National Information Standards Organization
File Specifications for the Digital Talking Book

Draft -- Version 3.8

February 1, 2001

Copyright © 2001 by National Information Standards Organization

Table of Contents

Foreword

(This foreword is not a part of the American National Standard for Digital Talking Books... . It is included for information only.)

This standard presents the file specifications for digital talking books (DTBs) for blind, visually impaired, physically handicapped, or otherwise print-disabled readers. For many years, "talking books" have been made available to print-disabled readers on analog media phonograph records and audio cassettes. Those media served their users well in providing human-speech recordings of a wide array of print material in increasingly robust and cost-effective formats. However, analog media are limited in several respects when compared to a print book. First, they are by their nature linear presentations, which while suitable for novels, leaves much to be desired when reading reference works, textbooks, magazines, and other materials which are often accessed randomly. Digital media offer readers the ability to move around a book or magazine as freely as (and more efficiently than) a sighted reader flips through a print book. Second, analog recordings do not allow users to interact with the book, placing bookmarks, highlighting material, and so forth. A DTB offers this capability, storing the bookmarks and highlights separate from, but associated with, the DTB itself. Third, talking book users have long complained that they did not have access to the spelling of the words their heard. As will be explained below, some DTBs will include a file containing the full text of the work, synchronized with the audio presentation, thereby allowing readers to locate specific words and hear them spelled. Finally, analog audio offers readers only one version of the document. If, for example, a book contains footnotes, they are either read where referenced, which burdens the casual reader with unwanted interruptions, or grouped at a location out of the flow of the text, making it difficult for interested readers to access them. A DTB allows the user to easily skip over or read footnotes. So the Digital Talking Book offers the print-disabled user a significantly enhanced reading experience -- one that is much closer to that of the sighted reader using a print book. This standard describes the various files that make up a DTB and specifies how each must be formatted.

DTBs go far beyond the limits imposed on analog audio books because they can include not just the audio rendition of the work, but the full text file and images as well. Because the text file is synchronized with the audio file, a DTB offers multiple sensory inputs to readers, a great benefit to learning-disabled readers, for example. Some visually impaired readers may choose to listen to most of the book, but find that inspecting the images provides information not available in the narrative flow. Others may opt to skip the audio presentation altogether and instead view the text file via screen-enlarging software. Braille readers may prefer to read some or all of the document via a refreshable braille display connected to their DTB player and accessing the text file.

Digital Talking Books are not tied to a single distribution medium. CD-ROMs will be used first but DTBs will be portable to any digital distribution medium capable of handling the large files associated with digital audio recordings. Regardless of how a DTB is distributed, however, it must be in the context of a digital rights management system whose functional requirements this standard describes.

The initiative behind this document grew from a desire to standardize DTB file structures, in the hope that it might prevent a recurrence of the multiple formats currently used for talking books throughout the world. This document benefitted greatly from the work of the DAISY Consortium, whose members had broken much of the ground covered in this standard and who contributed enormously to the solution of the many problems encountered.

NISO Voting Members

NISO Board of Directors

Standards Committee AQ

Standards Committee AQ on Digital Talking Books had the following members at the time this standard was approved:

- Mr. Donald J. Breda
  American Council of the Blind
- Mr. George Brummell

Blinded Veterans Association

- Mr. John Bryant
  National Library Service for the Blind and Physically Handicapped
  Library of Congress

- Mr. Glen Cavanaugh
  Telex Communications, Inc.

- Mr. Curtis Chong
  World Blind Union

- Mr. Thomas Kjellberg Christensen
  DAISY Consortium
  The Danish National Library for the Blind

- Mr. John Cookson
  National Library Service for the Blind and Physically Handicapped
  Library of Congress

- Mr. Frank Kurt Cylke
  National Library Service for the Blind and Physically Handicapped
  Library of Congress

- Mr. Jack Decker
  American Printing House for the Blind

- Dr. Judith Dixon
  National Library Service for the Blind and Physically Handicapped
  Library of Congress

- Mr. Jim Dust
  Telex Communications, Inc.

- Dr. Michael Gosse
  National Federation of the Blind

- Mr. Luis Gutierrez
  American Foundation for the Blind

- Mr. John Hedges
  American Printing House for the Blind

- Mr. Mark Hakkinen
  isSound, Inc.

- Ms. Vivian Juig
  The Hadley School for the Blind

- Ms. Rosemary Kavanagh
  Canadian National Institute for the Blind

- Mr. George Kerscher
  Recording for the Blind and Dyslexic and DAISY Consortium

- Mr. Wells "Brad" Kormann
  National Library Service for the Blind and Physically Handicapped
  Library of Congress

- Ms. Kathie Korpolinski
  Recording for the Blind and Dyslexic

- Dominic Labbé
  VisuAide, Inc.

- Ms. Mary-Frances Laughton
  Assistive Devices Industry Office
  Industry Canada

- Mr. Thomas McLaughlin

National Library Service for the Blind and Physically Handicapped
Library of Congress

- Mr. Michael Moodie, Chair
  National Library Service for the Blind and Physically Handicapped
  Library of Congress

- Ms. Freddie Peaco
  National Library Service for the Blind and Physically Handicapped
  Library of Congress

- Mr. Gilles Pepin
  VisuAide, Inc.

- Mr. Lloyd Rasmussen
  National Library Service for the Blind and Physically Handicapped
  Library of Congress

- Ms. Janina Sajka
  American Foundation for the Blind

- Mr. Rudy Savage
  Talking Book Publishers, Inc.

- Mr. Larry Skutchan
  American Printing House for the Blind

- Ms. Linda Stetson
  Association of Specialized and Cooperative Library Agencies American Library Association

- Mr. George Stockton
  National Library Service for the Blind and Physically Handicapped
  Library of Congress

- Ms. Karen Taylor
  Canadian National Institute for the Blind

[Contents](#)

Acknowledgements

[Contents](#)

1. General Information

(This section is informative)

1.1 Purpose and Scope of Standard

This standard establishes the file specifications for digital talking books (DTBs) for blind, visually impaired, physically handicapped, or otherwise print-disabled readers. Its purpose is to ensure interoperability across service organizations and vendors providing content and playback systems to the target population.

This standard provides specifications applicable to all aspects of digital talking book production and rendering, including authoring tools for DTBs, hardware- or software-based playback devices, and compliance-testing software.

Contents

1.2 Definitions

The following acronyms and terms are used in this standard as defined below. In the following definitions and throughout the standard, bracketed items correspond to entries in section 17, "References to Other Specifications/Documents," where the full URL is provided for each reference.

Accessible

> With respect to implementations, *accessible* refers to the design and functionality of the playback system where all features are usable by the target population.

CSS

> Cascading Style Sheets [CSS] is a mechanism for adding style (e.g. fonts, colors, spacing, formatting) to HTML or XML documents.

DRM

> *Digital Rights Management* is a system of tools and processes that protect intellectual property when it is encoded and distributed in digital form.

DTB

> The *Digital Talking Book* content data set that complies with the specifications in this standard.

DTD

> The *Document Type Definition* file contains machine-readable rules that define allowable XML markup for a particular application.

DTBook3

> DTBook3 is a unique DTD file (dtbook3.dtd) that defines the XML markup for the text content of a DTB.

Fragment Identifier

> A means to address a named place in a document. For reference within the current document, the reference part is to a named target, and begins with "#".See URI for addressing into another document.

Global navigation

> Efficient movement to user-selected portions of a document, with that movement enabled by the NCX. Navigation targets may be headings representing the hierarchical structure of the document or specific points such as pages, notes, sidebars, etc.

Guide

> A component of the Package File, the Guide lists the key structural features of a DTB, such as the table of contents, introduction, bibliography, etc. to enable playback devices to provide convenient access to them.

IMPLIED

> When used in definitions of attributes, means the attribute is optional, as opposed to REQUIRED.

Informative

> An explanatory part of this standard. Contrast with *normative*.

Local navigation

> Movement within a document at a granularity finer than that provided by the NCX. For example, navigation by

paragraph or sentence, or within a table or nested list. Precise local navigation is controlled by the text file; the granularity is limited by the degree to which the text file has been marked up. Time-based movement through a document (similar to fast-forward and rewind on an analog cassette) may also be implemented.

Manifest

A component of the Package File, the Manifest lists all files included in the DTB.

May

In this standard, the word "may" is to be interpreted as an optional feature that is not required but can be provided.

Must

In this standard, the word "must" is to be interpreted as a mandatory requirement on the content or implementation. The term "shall" has the same definition as "must".

Normative

A portion of the standard that supplies precise specifications rather than background or explanation. Contrast with *informative*. Notes within a normative section may be informative.

NCX

The *Navigation Control file for XML applications* (NCX) provides the reader efficient and flexible access to the hierarchical structure of a DTB as well as direct access to selected elements such as page numbers, notes, figures, etc.

OEBF (Open eBook Forum)

An organization formed to create and maintain standards and promote the successful adoption of electronic books. The *Open eBook Publication Structure Version 1.0.1* OEBPS provides a specification for representing the content of a book when it is converted from print to electronic form. This DTB standard utilizes a subset (the Package File) of that specification.

OPF

See "Package File."

Package File

The *Open eBook Forum Package File* (OPF) is an XML file conforming to the oebpkg1.dtd that contains administrative information about the DTB, the files that comprise it, and how these files interrelate.

Playback

With regard to implementations, playback refers to the methods used to render the DTB content. Playback may include audio, braille, large print, and synthetic speech as appropriate for the content and as supported by the playback system.

Player

The hardware/software platform which renders the contents of a DTB to a user. Synonymous with "Playback System."

Reader

The person reading the digital talking book. Synonymous with "user."

REQUIRED

When used in definitions of attributes, means the attribute is required, as opposed to IMPLIED.

SMIL

The Synchronized Multimedia Integration Language [SMIL] is a draft W3C specification (SMIL 2.0) utilized in this standard to control the synchronized presentation of content in multiple media.

Shall

See "Must."

Should

With respect to implementation, the word "should" is to be interpreted as an implementation recommendation, but not a requirement. With respect to content, the word "should" is to be interpreted as recommended programming practice for content.

Spine

A component of the Package File, the Spine lists the SMIL files included in the DTB in default reading order.

Target population

The target population consists of blind, visually impaired, physically handicapped and otherwise print-disabled readers.

TSM

Time-scale modification (TSM) is variable playback rate (both slower and faster than real time) while maintaining constant pitch.

User

See "Reader."

Text File

The content of the subject document in a character set specified by ISO/IEC 10646 to which XML markup has been applied.

URI

Uniform Resource Identifier, the means to uniquely identify a document and reference it. A URI may include a fragment identifier suffix beginning with "#" that matches some named anchor in the target document.

XML

A file conforming to the Extensible Markup Language 1.0 [XML] specification.

XSL

A file conforming to the Extensible Style Language [XSL] specification.

Contents

1.3 Strategy

This standard is based primarily on a variety of widely used standards and specifications, including several from the World Wide Web Consortium and the Open eBook Forum. Wherever applicable and appropriate standards or specifications existed they were used. The use of these specifications and technologies is intended to promote a fast and consistent adoption of this standard for the target population, while encouraging its extension into mainstream use.

Contents

1.4 Relationship to Other Specifications

This standard is based on the specific versions of the standards and specifications referenced herein, which are used as defined, except as noted by this document. Any refinement or replacement of a referenced specification by a newer or different version is not directly applicable to this standard. Conformance to this standard is based on the versions of the standards and specifications in effect at the time of this writing.

Contents

1.5 Patent Rights

It is possible that compliance with this standard may require the use of one or more inventions covered by patent rights. It is believed that all companies claiming such rights have agreed to grant a license under such rights that they hold on reasonable and non-discriminatory terms and conditions to any applicant.

Producers of DTB systems or any component thereof are responsible for obtaining the appropriate licenses for any and all technology defined by the relevant standards and specifications referenced by this standard.

Issues surrounding the protection of intellectual property embodied in the works distributed as digital talking books are discussed in section 14, Digital Rights Management.

Contents

2. Overview

(This section is informative)

A digital talking book (DTB) is a collection of electronic files arranged to present information to the target population via alternative media, namely, human or synthetic speech, refreshable braille, or visual display, e.g., large print. When these files are created and assembled into a DTB in accordance with this standard, they make possible a wide range of features such as rapid, flexible navigation; bookmarking and highlighting; keyword searching; spelling of words on demand; and user control over the presentation of selected items (e.g., footnotes, page numbers, etc.) For a full discussion of these capabilities, see the Document Navigation Features List [Navigation Features], developed as the user requirements document on which this standard was based. Appendix 7 "Theory Behind the DTBook3 DTD also describes the navigational capabilities of a DTB in some detail. The content of DTBs will range from audio alone, through a combination of audio, text, and images, to text alone.

DTB players will also take a variety of shapes. The simplest might be portable devices with audio-only capabilities. More complex portable players could include text-to-speech capabilities as well as audio output for recorded human speech. The most comprehensive playback systems are expected to be PC-based, supporting visual and audio output, text-to-speech capability, and output to a braille display.

The files comprising a DTB fall into ten categories, as described below:

Package File

> The Package File, drawn from the Open eBook Publication Structure 1.0.1, contains administrative information about the DTB and the files that comprise it. An XML version 1.0 file, it contains a complete set of metadata describing the DTB, a list (the *manifest*) of all files that make up the DTB, a *spine* that defines the default reading order of the document, and an optional *guide* to enable easy access to key points in the DTB. See section 3, "Package File."

Document Text File

> A DTB may contain part or all of the text of the document, as an XML 1.0 file tagged in accordance with the document type definition (DTD) defined for this standard, DTBook3.dtd. (See Appendix 5, "DTBook3 DTD".) The document text file enables a playback device to spell words on demand, carry out keyword searches, and permit finely-grained navigation. It may also be accessed directly via refreshable braille display, synthetic speech, or screen-enlarging software. See section 4, "Content Format for Text."

Audio Files

> A DTB may include human or synthetic speech recordings of the document, embodied in audio files encoded in one of a specified group of audio formats. Section 5, "Audio File Formats," presents the set of formats specified by this standard.

Other Media Files

> In addition to text and audio, DTBs may include images which can be presented on PC-based players. Section 6, "Other Media File Formats," lists the formats specified by this standard.

Synchronization Files

> To synchronize the different media files of a DTB during playback, this standard specifies the World Wide Web Consortium's (W3C) Synchronized Multimedia Integration Language (SMIL), SMIL 2.0 version, an XML 1.0 application. The DTB SMIL files define a sequence of media events. During each event, text elements and corresponding audio clips as well as any additional visual elements are presented simultaneously. DTB players utilize the synchronization information to both index into the audio presentation and to track, during audio playback, the corresponding position in the document text file. IDs on the SMIL elements match those on the corresponding elements in the text file. This standard utilizes a subset of the full SMIL 2.0 specification. See section 7, "Synchronization of Media Files," for discussion of these issues and Appendix 4, "DTB SMIL Profile and DTB-Specific DTD," for the DTDs and Modules that define the DTB SMIL application.

Navigation Control File

> The DTB system supports two modes of navigation, global and local. Global navigation -- movement by structure

(chapter, section, subsection) and by other selected points such as pages -- is effected through the Navigation Control file for XML applications (NCX) and the Lightweight Navigation file (LWN). The NCX and LWN present a dynamic view of the document's hierarchical structure, allowing the user to move through the document in large steps corresponding to its major divisions, or in progressively smaller steps down to a limit set by the document's detail. Text, audio, and image elements present to the user the document's headings, and id-based links point to the SMIL presentation at the corresponding locations. Appendix 1 contains the XML 1.0 DTDs for the NCX and LWN. Local (more finely-grained) navigation is not handled by the NCX but is enabled through the document text file or through time-based movement through the audio presentation, depending on the document and on the player. See section 8, "Navigation Control," and Appendix 1, "DTD for NCX." for specifications related to the NCX and LWN.

Bookmark/Highlight File

This standard supports user-set, exportable bookmarks and highlights to which text and audio notes may be applied. Specifications for the XML 1.0 file for portable bookmarks and highlights are presented in section 9, "Portable Bookmarks/Highlights" and Appendix 2, "DTD for Portable Bookmarks/Highlights."

Resource File

The resource file contains various text segments, audio clips, and/or images that provide alternative representations of navigational information -- for example, feedback on the user's current location in the document. It supplies information normally presented in a print book via typographical clues. It is an XML version 1.0 file based on the DTD "resource.dtd". See section 10, "Resource File," and Appendix 3, "DTD for Resource File" for file specifications.

Distribution

Given the great size of audio files, even when heavily compressed, it will be common for large books to span several distribution media. Section 11, "Packaging Files for Distribution," describes how DTB producers will map the location of each file to a specific media unit, e.g., disk 1 of 3. Alternatively, several small books may be distributed on the same media unit. Appendix 8, "Distribution Information DTD," presents the document type definition for "Distinfo" files.

Presentation Styles

Section 12, "Presentation Styles," discusses how presentation styles may be controlled for a variety of outputs modes through the use of optional style sheets. Contents

3. The DTB Package File

(This section is normative)

The Package File, drawn from the Open eBook Forum" (OEBF) Publication Structure 1.0.1, is a collection of files that contain administrative information about the DTB, the files that comprise it, and how these files interrelate. The Package File as utilized in this standard is identical in most respects to the Package File specified in the OEBF Publication Structure 1.0.1. Despite those few differences, the document type definition (DTD) that describes the OEBF package file also describes the DTB Package File. The first difference between the OEBF package file and the DTB application is simply an extension of section 2.4 of the Publication Structure to allow the spine element to refer to media types other than text/x-oeb-document. Specifically, DTBs require the spine to also allow SMIL files. The second difference is that items in a DTB manifest need not be restricted to OEBF core MIME types, or provide fallbacks that resolve to such MIME types.

A DTB conforming to this standard must include exactly one Package File which must be a valid XML 1.0 document that conforms to the OEBF package DTD (oebpkg1.dtd). Where multiple DTBs are distributed on a single piece of distribution media (e.g., CD-ROM) each DTB must include its own Package File. See section 11 for more information about multiple DTBs on one piece of media.

(This section is informative)

The full specification and DTD for the OEBF package file (section 2 of the OEBF Publication Structure 1.0.1 [OEBPS]) are available on the OEBF site. This section, drawn largely from the Publication Structure, provides only a

brief summary of the function of each section with an example illustrating how it is applied to the DTB. Please see section 2 of the full OEBF Publication Structure 1.0.1 for complete details on the Package File.

The Publication Structure describes the major parts of the Package File as follows:

- ❍ **PACKAGE IDENTITY** - a unique identifier for the OEB publication as a whole.
- ❍ **METADATA** - Publication metadata (title, author, publisher, etc.).
- ❍ **MANIFEST** - A list of files (documents, images, style sheets, etc.) that make up the publication. The manifest also includes fallback declarations for files of types not supported by this specification.
- ❍ **SPINE** - An arrangement of documents providing a linear reading order.
- ❍ **TOURS** - A set of alternate reading sequences through the publication, such as selective views for various reading purposes, reader expertise levels, etc.
- ❍ **GUIDE** - A set of references to fundamental structural features of the publication, such as table of contents, foreword, bibliography, etc.

Here is an informal outline of the package file:

```
<?xml version="1.0"?>
<!DOCTYPE package PUBLIC "+//ISBN 0-9673008-1-9//DTD OEB 1.0 Package//EN"
"http://openebook.org/dtds/oeb-1.0/oebpkg1.dtd">
<package>

     <metadata>...</metadata>
     <manifest>...</manifest>
     <spine>...</spine>
     <tours>...</tours>
     <guide>...</guide>

</package>
```

3.1 Package Identity

(This section is normative)

The root (the outermost) element in a package file is *package*. All other elements are nested inside it. The *package* must include a value for its *unique-identifier* attribute. This is required because more than one *dc:Identifier* may be present in a DTB's Package File metadata and the *unique-identifier* specifies which *dc:Identifier* element provides the package's preferred, or primary, identifier. The value of *unique-identifier* must match the id of the primary *dc:Identifier*.

The package file's author must ensure that the primary identifier is globally unique to the DTB, i.e., to the set of files listed in the package file's manifest. The *dc:Identifier* itself has an optional scheme attribute which names the system or authority that generated or assigned the text contained within the *dc:Identifier* element, for example "ISBN," or "DOI."

