0 128 0

            FUNCTION
                     RE            13
                     RS            10
                     SPACE         32
                     TAB SEPCHAR    9

            NAMING   LCNMSTRT ""
                     UCNMSTRT ""
                     LCNMCHAR ".-"
                     UCNMCHAR ".-"
                     NAMECASE GENERAL YES
                              ENTITY  NO
            DELIM    GENERAL  SGMLREF
```

```
                SHORTREF SGMLREF
          NAMES SGMLREF
          QUANTITY SGMLREF
                ATTSPLEN 2100
                LITLEN   1024
                NAMELEN  72    -- somewhat arbitrary; taken from
                                  internet line length conventions --
                PILEN    1024
                TAGLVL   100
                TAGLEN   2100
                GRPGTCNT 150
                GRPCNT   64

  FEATURES
    MINIMIZE
      DATATAG  NO
      OMITTAG  YES
      RANK     NO
      SHORTTAG YES
    LINK
      SIMPLE   NO
      IMPLICIT NO
      EXPLICIT NO
    OTHER
      CONCUR   NO
      SUBDOC   NO
      FORMAL   YES
    APPINFO    "SDA"  -- conforming SGML Document Access application
                      --
  >
```

7.3. Entity sets

7.3.1. ISO Latin 1 Character Entity Set

The following public text lists each of the characters specified in the Added Latin 1 entity set, along with its name, syntax for use, and description. This list is derived from ISO Standard 8879:1986//ENTITIES Added Latin 1//EN. HTML includes the entire entity set, and adds entities for all missing characters in the right part of ISO-8859-1.

```
 <!-- (C) International Organization for Standardization 1986
      Permission to copy in any form is granted for use with
      conforming SGML systems and applications as defined in
      ISO 8879, provided this notice is included in all copies.
   -->
 <!-- Character entity set. Typical invocation:
```

```
      <!ENTITY % ISOlat1 PUBLIC
        "ISO 8879-1986//ENTITIES Added Latin 1//EN//HTML">
      %ISOlat1;
   -->
 <!ENTITY nbsp   CDATA " " -- no-break space -->
 <!ENTITY iexcl  CDATA "&#161;" -- inverted exclamation mark -->
 <!ENTITY cent   CDATA "&#162;" -- cent sign -->
 <!ENTITY pound  CDATA "&#163;" -- pound sterling sign -->
 <!ENTITY curren CDATA "&#164;" -- general currency sign -->
 <!ENTITY yen    CDATA "&#165;" -- yen sign -->
 <!ENTITY brvbar CDATA "&#166;" -- broken (vertical) bar -->
 <!ENTITY sect   CDATA "&#167;" -- section sign -->
 <!ENTITY uml    CDATA "&#168;" -- umlaut (dieresis) -->
 <!ENTITY copy   CDATA "&#169;" -- copyright sign -->
 <!ENTITY ordf   CDATA "&#170;" -- ordinal indicator, feminine -->
 <!ENTITY laquo  CDATA "&#171;" -- angle quotation mark, left -->
 <!ENTITY not    CDATA "&#172;" -- not sign -->
 <!ENTITY shy    CDATA "&#173;" -- soft hyphen -->
 <!ENTITY reg    CDATA "&#174;" -- registered sign -->
 <!ENTITY macr   CDATA "&#175;" -- macron -->
 <!ENTITY deg    CDATA "&#176;" -- degree sign -->
 <!ENTITY plusmn CDATA "&#177;" -- plus-or-minus sign -->
 <!ENTITY sup2   CDATA "&#178;" -- superscript two -->
 <!ENTITY sup3   CDATA "&#179;" -- superscript three -->
 <!ENTITY acute  CDATA "&#180;" -- acute accent -->
 <!ENTITY micro  CDATA "&#181;" -- micro sign -->
 <!ENTITY para   CDATA "&#182;" -- pilcrow (paragraph sign) -->
 <!ENTITY middot CDATA "&#183;" -- middle dot -->
 <!ENTITY cedil  CDATA "&#184;" -- cedilla -->
 <!ENTITY sup1   CDATA "&#185;" -- superscript one -->
 <!ENTITY ordm   CDATA "&#186;" -- ordinal indicator, masculine -->
 <!ENTITY raquo  CDATA "&#187;" -- angle quotation mark, right -->
 <!ENTITY frac14 CDATA "&#188;" -- fraction one-quarter -->
 <!ENTITY frac12 CDATA "&#189;" -- fraction one-half -->
 <!ENTITY frac34 CDATA "&#190;" -- fraction three-quarters -->
 <!ENTITY iquest CDATA "&#191;" -- inverted question mark -->
 <!ENTITY Agrave CDATA "&#192;" -- capital A, grave accent -->
 <!ENTITY Aacute CDATA "&#193;" -- capital A, acute accent -->
 <!ENTITY Acirc  CDATA "&#194;" -- capital A, circumflex accent -->
 <!ENTITY Atilde CDATA "&#195;" -- capital A, tilde -->
 <!ENTITY Auml   CDATA "&#196;" -- capital A, dieresis or umlaut mark -->
 <!ENTITY Aring  CDATA "&#197;" -- capital A, ring -->
 <!ENTITY AElig  CDATA "&#198;" -- capital AE diphthong (ligature) -->
 <!ENTITY Ccedil CDATA "&#199;" -- capital C, cedilla -->
 <!ENTITY Egrave CDATA "&#200;" -- capital E, grave accent -->
 <!ENTITY Eacute CDATA "&#201;" -- capital E, acute accent -->
 <!ENTITY Ecirc  CDATA "&#202;" -- capital E, circumflex accent -->
 <!ENTITY Euml   CDATA "&#203;" -- capital E, dieresis or umlaut mark -->
```

