The IXIN Chronicles

Thien-Thi Nguyen

1 Specification

In the following description, `???` indicates points that are not very firm in the author's mind.

1.1 overall

Overall, the format is a series of text blocks. All SXML trees (none of which appear at the top-level, by the way) are *third-normal form*: with abutting character-data sequences[1] coalesced and attributes explicit in the tree's second position (i.e., CADR). Valid SXML elements and attributes, except for attribute `ixcc` (see ⟨undefined⟩ [no ixcc in DTD], page ⟨undefined⟩), are defined in the DTD with "public id":

```
-//GNU//DTD TexinfoML V5.0//EN
```

Additionally, the SXML leaf strings have the following context-sensitive properties, all related to "whitespace normalization".

trimmed whitespace

> The initial and final leaf strings of the following elements **should not** begin with newline and end with whitespace, respectively.

example	smallexample	verbatim
lisp	smalllisp	quotation
display	smalldisplay	para
format	smallformat	pre
flushleft	flushright	cartouche

> [**???** Any others? Any other way to specify this list?]
> Internal newlines, or newlines at the end of other elements are OK.

s/newline/space/ in `para`

> Leaf strings of the element `para`, unless exempted by the `xml:space` attribute of some interevening parent element having value 'preserve', **should** have newline (U+0A) replaced with space (U+20). For example:

```
(para (@)
      (b (@) "foo\n")
      "bar"
      (example (@ (xml:space "preserve"))
               "zow\n"
               (r (@) "oop\n")
               "ack\n"))
```

> Here, '`\n`' represents a newline character. Only the one after '`foo`' is replaced with space to make the SXML well-formed:

---

[1] Character entities should conform to the W3C policy of using XML "Normal Form C", as per http://www.w3.org/TR/xml-entity-names/. For example, this text: "mt ̸!" should be OK. (The '!' is EXCLAMATION MARK (U+21) combined/composed with COMBINING LONG SOLIDUS OVERLAY (U+0338).) If you are reading this in a PDF file, you might very well see only "mt !" — hopefully that can be fixed in Texinfo sooner or later.

```
(para (@)
      (b (@) "foo ")
      "bar"
      (example (@ (xml:space "preserve"))
               "zow\n"
               (r (@) "oop\n")
               "ack\n"))
```

The others are protected by the `xml:space` attribute of element `example`.

1.2 first line

The first line looks like:

```
ixin VERSION; -*- coding: CODING -*-
```

version   The format version number as a character string. For now: '1' (U+31).

coding   The coding system used for this file. It should normally be `utf-8`. See Section "Recognize Coding" in *The GNU Emacs Reference Manual*.

The rest of the line is exactly as above, with only one space between tokens, and a ':' (colon) as part of the third token.

1.3 meta

The list:

```
(XID VARS SETTINGS COPYING TITLEPAGE TOC)
```

that comprises all the information prior to the first `@node`.

1.3.1 xid

The list:

```
(FILENAME LANG TITLE INVITATIONS)
```

These are from the `@setfilename`, `@documentlanguage`, `@settitle`, `@dircategory` and `@direntry` commands, respectively. "XID" stands for "external identifiers".

filename   A string, or the symbol - (hyphen).

lang   A symbol or a list of two symbols. If `@documentlanguage` is unspecified, defaults to (`en US`).

title   A string, or a renderable sequence, or the symbol - (hyphen).

invitations
    If there are no `@dircategory` or `@direntry` commands, the symbol - (hyphen). Otherwise, a list of the form:

    ```
    (CATEGORY [ENTRY...])
    ```

    where *category* is a single-line string, or - (hyphen); and each *entry* has the form:

    ```
    (TITLE NODE [DESCRIPTION...])
    ```

title   A single-line string.

node   A single-line string of the form:

        `(IXIN-FILE-NAME)[NODE-NAME]`

That is, open-paren, file name, close-paren and the rest, if any, the name of the node. A space may appear between the close-paren and the node name.

description
        Either a string or a renderable sequence.