(This section is informative)

```
Example 3.1: ...  <package unique-identifier="uid">

     <metadata>

         <dc-metadata ... >
         <dc:Identifier id="uid" scheme="DTB">US-NLS-DTB00001</dc:Identifier>

     ...

</package>
```

3.2 Publication Metadata

(This section is normative)

This portion of the Package File contains the information about a DTB that would normally be found in a library catalog record. It includes data about the DTB itself (e.g., title, author, producer, format, and narrator) as well as information about the source publication (usually a print book) such as publisher, edition, copyright statement, etc.

The Package File must contain exactly one metadata element which must contain one and only one *dc-metadata* element holding Dublin Core [DC] metadata and may contain supplemental metadata in an *x-metadata* element. If used, the *x-metadata* element must contain one or more instances of the *meta* element, which uses "name" and "content" attributes to define its value.

3.2.1 Dublin Core Metadata

(This section is normative)

The Publication Structure describes the use of Dublin Core metadata in the following three paragraphs.

" The dc-metadata element can contain any number of instances of any Dublin Core elements. Dublin Core element names begin with the "dc:" prefix followed by a leading uppercase letter. Dublin Core metadata elements may occur in any order; in fact, multiple instances of the same element type (multiple dc:Creator elements, for example) can be interspersed with other metadata elements without change of meaning.

"For upwards-compatibility, the element metadata in an OEB package is required to have an attribute of xmlns:dc="http://purl.org/dc/elements/1.0/" and xmlns:oebpackage="http://openebook.org/namespaces/oeb-package/1.0/".

"Each Dublin Core field is represented by an element whose content is the field's value. The dc:Title and at least one dc:Identifier must be included in the dc-metadata element. Dublin Core elements, like any other elements in the OEB package file, may have an id attribute specified. At least one dc:Identifier, that which is referenced from the package unique-identifier attribute, must have an id specified."

Following are brief definitions of the Dublin Core elements. See the Publication Structure and the Dublin Core itself for more complete descriptions. The attributes "xml:lang" and "id" can be applied to all "dc:..." elements. Additional attributes can be used with several elements as detailed below.

- ❍ dc:Title
    - ■ Content: The title of the DTB.
    - ■ Occurrence: Required
- ❍ dc:Creator
    - ■ Content: Names(s) of primary author(s) or creator(s) of the intellectual content of the publication.
    - ■ Occurrence: Optional (not all documents have known creators) - recommended
    - ■ Added attributes:
        - ■ role -- (optional) The function performed by the creator (e.g., author, editor). See Publication Structure for details on normative list of values.
        - ■ file-as -- (optional) A normalized form of the contents, suitable for machine processing.
- ❍ dc:Subject
    - ■ Content: The topic of the content of the publication.
    - ■ Occurrence: Optional -recommended
- ❍ dc:Description
    - ■ Content: Plain text describing the publication's content.
    - ■ Occurrence: Optional

- ❍ dc:Publisher
  - ■ Content: The agency responsible for making the DTB available. (Compare dtb:sourcePublisher and dtb:producer.)
  - ■ Occurrence: Required
- ❍ dc:Contributor
  - ■ Content: A party whose contribution to the publication is secondary to those named in dc:Creator.
  - ■ Occurrence: Optional
  - ■ Added attributes:
    - ■ role -- (optional) The function performed by the contributor (e.g., translator, compiler). See Publication Structure for details on normative list of values.
    - ■ file-as -- (optional) A normalized form of the contents, suitable for machine processing.
- ❍ dc:Date
  - ■ Content: Date of publication of the DTB. (Compare dtb:sourceDate and dtb:producedDate.)
  - ■ Occurrence: Required
  - ■ Added attributes:
    - ■ event -- (optional) Significant occurrence related to publication of the DTB. Allows repetition of dc:Date to describe, for example, multiple revisions. Best practice is to use dtb:revision and dtb:revisionDate instead.
  - ■ Scheme: [W3C/ISO8601]; the syntax is YYYY-[MM-[DD]], with a mandatory 4-digit year, an optional 2-digit month, and if the month is present, an optional 2-digit day of month.
- ❍ dc:Type
  - ■ Content: The nature of the content of the DTB (e.g., sound, text, image). Best practice is to draw from an enumerated list, although none has been finalized as of this writing. Consult Dublin Core.
  - ■ Occurrence: Optional
- ❍ dc:Format
  - ■ Content: The standard or specification to which the DTB was produced (e.g., DAISY 2.02).
  - ■ Occurrence: Required
- ❍ dc:Identifier
  - ■ Content: A string or number that is globally unique to the DTB.
  - ■ Occurrence: Required
  - ■ Added attributes:
    - ■ scheme -- (optional) The name of the system or authority that generated or assigned the identifier. For example, "ISBN," or "DOI."
- ❍ dc:Source
  - ■ Content: A reference to a resource (generally a print original) from which the DTB is derived. Best practice is to use the ISBN when available.
  - ■ Occurrence: Optional - recommended
- ❍ dc:Language
  - ■ Content: Language of the content of the publication. May be repeated to describe multiple languages.
  - ■ Occurrence: Required
  - ■ Scheme: [ISO 639] language code optionally followed by a two letter country code as in [ISO 3166]. For Sweden: "sv" or "sv-se", for UK: "en" or "en-uk," etc.
- ❍ dc:Relation
  - ■ Content: A reference to a related resource.
  - ■ Occurrence: Optional

❍ dc:Coverage

■ Content: The extent or scope of the content of the resource. Not expected to be used for DTBs.

■ Occurrence: Optional

❍ dc:Rights

■ Content: Information about rights held in and over the DTB. (Compare dtb:sourceRights.)

■ Occurrence: Optional

3.2.2 X-Metadata

(This section is normative)

The following elements were developed for the DTB application to supply information that the Dublin Core element set does not cover. The following elements would appear within the *x-metadata* containing element.

❍ dtb:sourceDate

■ Content: Date of publication of the resource (generally a print original) from which the DTB is derived.

■ Occurrence: Optional - recommended

■ Scheme: [W3C/ISO8601]; the syntax is YYYY-[MM-[DD]], with a mandatory 4-digit year, an optional 2-digit month, and if the month is present, an optional 2-digit day of month.

❍ dtb:sourceEdition

■ Content: Integer describing the edition of the resource (generally a print original) from which the DTB is derived.

■ Occurrence: Optional - recommended

❍ dtb:sourcePublisher

■ Content: The agency responsible for making available the resource (generally a print original) from which the DTB is derived. (Compare dc:Publisher.

■ Occurrence: Optional - recommended

❍ dtb:sourceRights

■ Content: Information about rights held in and over the resource (generally a print original) from which the DTB is derived. (Compare dc:Rights.

■ Occurrence: Required

❍ dtb:sourceTitle

■ Content: The title of the resource (generally a print original) from which the DTB is derived. To be used only if different from dc:Title.

■ Occurrence: Optional

❍ dtb:multimediaType

■ Content: One of the seven types of DTB described in section 13. Values are: audioOnly, audioNCX, audioPartText, audioFullText, textPartAudio, textNCX, textOnly.

■ Occurrence: Optional

❍ dtb:narrator

■ Content: Name of the person whose recorded voice is embodied in the DTB.

■ Occurrence: Optional

❍ dtb:producer

■ Content: Name of the organization/production unit that created the DTB. (Compare dc:Publisher.)

■ Occurrence: Optional

❍ dtb:producedDate

■ Content: Date of first generation of the complete DTB. (Compare dc:Date.)

- Scheme: [W3C/ISO8601]; the syntax is YYYY-[MM-[DD]], with a mandatory 4-digit year, an optional 2-digit month, and if the month is present, an optional 2-digit day of month.
- Occurrence: Optional

❍ dtb:revision

- Content: Integer value of revision number.
- Occurrence: Optional

❍ dtb:revisionDate

- Content: Date associated with specific dtb:revision.
- Scheme: [W3C/ISO8601]; the syntax is YYYY-[MM-[DD]], with a mandatory 4-digit year, an optional 2-digit month, and if the month is present, an optional 2-digit day of month.
- Occurrence: Optional

❍ dtb:totalTime

- Content: Total elapsed time of all audio files comprising the content of the DTB.
- Scheme: [W3C/ISO8601]; using "hh:mm:ss" notation.
- Occurrence: Required

❍ dtb:charset

- Content: The character set in which the files in the DTB file set are written (e.g., ISO-8859-1).
- Occurrence: Required

Authoring tools must enforce the use of required elements.

(This section is informative)

Example 3.2:

```
...
<metadata>
    <dc-metadata xmlns:dc="http://purl.org/dc/elements/1.0/"
    xmlns:oebpackage="http://openebook.org/namespaces/oeb-package/1.0/">
        <dc:Title>Revised Standards and Guidelines of Service for the Library
        of Congress Network of Libraries for the Blind and Physically
        Handicapped 1995</dc:Title>
        <dc:Subject>library information networks</dc:Subject>
        <dc:Subject>libraries and the physically
        handicapped--standards--U.S.</dc:Subject>
        <dc:Subject>libraries and the blind--standards--U.S.</dc:Subject>
        <dc:Identifier id="uid" scheme="DTB">us-nls-DTB00001</dc:Identifier>
        <dc:Creator role="aut">American Library Association. Association of
        Specialized and Cooperative Library Agencies</dc:Creator>
        <dc:Publisher>National Library Service for the Blind and Physically
        Handicapped, Library of Congress</dc:Publisher>
        <dc:Date>2000-06-22</dc:Date>
        <dc:Source>0-8389-7797-9</dc:Source>
        <dc:Language>en</dc:Language>
        <dc:Format>NISODTB1.0</dc:Format>
        <dc:Description>A document developed to improve library service for
        blind and physically disabled persons by providing a tool for assessing
        the current status of those services and for developing long-range
        plans.</dc:Description>
        <dc:Language>en</dc:Language>
    </dc-metadata>
```

```
    <x-metadata>
        <meta name="dtb:sourceDate" content="1995" />
        <meta name="dtb:sourcePublisher" content="American Library Association"
        />
        <meta name="dtb:sourceRights" content="copyright 1995, American Library
        Association" />
        <meta name="dtb:narrator" content="Lowenstein, Ralph" />
        <meta name="dtb:producer" content="American Foundation for the Blind"
        />
        <meta name="dtb:multimediaType" content="audioNcx" />
        <meta name="dtb:totalTime" content="06:22:34" />
        <meta name="dtb:charset" content="ISO-8859-1" />
    </x-metadata>
</metadata>
...
```

3.3 Manifest

(This section is normative)

The *manifest* contains a list of all of the files (documents, images, style sheets, etc.) that make up the DTB, including the package file itself. Each file is referenced by an *item* element. Each *item* must have valued an *href* attribute (the URI of the referenced file; it must not include fragment identifiers), a *media-type* attribute containing the MIME media type of the file, and an *id* attribute. The *id* is specifically utilized when a manifest *item* is referenced by the *spine*. The *manifest* also includes fallback declarations for files of types not supported by this standard (see OEBF Publication Structure for details). A sample *manifest* for a DTB with audio, structure, and text follows (multimediaType=audioFullText):

(This section is informative)

Example 3.3:

```
...
<manifest>
    <item id="opf" href="rs.opf" media-type="text/xml" />
    <item id="distinfo" href="distinfo" media-type="text/xml" />
    <item id="text" href="rs.xml" media-type="text/xml" />
    <item id="text_style" href="dtbbase.css" media-type="text/css2" />
    <item id="ncx" href="rs_ncx.xml" media-type="text/xml" />
    <item id="ncx_style" href="ncx16.css" media-type="text/css2" />
    <item id="SMIL" href="rs.smil" media-type="application/smil" />
    <item id="foreword" href="rs_fwdx.mp3" media-type="audio/mp3" />
    <item id="standards" href="rs_stdx.mp3" media-type="audio/mp3" />
    <item id="appendices" href="rs_app.mp3" media-type="audio/mp3" />
    <item id="index" href="rs_index.mp3" media-type="audio/mp3" />
    <item id="fig_01" href="fig1.png" media-type="image/png" />
    <item id="resource" href="rs.res" media-type="text/xml" />
    <item id="resource_audio" href="res.mp3" media-type="audio/mp3" />
</manifest>
...
```

Here is a *manifest* for an audio-only version of the above DTB (multimediaType=audioNcx), where separate SMIL files were created for each segment of the book.

Example 3.4:

```
...
<manifest>
    <item id="ncx" href="rs_ncx.xml" media-type="text/xml" />
    <item id="foreword" href="rs_fwdx.mp3" media-type="audio/mp3" />
    <item id="standards" href="rs_stdx.mp3" media-type="audio/mp3" />
    <item id="appendices" href="rs_app.mp3" media-type="audio/mp3" />
    <item id="index" href="rs_index.mp3" media-type="audio/mp3" />
    <item id="SMIL1" href="rsfwd.smil" media-type="application/smil" />
    <item id="SMIL3" href="rsapp.smil" media-type="application/smil" />
    <item id="SMIL4" href="rsind.smil" media-type="application/smil" />
    <item id="SMIL2" href="rsstd.smil" media-type="application/smil" />
</manifest>
...
```

3.4 Spine

(This section is normative)

The *spine* consists of a list of one or more *itemref* elements whose order defines the default linear reading order for the DTB. Each *itemref* contains an *idref* which points to the *id* of a file listed in the manifest. The SMIL files for the DTB must be listed in the spine in order of presentation. The player must consult the *spine* when it reaches the end of a SMIL file to determine which file to render next. The first of the following examples shows the *spine* that corresponds to the first of the two *manifest* examples above:

(This section is informative)

Example 3.5:

```
<spine>
    <itemref idref="SMIL" />
</spine>
```

The following </spine> matches the second *manifest* example above. The correct reading order is presented here. Note that it does not match the order of files in the *manifest*, where order is not significant.

Example 3.6:

```
<spine>
    <itemref idref="SMIL1" />
    <itemref idref="SMIL2" />
    <itemref idref="SMIL3" />
    <itemref idref="SMIL4" />
</spine>
```

3.5 Tours

(This section is informative)

The *tours* section of the Package File is described in the OEBF Publication Structure as follows: "Much as a tour guide might assemble points of interest into a set of sightseers' tours, a content provider may assemble selected parts of a publication into a set of tours to enable convenient navigation. ... Reading systems may use tours to provide various access sequences to parts of the publication, such as selective views for various reading purposes, reader expertise

levels, etc." Because of inherent differences between the structure of a DTB and the OEBF *tours*, it is not feasible to implement *tours* in a DTB prepared in accordance with this standard. If a producer wishes to provide alternate versions of a DTB that supply the functionality described above, they may do so by producing versions with different NCXs.

3.6 Guide

(This section is informative)

As specified in the OEBF Publication Structure, the *guide* lists the key structural features of the DTB, such as the table of contents, introduction, bibliography, etc. to enable playback devices to provide convenient access to them. Because DTBs include a mandatory NCX that satisfies a more rigorous and detailed access requirement, the *guide* may not be used frequently in DTBs. The Publication Structure defines a limited set of recognized "types."

Example 3.7:

```
<guide>
    <reference type="glossary" title="Glossary" href="glos.xml" />
    <reference type="bibliography" title="Bibliography" href="bib.xml" />
</guide>
```
Contents

4. Content Format for Text

4.1 Introduction

(This section is normative)

This standard defines an XML 1.0 element set (DTBook3) for markup of the text files of books and other publications presented in digital talking book (DTB) format. A Document Type Definition (DTD)-- dtbook3.dtd -- specifies the markup elements ("tags") and attributes needed to define a document's structure and the semantics of the tagged items. This DTD can be found in Appendix 5, "DTBook3 DTD." To be compliant with this standard, a text file of a DTB must be valid to dtbook3.dtd.

Any element that is to be referenced from the navigation file or synchronization file must contain a unique id.

DTB content producers may extend the base DTD by including one or more new elements or full modules for special situations. To remain conformant with this standard, such extensions must employ the mechanisms specified by XML 1.0.

4.2 Using the DTBook3 Element Set

(This section is informative)

A discussion of the rationale underlying the DTBook3 element set and the benefits it provides to digital talking book applications is located in Appendix 7, "Theory Behind the DTBook3 DTD."

An alphabetical listing of the DTBook3 tags, with definitions, is included in section 4.4. Two documents external to this standard provide detailed information on the use of the element set. First, an expanded version of the DTD, in HTML format, (see [DTBook3 HTML] provides full detail on each element, describing where it can be used and which elements can be used within it, along with an expanded list of attributes.

Second, a comprehensive set of guidelines [StructGuide] for applying the DTBook3 markup is available from the

DAISY Consortium. These **Structure Guidelines** describe the correct application of the DTBook3 element set, emphasizing the importance of capturing the structure of the text content and providing detailed examples of the use of nearly all elements. The expanded DTD and guidelines are not normative.

For more information on XML 1.0 markup and DTD usage, see the W3C XML site [XML].

4.2.1 Modular Extension of the DTD

(This section is informative)

The DTBook3 DTD includes a base set of 83 elements for use in tagging a broad range of material. Additional modules containing tags for specialized applications such as poetry, plays, dictionaries, mathematics, etc. can be "invoked" from within a DTBook3 document when needed, as described below.

A DTBook3 document is an XML application. Therefore it should begin with the XML processing instruction identifying the version of XML, and the character set encoding (see Appendix 5 - DTBook3 DTD for more information):
<?xml version="1.0" encoding="ISO-8859-1" ?>

This is followed by the document type declaration, the DOCTYPE:

```
<!DOCTYPE dtbook3 PUBLIC
          "-//NISO//DTD dtbook3.dtd Version 3-07 2001-01-29//EN"
          "http://www.loc.gov/nls/niso/dtbook3.dtd"
          "dtbook3.dtd">
```

A book can invoke other DTDs or modules to augment the DTBook3 DTD by adding instructions in square brackets before the concluding ">" of the document type declaration. Such instructions in square brackets are called the "internal subset of declarations." For example:

```
<!DOCTYPE dtbook3 PUBLIC
        "-//NISO//DTD dtbook3.dtd//EN"
        "http://www.loc.gov/nls/niso/dtbook3.dtd"
        "dtbook3.dtd"
        [
            <!ENTITY % dramaModule
"http://www.loc.gov/nls/niso/drama.dtd" >
            %dramaModule;
            <!ENTITY % externalblock "| drama">
            <!ENTITY % externalinline "| stagedir">
        ]>
```

The first line of the internal subset declares an entity known as "&dramaModule" and provides the URI where that module can be found. The second line invokes the entity, that is "brings it into" the current document, just as the DOCTYPE declaration invoked the base DTD (DTBook3). The third line declares the entity "&externalblock" and gives it the value "drama." Since DTBook3 contains an entity of the same name, and the internal subset overrules the base DTD in areas of conflict, everywhere in DTBook3 where &externalblock appears (wherever block elements are allowed), the value "drama" is substituted. Since "drama" is the root element in the drama module, the full drama module can be used there. Similarly, the last line effectively allows the element "stagedir" to be used anywhere %externalinline is allowed in DTBook3 (wherever inline elements can be used).

More than one module may be needed and included in a book. In the following example, both a poetry and drama module are invoked, as well as one inline element from the drama module.

```
        [
            <!ENTITY % poemModule
```

```
"http://www.loc.gov/nls/niso/poem.dtd" >
            %poemModule;
            <!ENTITY % dramaModule
"http://www.loc.gov/nls/niso/drama.dtd" >
            %dramaModule;
            <!ENTITY % externalblock "| poem | drama" >
            <!ENTITY % externalinline "| stagedir">
        ]>
```

See Appendix 5 - DTBook3 DTD for further details.

This standard does not mandate the level of markup to be applied to a text file. However, the richer the tagging, the greater the functionality available to the reader.

4.3 Playback Systems and DTBook3

(This section is informative)

The DTBook3 DTD is an XML application defining the structure and content allowed in the textual portion of a digital talking book. The content is provided within semantically rich elements. The functions as outlined in the DTBook3 DTD should be implemented in playback systems to provide the efficient document navigation required by the target population.