```
<!ENTITY Igrave CDATA "&#204;" -- capital I, grave accent -->
<!ENTITY Iacute CDATA "&#205;" -- capital I, acute accent -->
<!ENTITY Icirc  CDATA "&#206;" -- capital I, circumflex accent -->
<!ENTITY Iuml   CDATA "&#207;" -- capital I, dieresis or umlaut mark -->
<!ENTITY ETH    CDATA "&#208;" -- capital Eth, Icelandic -->
<!ENTITY Ntilde CDATA "&#209;" -- capital N, tilde -->
<!ENTITY Ograve CDATA "&#210;" -- capital O, grave accent -->
<!ENTITY Oacute CDATA "&#211;" -- capital O, acute accent -->
<!ENTITY Ocirc  CDATA "&#212;" -- capital O, circumflex accent -->
<!ENTITY Otilde CDATA "&#213;" -- capital O, tilde -->
<!ENTITY Ouml   CDATA "&#214;" -- capital O, dieresis or umlaut mark -->
<!ENTITY times  CDATA "&#215;" -- multiply sign -->
<!ENTITY Oslash CDATA "&#216;" -- capital O, slash -->
<!ENTITY Ugrave CDATA "&#217;" -- capital U, grave accent -->
<!ENTITY Uacute CDATA "&#218;" -- capital U, acute accent -->
<!ENTITY Ucirc  CDATA "&#219;" -- capital U, circumflex accent -->
<!ENTITY Uuml   CDATA "&#220;" -- capital U, dieresis or umlaut mark -->
<!ENTITY Yacute CDATA "&#221;" -- capital Y, acute accent -->
<!ENTITY THORN  CDATA "&#222;" -- capital Thorn, Icelandic -->
<!ENTITY szlig  CDATA "&#223;" -- small sharp s, German (sz ligature) -->
<!ENTITY agrave CDATA "&#224;" -- small a, grave accent -->
<!ENTITY aacute CDATA "&#225;" -- small a, acute accent -->
<!ENTITY acirc  CDATA "&#226;" -- small a, circumflex accent -->
<!ENTITY atilde CDATA "&#227;" -- small a, tilde -->
<!ENTITY auml   CDATA "&#228;" -- small a, dieresis or umlaut mark -->
<!ENTITY aring  CDATA "&#229;" -- small a, ring -->
<!ENTITY aelig  CDATA "&#230;" -- small ae diphthong (ligature) -->
<!ENTITY ccedil CDATA "&#231;" -- small c, cedilla -->
<!ENTITY egrave CDATA "&#232;" -- small e, grave accent -->
<!ENTITY eacute CDATA "&#233;" -- small e, acute accent -->
<!ENTITY ecirc  CDATA "&#234;" -- small e, circumflex accent -->
<!ENTITY euml   CDATA "&#235;" -- small e, dieresis or umlaut mark -->
<!ENTITY igrave CDATA "&#236;" -- small i, grave accent -->
<!ENTITY iacute CDATA "&#237;" -- small i, acute accent -->
<!ENTITY icirc  CDATA "&#238;" -- small i, circumflex accent -->
<!ENTITY iuml   CDATA "&#239;" -- small i, dieresis or umlaut mark -->
<!ENTITY eth    CDATA "&#240;" -- small eth, Icelandic -->
<!ENTITY ntilde CDATA "&#241;" -- small n, tilde -->
<!ENTITY ograve CDATA "&#242;" -- small o, grave accent -->
<!ENTITY oacute CDATA "&#243;" -- small o, acute accent -->
<!ENTITY ocirc  CDATA "&#244;" -- small o, circumflex accent -->
<!ENTITY otilde CDATA "&#245;" -- small o, tilde -->
<!ENTITY ouml   CDATA "&#246;" -- small o, dieresis or umlaut mark -->
<!ENTITY divide CDATA "&#247;" -- divide sign -->
<!ENTITY oslash CDATA "&#248;" -- small o, slash -->
<!ENTITY ugrave CDATA "&#249;" -- small u, grave accent -->
<!ENTITY uacute CDATA "&#250;" -- small u, acute accent -->
<!ENTITY ucirc  CDATA "&#251;" -- small u, circumflex accent -->
```