Global navigation is controlled by the NCX and provides direct access to the hierarchical structure of the book as defined by the author and publisher -- parts, chapters, sections, subsections, etc., as well as to pages, notes, sidebars, etc. Section 8 describes how this is implemented. Local navigation based on the marked-up text file, when present, should also be supported, enabling movement through the document at a finer granularity than allowed by the NCX. Examples are navigating through tables and nested lists, movement by paragraph, sentence, or word (depending on the degree of markup), and automatically skipping predefined classes of objects such as notes or sidebars. For more information see the Document Navigation Features List [Navigation Features].

Players should have an accessible interface to provide efficient navigational capabilities to the target population. The implementation of navigation between NCX references (global) and inside markup content (local) should be transparent to the target population.

4.4 DTBook3 Tags

(This section is informative)

The element names from DTBook3 are listed below in alphabetical order. The description provided for each element is taken directly from the DTBook3 DTD. In the following list, elements labeled "HTML 4.0" were drawn from the HTML 4.0 specification when creating DTBook3.

Element	Description
a	contains an anchor, which is used in two ways: A name anchor identifies an exact position in a given document. A link anchor, when activated, "links to" (repositions the focus to) another location within that document or another document. [HTML 4.0]
abbr	designates an abbreviation, a shortened form of a word. For example: Mr., approx ., lbs., rec'd.
acronym	marks a word formed from key letters (usually initials) of a group of words. For example: UNESCO, NATO, XML.
address	contains a location at which a person or agency may be contacted. [HTML 4.0]
author	identifies the writer of a given work.
base	contains the base URI from which local references start. It acts as an absolute URI that serves as the base URI for resolving relative URIs found within the document. It is an empty element that may appear only in <head>. [HTML 4.0]

bdo	is used in special cases where the automatic actions of the bidirectional algorithm would result in incorrect display. [HTML 4.0]
blockquote	indicates a block of quoted content that is set off from the surrounding text by paragraph breaks. Compare with <q> which marks short, inline quotations. [HTML 4.0]
bodymatter	consists of the text proper of a book, as opposed to preliminary material <frontmatter> or supplementary information <rearmatter>.
book	surrounds the actual content of the document, which is divided into <frontmatter>, <bodymatter>, and <rearmatter>. <head>, which contains metadata, precedes <book>.
br	marks a forced line break. [HTML 4.0]
caption	describes a table. If used, it must follow immediately after the table start tag. [HTML 4.0]
citation	marks a reference to another document.
code	designates a fragment of computer code. [HTML 4.0]
col	is a means to apply attribute values to table columns. [HTML 4.0]
colgroup	is a group of columns that may share attribute values within a table. [HTML 4.0]
dd	marks a definition of a term within a definition list. [HTML 4.0]
dfn	marks the first occurrence of a word or term that is defined or explained elsewhere in a book. [HTML 4.0]
div	is a generic container for subdivisions of a book. The <level1> ... <level6> hierarchy, or the <level> tag used recursively, should mark the major hierarchical structures of a book, while <div> is used in less formal circumstances or when for production purposes it is desired that a structure should be treated differently. The class attribute identifies the actual name (e.g., part, chapter, letter) of the structure it marks. Compare with <span> which is used in inline settings. [HTML 4.0]
dl	contains a definition list, usually consisting of pairs of terms <dt> and definitions <dd>. Any definition can contain another definition list. [HTML 4.0]
doctitle	marks the title of the book, as the first tag within <frontmatter>. It is used to quickly identify the book.
dt	marks a term in a definition list. [HTML 4.0]
dtbook3	is the root element in the Digital Talking Book 3.0 DTD. Contains metadata in <head> and the document itself in <book>.
em	indicates emphasis. Compare with <strong>. [HTML 4.0]
frontmatter	contains preliminary material such as the copyright notice, foreword, acknowledgments, table of contents, etc. which serves as a guide to the contents and nature of a book.
h1	contains the text of the heading for a <level1> structure. [HTML 4.0 but nested]
h2	contains the text of the heading for a <level2> structure. [HTML 4.0 but nested]
h3	contains the text of the heading for a <level3> structure. [HTML 4.0 but nested]
h4	contains the text of the heading for a <level4> structure. [HTML 4.0 but nested]
h5	contains the text of the heading for a <level5> structure. [HTML 4.0 but nested]
h6	contains the text of the heading for a <level6> structure. [HTML 4.0 but nested]
hd	marks the text of a heading in a <list> or <sidebar>.
head	contains metainformation about the book but no actual content of the book itself, which is placed in <book>. This information is consonant with the head information in xhtml. See: http://www.w3.org/ [HTML 4.0]
hr	is an empty element indicating a horizontal rule. May be used to indicate a break in the text where only blank lines, a row of asterisks, a horizontal line, etc. are used in the print book. [HTML 4.0]
img	marks a visual image. The "src" attribute specifies the location of the image file. The "alt" and "longdesc" attributes may be used to supply short and long descriptions, respectively. may be used to supply short and long descriptions, respectively, although prodnote will generally contain the latter. Longdesc may contain a pointer to the prodnote. The referencing is typically of the form <imgcaption imgref="#yyy">The Caption</imgcaption> containing the printed caption for the <img id="yyy">. [HTML 4.0]
imgcaption	contains the caption for one or more <img>. If the caption applies to more than one <img>, each idref in the list of IDs in the imgref is separated by whitespace.

imggroup	provides a container for <img> and its associated <imgcaption> and <prodnote>. <prodnote> will contain a description of the image. Content model allows multiple <img> if they share a caption, multiple <imgcaption> if several captions refer to a single <img>, and multiple <prodnote> if different versions are needed for different media (e.g., large print, braille, etc.)
kbd	designates information that the reader is to input directly into a computer using the keyboard. [HTML 4.0]
level	is an alternative tag for marking the major structures in a book. It may be used recursively, i.e., repeated indefinitely with each successive occurrence nesting within the previous. It may also be included in a subsequent higher level. The class attribute identifies the actual name (e.g., part, chapter, section, subsection) of the structure it marks. the depth attribute indicates the nesting depth, starting at 1. Subordinate levels have greater depth.
level1	is the highest level container of major divisions of a book. Used in <frontmatter>, <bodymatter>, and <rearmatter> to mark the largest divisions of the book (usually parts or chapters), inside which level2 subdivisions (often sections) may nest. The class attribute identifies the actual name (e.g., part, chapter) of the structure it marks.
level2	contains subdivisions that nest within <level1> divisions. The class attribute identifies the actual name (e.g., subpart, chapter, subsection) of the structure it marks.
level3	contains subdivisions that nest within <level2> subdivisions (e.g., subsections within subsections). The class attribute identifies the actual name (e.g., section, subpart, subsubsection) of the subordinate structure it marks.
level4	contains subdivisions that nest within <level3> subdivisions. The class attribute identifies the actual name of the subordinate structure it marks.
level5	contains subdivisions that nest within <level4> subdivisions. The class attribute identifies the actual name of the subordinate structure it marks.
level6	contains subdivisions that nest within <level5> subdivisions. The class attribute identifies the actual name of the subordinate structure it marks.
levelhd	contains the text of a heading within <level>. Corresponds to <h1> through <h6> used in <level1> through <level6>.
li	marks each list item in a <list>. content may be either inline or block and may include other nested lists. Alternatively it may contain a sequence of list item components, <lic>, that identify regularly occurring content, such as the heading and page number of each entry in a table of contents. [HTML 4.0]
lic	("list item component") allows ordered substructure within a list item <li>. Used when a list item is made up of two or more components, as in a table of contents entry.
line	marks a single logical line of text. Often used in conjunction with <linenum> in documents with numbered lines.
linenum	contains a line number in, for example, legal text.
link	is an empty element appearing in the <head> section of a document that establishes a connection between the current document and another document(s). The <link> element conveys relationship information (for example, "next" and "previous") that may be rendered by user agents in a variety of ways. [HTML 4.0]
list	contains a list. The "type" attribute can indicate whether a list is ordered or unordered.
meta	indicates metadata about the book. It is an empty element that may appear only in <head>. [HTML 4.0]
noscript	identifies an alternate method for carrying out a function when a playback device cannot execute a <script>. See <script>. [HTML 4.0]
note	marks a footnote, endnote, annotation, etc. The reference to the note within the text is marked with a <noteref>.
noteref	marks one or more characters that reference a footnote, endnote, or annotation <note>.
notice	contains a warning, caution, or other type of admonition normally found in the margin of a book. Differs from a sidebar in that a notice must be presented at a specific location within the text and its presentation is not optional.
object	marks an embedded object, which may consist of scripts, applets, images, etc. [HTML 4.0]

p	contains a paragraph. [HTML 4.0]
pagenum	contains a page number from the print document, recorded as the first text object on a page. The"page" attribute allows three types of page numbering schemes to be identified: "normal" arabic numbering in the body of the book, "front" pages (from the frontmatter), and "special" pagination schemes such as hyphenated numbers in appendices.
param	provides a named property for <object>. [HTML 4.0]
prodnote	contains language added to the alternative-format version by the producers; commonly used to provide verbal descriptions of visual elements such as charts, graphs, etc.; to supply operating instructions; or to describe differences between the print book and the audio version.
q	contains a short, inline quotation. Compare with <blockquote> which marks a longer quotation set off from the surrounding text. [HTML 4.0]
rearmatter	contains supplementary material such as appendices, glossaries, bibliographies, and indices following the <bodymatter> of the book.
samp	contains a sample of work created by the author for use as an example or template. For example, a sample business letter, resume, computer program output, or form. [HTML 4.0]
script	contains a script, a program that may accompany a document or be embedded directly in it. The program executes on the client's machine when the document loads, or at some other time such as when a link is activated. See <noscript> for an alternative in case the <script> cannot be executed. [HTML 4.0]
sent	marks a sentence.
sidebar	contains information supplementary to the main text and/or narrative flow and is often boxed and printed apart from the main text block on a page.
span	is a generic container for use in inline settings when no specific tag exists for a given situation. The class attribute may describe the nature of the text it marks (e.g., a typographical error). May be used to mark a class of items to which styles are to be applied. Compare with <div> which is used in block settings. [HTML 4.0]
strong	marks stronger emphasis than <em>. [HTML 4.0]
style	is means to include styling information that applies to the book. It may appear only in <head>. [HTML 4.0]
sub	indicates a subscript character (printed below a character's normal baseline). Can be used recursively and/or intermixed with <sup>. [HTML 4.0]
sup	marks a superscript character (printed above a character's normal baseline). Can be used recursively and/or intermixed with <sub>. [HTML 4.0]
table	contains a table data arranged in rows and columns. [HTML 4.0]
tbody	marks a group of rows in the main body of a table. If the table is divided into several sections, each consisting of a number of rows, each section would be separately tagged with tbody. [HTML 4.0]
td	indicates an individual data cell in the body of a table. [HTML 4.0]
tfoot	marks table footer information, consisting of one or more rows (each marked with the tr tag). [HTML 4.0]
th	indicates a table cell containing header information. [HTML 4.0]
thead	marks header information in a table, consisting of one or more rows (each marked with the tr tag) of <th> cells. [HTML 4.0]
title	contains the title of the book but is used only as metainformation in <head>. Use <doctitle> within <book> for the actual book title, which will usually be the same. [HTML 4.0]
tr	marks one row of a table containing <th> or <td> cells. [HTML 4.0]
var	indicates an instance of a variable or program argument. Commonly used as a placeholder for text to be entered by the user. [HTML 4.0]
w	marks a word.

[Contents]

5. Audio File Formats

(This section is normative)

5.1 Introduction

(This section is normative)

The limitations placed upon currently feasible DTB distribution systems by the size of audio files necessitate some form of data compression. A set of audio file formats are listed below; audio players should be capable of decoding all of the applicable formats listed, while content must be delivered in one of these formats, or any mixture of them.

5.2 Distribution File Formats

(This section is normative)

Digital Talking Book producers must use only a selected set of standard audio encoding formats for distribution of content to the user. These formats are based on stable and widely-used standards.

It is permissible for parts of a single document to be encoded in different audio formats. For example, a producer may choose to encode a lengthy bibliography at a lower bitrate or with a different codec than the main body of the book. Players must support transitions between differently encoded sections smoothly.

Support for the decoding of stereo or multi-channel signals is not required.

While the ISO standards for MP3 and AAC require support for variable bitrate playback, DTB players will only be required to support constant bitrate playback.

A compliant DTB player that provides audio output should be capable of decoding the following audio formats:

- MPEG-4 AAC [MPEG] - ISO/IEC 14496-3

  Best currently available high-quality standardized audio coder. Is intended to cover bitrates from 16 to 64 kbps, and can be bitrate scalable.

- MPEG-1/2 Layer III (MP3) [MPEG] - ISO/IEC 11172-3, ISO/IEC 13818-3

  Included because of its current ubiquity. It is expected that MP3 will be deprecated in favor of MPEG-4 AAC, which is capable of better sound quality at lower bitrates with comparable decoder complexity.

- Linear PCM - RIFF WAVE format [RIFFWAV]

  File header contains information about sample rate, number of channels, etc.
  Not compressed; may be used for titles, headings, or other short sections.
  Players are required to handle monaural WAVE files with single 'fmt' and 'data' subchunks only, and are not required to support any other RIFF media types. Players must support sample rates of 44.1, 22.05, and 11.025 kHz.

Audio players capable of recording and exporting audio notes for bookmarks and highlights must support encoding in the following format. Similarly, audio players capable of importing bookmarks and highlights must support decoding of the following format.

- ADPCM - ITU-T G.726

Communication quality at 40,32,24 and 16 kbps, usually encoded at 32 kbps from 16 bit 8 kHz data. Encoder and decoder are simple to implement; primarily intended for recording and playback of user notes on bookmarks and highlights. Contents

6. Other Media File Formats

(This section is normative)

Playback devices that support image display must be capable of displaying the following image formats: jpeg (RFC 2046) and png (RFC 2083).

Contents

7. Synchronization of Media Files

7.1 Introduction

(This section is informative)

The Synchronized Multimedia Integration Language (SMIL) was developed by the World Wide Web Consortium as a standard for definition and playback of multimedia presentations over the Internet. SMIL defines the sequence of playback for one or more media objects. In the case of DTBs, the primary media objects are audio and text files; SMIL provides for their parallel and synchronized presentation. Any DTB constructed using SMIL, and utilizing content encoded in standard text and audio media types, is playable on any device or platform which has implemented a SMIL-conformant player of the same or later SMIL version, so long as the necessary audio, image, and textual rendering decoders are present.

What distinguishes a DTB playback system from a basic SMIL player is the inclusion of specific navigation and presentational capabilities set out in the user requirements for DTBs ([Navigation Features]). These capabilities can utilize information from an NCX file, from the textual content, and/or from the SMIL file itself. The key to this information is the inclusion of unique identifiers within the textual content (when present) and SMIL files. Audio files are indexed by time-based positions and in themselves contain no embedded semantic structure. To provide semantic structure to audio content, it is necessary to associate time-points in the audio file with the corresponding position within the textual content. This is achieved using SMIL through the pairing of a pointer to a specific position within a text file (referenced by a Uniform Resource Identifier (URI)) with its corresponding time position in the audio content. In the case of the DTB SMIL application, each synchronization point within the SMIL file is assigned a unique identifier. The presence of these identifiers within both the textual content and the SMIL allows navigation to occur by several different methods, as determined by the playback system.

SMIL incorporates a control structure called *customTest*s, which allows SMIL authors to define optional content selection during playback. This capability permits multiple auditory "views" of a DTB to be selected by a reader. Because a SMIL-based DTB is composed of a sequence of structured audio elements, it is possible to tag individual elements with a structure classification, such as notes or page numbers. Playback systems should expose to the user the presence of these *customTest*s and allow the user to select whether a given structural element is to be read during sequential playback or not.

The DTB producer determines granularity of the synchronization events. Synchronization events may be limited to the primary structural elements (those indicated in the NCX) or may be augmented in books with full textual content to include synchronization down to paragraph, sentence, or even word level. The requirement for this level of synchronization is that the textual content include mark-up tags for the desired elements, and that those elements include unique identifiers that can be referenced in the SMIL.

The SMIL file for a DTB consists of a sequence of typically parallel events, e.g., text and audio (and possibly image) events occurring simultaneously. SMIL represents this structure through the use of the "time containers" *seq* (sequence of events) and *par* (parallel time grouping in which multiple elements play back at the same time). A simple form of DTB SMIL file would be as follows, where the three *pars* shown are played one after the other:

```
<smil>
...
```

```
<seq>
<par>...</par>
<par>...</par>
<par>...</par>
</seq>
...
</smil>
```

7.2 SMIL Modules

(This section is informative)

This standard is based on the SMIL 2.0 Specification. [*Note: At the time of this writing, SMIL 2.0 is in last call status (21 September 2000 draft) so parts of this standard may change if the applicable sections of SMIL 2.0 are modified before the draft achieves recommendation status.*] Developers are requested to reference the SMIL 2.0 specification for complete background and details. Only a small subset of the SMIL specification is utilized in this implementation, drawing from the following modules, which are grouped by functional area. Modules marked with asterisks are used in whole or in part in this application; the others are included because they are part of a core set of modules required for *host language conformance* under W3C modularization guidelines.

- ❍ Timing
  - ■ *BasicInlineTiming
  - ■ MinMaxTiming
  - ■ SyncbaseTiming
  - ■ EventTiming
  - ■ *BasicTimeContainers
- ❍ Content Control
  - ■ *BasicContentControl
  - ■ *CustomTestAttributes
  - ■ SkipContentControl
- ❍ Layout
  - ■ *BasicLayout
- ❍ Linking
  - ■ *BasicLinking
  - ■ *LinkingAttributes
- ❍ Media Objects
  - ■ *BasicMedia
  - ■ *MediaClipping
- ❍ Metainformation
  - ■ *Metainformation
- ❍ Structure
  - ■ *Structure

(This section is normative)

The above modules have been chosen to provide the functionality required for the DTB application. Together they form a *profile* defined in Appendices 4.1 and 4.2, which consist of a DTD and an associated module, as follows:

- ❍ DTB-SMIL20.dtd (The driver file for DTB SMIL. See Appendix 4.1, "DTB SMIL 2.0 DTD")
- ❍ DTB-SMIL-model-1.mod (This file specifies the content models and attributes for the modules called by the driver file.

See Appendix 4.2, "DTB SMIL Document Model")

Authoring tools using the above files to validate DTB SMIL files must also reference the module files listed in Appendix 4.3, "SMIL 2.0 Modules Included in DTB SMIL Profile."

To simplify validation using commonly available parsers and to lessen the complexity of determining content models and attribute lists, a DTB-specific SMIL DTD is included in Appendix 4.4.

A compliant DTB must contain at least one SMIL file. All SMIL files that comprise a DTB must be valid XML and must validate to either the DTB-Specific SMIL DTD or to the DTB SMIL Profile. Any player that complies with the DTB SMIL Profile will be able to play a compliant DTB SMIL presentation.

7.3 SMIL Elements

(This section is informative)

As mentioned above, the DTB application utilizes only a portion of the elements and attributes that make up the modules in the DTB SMIL Profile. Playback devices compliant with this standard need support only the following SMIL elements and attributes.

- ❍ **<smil>**
  Description: The root element of a SMIL 2.0 file is *smil*. The *smil* element contains exactly one *head* and exactly one *body*.
  Declaration:<!ELEMENT smil (head?, body?) >
  Syntax: <smil>...content...</smil>
  Attributes:
    - ■ **%Core.attrib;** See section 7.3.1.
    - ■ **xml:lang (NMTOKEN, IMPLIED)**
  Valid inside: None

- ❍ **<head>**
  Description: Contains information not directly related to the temporal presentation: metadata (using *meta* element), optional *layout*, and optional *customAttributes*, in that order.
  Declaration:<!ELEMENT head ((meta)*, (layout)?, (customAttributes)?) >
  Syntax: <head>...content...</head>
  Attributes:
    - ■ **%Core.attrib;** See section 7.3.1.
    - ■ **xml:lang (NMTOKEN, IMPLIED)**
  Valid inside: <smil>

- ❍ **<meta>**
  Description: Contains information describing the SMIL document.
  Syntax: <meta ...attributes... />
  Declaration:<!ELEMENT meta EMPTY >
  Attributes:
    - ■ **content (CDATA, #IMPLIED)**
    - ■ **name (CDATA, #REQUIRED)**
  Valid inside: <head>

- ❍ **<layout>**
  Description: Controls (through the *region* elements it contains) where on a visual, audio, or tactile rendering space

various producer-defined elements, e.g., figures, text, footnotes, etc. are displayed.
Declaration:<!ELEMENT layout (region)* >
Syntax: <layout>...content...</layout>
Attributes:

- **%Core.attrib;**
- **xml:lang (NMTOKEN, IMPLIED)**Specifies an RFC1766 language code.

Valid inside: <head>
Comments: This standard allows only SMIL 2.0 Basic Layout syntax (i.e., CSS2 syntax and others are not permitted).

❍ **<region>**
Description: Controls the position, size, and scaling of media objects (e.g., text, img).
Declaration:<!ELEMENT region EMPTY >
Syntax:
Attributes:

- **id (ID, REQUIRED)** Value of *region* attribute on media object matches value of id on appropriate *region* element.
- **bottom (CDATA, 'auto')** Locates region in display space. See SMIL 2.0 for details.
- **right (CDATA, 'auto')** Locates region in display space. See SMIL 2.0 for details.
- **height (CDATA, 'auto')** Locates region in display space. See SMIL 2.0 for details.
- **fit ((hidden|fill|meet|scroll|slice) 'hidden')** Specifies behavior if the intrinsic height and width of a visual media object differ from those of the region is which they are displayed. See SMIL 2.0 for definitions of attribute values.
- **z-index (CDATA, IMPLIED)** Used for control of multilayered displays.

Valid inside: <layout>
Comments: All media objects whose *region* attribute matches the id on a given *region* element will be displayed in that region.

❍ **<customAttributes>**
Description: Contains one or more *customTests* which allow the producer to specify kinds of structures that the user can choose to have automatically rendered or skipped.
Declaration:<!ELEMENT customAttributes (customTest)+ >
Syntax: <customAttributes>...content...</customAttributes>
Attributes:

- **%Core.attrib;** See section 7.3.1.
- **xml:lang (NMTOKEN, IMPLIED)**

Valid inside: <head>
Comments: See section 7.4.5 for normative content.

❍ **<customTest>**
Description: Defines the kinds of structures (e.g., page numbers, notes, line numbers, etc.) which the user can choose to have presented or skipped during normal playback of a DTB. See definition of *customTest* attribute for *par* and *seq* below.
Declaration:<!ELEMENT customTest EMPTY >
Syntax:
Attributes:

- **id (ID, REQUIRED)** Id here serves as a unique identifier linking *customTest* declaration in *head* with corresponding structures identified by a *customTest* attribute on *par* or *seq* in *body* of SMIL.
- **class (CDATA, IMPLIED)**

- **title (CDATA, IMPLIED)**
- **xml:lang (NMTOKEN, IMPLIED)**
- **defaultState (Values: true, false. Default: false)** Specifies whether player will present (value = true) or skip (value = false) the structure during sequential playback. If no value is present, the default is false and the content is skipped.
- **override (Values: allowed, not-allowed. Default: not-allowed)** Specifies whether reader may override the setting of defaultState. In DTB applications, the element *customTest* will only be used when the producer wishes to allow the reader to turn a class of structures on or off, so override should always be set to "allowed."

Valid inside: <customAttributes>

❍ **<body>**
Description: Contains the time containers that define the temporal presentation.
Declaration:<!ELEMENT body (par|seq|text|audio|img|a)* >
Syntax: <body>...content...</body>
Attributes:

- **%Core.attrib;** See section 7.3.1.
- **xml:lang (NMTOKEN, IMPLIED)**

Valid inside: <smil>
Comments: The *body* contains zero or more *seq*s or *par*s and may also directly contain zero or more media objects (<text>, <audio>, <img>), or links (<a>).

❍ **<seq>**
Description: Container for a sequence of SMIL events, e.g., a series of *pars* and *seqs*.
Declaration:<!ELEMENT seq (par|seq|text|audio|img|a)* >
Syntax: <seq>...content...</seq>
Attributes:

- **id (ID, REQUIRED):** Must be valued for all *seqs*. Matches *id* of corresponding fragment of text file, if text file is present. When *seq* is target of an NCX navObject, *id* on *seq* is referenced by fragment identifier in value of *src* attribute on NCX *content* element.
- **class (CDATA, IMPLIED):** Used for several purposes: First, to identify structures such as tables, lists, and notes, from which a reader can "escape" with a single action. Second, to identify structures such as tables and lists for which special navigation functions should be automatically invoked when entered. To be effective for these applications, the class attribute will be valued with standard names drawn from the DTBook3 DTD, e.g., "table," "list," and "note."
- **customTest (IDREF, IMPLIED):** ID reference linking *seq* with matching *customTest* declaration in *head*.
- **begin (CDATA, IMPLIED)** The time at which the seq becomes active. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module [SMILCLOCK] See Section 7.7, Clock Values.
- **dur (CDATA, IMPLIED)** The duration of the seq. It uses the same attribute value syntax as *begin*.
- **end (CDATA, IMPLIED)** The time at which the seq ceases to be active. It uses the same attribute value syntax as *begin*.

Valid inside: <body>, <seq>, <par>
Comments: It is permissable to nest *seqs*.

❍ **<par>**
Description: Parallel time grouping in which multiple elements (e.g., text, audio, and image) play back simultaneously.
Declaration:<!ELEMENT par (seq|text|audio|img|a)* >
Syntax: <par>...content...</par>
Attributes:

- **id (ID, REQUIRED):** Must be valued for all *pars*. Matches *id* of corresponding fragment of text file, if text file

is present. When *par* is target of an NCX navObject, *id* on *par* is referenced by fragment identifier in value of *src* attribute on NCX *content* element.

- **class (CDATA, IMPLIED):** Used for several purposes: First, to identify structures such as tables, lists, and notes, from which a reader can "escape" with a single action. Second, to identify structures such as tables and lists for which special navigation functions should be automatically invoked when entered. To be effective for these applications, the class attribute will be valued with standard names drawn from the DTBook3 DTD, e.g., "table," "list," and "note."

- **customTest (IDREF, IMPLIED):** ID reference linking *par* with matching *customTest* declaration in *head*.

- **begin (CDATA, IMPLIED)** The time at which the par becomes active. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module [SMILCLOCK] See Section 7.7, Clock Values.

- **dur (CDATA, IMPLIED)** The duration of the par. It uses the same attribute value syntax as *begin*.

- **end (CDATA, IMPLIED)** The time at which the par ceases to be active. It uses the same attribute value syntax as *begin*.

Valid inside: <body>, <seq>

Comments: Each *par* contains at most one each of *text*, *audio*, and *image* but there is no limit on the number of *seq*s. Because of resource limitations on portable DTB players, nesting of *pars* within *pars* is strongly discouraged. Authoring tools should not support the generation of nested *pars* and their validators should issue warnings if any are encountered. However, playback devices should not fail but fall back gracefully when confronting nested *pars*.

❍ **<text>**
Description: Points to segment of textual content to be rendered.
Declaration:<!ELEMENT text EMPTY >
Syntax: <text ...attributes... />
Attributes:

- **src (CDATA, IMPLIED):** URI of fragment of text file to be rendered.

- **region (CDATA, IMPLIED):** Specifies the *region* (defined in *layout* in document *head*) in which the text will be presented. References the id of the *region* specified. All types of text objects which are to appear in the same rendering space should be assigned the same value for *region*. For example, page numbers and producer's notes might both be displayed in the main text area of a screen (region="text"), while notes (e.g., footnotes) might be displayed in a separate area at the bottom of the screen (region="notes").

- **begin (CDATA, IMPLIED)** The time at which the media object becomes active. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module [SMILCLOCK] See Section 7.7, Clock Values.

- **dur (CDATA, IMPLIED)** The duration of the media object. It uses the same attribute value syntax as *begin*.

- **end (CDATA, IMPLIED)** The time at which the media object ceases to be active. It uses the same attribute value syntax as *begin*.

Valid inside: <body>, <par>, <seq>

❍ **<audio>**
Description: Points to segment of audio content to be rendered.
Declaration:<!ELEMENT audio EMPTY >
Syntax:
Attributes:

- **src (CDATA, IMPLIED):** URI of audio file containing clip to be rendered.

- **clipBegin (CDATA, IMPLIED):** The clipBegin attribute specifies the beginning of a segment of a continuous audio file as a time offset from the start of the audio file. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module [SMILCLOCK] See Section 7.7, Clock Values.

- **clipEnd (CDATA, IMPLIED):** The clipEnd attribute specifies the end of a segment of a continuous audio file as a time offset from the start of the audio file. It uses the same attribute value syntax as clipBegin.

- **begin (CDATA, IMPLIED)** The time at which the audio file becomes active. The value syntax is defined by the

SMIL 2.0 Timing and Synchronization Module [SMILCLOCK] See Section 7.7, Clock Values.

- **dur (CDATA, IMPLIED)** The duration of the audio file. It uses the same attribute value syntax as *begin*.
- **end (CDATA, IMPLIED)** The time at which the audio file ceases to be active. It uses the same attribute value syntax as *begin*.
- **region (CDATA, IMPLIED):** Specifies the *region* (defined in *layout* in document *head*) in which the audio object will be presented. References the id of the *region* specified.

Valid inside: <body>, <par>, <seq>

Comments: If clipBegin is not present the audio file is played from its beginning. If clipEnd is not present, the audio file is played to its end. If the value of the clipEnd attribute exceeds the duration of the audio file, the value is ignored, and the audio file is played to its end.

❍ **<img>**
Description: Points to image to be rendered.
Declaration:<!ELEMENT img EMPTY >
Syntax:
Attributes:

- **src (CDATA, IMPLIED):** URI of image file to be rendered.
- **region (CDATA, IMPLIED):** Specifies the *region* (defined in *layout* in document *head*) in which the image will be presented. References the id of the *region* specified.
- **begin (CDATA, IMPLIED)** The time at which the media object becomes active. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module [SMILCLOCK] See Section 7.7, Clock Values.
- **dur (CDATA, IMPLIED)** The duration of the media object. It uses the same attribute value syntax as *begin*.
- **end (CDATA, IMPLIED)** The time at which the media object ceases to be active. It uses the same attribute value syntax as *begin*.

Valid inside: <body>, <par>, <seq>

❍ **<a>**
Description: Defines a link, which can be traversed during a defined period of time.
Declaration:<!ELEMENT a (text|audio|img)* >
Syntax: <a>...content...>
Attributes:

- **%Core.attrib;** See section 7.3.1.
- **xml:lang (NMTOKEN, IMPLIED)**
- **href (%URI;, REQUIRED)** Specifies the URI of the target of the link. The URI may include a fragment identifier.
- **begin (CDATA, IMPLIED)** The time at which the link becomes active, relative to the beginning of the audio file referenced by the *audio* element contained within the *a* element. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module [SMILCLOCK] See Section 7.7, Clock Values.
- **dur (CDATA, IMPLIED)** The duration during which the link is active. It uses the same attribute value syntax as *begin*.
- **end (CDATA, IMPLIED)** The time at which the link ceases to be active, relative to the beginning of the audio file referenced by the *audio* element contained within the *a* element. It uses the same attribute value syntax as *begin*.

Valid inside: <body>, <par>, <seq>
Comments: See section 7.4.6 for normative content.

7.3.1 Common Attributes

(This section is informative)

The following attributes are allowed when the entity %Core.attrib; is listed above:
- **id (ID, IMPLIED)**
- **class (CDATA, IMPLIED)**
- **title (CDATA, IMPLIED)**

7.4 SMIL Production Issues

7.4.1 "Escapable" Structures

(This section is normative)

DTB players should provide the functionality to allow readers to escape from specific structures (at a minimum tables, lists, and notes) with a single action. To support this functionality, producers must ensure that the beginning and end of each such structure is indicated in SMIL by wrapping in a <seq> any such structure consisting of multiple time containers (i.e., <seq>s and <par>s). In addition, producers must include a class attribute on the <seq> or <par> containing a table, list, or note, using element names drawn from the DTBook3 DTD (i.e., "table," "list," and "note").

7.4.2 Automatic Invocation of Special Navigation Modes

(This section is normative)

DTB players should automatically invoke special navigation modes when the reader enters a table or list. To support this functionality, producers must include a class attribute on the <seq> or <par> containing a table or list, using element names drawn from the DTBook3 DTD (i.e., "table" and "list.") Producers and players may also support this functionality for other structures using the same mechanism.

7.4.3 "Skippable" Structures

(This section is normative)

Players should offer the user the option to "turn off" selected structures in a DTB, that is, identify certain structures such as notes or line numbers that the player will automatically skip over during sequential playback. To support this capability, producers must include customTest attributes on <seq>s and <par>s containing those structures. In addition, <customAttributes>/<customTest> elements must be valued in the <head> of each SMIL file. At a minimum, producers must offer readers the option to skip over <linenum>, <note>, <noteref>, <pagenum>, optional <prodnote>, and <sidebar>.

7.4.4 Packaging Files across Several Distribution Media

(This section is normative)

When a DTB spans several distribution media, producers must package SMIL and other files correctly to ensure a complete DTB presentation. See section 11.2, "Distribution Requirements" for details.

7.4.5 Use of CustomTest Element and Attribute

(This section is normative)

Players should expose to the user the presence of *customTests* and allow the user to select whether a given class of structure (e.g., page number, sidebar, optional producer's note, etc.) will be read during sequential playback.

Authoring tools should be able to create different customTests for a single element, depending on the element's attributes. For example, <prodnote render="optional"> might be assigned the customTest "prodnote_opt", while <prodnote render="required"> would not need to be assigned a customTest as the user should not have to option of turning them off.

If customAttributes are to be included in a SMIL presentation, they must be present in the *head* of each SMIL file in the DTB.

7.4.6 Links

(This section is normative)

If links are present in the document text file of a DTB, producers must also include them in SMIL. The default behavior of a link is to be active for the duration of the media object it contains. If producers specify the precise active duration of a link they must use either *dur* or *end*, but not both.

7.5 SMIL Metadata

(This section is normative)

Metadata is included in the <head> element using the <meta> tag. Content producers may introduce other metadata if needed.
- ❍ Required metadata:
    - ■ charset - The character set in which the SMIL file is written (e.g., ISO-8859-1).
    - ■ uid - The globally unique identifier for the DTB.
    - ■ dtb:totalElapsedTime - The total time elapsed up to the beginning of this SMIL file, using "hh:mm:ss" notation from W3C/ISO8601.
- ● Optional metadata:
    - ❍ dtb:generator - The software used to generate the SMIL file.

7.6 SMIL Examples

(This section is informative)

The following example illustrates the use of *head* and its contents. The *meta* element contains the unique id of the DTB as well as the title. The *root-layout* element defines the size of the rendering window. The visual display location of any text elements with region ="text" or region="notes" is specified by the *region* elements within *layout*. The text region occupies most of the screen (the bottom edge of the "text" region is 15% from the bottom of the overall rendering window), while the notes regions occupies only the bottom 15%. The *customAttributes* indicate that any *pars* with customTest="pagenum" will not be rendered by default, while *pars* with customTest="notes" will automatically be played. If the user interface of the playback device supports it, the user may change these settings.

Example 7.1:
```
<smil>
    <head>
        <meta name="uid" content="us-nls-00065" />
        <meta name="charset" content="ISO-8859-1" />
        <meta name="dtb:totalElapsedTime" content="01:33:56.2" />
```

```
        <layout>
            <root-layout width="640px" height="480px"/>
            <region id="text" top="0%" left="0% right="0% bottom="15%"/>
            <region id="notes" top=85%" left="0% right="0% bottom="0%"/>
        </layout>
        <customAttributes>
            <customTest id="pagenum" defaultState="false" />
            <customTest id="note" defaultState="true" />
        </customAttributes>
    </head>
    <body>
    ...
    </body>
</smil>
```

Example 7.2 shows the use of SMIL elements within *body*. The initial *seq* includes the attribute "dur" which specifies that the entire SMIL file is one hour, three minutes, 24.9 seconds long. Each *par* (a page number, a heading, and two paragraphs are shown) includes the segment of text and corresponding audio clip that are to be rendered simultaneously. The last *par* includes a link that "wraps" the *audio* element. The link becomes active at 2 minutes 12.6 seconds (relative to the beginning of the audio file referenced by the *audio* element) and becomes inactive 15 seconds later. Alternatively, the producer could have chosen to specify when the link would become inactive with the *end* attribute, perhaps in a table of contents where each entry is a link and the producer wishes to make each link active only until the next begins. However, as mentioned above, the default behavior of a link is to be active for the duration of the media object it contains.

Example 7.2:

```
<smil>
    <head>
    ...
    </head>
    <body>
        <seq dur="1:03:24.9">
            <par id="p1" customTest="pagenum"> class="pagenum"
                <text region="text" src="rs.xml#p1" />
                <audio src="rs_fwdx.mp3" clipBegin="00:00:00" clipEnd="00:00:00.9"
                />
            </par>

            <par id="h1_1">
                <text region="text" src="rs.xml#h1_1" />
                <audio src="rs_fwdx.mp3" clipBegin="00:00:01.6"
                clipEnd="00:00:02.5" />
            </par>

            <par id="para_1">
                <text region="text" src="rs.xml#para_1" />
                <audio src="rs_fwdx.mp3" clipBegin="00:00:03.5"
                clipEnd="00:01:45.3" />
            </par>

            <par id="para_2">
                <text region="text" src="rs.xml#para_2" />
```

```
                        <a href="rs12.smil#h2_9" begin="00:02:12.6" dur="15.0"> <audio
                        src="rs_fwdx.mp3" clipBegin="00:01:45.9" clipEnd="00:04:03.7" />
                        </a>
                </par>
                ...


        </seq>
    </body>
</smil>
```

As mentioned earlier, *seqs* may be nested in a DTB SMIL file. Notes or sidebars containing multiple paragraphs will need to be represented as a series of *pars* wrapped in a *seq*, so that a *customTest* can be applied to the *seq*, permitting the user to skip the entire sequence. In addition, note references occuring in the middle of a paragraph will require this special syntax so that the playback device can properly render the text with or without either the note reference or the note. In Example 7.3, the first *par* contains the portion of paragraph 12 preceding the note reference. The second *par* holds the note reference itself (e.g., "footnote 1"). The third *par* contains the contents of footnote 1 and the last holds the remainder of paragraph 12. Note that the *seq* and each *par* contains a unique *id*. The *region* attribute on *text* will control whether each segment is displayed in the text or notes region.

Example 7.3:

```
...
<body>
    <seq dur="14:34.1">
        ... (a series of pars)
        <seq id="para_12">
            <par id="para_12a">
                <text region="text" src="rs.xml#para_12" />
                <audio src="rs_fwdx.mp3" clipBegin="00:58.7" clipEnd="01:21.6" />
            </par>

            <par id="nref_1" customTest="noteref" class="noteref">
                <text region="text" src="rs.xml#nref_1" />
                <audio src="rs_fwdx.mp3" clipBegin="01:22.6" clipEnd="01:23.5" />
            </par>

            <par id="ftn_1" customTest="note"> class="note"
                <text region="notes" src="rs.xml#ftn_1" />
                <audio src="rs_fwdx.mp3" clipBegin="01:23.9" clipEnd="01:34.7" />
            </par>

            <par id="para_12b">
                <text region="text" src="rs.xml#para_12" />
                <audio src="rs_fwdx.mp3" clipBegin="01:35.5" clipEnd="01:48" />
            </par>
        </seq>
    ... (a series of pars)
</seq>
</body>
...
```

7.7 Clock Values

(This section is normative)

The SMIL 2.0 Timing and Synchronization Module describes several different formats in which "clock values" (timing) may be represented. See *Clock Values* [SMILclock] in that module. Playback devices must support all of these formats. Examples of the three different formats follow:

Full-clock-val (hours, minutes, seconds, and fractions of seconds: 3:22:55.9

Partial-clock-val (minutes, seconds, and fractions of seconds: 43:15.0

Timecount-val (one or more digits, plus an optional fraction and unit of measurement -- h=hours, min=minutes, s=seconds, ms=milliseconds): 34.6s or 356ms or 58.2. (For Timecount values, if no unit is shown, the default is "s" (for seconds).)

If either of the first two formats is used, authoring tools must add leading zeroes to single-digit values for minutes and seconds.