```
<!ENTITY uuml   CDATA "&#252;" -- small u, dieresis or umlaut mark -->
<!ENTITY yacute CDATA "&#253;" -- small y, acute accent -->
<!ENTITY thorn  CDATA "&#254;" -- small thorn, Icelandic -->
<!ENTITY yuml   CDATA "&#255;" -- small y, dieresis or umlaut mark -->
```

Bibliography

[BRYAN88] M. Bryan, "SGML -- An Author's Guide to the Standard Generalized Markup Language", Addison-Wesley, Reading, 1988.

[ERCS] Extended Reference Concrete Syntax for SGML. <http://www.sgmlopen.org/sgml/docs/ercs/ercs-home.html>

[ETHNO] "Ethnologue, Languages of the World", 12th Edition, Barbara F. Grimes editor, Summer Institute of Linguistics, Dallas, 1992.

[FILE-UPLOAD] E. Nebel and L. Masinter, "Form-based File Upload in HTML", Work in progress (draft-ietf-html-fileupload-03.txt), Xerox Corporation, August 1995.

[GOLD90] C. F. Goldfarb, "The SGML Handbook", Y. Rubinsky, Ed., Oxford University Press, 1990.

[HTML-2] T. Berners-Lee and D. Connolly, "Hypertext Markup Language - 2.0", Work in progress (draft-ietf-html-spec-05.txt), MIT/W3C, August 1995.

[HTTP] T. Berners-Lee, R. T. Fielding, and H. Frystyk Nielsen, "Hypertext Transfer Protocol - HTTP/1.0", Work in progress (draft-ietf-http-v10-spec-00.ps), MIT, UC Irvine, CERN, March 1995.

[ISO-639] ISO 639:1988. Codes pour la reprsentation des noms de langue. Technical content in <http://www.sil.org/sgml/iso639a.html>

[ISO-1000] ISO 1000:1992. Units SI et recommandations pour l'emploi de leurs multiples et de certaines autres units.

[ISO-3166] ISO 3166:1993. Codes pour la reprsentation des noms de pays.

[ISO-4217] ISO 4217:1990. Codes pour la reprsentation des

monnaies et types des fonds.

[ISO-8601] ISO 8601:1988. lments de donnes et formats d'change -- change d'information -- Reprsentation de la date et de l'heure.

[ISO-8859-1] ISO 8859-1:1987. International Standard -- Information Processing -- 8-bit Single-Byte Coded Graphic Character Sets -- Part 1: Latin Alphabet No. 1.

[ISO-8879] ISO 8879:1986. International Standard -- Information Processing -- Text and Office Systems -- Standard Generalized Markup Language (SGML).

[ISO-10646] ISO/IEC 10646-1:1993. International Standard -- Information technology -- Universal Multiple-Octet Coded Character Set (UCS) -- Part 1: Architecture and Basic Multilingual Plane.