Contents

8. Navigation Control (NCX and LWN)

8.1 Introduction

(This section is informative)

Navigation is controlled by one of two files: the NCX file, an XML file that complies with the DTD for NCX found in Appendix 1.1, or the LWN file, a transformation of the NCX file that simplifies navigation for lightweight players, that complies with the DTD for Navigation with Lightweight Players found in Appendix 1.2. Both files are required, and a player may choose to navigate using either.

The NCX file exposes the hierarchical structure of a document to allow the user to navigate through it. The NCX is similar to a table of contents in that it enables the reader to jump directly to any of the major structural elements of the document, i.e. part, chapter, or section, but it will often contain more elements of the document than the publisher chooses to include in the original print table of contents. It can be visualized as a collapsible tree familiar to users of Windows. Its development was motivated by the need to provide quick access to the main structural elements of the document without the need to parse the entire marked-up text file, which in many cases may not be present at all. Other elements such as pages, footnotes, figures, tables, etc. can be included in separate, non-hierarchical lists and may be accessed by the user as well.

The LWN file contains the same information as the NCX file, transformed to a non-hierarchical, sequential list. Hierarchical nesting information is preserved by the inclusion of levelNumber attributes, and all elements that will be available for navigation are merged in the proper sequence into a single list. This file will allow a player with limited resources to jump to any arbitrary location in the book and begin playing without needing to parse the entire NCX or synchronize multiple lists. It is planned to define a compiled format for the LWN in a future version of this standard.

It is important to emphasize that these navigation features are intended as a convenience for users who want them, and not a burden to those who do not. Players should be able to present the document in the default play sequence defined by the package file's spine without requiring user input beyond play and stop controls.

8.2 Navigation Control Files

8.2.1 NCX

(This section is normative)

Every DTB must contain exactly one NCX file. The NCX file must comply with the NCX DTD (see Appendix 1.1, "DTD for NCX") The NCX entry in the Package File manifest must have an id value equal to "ncx". The NCX file itself must be named with the extension ".ncx".

(This section is informative)

Brief descriptions of the NCX elements follow. Each includes the element declaration extracted from the NCX DTD, along with descriptions of any applicable attributes.

❍ **<ncx>**
Description: The root element
Declaration: <!ELEMENT ncx (head, docTitle, navStruct, navList*)>
Syntax: <ncx ...attributes...>...content...</ncx>
Attributes:

- **lang (NMTOKEN, IMPLIED):** Specifies the RFC1766 language code of the language of the document.
- **dir ((ltr|rtl), IMPLIED):** The dir attribute specifies the direction of the text, where ltr is left to right and rtl is right to left.

Valid inside: None

❍ **<head>**
Description: Contains the uid, character set declaration, and metadata.
Declaration: <!ELEMENT head (uid, charset, meta*)>
Syntax: <head>...content...</head>
Attributes: None
Valid inside: <ncx>

❍ **<uid>**
Description: Contains the globally unique identifier for the DTB.
Declaration: <!ELEMENT uid (#PCDATA) >
Syntax: <uid>...content...</uid>
Attributes: None
Valid inside: <head>

❍ **<charset>**
Description: Character set utilized in the NCX.
Declaration: <!ELEMENT charset (#PCDATA)>
Syntax: <charset>...content...</charset>
Attributes: None
Valid inside: <head>

❍ **<meta>**
Description: Contains metadata applicable to the SMIL file.
Declaration: <!ELEMENT meta EMPTY>
Syntax:
Attributes:

- **name (CDATA, REQUIRED)**
- **content (CDATA, REQUIRED)**
- **scheme (CDATA, IMPLIED)**

Valid inside: <head>
Comments: Required and optional metadata are listed in section 8.5.

❍ **<docTitle>**
Description: The title of the document. Contains text and an optional pointer to an audio rendering of the title.
Declaration: <!ELEMENT docTitle (text, audio?)>
Syntax: <docTitle ...attributes...>...content...</docTitle>
Attributes:

■ **id (ID, IMPLIED):** Optional unique identifier.

Valid inside: <ncx>

❍ **<text>**
Description: Contains the text of the heading or title referenced by this element.
Declaration: <!ELEMENT text (#PCDATA)>
Syntax: <text ...attributes...>...content...</text>
Attributes:

■ **id (ID, IMPLIED):** Optional unique identifier.

■ **class (CDATA, IMPLIED):** Optional descriptor of this instance of the element.

Valid inside: <descr>, <docTitle>

❍ **<audio>**
Description: Contains a pointer to an audio version of the heading or title referenced by this element.
Declaration: <!ELEMENT audio EMPTY>
Syntax: <audio...attributes... />
Attributes:

■ **id (ID, IMPLIED):** Optional unique identifier.

■ **class (CDATA, IMPLIED):** Optional descriptor of this instance of the element.

■ **src (CDATA, REQUIRED):** The URI of the audio media object.

■ **clipBegin (CDATA, IMPLIED):** The clipBegin attribute specifies the beginning of a segment of a continuous media object as a time offset from the start of the media object. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module [SMILCLOCK] See Section 7.7, Clock Values.

■ **clipEnd (CDATA, IMPLIED):** The clipEnd attribute specifies the end of a segment of a continuous media object as a time offset from the start of the media object. It uses the same attribute value syntax as clipBegin.

Valid inside: <descr>, <docTitle>
Comments: If clipBegin is not present the media object is played from its beginning. If clipEnd is not present, the media object is played to its end. If the value of the clipEnd attribute exceeds the duration of the media object, the value is ignored, and the media object is played to its end.

❍ **<navStruct>**
Description: Container for primary navigation information.
Declaration: <!ELEMENT navStruct (descr+, navObject+)>
Syntax: <navStruct ...attributes...>...content...</navStruct>
Attributes:

■ **id (ID, IMPLIED):** Optional unique identifier.

Valid inside: <ncx>
Comments: The *navStruct* element contains the primary navigation information, pointing to each of the major structural elements of the document. Players should begin navigation here by default. Secondary navigation elements such as page numbers, etc. are not included in *navStruct*, but are contained in *navList*s.

❍ **<navObject>**

Description: Contains description(s) of target and pointer to content.
Declaration: <!ELEMENT navObject (descr+, content, navObject)>
Syntax: <navObject ...attributes...>...content...</navObject>
Attributes:

- **id (ID, REQUIRED):** Unique identifier matching the id attribute of the corresponding element in the XML and/or SMIL files. See section 8.7.1 for normative content.
- **onFocus (%script, IMPLIED):** May be used by player to provide location feedback to the user, e.g., highlight the current navObject on a visual display of the NCX.
- **onBlur (%script, IMPLIED):** May be used by player to provide location feedback to the user, e.g., remove highlighting when focus has shifted to another navObject on a visual display.
- **navType (CDATA, IMPLIED):** Describes the kind of structural object represented by this navObject, e.g. chapter, section.
- **levelNumber (CDATA, IMPLIED):** Indicates the nesting depth of this navObject.
- **value (CDATA, IMPLIED):** An integer representing the text content of a numeric label, e.g. a page number.
- **pageRef (IDREF, IMPLIED):** The id of the navTarget representing the page on which this navObject begins.

Valid inside: <navStruct>, <navObject>
Comments: The *navObject* element has one or more of *text*, *audio*, or *image* representing the referenced part of the document, e.g. chapter title, section number, along with a pointer to *content*.

○ **<descr>**
Description: Contains the description of this object in the various media for presentation to the user. Can be repeated so descriptions can be provided in multiple languages.
Declaration: <!ELEMENT descr (text?, audio?, image?)>
Syntax: <text ...attributes...>...content...</text>
Attributes:

- **lang (NMTOKEN, IMPLIED):** Specifies the RFC1766 language code of the language of the heading. Can be used to control presentation of headings in a specified language.
- **dir ((rtl|ltr), IMPLIED):** Specifies the direction of the text, where ltr is left to right and rtl is right to left.

Valid inside: <navStruct>, <navObject>, <navList>, <navTarget>

○ **<image>**
Description: Information about the location of an image file.
Declaration: <!ELEMENT image EMPTY>
Syntax: <image...attributes... />
Attributes:

- **id (ID, IMPLIED):** Optional unique identifier.
- **class (CDATA, IMPLIED):** Optional descriptor of this instance of the element.
- **src (CDATA, REQUIRED):** The URI of the media object.

Valid inside: <descr>

○ **<content>**
Description: Pointer into SMIL file to beginning of the item referenced by the *navObject* or *navTarget*.
Declaration: Syntax: <content...attributes... />
Attributes:

- **id (ID, IMPLIED):** Optional unique identifier.
- **src (CDATA, REQUIRED):** The URI of the SMIL time container corresponding to the start of the referenced part of the document.

Valid inside: <navObject>, <navTarget>

❍ **<navList>**
Description: Container for secondary navigational information.
Declaration: <!ELEMENT navList (navTarget)+>
Syntax: <navList ...attributes...>...content...</navList>
Attributes:

- **id (ID, IMPLIED):** Optional unique identifier.
- **navType (CDATA, IMPLIED):** Describes the kind of structural objects represented by this navList, e.g. pages, notes.

Valid inside: <ncx>
Comments: The *navList* element contains secondary navigation information within *navTarget*s. It is similar to *navStruct* except *navTarget*s may not nest, whereas *navObject*s can. Used for lists of elements such as page numbers, footnotes, figures, tables, etc. that the user may want to access directly but would clutter up the primary navigation information.

❍ **<navTarget>**
Description: Container for text, audio, image and content elements containing secondary navigational information.
Declaration: <!ELEMENT navTarget (descr+, content)>
Syntax: <navTarget ...attributes...>...content...</navTarget>
Attributes:

- **id (ID, REQUIRED):** Unique identifier matching the id attribute of the corresponding element in the XML and/or SMIL files. See section 8.7.1 for normative content.
- **onFocus (CDATA, IMPLIED):** May be used by player to provide location feedback to the user, e.g., highlight the current navTarget on a visual display of the NCX.
- **onBlur (CDATA, IMPLIED):** May be used by player to provide location feedback to the user, e.g., remove highlighting when focus has shifted to another navTarget on a visual display.
- **value (CDATA, IMPLIED):** An integer representing the text content of a numeric label, e.g. a page number.
- **navType (CDATA, IMPLIED):** Describes the kind of structural object represented by this navTarget, e.g. page, note.
- **structRef (IDREF, REQUIRED):** Pointer to the *navObject* that contains this *navTarget*.

Valid inside: <navList>
Comments: The *navTarget* element has one or more of *text*, *audio*, or *image* representing the referenced part of the document; points to page numbers, footnotes, figures, tables, etc. The structRef attribute is used to synchronize the SMIL and NCX.

8.2.2 LWN Elements

(This section is normative)

Every DTB must contain exactly one LWN file. The LWN file must comply with the LWN DTD (see Appendix 1.2 "DTD for Navigation with Lightweight Players"). The LWN file must contain the identical information contained in the NCX file, transformed into a non-hierarchical list with all elements in the proper sequence. The LWN entry in the Package File manifest must have an id value equal to "lwn". The LWN file itself must be named with the extension ".lwn".

(This section is informative)

LWN elements are identical to the corresponding NCX elements with the following exceptions:

1. The root element is <lwn>.
2. <navObject>s may not nest.
3. <navStruct> may contain a mixture of <navObject>s and <navTarget>s.

4. <navList> is not allowed.

Brief descriptions of LWN elements whose content models differ from the corresponding NCX elements are included below. See section 8.2.1 for descriptions of all other elements.

- ❍ **<lwn>**
  Description: The root element
  Declaration: <!ELEMENT lwn (head, docTitle, navStruct)>
  Syntax: <lwn ...attributes...>...content...</lwn>
  Attributes:
  - ■ **lang (NMTOKEN, IMPLIED):** Specifies the RFC1766 language code of the language of the document.
  - ■ **dir ((ltr|rtl), IMPLIED):** The dir attribute specifies the direction of the text, where ltr is left to right and rtl is right to left.

  Valid inside: None

- ❍ **<navStruct>**
  Description: Container for primary navigation information.
  Declaration: <!ELEMENT navStruct (descr+, (navObject | navTarget)+)>
  Syntax: <navStruct ...attributes...>...content...</navStruct>
  Attributes:
  - ■ **id (ID, IMPLIED):** Optional unique identifier.

  Valid inside: <lwn>

- ❍ **<navObject>**
  Description: Contains description(s) of target and pointer to content.
  Declaration: <!ELEMENT navObject (descr+, content)>
  Syntax: <navObject ...attributes...>...content...</navObject>
  Attributes:
  - ■ **id (ID, REQUIRED):** Unique identifier matching the id attribute of the corresponding element in the XML and/or SMIL files. See section 8.7.1 for normative content.
  - ■ **onFocus (%script, IMPLIED):** May be used by player to provide location feedback to the user, e.g., highlight the current navObject on a visual display of the NCX.
  - ■ **onBlur (%script, IMPLIED):** May be used by player to provide location feedback to the user, e.g., remove highlighting when focus has shifted to another navObject on a visual display.
  - ■ **navType (CDATA, IMPLIED):** Describes the kind of structural object represented by this navObject, e.g. chapter, section.
  - ■ **levelNumber (CDATA, IMPLIED):** Indicates the nesting depth of this navObject.
  - ■ **value (CDATA, IMPLIED):** An integer representing the text content of a numeric label, e.g. a page number.
  - ■ **pageRef (IDREF, IMPLIED):** The id of the navTarget representing the page on which this navObject begins.

  Valid inside: <navStruct>

8.3 Navigation Modes

(This section is informative)

The digital talking book system supports two modes of navigation:

- ❍ **Global navigation** is controlled by the NCX and provides access primarily to relatively large chunks of the document. If the NCX represents a nested structure, it is possible to step forward or back through items located at the same level, and into smaller items or out to larger ones. The NCX also offers direct access to selected structures in the book such as page numbers, notes, figures, etc.

❍ **Local navigation** occurs after an NCX item has been selected and the user has begun reading. It comprises such actions as movement within a list or table, or among a group of words, sentences, or paragraphs. Local navigation is more implementation dependent; a simple player would allow jumping forward or back in time by a preset increment, or fast forward and rewind with audible feedback. A more complex version could allow jumps in all directions to selected elements in the marked-up text or SMIL files.

Both modes can use the same set of controls, and the user should be able to use the same methods to move through the material in either mode.

8.4 How NCX Works

8.4.1 NCX (Hierarchical) Version

(This section is informative)

Upon opening a document, a player will by default use the NCX *navStruct* to define the user's choices for navigation. The *navStruct* contains nested *navObject*s that represent the major divisions of the document. For example, the structure of the book whose NCX is shown in Section 8.5, Example 8.1 would look like this:

❍ Foreword..(Level 1)
   ■ History...................................(Level 2)
   ■ Development of Standards.....(Level 2)
❍ Standards..(Level 1)
   ■ 1 Core Services.....................(Level 2)
      ■ 1.1..............................(Level 3)
         ■ a........................(Level 4)
      ■ 1.2..............................(Level 3)

Foreword and Standards are at the same level, in this case the highest level, level 1. The nesting of *navObject*s allows the user to move directly between these objects without passing through the lower level divisions in between. From Foreword, the user can move to level 2 and step to any of the sections of Foreword. Since there is no level3 under Foreword, no smaller divisions can be accessed from the NCX. Such smaller divisions may be present, but they can only be reached through local navigation. The a. division of Standards is at level4, and can be reached by stepping through 1 Core Services and 1.1.

The user will also have the option of navigating to items that do not fit easily into the hierarchical structure of a document, e.g. pages, footnotes or sidebars. This function is provided by *navList*s. There is no nesting in *navList*s, all *navTarget*s are at the same level. In example 1, there are two *navLists*: the first contains three *navTarget*s representing page numbers, and the second contains three *navTarget*s representing notes.

Each *navObject* or *navTarget* provides navigation information about one piece of the document, e.g. a chapter heading, section number, page number, figure, etc. The *text* element contains the actual heading, page number, etc. for visual or text-to-speech presentation; the *audio* element uses SMIL 2.0 syntax to point to a clip containing the audio presentation of the same information. One or both should be used to give location feedback to the user. The *content* element provides a pointer to an ID within a SMIL file.

The required *structRef* attribute of *navTarget* allows synchronization of *navList*s with the *navStruct*. *structRef* points to the *navObject* that contains the page number, note, or other element referenced by the *navTarget*. Similarly, the *pageRef* attribute of *navObject* points to the *navTarget* representing the page on which the *navObject* begins.

8.4.2 LWN (Non-hierarchical) Version

(This section is normative)

The LWN file consists of a single list with references to every navigable element of the book. That is, the *navObject*s and *navTarget*s are intermingled, with each element in its proper order. Since players may choose to navigate with either NCX or LWN in different situations, the LWN file is required to contain the same information as the NCX file and it should be derived from a valid NCX file; the marked up text file or SMIL file(s) will often also be needed to achieve proper sequencing of non-hierarchical elements.

8.5 Navigation Metadata

(This section is normative)

Metadata is included in the <head> element using the <meta> tag. Content producers may introduce other metadata if needed.
- ❍ Required metadata:
    - ■ dtb:pageNormal - positive integer indicating the number of 'normal' pages in the DTB.
    - ■ dtb:pageSpecial - non-negative integer indicating the number of 'special' pages in the DTB.
    - ■ dtb:pageFront - non-negative integer indicating the number of 'front' pages in the DTB.
- ❍ Optional metadata:
    - ■ dtb:depth - positive integer indicating depth of structure of the DTB.
    - ■ dtb:maxPageNormal - non-negative integer indicating the content of the highest 'normal' <pagenum> in the DTB.
    - ■ dtb:generator - name and version of software that generated the NCX.

8.6 Examples

(This section is informative)

Example 8.1: NCX

```
<ncx>
    <head>
      <UID>us-nls-00001</UID>
        <charset>ISO-8859-1</charset>
        <meta name="dtb:pageNormal" content="47"/>
        <meta name="dtb:pageSpecial" content="0"/>
        <meta name="dtb:pageFront" content="5"/>
    </head>

    <docTitle>
        <text>Revised Standards and Guidelines of Service for the
Library of Congress
Network of Libraries for the Blind and Physically Handicapped 1995</text>
        <audio src="rs_title.mp3" />
    </docTitle>


    <navStruct>
        <navObject class="chapter" id="lvl1_3" pageRef="1" levelNumber="1">
          <descr>
            <text>Foreword</text>
            <audio src="rs_fwdx.mp3" clipBegin="00:01.5"
```

```
clipEnd="00:02.0" />
          </descr>
          <content src="sample.smil#h1_3" />
          <navObject class="section" id="lvl2_1" pageRef="1"
levelNumber="2">
            <descr>
              <text>History</text>
              <audio src="rs_fwdx.mp3" clipBegin="00:03.4"
clipEnd="00:03.9" />
            </descr>
            <content src="sample.smil#h2_1" />
          </navObject>
          <navObject class="section" id="lvl2_2" pageRef="2"
levelNumber="2">
            <descr>
              <text>Development of Standards</text>
              <audio src="rs_fwdx.mp3" clipBegin="00:56.3"
clipEnd="00:57.7" />
            </descr>
            <content src="sample.smil#h2_2" />
          </navObject>
        </navObject>
        ...
        <navObject class="chapter" id="lvl1_7" pageRef="16"
levelNumber="1">
          <descr>
            <text>Standards</text>
            <audio src="rs_stdx.mp3" clipBegin="00:01.3"
clipEnd="00:02.1" />
          </descr>
          <content src="sample.smil#h1_7" />
          <navObject class="section" id="lvl2_11" pageRef="16"
levelNumber="2">
            <descr>
              <text>1 Core Services</text>
              <audio src="rs_stdx.mp3" clipBegin="00:02.9"
clipEnd="00:04.9" />
            </descr>
            <content src="sample.smil#h2_10" />
            <navObject class="subsection" id="lvl3_1" pageRef="16"
levelNumber="3">
              <descr>
                <text>1.1</text>
                <audio src="rs_stdx.mp3" clipBegin="00:05.7"
clipEnd="00:06.7" />
              </descr>
              <content src="sample.smil#h3_1" />
              <navObject class="sub-subsection" id="lvl4_1"
pageRef="16" levelNumber="4">
                <descr>
                  <text>a.</text>
                  <audio src="rs_stdx.mp3" clipBegin="00:18.7"
clipEnd="00:19.1" />
                </descr>
```

```
                <content src="sample.smil#h4_1" />
              </navObject>
            </navObject>
            <navObject class="subsection" id="lvl3_2" pageRef="16"
levelNumber="3">
              <descr>
                <text>1.2</text>
                <audio src="rs_stdx.mp3" clipBegin="00:50.5"
clipEnd="00:51.4" />
              </descr>
              <content src="sample.smil#h3_2" />
            </navObject>
          </navObject>
        </navObject>
    </navStruct>


    <navList id="pages" class="pages">
      <navTarget class="page" id="p1" value="1" structRef="lvl1_3">
        <descr>
          <text>1</text>
          <audio src="rs_fwdx.mp3" clipBegin="00:00" clipEnd="00:00.9" />
        </descr>
        <content src="sample.smil#p1" />
      </navTarget>
      <navTarget class="page" id="p2" value="2" structRef="lvl2_2">
        <descr>
          <text>2</text>
          <audio src="rs_fwdx.mp3" clipBegin="00:53.9" clipEnd="00:54.6" />
        </descr>
        <content src="sample.smil#p2" />
      </navTarget>
      ...
      <navTarget class="page" id="p16" value="16" structRef="lvl1_7">
        <descr>
          <text>16</text>
          <audio src="rs_stdx.mp3" clipBegin="00:00.0" clipEnd="00:00.7" />
        </descr>
        <content src="sample.smil#p3" />
      </navTarget>
      ...
    </navList>


    <navList id="notes" class="notes">
      <navTarget class="note" id="nref_1" structRef="lvl2_2">
        <descr>
          <text>1</text>
          <audio src="rs_fwdx.mp3" clipBegin="01:22.6" clipEnd="01:23.5" />
        </descr>
        <content src="sample.smil#nref_1" />
      </navTarget>
      <navTarget class="note" id="nref_2" structRef="lvl2_2">
        <descr>
```

```
          <text>2</text>
          <audio src="rs_fwdx.mp3" clipBegin="02:00.6" clipEnd="02:01.4" />
        </descr>
        <content src="sample.smil#nref_2" />
      </navTarget>
      <navTarget class="note" id="nref_3" structRef="lvl2_2">
        <descr>
          <text>3</text>
          <audio src="rs_fwdx.mp3" clipBegin="03:13.3" clipEnd="03:14.1" />
        </descr>
        <content src="sample.smil#nref_3" />
      </navTarget>
    </navList>
</ncx>
```

Example 8.2: LWN

```
<lwn>
    <head>
      <UID>us-nls-00001</UID>
        <charset>ISO-8859-1</charset>
        <meta name="dtb:pageNormal" content="47"/>
        <meta name="dtb:pageSpecial" content="0"/>
        <meta name="dtb:pageFront" content="5"/>
    </head>

    <docTitle>
       <text>Revised Standards and Guidelines of Service for the
Library of Congress
Network of Libraries for the Blind and Physically Handicapped 1995</text>
       <audio src="rs_title.mp3" />
    </docTitle>


    <navStruct>
      <navTarget class="page" id="p1" value="1" structRef="lvl1_3">
        <descr>
          <text>1</text>
          <audio src="rs_fwdx.mp3" clipBegin="00:00" clipEnd="00:00.9" />
        </descr>
        <content src="sample.smil#p1" />
      </navTarget>
      <navObject class="chapter" id="lvl1_3" pageRef="1" levelNumber="1">
        <descr>
          <text>Foreword</text>
          <audio src="rs_fwdx.mp3" clipBegin="00:01.5" clipEnd="00:02.0" />
        </descr>
        <content src="sample.smil#h1_3" />
      </navObject>
      <navObject class="section" id="lvl2_1" pageRef="1" levelNumber="2">
        <descr>
          <text>History</text>
          <audio src="rs_fwdx.mp3" clipBegin="00:03.4" clipEnd="00:03.9" />
        </descr>
        <content src="sample.smil#h2_1" />
```

```
      </navObject>
      <navTarget class="page" id="p2" value="2" structRef="lvl2_2">
        <descr>
          <text>2</text>
          <audio src="rs_fwdx.mp3" clipBegin="00:53.9" clipEnd="00:54.6" />
        </descr>
        <content src="sample.smil#p2" />
      </navTarget>
      <navObject class="section" id="lvl2_2" pageRef="2" levelNumber="2">
        <descr>
          <text>Development of Standards</text>
          <audio src="rs_fwdx.mp3" clipBegin="00:56.3" clipEnd="00:57.7" />
        </descr>
        <content src="sample.smil#h2_2" />
      </navObject>
      <navTarget class="note" id="nref_1" structRef="lvl2_2">
        <descr>
          <text>1</text>
          <audio src="rs_fwdx.mp3" clipBegin="01:22.6" clipEnd="01:23.5" />
        </descr>
        <content src="sample.smil#nref_1" />
      </navTarget>
      <navTarget class="note" id="nref_2" structRef="lvl2_2">
        <descr>
          <text>2</text>
          <audio src="rs_fwdx.mp3" clipBegin="02:00.6" clipEnd="02:01.4" />
        </descr>
        <content src="sample.smil#nref_2" />
      </navTarget>
      <navTarget class="note" id="nref_3" structRef="lvl2_2">
        <descr>
          <text>3</text>
          <audio src="rs_fwdx.mp3" clipBegin="03:13.3" clipEnd="03:14.1" />
        </descr>
        <content src="sample.smil#nref_3" />
      </navTarget>
      ...
      <navTarget class="page" id="p16" value="16" structRef="lvl1_7">
        <descr>
          <text>16</text>
          <audio src="rs_stdx.mp3" clipBegin="00:00.0" clipEnd="00:00.7" />
        </descr>
        <content src="sample.smil#p3" />
      </navTarget>
      <navObject class="chapter" id="lvl1_7" pageRef="16" levelNumber="1">
        <descr>
          <text>Standards</text>
          <audio src="rs_stdx.mp3" clipBegin="00:01.3" clipEnd="00:02.1" />
        </descr>
        <content src="sample.smil#h1_7" />
      </navObject>
          <navObject class="section" id="lvl2_11" pageRef="16"
levelNumber="2">
        <descr>
```

```
        <text>1 Core Services</text>
        <audio src="rs_stdx.mp3" clipBegin="00:02.9" clipEnd="00:04.9" />
      </descr>
      <content src="sample.smil#h2_10" />
    </navObject>
    <navObject class="subsection" id="lvl3_1" pageRef="16"
levelNumber="3">
      <descr>
        <text>1.1</text>
        <audio src="rs_stdx.mp3" clipBegin="00:05.7" clipEnd="00:06.7" />
      </descr>
      <content src="sample.smil#h3_1" />
    </navObject>
    <navObject class="sub-subsection" id="lvl4_1" pageRef="16"
levelNumber="4">
      <descr>
        <text>a.</text>
        <audio src="rs_stdx.mp3" clipBegin="00:18.7" clipEnd="00:19.1" />
      </descr>
      <content src="sample.smil#h4_1" />
    </navObject>
    <navObject class="subsection" id="lvl3_2" pageRef="16"
levelNumber="3">
      <descr>
        <text>1.2</text>
        <audio src="rs_stdx.mp3" clipBegin="00:50.5" clipEnd="00:51.4" />
      </descr>
      <content src="sample.smil#h3_2" />
    </navObject>
  </navStruct>
</lwn>
```

8.7 NCX Production Implications

8.7.1 IDs

(This section is normative)

The id attribute on *navObject* and *navTarget* must be equal to the id attribute of the corresponding element in the XML and/or SMIL files.

8.7.2 DTBs Spanning Multiple Media Units

(This section is normative)

When a DTB spans several distribution media (e.g., multiple CD-ROMs), the audio clips referenced in the NCX must be copied from the audio content files into a single file that is replicated on each piece of distribution media. Creating such a file will ensure that the full NCX is functional on each piece of media. The NCX must also be constructed in a slightly different manner than ordinarily, in that the *audio* element must point to clips in a special file, instead of to clips in the audio content file(s). See Section 11, Distribution Information for further discussion of this issue.

Contents

9. Portable Bookmarks and Highlights

9.1 Introduction

(This section is normative)

This standard establishes a specific XML file format to support bookmark and highlight export and import. A playback system may allow readers to set bookmarks and to highlight passages in a document, label the marked sections with text or audio notes, and export the resulting collection of marks and notes to other compliant playback devices. Bookmarks and highlights, and their associated notes, if any, are stored within the player, separate from the DTB itself.

This standard does not require that all compliant players support all of the functionality described above. In addition, this standard places no constraints on a playback system's internal system for storing or manipulating the information in the bookmark file. However, if a player supports the export of bookmarks and highlights and their associated audio notes, the player must format the information as described below when it is exported. See Appendix 2 -- DTD for Portable Bookmarks/Highlights.

If a playback device supports user-recording of audio notes on bookmarks or highlights that may be exported, the recording may be in any format supported by the standard. The format is implicit in the suffix of the filename. The playback device must generate this suffix when generating the filename.

Bookmark files shall be named, by default, with the value from *uid* and the extension ".bmk". For example: "us-nls-14339.bmk". Players may allow users to apply their own filenames to accommodate character limitations in other filesystems and to avoid filename collisions. To accommodate user-supplied names, players with bookmark import capabilities must be able to open bookmark files and read the *uid* value to match the correct bookmark file with the current DTB.

Players may implement a variety of systems for numbering or otherwise identifying bookmarks or highlighted sections so the user can step through and choose from a group of them, but the default order in which they are numbered must be the order in which they fall in the document. When exported, bookmarks and highlights shall be in the default order.

This standard prescribes no methods for setting or accessing bookmarks or highlights, for notifying the user of their presence, or for controlling the export or import process. Player manufacturers are free to design the user interface for bookmarks/highlights in whatever manner integrates most effectively into the overall user interface of the playback device.

9.2 Bookmark/Highlight Elements

(This section is informative)

Brief descriptions of the Bookmark/Highlight elements follow. Each includes the element declaration extracted from the Bookmark DTD found in Appendix 2, along with descriptions of any applicable attributes.

  ❍ **<bookmarkSet>**
    Description: The root element in the Bookmark/Highlight DTD. Contains all data pertaining to bookmarks, highlights, and lastmarks for a given DTB.
    Declaration: <!ELEMENT bookmarkSet (title, uid, lastmark, (bookmark | hilite)*) >
    Syntax: <bookmarkSet>...content...</bookmarkSet>
    Attributes: None
    Valid inside: None

  ❍ **<title>**
    Description: Contains the title, in text and in an optional audio clip, of the DTB for which the bookmark set was created.
    Declaration: <!ELEMENT title (text, audio?) >

Syntax: <title>...content...</title>
Attributes: None
Valid inside: <bookmarkSet>
Comments: When bookmark sets are exported to other compliant playback devices, the title will allow users to identify and manage them.

❍ **<text>**
Description: Text of title or note.
Declaration: <!ELEMENT text (#PCDATA)>
Syntax: <text>...content...</text>
Attributes: None
Valid inside: <title>, <note>

❍ **<audio>**
Description: Audio clip of title or note, in any format supported by standard.
Declaration: <!ELEMENT audio EMPTY >
Syntax: <audio...attributes... />
Attributes:

■ **src (%uri, #REQUIRED):** The src attribute holds the URI of the audio file that contains the referenced clip.
■ **clipBegin (CDATA, IMPLIED):** The clipBegin attribute specifies the beginning of a segment of a continuous media object as a time offset from the start of the media object. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module [SMILCLOCK].
■ **clipEnd (CDATA, IMPLIED):** The clipEnd attribute specifies the end of a segment of a continuous media object as a time offset from the start of the media object. It uses the same attribute value syntax as clipBegin.

Valid inside: <title>, <note>
Comments: If clipBegin is not present the media object is played from its beginning. If clipEnd is not present, the media object is played to its end. If the value of the clipEnd attribute exceeds the duration of the media object, the value is ignored, and the media object is played to its end.

❍ **<uid>**
Description: Globally unique identifier for the book, drawn from the package file. Matches the identifier referenced by the "unique-identifier" attribute on <package> element.
Declaration: Syntax:<uid>...content...</uid>
Attributes: None
Valid inside: <bookmarkSet>

❍ **<lastmark>**
Description: Location where reader left off and where player will resume play when restarted. Location consists of a URI pointing to the id attribute of the <par> or <seq> element in the SMIL file that contains the lastmark, plus a time offset or character offset to the exact point.
Declaration: <!ELEMENT lastmark (ncxRef, uri, (timeoffset | charoffset)) >
Syntax:<lastmark>...content...</lastmark>
Attributes: None
Valid inside: <bookmarkSet>
Comments: The <lastmark> is set automatically by the playback device.

❍ **<ncxRef>**
Description: Captures current location in NCX (the id of the current navObject)at time lastmark, bookmark, or highlight is set. Ensures that current location in NCX and SMIL are synchronized after moving to a lastmark, bookmark, or highlight so that any global navigation commands issued by the user will start from the current location.
Declaration: <!ELEMENT ncxRef (#PCDATA)>
Syntax:<ncxRef>...content...</ncxRef>

Attributes: None
Valid inside: <lastmark>, <bookmark>, <hilStart>, <hilEnd>

❍ **<uri>**
Description: Pointer to id of <par> or <seq> in SMIL, to id in text-only file, or to audio file that contains the <lastmark>, <bookmark>, <hilStart>, or <hilEnd>.
Declaration: <!ELEMENT uri (#PCDATA)>
Syntax:<uri>...content...</uri>
Attributes: None
Valid inside: <lastmark>, <bookmark>, <hilStart>, <hilEnd>
Comments: The <uri> points to the id of the <par> or <seq> in the SMIL file that contains the <lastmark>, <bookmark>, <hilStart>, or <hilEnd>.

❍ **<timeoffset>**
Description: Exact position of <lastmark>, <bookmark>, <hilStart> or <hilEnd> in audio file referenced (via the SMIL file) by the uri ; in seconds, measured from beginning of audio file.
Declaration: <!ELEMENT timeoffset (#PCDATA) >
Syntax:<timeoffset>...seconds.fraction...</timeoffset>

      Seconds ::= DIGIT+
      Fraction ::= 3DIGIT

Attributes: None
Valid inside: <lastmark>, <bookmark>, <hilStart>, <hilEnd>

❍ **<charoffset>**
Description: Exact position of bookmark, lastmark, hilStart, or hilEnd in text-only file referenced by the uri: in bytes, measured from beginning of referenced segment.
Declaration: <!ELEMENT charoffset (#PCDATA) >
Syntax:<charoffset>...content...</charoffset>
Attributes: None
Valid inside: <lastmark>, <bookmark>, <hilStart>, <hilEnd>

❍ **<bookmark>**
Description: Point in document marked by user for direct access in future. Bookmark consists of location and optional note. Location consists of a uri pointing to the id attribute of the <par> or <seq> element in the SMIL file that contains the bookmark, plus a time offset or character offset to the exact point.
Declaration: <!ELEMENT bookmark (ncxRef, uri, (timeoffset | charoffset), note?) >
Syntax:<bookmark>...content...</bookmark>
Attributes:

    ■ **label (CDATA, #IMPLIED):** optional attribute for use in storing identifying label to assist user in choosing among a set of bookmarks.

Valid inside: <bookmarkSet>
Comments: Player shall by default automatically number bookmarks and hilites in the order in which they fall in the book.

❍ **<note>**
Description: Holds the user's label for or thoughts about a bookmark or highlighted section. It can be text or audio or both.
Declaration: <!ELEMENT note (text?, audio?) >
Syntax:<note>...content...</note>
Attributes: None
Valid inside: <hilite>, <bookmark>

Comments: Playback devices supporting recording of audio <notes> need not support recording in all of the codecs allowed by this standard.

❍ **<hilite>**
Description: A block of text marked by the user with an optional note attached.
Declaration: <!ELEMENT hilite (hilStart, hilEnd, note?) >
Syntax: <hilite>...content...</hilite>
Attributes:

■ **label (CDATA, #IMPLIED):** optional attribute for use in storing identifying label to assist user in choosing among a set of highlights.

Valid inside: <bookmarkSet>
Comments: Player shall by default automatically number bookmarks and hilites in the order in which they fall in the book.

❍ **<hilStart>**
Description: Starting point of highlighted block. Location consists of a uri pointing to the id attribute of the <par> or <seq> element in the SMIL file that contains the beginning of the highlighted section, plus a time offset or character offset to the exact point.
Declaration: <!ELEMENT hilStart (ncxRef, uri, (timeoffset | charoffset)) >
Syntax: <hilStart>...content...</hilStart>
Attributes: None
Valid inside: <hilite>

❍ **<hilEnd>**
Description: End of highlighted block. Location consists of a uri pointing to the id attribute of the <par> or <seq> element in the SMIL file that contains the end of the highlighted section, plus a time offset or character offset to the exact point.
Declaration: <!ELEMENT hilEnd (ncxRef, uri, (timeoffset | charoffset)) >
Syntax: <hilEnd>...content...</hilEnd>
Attributes: None
Valid inside: <hilite>

9.3 Examples

(This section is informative)

In Example 9.1, the reader has set two bookmarks, one in chapter 1, 22 seconds from the start of paragraph 8 and the other in chapter 3, 88 seconds from the start of paragraph 12. The reader has added the text note "Atlanta burns" to the second bookmark. The user has also highlighted a passage in chapter 4 beginning at the start of paragraph 1 and ending 246 seconds after the start of paragraph 6, labeling it with a ten-second audio comment. The reader last stopped reading (as indicated by the <lastmark>) in chapter 5, paragraph 23. The default filename for this bookmark file would be "us-nls-00065.bmk."

Example 9.1:

```
<?xml version="1.0" encoding="ISO-8859-1" ?> <!DOCTYPE bookmarkSet SYSTEM
"bookmark.dtd"> <bookmarkSet>
    <title>
        <text>Gone with the Wind</text>
        <audio src="gwtw_title.mp3" />
    </title>
    <uid>us-nls-00065</uid>
```

```
    <lastmark>
        <ncxRef>gwtw.ncx#lvl1_5</ncxRef>
        <uri>gwtw_ch5.smil#para023</uri>
        <timeoffset>173</timeoffset>
    </lastmark>

    <bookmark>
        <ncxRef>gwtw.ncx#lvl1_1</ncxRef>
        <uri>gwtw_ch1.smil#para008</uri>
        <timeoffset>22</timeoffset>
    </bookmark>

    <bookmark>
        <ncxRef>gwtw.ncx#lvl1_3</ncxRef>
        <uri>gwtw_ch3.smil#para012</uri>
        <timeoffset>88</timeoffset>
        <note>
            <text>Atlanta burns.</text>
        </note>
    </bookmark>

    <hilite>
        <hilStart>
            <ncxRef>gwtw.ncx#lvl1_4</ncxRef>
            <uri>gwtw_ch4.smil#para001</uri>
            <timeoffset>0</timeoffset>
        </hilStart>
        <hilEnd>
            <ncxRef>gwtw.ncx#lvl1_4</ncxRef> <uri>gwtw_ch4.smil#para006</uri>
            <timeoffset>246</timeoffset>
        </hilEnd>
        <note>
            <audio src="us-nls-00065.wav" clipBegin="00:00.00"
            clipEnd="00:10.00" />
        </note>
    </hilite>
</bookmarkSet>
```

Example 9.2 shows a text-only file in which the reader last stopped reading 130 characters after the start of paragraph 297.

Example 9.2:

```
...
<bookmarkSet>
    <title>
        <text>Chemistry Today</text>
    </title>
    <uid>us-rfbd-MM499</uid>
```

```
    <lastmark>
        <ncxRef>chemtd.ncx#lvl1_3</ncxRef>
        <uri>chemtd.xml#para297</uri>
        <charoffset>130</charoffset>
    </lastmark>
</bookmarkSet>
```

[Contents](#)

10. Resource File

10.1 Introduction

(This section is informative)

The Resource File supplies text segments, audio clips, or images that may assist the user in navigating a document. These media objects or "resources" provide information missing from a document or present only in a form inaccessible to the reader. Some examples of applications are:

1. Documents with definite structures, but missing headings, such as books with multi-level structures but no headings on items below the level of sections. The resource file could contain the word "subsection" for presentation to the reader when stepping through the document via the NCX.