[NICOL] G.T. Nicol, "The Multilingual World Wide Web", Electronic Book Technologies, 1995, <http://www.ebt.com/docs/multling.html>

[RFC1468] J. Murai, M. Crispin and E. van der Poel, "Japanese Character Encoding for Internet Messages", RFC 1468, Keio University, Panda Programming, June 1993.

[RFC1521] N. Borenstein and N. Freed, "MIME (Multipurpose Internet Mail Extensions) Part One: Mechanisms for Specifying and Describing the Format of Internet Message Bodies", RFC 1521, Bellcore, Innosoft, September 1993.

[RFC1590] J. Postel, "Media Type Registration Procedure", RFC 1590, USC/ISI, March 1994.

[RFC1738] T. Berners-Lee, L. Masinter, and M. McCahill, "Uniform Resource Locators (URL)", RFC 1738, CERN, Xerox PARC, University of Minnesota, October 1994.

[RFC1766] H. Alverstrand, "Tags for the Identification of Languages", RFC 1766, UNINETT, March 1995.

[SQ91] SoftQuad, "The SGML Primer", 3rd ed., SoftQuad Inc., 1991.

[TAKADA] Toshihiro Takada, "Multilingual Information Exchange through the World-Wide Web", Computer Networks and ISDN Systems, Vol. 27, No. 2, Nov. 1994 , p. 235-241.

[TEI] TEI Guidelines for Electronic Text Encoding and Interchange. <http://etext.virgina.edu/TEI.html>

[UNICODE] The Unicode Consortium, "The Unicode Standard -- Worldwide Character Encoding -- Version 1.0", Addison-Wesley, Volume 1, 1991, Volume 2, 1992. The BIDI algorithm is in appendix A of volume 1, with corrections in appendix D of volume 2.

[VANH90] E. van Hervijnen, "Practical SGML", Kluwer Academicq Publishers Group, Norwell and Dordrecht, 1990.

Authors' Addresses

Franois Yergeau
Alis Technologies
3410, rue Griffith
Montral QC H4T 1A7
Canada

Tel: +1 (514) 738-9171
Fax: +1 (514) 342-0318
EMail: yergeau@alis.ca

Gavin Thomas Nicol
Electronic Book Technologies, Japan
1-29-9 Tsurumaki,
Setagaya-ku,
Tokyo
Japan

Tel + Fax: +81-3-3706-7351
EMail: gtn@ebt.com, gtn@twics.co.jp

Glenn Adams
Stonehand
118 Magazine Street
Cambridge, MA 02139
U.S.A.

Tel: +1 (617) 864-5524
Fax: +1 (617) 864-4965
EMail: glenn@stonehand.com

Martin J. Duerst

Multimedia-Laboratory
Departement of Computer Science
University of Zurich
Winterthurerstrasse 190
CH-8057 Zurich
Switzerland

Tel: +41 1 257 43 16
Fax: +41 1 363 00 35
E-mail: mduerst@ifi.unizh.ch