2. Player implementations that present generic labels such as "level 2" when the user changes levels in the NCX. The resource file could present the actual name of items found at that level in that specific context, e.g., "chapter."

3. "Where Am I?" applications, especially in situations where headings are absent.

4. Detailed information about the marked-up text file; useful when exact knowledge of a document's finest structure is essential. The resource file could contain text or audio clips of all of the element names in the DTBook3 DTD, alerting the user when crossing the boundaries of paragraphs, list items, table cells, etc.

5. Information about the types of text structures that the user may choose to "turn off" via the SMIL file (see section 7.4.3 "Skippable Structures"). The Resource File could contain text segments or audio clips of the names of the structures affected; for example, page numbers, notes, or sidebars.

6. Names of the items listed in the Guide.

The Resource File, then, can contain three types of information:

1. Resources tailored to a given document, for use primarily during global navigation. As the user navigates via the NCX, the player, when necessary, will look in the Resource File to locate the *navTypeResource* with the same *class* attribute as the current NCX *navObject*.

2. Generic representations of the names of elements from the DTBook3 DTD, for use during local navigation. As the reader issues local navigation commands referencing the text file, the player will use the name of the current element in the text file to locate the *elementResource* with that element name in its *elementRef* attribute. For example, encountering a paragraph (tagged with <p>...</p>) would call the *elementResource* with *elementRef* equal to "p".

In addition, the *classRef* attribute on *elementResource* allows the DTB producer to create *elementResource*s tailored to elements with specific class names. For example, different *elementResource*s could be created for <w class="reservedword">...</w> and <w class="variablename">...</w>.

3. Representations of components of the Guide or SMIL file. The player will use the "id" of a customTest element in SMIL or the "type" of an entry in the Guide to locate the corresponding *idResource*. For example, the customTest element tagged <customTest id="pagenum" />) would call the *idResource* with *idRef* equal to "pagenum".

The text, audio, and image alternatives allow a *navTypeResource*, *elementResource* or *idResource* to be presented in a

medium appropriate to the playback system's capabilities and the user's preferences. Images are conceived as holding iconic representations of heading types. The Resource File, if present, will be created by the DTB producer and distributed as part of the DTB.

10.2 Resource Elements

(This section is normative)

The Resource File is an XML version 1.0 file defined by the Document Type Definition resource.dtd, to which it must comply. See [Appendix 3, "DTD for Resource File."](#) The resourcefile shall be named with the extension ".res". The Resource File entry in the Package File manifest must have an id value equal to "resource".

Brief descriptions of the elements follow. Each includes the element declaration extracted from the Resource DTD, along with descriptions of any applicable attributes.

- **\<navResource\>**
Description: The root element in the Resource DTD
Declaration: <!ELEMENT navResource (navTypeResource | elementResource | idResource)+ >
Syntax: <navResource>...content...</navResource>
Attributes: None
Valid inside: None

- **\<navTypeResource\>**
Description: Contains text, audio, or image serving as an alternative representation of navObjects with a common navType attribute in the NCX.
Declaration: <!ELEMENT navTypeResource (text?,audio?,image?) >
Syntax: <navTypeResource...attributes...>...content...</navTypeResource >
Attributes:
    - **navTypeRef (CDATA, REQUIRED):** Specifies the name of the navType attribute on the NCX element for which the *navTypeResource* is supplied.
    - **lang (NMTOKEN, IMPLIED):** Specifies the language of the resource item, using an RFC1766 language code.
Valid inside: <navResource>
Comments: Alternative representation can be used to convey navigational information, e.g., text can be used for screen or braille display, audio for digital talking book players, and image for screen display.

- **\<elementResource\>**
Description: Contains text, audio, or image serving as alternative representation of a given XML element in the document text file.
Declaration: <!ELEMENT elementResource (text?,audio?,image?) >
Syntax: <elementResource...attributes...>...content...</elementResource >
Attributes:
    - **elementRef (CDATA, REQUIRED):** Specifies the name of the XML element for which the *elementResource* is supplied.
    - **classRef (CDATA, IMPLIED):** Specifies the class name of the XML element for which the *elementResource* is supplied.
    - **lang (NMTOKEN, IMPLIED):** Specifies the language of the resource item, using an RFC1766 language code.
Valid inside: <navResource>
Comments: Alternative representation can be used to convey navigational information, e.g., text can be used for screen or braille display, audio for digital talking book players, and image for screen display.

- **<idResource>**
  Description: Contains text, audio, or image serving as alternative representation of a given XML element in the Guide or SMIL file.
  Declaration: <!ELEMENT idResource (text?,audio?,image?) >
  Syntax: <idResource...attributes...>...content...</idResource >
  Attributes:
  - **idRef (CDATA, REQUIRED):** Specifies the name of the id attribute on the customTest element in SMIL or, alternatively, the type attribute on an entry in the Guide, for which the *idResource* is supplied.
  - **lang (NMTOKEN, IMPLIED):** Specifies the language of the resource item, using an RFC1766 language code.

  Valid inside: <navResource>
  Comments: Alternative representation can be used to convey navigational information, e.g., text can be used for screen or braille display, audio for digital talking book players, and image for screen display.

- **<text>**
  Description: Contains the text used for alternative representation.
  Declaration: <!ELEMENT text (#PCDATA) >
  Syntax: <text>...content...</text>
  Attributes: None
  Valid inside: <navTypeResource>, <elementResource>, <idResource>

- **<audio>**
  Description: Points to file containing audio representation and provides time offsets of beginning and end of clip.
  Declaration: <!ELEMENT audio EMPTY >
  Syntax: <audio...attributes... />
  Attributes:
  - **src (CDATA, REQUIRED):** URI of the audio file.
  - **clipBegin (CDATA, IMPLIED):** Specifies the beginning of a segment of a continuous media object as offset from the start of the media object. This offset is a real-time measurement of normal media playback. If clipBegin is not used the media object is played from the beginning of source. See section 7.5 Clock Values for valid syntax.
  - **clipEnd (CDATA, IMPLIED):** Specifies the end of a segment of a continuous media object as offset from the start of the media object. This offset is a real-time measurement of normal media playback. It uses the same attribute value syntax as the clipBegin attribute. If clipEnd is not used the media object is played to the end of the source. If the value of the clipEnd attribute exceeds the duration of the media object, the value is ignored, and the clip end is set equal to the effective end of the media object. Reference for clipBegin and clipEnd: SMIL 2.0, Module 11, [SMILtiming] The SMIL Timing and Synchronization Module. Copyright © 2000 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. http://www.w3.org/Consortium/Legal/.

  Valid inside: <navTypeResource>, <elementResource> ;, <idResource>

- **<image>**
  Description: Points to file containing the image.
  Declaration: <!ELEMENT image EMPTY >
  Syntax: <image...attributes... />
  Attributes:
  - **src (CDATA, REQUIRED):** URI of the image file.

  Valid inside: <navTypeResource>, <elementResource>;, <idResource>

10.3 Examples

(This section is informative)

In Example 10.1, the Resource File contains a navTypeResource for the word "chapter," and elementResources for four selected DTBook3 elements. The fourth elementResource uses the classRef attribute to specify a given class of the element <code>.

Example 10.1:

```
<navResource>
      <navTypeResource navTypeRef="chapter" lang="en">
           <text>Chapter</text> <audio src="chapter.mp3" /> <image src="chapter.gif" />
      </navTypeResource>
      <elementResource elementRef="li" lang="en">
           <text>list item</text> <audio src="elemres.mp3"
                 clipBegin="00:36" clipEnd="00:38.14" />
      </elementResource>
      <elementResource elementRef="p" lang="en">
           <text>paragraph</text> <audio src="elemres.mp3"
                 clipBegin="00:47.51" clipEnd="00:49.34" />
      </elementResource>
      <elementResource elementRef="td" lang="en">
           <text>table cell</text> <audio src="elemres.mp3"
                 clipBegin="01:22.12" clipEnd="01:24.01" />
      </elementResource>
      <elementResource elementRef="code" classRef="javascript" lang="en">
           <text>javascript</text> <audio src="elemres.mp3"
                 clipBegin="01:45.15" clipEnd="01:47.01" />
      </elementResource>
... </navResource>
```

In Example 10.2, navTypeResources are supplied in both English and Danish for a book whose NCX carries English class names on navObjects (e.g., "chapter"). The "lang" attribute on navTypeResource controls which will be presented to the reader.

Example 10.2:

```
      <navTypeResource navTypeRef="chapter" lang="da">
           <text>Kapitel</text>
           <audio src="kapitel.mp3" clipBegin="00:00" clipEnd="00:02.23" />
           <image src="Kapitel.png" />
      </navTypeResource>
      <navTypeResource navTypeRef="chapter" lang="en">
           <text>Chapter</text>
           <audio src="chapter.mp3" clipBegin="00:00" clipEnd="00:02.01" />
           <image src="chapter.png" />
```

</navTypeResource>

In Example 10.3, the Resource File contains idResources for two customTest elements with ids of "pagenum" and "note".

Example 10.3:

<idResource idRef="pagenum" lang="en">

    <text>page numbers</text>

    <audio src="idres.mp3" clipBegin="00:00" clipEnd="00:01.17" />

</idResource>

<idResource idRef="note" lang="en">

    <text>notes</text>

    <audio src="idres.mp3" clipBegin="00:02" clipEnd="00:02.72" />

</idResource>

Contents

11. Packaging Files for Distribution

11.1 Introduction

(This section is informative)

If DTBs are distributed on a physical medium such as CD-ROM, producers will sometimes put more than one book on a disk and sometimes use more than one disk to hold a single book. When multiple DTBs are included on a single distribution medium, a simple method of storing this information for easy access by the player is needed, to present to the reader a "bookshelf" of books. When a single DTB spans several distribution media, the player needs access to specific information so that it can provide correct instructions to the reader, e.g., "Insert disk 2," when required. The "Distribution Information File" (or "Distinfo File") stores the data needed for these purposes.

In the following scenarios, the player would need accurate "distribution information" to respond appropriately:

1. Trying to reach an NCX target that lies on another disk.
2. Trying to reach a bookmark or highlight that is on another disk.
3. Resuming reading on a different disk than last ended on. "Lastmark" will point to another disk.
4. Following a cross-reference or other link pointing to a target on another disk.
5. Hitting the "Retrace" or "Back" button after inserting a new disk to follow a link, . The "history file" will point back to the first disk.
6. Reading notes during normal playback. If the notes are printed at the end of the chapter or book, and are recorded separately from the text where they are referenced, they may fall on a different disk from the noterefs.
7. Reaching the end of a disk of a multidisk book. This might be handled in another way, but could be implemented using the Distinfo File.

A Distinfo File would normally be created for each type of distribution medium, whereas other DTB files would be unchanged regardless of how a DTB is distributed.

11.2 Distinfo Requirements

(This section is normative)

When a DTB spans several distribution media, all files referenced by a SMIL file must be included on the same disk as the given SMIL file. This requirement ensures that players need only track the location of SMIL files in order to provide a complete DTB presentation.

The Distinfo File is required when a DTB spans several distribution media or when multiple DTBs are contained on one piece of media. Otherwise, a Distinfo File is optional. If present on one piece of media, it must be present on all. Players must be able to locate a DTB's package file even if a Distinfo File is not present.

The Distinfo File, if present, shall be named "distinfo," with no extension. There shall be no more than one Distinfo File per media unit (e.g., CD-ROM disk).

Distribution on multiple media units has implications for the production of the NCX. See section 8.7.2 DTBs Spanning Multiple Media Units.

Optional *changemsg*s may be used to supply customized messages instructing user how to proceed when another piece of distribution media is needed to continue reading. If no *changemsg* is present when required, the player must render a default announcement (e.g., "please insert disk 2").

11.3 Distinfo Elements

(This section is normative)

❍ **<distinfo>**
Description: The root element of a Distinfo File.
Declaration: <!ELEMENT distinfo (book+) >
Syntax: <distinfo>...content...</distinfo>
Attributes: None
Valid inside: None

❍ **<book>**
Description: Identifies a DTB that is present, in part or whole, on this piece of distribution media.
Declaration: <!ELEMENT book (distmap?) >
Syntax: <book ...attributes...>...content...</book>
Attributes:

- **uid (CDATA, REQUIRED):** Matches the unique identifier for the given book, drawn from the package file.
- **pkgref (CDATA, REQUIRED):** The URI of the book's package file.
- **media (CDATA, IMPLIED):** If the book spans two or more pieces of media, the *media* attribute identifies the piece of media in hand, in the format "x:y", where x is the sequence number of this piece of media, and y is the total number of media pieces in the distribution of this book.

Valid inside: <distinfo>
Comments: Contains zero or one *distmap*s.

❍ **<distmap>**
Description: A map identifying which piece of media the various SMIL files reside upon.
Declaration: <!ELEMENT distmap (smilref+) >
Syntax: <distmap>...content...</distmap>
Attributes: None
Valid inside: <book>
Comments: Contains one or more *smilref*s.

❍ **<smilref>**

Description: A reference to a DTB SMIL file.
Declaration: <!ELEMENT smilref EMPTY >
Syntax:
Attributes:

- **file (CDATA, REQUIRED):** The filename of the given SMIL file.
- **mediaref (CDATA, REQUIRED):** Identifies the piece of media on which the given SMIL file resides, in the format "x:y", where x is the sequence number of this piece of media, and y is the total number of media pieces in the distribution of this book.

Valid inside: <distmap>
Comments: Contains the filenames of the SMIL files as they appear in the package file manifest.

❍ **<changemsg>**
Description: A pointer to a custom message to be read when a new disk is requested by the reading system.
Declaration: <!ELEMENT changemsg EMPTY>
Syntax:
Attributes:

- **href(CDATA, REQUIRED):** URI pointing to a SMIL file or file fragment that represents the message.
- **mediaref (CDATA, REQUIRED):** Identifies the piece of media for which this message is intended and matches the "media" attribute of and the "mediaref" attribute of ; in the format "x:y", where x is the sequence number of this piece of media, and y is the total number of media pieces in the distribution of this book.

Valid inside: <book>

11.4 Examples

(This section is informative)

Example 11.1 shows the Distinfo File for the first disk of a book that spans two CD-ROMs. The book element identifies the book through the *uid* attribute, points to the package file via *pkgref* and indicates in the *media* attribute that this disk is the first of two. Players would parse the package file to obtain book metadata, etc. The distmap element contains a *smilref* for each SMIL file in the book (there are 10 in this particular case). The *file* attribute gives the name of each individual SMIL file. The *mediaref* attribute indicates which disk that particular SMIL file (and all audio/text/image files referenced by it) resides upon (the first 5 are on disk 1, the remainder on disk 2).

Players would refer to this map when a particular SMIL file is targeted for playback; if the file is not present on the current disk, a "please insert disk X" announcement would be played.

Example 11.1:
```
<distinfo>
    <book uid="US-RFBD-TBFZ284" pkgref="./FZ284.opf" media="1:2">
        <distmap>
            <smilref file="FZ284_0001d.smil" mediaref="1:2"/>
            <smilref file="FZ284_0002d.smil" mediaref="1:2"/>
            <smilref file="FZ284_0003d.smil" mediaref="1:2"/>
            <smilref file="FZ284_0004d.smil" mediaref="1:2"/>
            <smilref file="FZ284_0005d.smil" mediaref="1:2"/>
            <smilref file="FZ284_006d.smil" mediaref="2:2"/>
            <smilref file="FZ284_007d.smil" mediaref="2:2"/>
            <smilref file="FZ284_008d.smil" mediaref="2:2"/>
            <smilref file="FZ284_009d.smil" mediaref="2:2"/>
            <smilref file="FZ284_0010d.smil" mediaref="2:2"/>
        </distmap>
```

```
        </book>
</distinfo>
```

In Example 11.2, a sample Distinfo File is presented for a case where two books are included on one CD-ROM. The file contains pointers to two book package files. Both books are complete on this one piece of media so the *media* attribute is omitted. Players would parse the package files to obtain book metadata, etc.

Example 11.2:
```
<distinfo>
    <book uid="US-NLS-DTB00001" pkgref="./book1/AllAboutDogs.opf" />
    <book uid="US-NLS-DTB98765" pkgref="./book2/AllAboutCats.opf" />
</distinfo>
```
Contents

12. Presentation Styles

(This section is informative)

The W3C has defined mechanisms for separating content from presentation called the Cascading Style Sheet [CSS]and Extensible Style Language [XSL]. CSS (for which two levels of functionality are defined, Level 1 [CSS1] and Level 2 [CSS2]) and XSL allow specific formatting rules for mark-up to be defined and stored independent of the actual content. Default rules are normally applied by the specific playback or rendering system. The CSS *Cascade* provides a defined mechanism in which style rules may be applied by the content author as well as by the user.

CSS or XSL files may be provided by the content producer to control visual formatting of textual content when a DTB is played on a system that incorporates a visual display.

If a refreshable braille display is connected to a DTB player, a braille style sheet can control formatting so that the document is more easily navigable.

Audio CSS (ACSS, part of CSS2) and XSL also support the aural equivalent of visual formatting, and allow for audio cues to be associated with textual content mark-up. For example, chapter starts or page breaks may be indicated with a specific audio cue. As with visually oriented CSS, an author- or producer-supplied audio style sheet is optional.

Style sheets are optional components of DTBs and DTB distribution systems. DTB producers may choose to supply default style sheets for any of the above three categories. DTBs referencing style sheets should do so using standard W3C mechanisms to link an XML source to its style sheet.

Playback systems that utilize common PC-based browsers should support presentation styles to the extent the browser itself does. Portable players will not normally provide full support for style sheets but may implement a subset of CSS or XSL sufficient for DTB use and the media presented on the player. For example, an audio-only player that is aware of the textual content might support only the audio styles described above.

Developers of playback systems may implement user interface features that support local control of style sheets, that is, allow the user to define styles that supersede default or producer-defined styles; however, this standard does not require it.

Contents

13. Types of DTB

13.1 Types

(This section is informative)

Digital talking books produced in compliance with this standard fall into seven types representing the proportions in which six key file types are present. A DTB which incorporates audio and text files for the full contents of the document, as well as a structured navigation control file (NCX) for efficient navigation, offers the most features to a reader.

Type 1 -- Full audio only

> The full contents of the document are present in recorded speech. The DTB contains no structure. The navigation control file (NCX) contains only the title of the book and a pointer to the first SMIL file. The Package File *spine* defines the linear reading order. This Type of DTB will be represented primarily by "legacy" titles transferred from analog to digital form. Direct navigation to points within the DTB is not possible for the reader.

Type 2 -- Full audio with structure

> The full contents of the document are present in recorded speech. The NCX contains (via SMIL) links to the structural elements of the book and may contain links to features such as page numbers, narrated footnotes, etc. The reader can navigate directly only to items included in the NCX. The Package File *spine* defines the linear reading order.

Type 3 -- Full audio with structure and partial text

> The full contents of the document are present in recorded speech. The NCX contains (via SMIL) links to the structural elements of the book and may contain links to features such as page numbers, narrated footnotes, etc. In addition, part of the document is present as a text file marked up with the DTBook3 element set. The segments included in the text file might be chosen either because keyword searching, spelling, and direct access to the text would be beneficial in those segments (e.g., glossaries), or because a synthetic speech rendering would be as useful as a human speech version (e.g., indexes). Images may also be included. The reader can navigate directly to items in the NCX and to tagged items in the text file. Where both text and audio renditions of a segment are present, they are synchronized. The Package File *spine* defines the linear reading order.

Type 4 -- Full audio with structure and full text

> The full contents of the document are present in recorded speech. The NCX contains (via SMIL) links to the structural elements of the book and may contain links to features such as page numbers, narrated footnotes, etc. The full text of the document is present as a text file marked up with the DTBook3 element set. Images may also be included. The reader can navigate directly to items in the NCX and to tagged items in the text file. Where both text and audio renditions of a segment are present, they are synchronized. The Package File *spine* defines the linear reading order.

Type 5 -- Full text with structure and partial audio

> The full text of the document is present as a text file marked up with the DTBook3 element set but only part of the document is present as recorded speech. Images may also be included. The NCX contains (via SMIL) links to the structural elements of the book and may contain links to features such as page numbers, narrated footnotes, etc. The reader can navigate directly to items in the NCX and to tagged items in the text file. Where both text and audio renditions of a segment are present, they are synchronized. The Package File *spine* defines the linear reading order. This type includes books such as dictionaries, where the full text is present but the only audio contains human speech recordings of word pronunciations

Type 6 -- Full text with structure but no audio

The full text of the document is present as a text file marked up with the DTBook3 element set. The NCX contains (via SMIL) links to the structural elements of the book and may contain links to features such as page numbers. The reader can navigate directly to items in the NCX and to tagged items in the text file. The Package File *spine* defines the linear reading order. Images may be included. There are no audio files.

Type 7 -- Full text only

The full text of the document is present as a text file marked up with the DTBook3 element set. The NCX contains only the title of the book and a pointer to the first SMIL file. The reader can navigate directly to tagged items in the text file. The Package File *spine* defines the linear reading order. Images may be included. There are no audio files.

The following table shows the seven types of DTB and whether each of six file types is required (R), optional (O), or not applicable (N/A). Note that the Open eBook Package File (OPF), the navigation control file (NCX), and the SMIL file(s) are required for all types, although the latter two may serve merely as pointers to other files in some cases.

DTB Type	OPF	NCX	Audio	Text	SMIL	Image
Full audio only	R	R	R	N/A	R	N/A
Full audio+structure	R	R	R	N/A	R	N/A
Audio+structure+partial text	R	R	R	R	R	O
Audio+structure+full text	R	R	R	R	R	O
Full text+structure+partial audio	R	R	R	R	R	O
Full text+structure, no audio	R	R	N/A	R	R	O
Full text only	R	R	N/A	R	R	O

13.2 Content Rendering

(This section is normative)

Players must be able to determine how to render content from the types of files present. If only a text file is found, a synthetic speech rendering and output to a braille display and/or screen may be presented, according to the user's preferences and the features provided on the playback system. If only an audio file is present, straight audio playback should be initiated. Players that support only a subset of the media included in DTBs must not fail when encountering an unsupported medium, but must fall back gracefully to those it does support.

Contents

14. Digital Rights Management

14.1 Introduction

This section is informative

This standard was developed to enhance and extend certain specialized technologies which have historically provided an alternative mode of access to published information for persons who are blind and print disabled-- who cannot utilize commercially published print titles. These specialized technologies and formats were developed and maintained in ways that have clearly distinguished them from their contemporary commercial counterparts. In particular, the "Talking Book" has been developed and maintained as a format incompatible with its commercial counterparts expressly so that the non-commercial, non-governmental organizations and national libraries which serve this population might keep production costs minimized and the number of titles produced maximized. In this section we describe the normative performance requirements a Digital Rights Management (DRM) system must implement for protected content products in order to comply with the standard. This maintains titles produced under this standard as separate and set aside expressly for the exclusive use of this population, thus protecting the intellectual property rights

of others in the authoring, publishing, production, and distribution chain.

14.2 Production

This section is normative

The production process must include effective protocols to prevent the unauthorized dissemination of any content files, whether intermediate (in-process) files, archive masters, or finished product files. All entities involved in producing content shall meet the following requirements:

1. Specific written protocols and procedures shall be created to ensure security in the production process;
2. Only authorized staff shall have access to files and archive masters;
3. Staff engaged in production shall be advised in writing of their obligations and responsibilities for maintaining security;
4. Only authorized and secure transport services shall be utilized to convey content among production entities (including publishers and distributors) whether that content is intermediate files, archive masters, or finished product;
5. Regular security audits and other monitoring procedures shall be utilized to ensure the effectiveness of security protocols and procedures.

14.2.1 Sample Approach in Production

This section is informative

A producer might implement the following policies and procedures to ensure compliance under this section.

- Staff would be required to sign agreements that clearly describe their need to safeguard the files;
- login and passwords would be required on networks for users that access files and file access times could be logged;
- Access to files would be restricted to those staff members who actually need to manipulate the data during production;
- Files that are transferred from one physical location to another would use some form of secure transfer mechanism.

14.3 Distribution

This section is normative

Distribution to end users must use DRM solutions that contain an education component that informs users of their rights and obligations; and a follow-up component capable of discovering unauthorized distribution, identifying perpetrators, and imposing remedies--without adversely affecting the straightforward user-friendly environment which is a hallmark of lawful use.

14.3.1 User Education

This section is normative

Education means that the DRM system must clearly and emphatically inform users that:

1. Content is provided to users by special provision and may not be given to others;
2. Procedures are in effect to discover unauthorized distribution of content and to identify users responsible for such unauthorized distribution;
3. Irresponsible users face consequences;

4. Users will be informed of the privacy policy of the distributing agency.

14.3.1.1 Sample Language

This section is informative

"This DTB is provided with the kind permission of the copyright holder and by provision of law. It is restricted for your use and may not be given to others. Distribution of this material to unauthorized persons is a violation of copyright law. Please be advised that this DTB contains technology which can identify violators, who may face legal penalty or administrative revocation of access to DTBs. To protect your privacy, however, this technology will not be used to disclose any information about you or your legal use of the product."

14.3.2 User Follow-up

This section is normative

Follow-up means that the DRM system includes the following three safeguards:

1. Systems are implemented to identify, apprehend, and punish violators;
2. The system must respect personal privacy of lawful users;
3. System performance must be verifiable.

14.3.2.1 Sample Approach to Follow-up

This section is informative

This example describes what an organization might do to evaluate the effectiveness of its DRM system.

- Periodically search the Internet for titles the organization distributes;
- When a title is detected, the data can be retrieved to determine whether or not the title is an unauthorized publication of the organization's edition of that title;
- When a title is found to be illegally distributed, immediate measures must be taken to remove the title from unauthorized circulation;
- Appropriate remedies against violators must be pursued;
- To test follow-up system effectiveness, freely distributable files might be placed in random locations on the Internet from time to time.

14.3.3 Protection

This section is normative

Protection means that the DRM system has the following three characteristics:

1. It uses technology that prevents usage of any part of the DTB on unauthorized hardware or software player devices;
2. For downloaded material the system must be capable of coding the content such that the material can be decoded only on personalized, uniquely identified devices;
3. For material distributed on physical media such as CD-ROM, DVD, or solid-state memory the system must be capable of coding such that the material can be decoded only on personalized identified classes of players;

14.3.3.1 Sample Protection Approach

This section is informative

This sample approach employs a protection system that is in widespread use in the technology industry today.

- The distributor could encrypt the DTB to make the data unusable by anyone who does not possess the appropriate key to unlock the data;
- Keys could be transmitted via a secure transmission from the distributor to users for installation on a particular reading system or class of reading systems;
- The key would be enclosed in a voucher that identifies the user, the distributor, the expiration date for the voucher, and the reading system for which it is valid;
- The voucher could be secured via a public key encryption system so it could only be used by a specific authorized reading system or class of reading systems.

Contents

15. Time-Scale Modification

(This section is normative)

Playback systems should implement Time-Scale Modification (TSM) to enable user control of playback speed with or without pitch correction. This standard does not address TSM implementation methodology.

All time offsets in a DTB (e.g., SMIL and NCX *clipBegin/clipEnd*, bookmark *timeoffset*s, etc.), are based on normal play speed. In order to maintain synchronization, a player must process time offsets independently of actual playback speed.

Contents

16. Conformance

This standard defines two kinds of conformance: file conformance and player conformance. Conformant Digital Talking Books and DTB playback systems must meet all of the applicable requirements specified in the normative sections of this standard. Requirements will vary depending on the media included in a DTB and the functions supported by a DTB player. Contents

17. References to Other Specifications/Documents

The following standards, recommendations, and guidelines are referenced by this standard:

ATAG

Authoring Tool Accessibility Guidelines: http://www.w3.org/TR/WAI-AUTOOLS

CSS

Cascading Style Sheets: http://www.w3.org/Style/css

CSS1

Cascading Style Sheets, Level 1: http://www.w3.org/TR/REC-CSS1

CSS2

Cascading Style Sheets, Level 2: http://www.w3.org/TR/REC-CSS2

DC

Dublin Core Metadata Initiative: http://www.purl.org/dc/

DTBook3 HTML

HTML version of expanded DTBook3: http://www.loc.gov/nls/niso/dtbook3doc.htm

ISO

International Organization for Standardization homepage: http://www.iso.ch/

ISO 639

ISO 639 - Code for Representation of Names of Languages: http://www.oasis-open.org/cover/iso639a.html

ISO 3166

ISO 3166 - Codes for the Representation of Names of Countries and their Subdivisions: http://www.din.de/gremien/nas/nabd/iso3166ma/codlstp1/

ISO 8601

ISO/FDIS 8601 - Representation of Dates and Times: http://www.iso.ch

ISO 10646

ISO/IEC 10646 - Universal Multiple-Octet Coded Character Set: http://www.iso.ch

ISO 8859-1

ISO 8859-1 - 8-bit single-byte coded graphic character sets -- Part 1: Latin alphabet No. 1 (HTML character set):http://www.iso.ch

MPEG

Copies of these MPEG standards:

- MPEG-1 Audio, ISO/IEC 11172-3
- MPEG-4 Audio, ISO/IEC 14496-3

can be obtained from the International Organization for Standardization homepage: http://www.iso.ch or from your national standards body. In the United States, this is the American National Standards Institute: http://www.ansi.org

Navigation Features

Document Navigation Features List: http://www.loc.gov/nls/niso/navigation.htm

NS

XML NameSpaces: http://www.w3.org/TR/REC-xml-names

OEBPS

the Open eBook Forum Publication Structure, version 1.0: http://openebook.org/oebpsdownload.htm

RFC 1766

RFC 1766 - Tags for the Identification of Languages: http://www.cis.ohio-state.edu/htbin/rfc1766.html

RFC 2046

RFC 2046 - Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types: http://www.cis.ohio-state.edu/htbin/rfc2046.html

RFC2083

Portable Network Graphics, Version 1: http://www.cis.ohio-state.edu/htbin/rfc2083.html

RIFFWAV

RIFF WAV format information: ftp://ftp.cwi.nl/pub/audio/RIFF-format

SMIL

SMIL 2.0 Draft: http://www.w3.org/TR/smil20

SMILclock

Clock Values section of SMIL Timing and Synchronization Module:

http://www.w3.org/TR/smil20/smil-timing.html#Timing-ClockValueSyntax

SMILmedia

SMIL Media Object Module: http://www.w3.org/TR/smil20/extended-media-object.html

SMILtiming

SMIL Timing and Synchronization Module: http://www.w3.org/TR/smil20/smil-timing.html

StructGuide

Structure Guidelines: http://www.daisy.org/products/menupps.htm

UAAG

User Agent Accessibility Guidelines: http://www.w3.org/TR/UAAG

WCAG

Web Content Accessibility Guidelines: http://www.w3.org/TR/WAI-WEBCONTENT

XML

XML Version 1.0: http://www.w3.org/TR/1998/REC-xml-19980210.html

XSL

XSL, the Extensible Style Language: http://www.w3.org/TR/xsl/

Contents

Appendix 1 - DTDs for Navigation

Appendix 1.1 - DTD for NCX

(This section is normative)

```
<!-- NCX 1.93 DTD DRAFT  (ncx193.dtd)
    Authors: M. Hakkinen, G. Kerscher, T. McLaughlin, J. Pritchett, and M. Moodie
    Last Revised:   30 January 2001
    Origin Date:    2 February 2000
    Copyright (c)2000-2001 by the Daisy Consortium and NISO


    Description:

    NCX (Navigation Control for XML) is a generalized navigation
definition DTD for application
to Digital Talking Books, eBooks, and general web content models.



    This DTD is an XML application that layers navigation functionality
on top of SMIL 2.0
(Boston) content.

    The NCX defines a navigation path/model which may be applied upon
existing publications,
without modification of the existing publication source, so long as
the navigation targets within
the source publication can be directly referenced via a URI.
```

```
    DTD Change List:
30 January 2001 MM
        Added value attribute on navTarget
25 January 2001 0930 TM
         Camelcased onFocus and onBlur.
         Made lang attribute NMTOKEN.
23 January 2001 5 p.m.
      Dropped all dtb-prefixed metadata elements.  Retained
<uid>, <charset>, and <meta>*.
      Corrected spelling of docTitle in content model for <ncx>.
      Moved comment defining <text> to immediately preceding
<text> declaration.
      Defined <uid>      and <charset>.
23 January 2001
      Restored navList and navTarget
      Wrapped navTarget audio, text, etc. in a <descr> element
      Added <descr> to navList
      navStruct is no longer repeatable
      Removed "type" attribute for navStruct
      Removed "structRef" attribute for navObject
      Camel-cased levelNumber attribute and <docTitle> element
      Added prodnote and sidebar to navList navType values
      Drop style attribute from all elements
      Change class attribute on navObject, navTarget to navType
      Drop class attribute from everything except text, audio, image elements
      Created specific elements for each required metadata item
      Reordered element declarations to be a little more intuitive

18 January 2001 - JP - Changes to more closely match latest version of OEB NCX:
      Allowed 1 or more navStructs per ncx
      Removed navList and navTarget
      Removed navLevel and allowed nesting of navObject instead
      Wrapped audio, text, etc. in a <descr> element
      Allowed multiple <descr> elements per navObject (to support
multiple languages)
      Added <descr> (one or more) to navStruct to describe
overall structure
      Added "type" attribute to navStruct to store hierarchical/flat information
      Added rules and reserved values for "class" attribute of navStruct
      Added "pageRef" attribute to navObject
      Added "levelnumber" and "value" attributes to navObject
      Removed <title> from <head>
      Added <meta> to <head>

(See version 1.6 for changes prior to this version.)
22 August 2000 - TM/MM - Dropped entity %coreattrs.  Replaced it with
individual attributes id, class, and style (dropped title attribute)
to allow id to be REQUIRED in some instances.
      Changed attribute onlostfocus to onblur.
      Changed structref to structRef.


-->
```

```
<!-- Basic Entities -->

<!ENTITY % i18n
    "lang               NMTOKEN          #IMPLIED
    dir         (ltr|rtl)        #IMPLIED" >

<!ENTITY % uri          "CDATA" >
<!ENTITY % script       "CDATA" >

<!-- ELEMENTS -->

<!-- Top Level NCX Container. -->
<!ELEMENT ncx (head, docTitle, navStruct, navList*)>
<!ATTLIST ncx
    %i18n;
>

<!-- Document Head - Contains all NCX metadata.
-->

<!ELEMENT head (uid, charset, meta*)>

<!-- DocTitle - the title of the document, required and must
immediately follow head.
-->

<!ELEMENT docTitle (text, audio?)>
<!ATTLIST docTitle
    id          ID              #IMPLIED
>

<!-- Navigation Structure - container for all of the NCX objects
that are part of the hierarchical structure of the document.
-->

<!ELEMENT navStruct (descr+, navObject+)>
<!ATTLIST navStruct
    id          ID              #IMPLIED
>

<!-- Navigation Object - contains description(s) of target, as
well as a pointer to entire content of target.
Hierarchy is represented by nesting navObjects.  "navType" attribute
describes the kind of structural unit this object represents (e.g.,
"chapter", "section").  "levelNumber" attribute is the nesting level
of this object
(if within a hierarchy).  "value" attribute is a numerical
representation of the text content of the
label if this is a purely numerical (integer only) label (e.g., a
page number).  "pageRef" is the id of the page navTarget on which
this structure target begins.
```

```
   -->
   <!ELEMENT navObject (descr+, content, navObject*)>
   <!ATTLIST navObject
      id           ID              #REQUIRED
      onFocus      %script;        #IMPLIED
      onBlur       %script;        #IMPLIED
      navType      CDATA           #IMPLIED
      levelNumber  CDATA           #IMPLIED
      value                 CDATA               #IMPLIED
      pageRef      IDREF           #IMPLIED
   >


   <!-- Navigation List - container for distinct, flat sets of
   navigable elements, e.g.  page numbers,
   notes, figures, tables, etc.  Essentially a flat version of
   navStruct.  The "navType" attribute describes the type of list this
   navList represents.  Acceptable values are:  "pages", "notes",
   "tables", "figures", "prodnotes", "sidebars", and "other.xxx" (where
   "xxx" is defined by the producer).
   -->


   <!ELEMENT navList   (descr+, navTarget+) >
   <!ATTLIST navList
      id           ID              #IMPLIED
      navType      CDATA           #IMPLIED
   >


   <!-- Navigation Target - contains description(s) of target, as
   well as a pointer to entire content of target.
   navTargets are the equivalent of navObjects for use in navLists.
   "structRef" is the id of another navObject within this NCX that
   contains this navTarget.  "navType" attribute describes the kind of
   structure this target represents.  Acceptable values are:  "page",
   "note", "table", "figure", "prodnote", "sidebar", and "other.xxx"
   (where "xxx" is defined by the producer).
   -->


   <!ELEMENT navTarget  (descr+, content) >
   <!ATTLIST navTarget
      id           ID              #REQUIRED
      onFocus      %script;        #IMPLIED
      onBlur       %script;        #IMPLIED
      navType      CDATA           #IMPLIED
      value                 CDATA               #IMPLIED
      structRef    IDREF           #REQUIRED
   >



   <!-- Description of this structure in the various media -->
   <!ELEMENT descr (text?, audio?, image?)>
   <!ATTLIST descr
         %i18n;
   >
```

```
<!-- Content Element - pointer into SMIL to beginning of navObject. -->
<!ELEMENT content EMPTY>
<!ATTLIST content
   id              ID              #IMPLIED
   src             %uri;           #REQUIRED
>


<!-- Text Element - Contains text of navObject heading or docTitle. -->
<!ELEMENT text (#PCDATA)>
<!ATTLIST text
   id              ID              #IMPLIED
   class                CDATA           #IMPLIED
>


<!-- Audio Element - audio clip of navObject heading. -->
<!ELEMENT audio EMPTY>
<!ATTLIST audio
   id              ID              #IMPLIED
   class                CDATA           #IMPLIED
   src             %uri;           #REQUIRED
   clipBegin    CDATA           #IMPLIED
   clipEnd      CDATA           #IMPLIED
>


<!-- Image Element - image that may accompany heading. -->
<!ELEMENT image EMPTY>
<!ATTLIST image
   id              ID              #IMPLIED
   class                CDATA           #IMPLIED
   src             %uri;           #REQUIRED
>


<!-- METADATA ELEMENTS  -->

<!-- Uid Element - Globally unique identifier for DTB. -->
<!ELEMENT uid (#PCDATA) >

<!-- Charset Element - Character set utilized in NCX. -->
<!ELEMENT charset (#PCDATA)>

<!-- Meta Element - producer-defined metadata about this NCX -->
<!ELEMENT meta EMPTY>
<!ATTLIST meta
   name            CDATA           #REQUIRED
   content         CDATA           #REQUIRED
   scheme          CDATA           #IMPLIED
>
```

Appendix 1.2 - DTD for Navigation by Lightweight Players

(This section is normative)

```
<!-- LWN 0.11 DTD DRAFT  (lwn011.dtd)
    Author: T. McLaughlin
    Last Revised:   30 January 2001 MM
    Origin Date:    25 January 2001
    Copyright (c)2000-2001 by the Daisy Consortium and NISO

    Description:

    LWN (Light Weight Navigation) is intended to allow players with
sparse resources access to all navigation information included in the
hierarchical NCX file.



DTD Change List:
30 January 2001 MM
        Added value attribute on navTarget
25 January 2001 TM
        Built from NCX 1.92 DTD
-->


<!-- Basic Entities -->

<!ENTITY % i18n
    "lang                CDATA               #IMPLIED
    dir          (ltr|rtl)       #IMPLIED" >

<!ENTITY % uri          "CDATA" >
<!ENTITY % script       "CDATA" >

<!-- ELEMENTS -->

<!-- Top Level LWN Container. -->
<!ELEMENT lwn (head, docTitle, navStruct)>
<!ATTLIST lwn
    %i18n;
>

<!-- Document Head - Contains all LWN metadata.
-->

<!ELEMENT head (uid, charset, meta*)>


<!-- DocTitle - the title of the document, required and must
immediately follow head.
-->
```

```
    <!ELEMENT docTitle (text, audio?)>
    <!ATTLIST docTitle
        id            ID              #IMPLIED
    >


    <!-- Navigation Structure - container for all of the LWN objects
    that are part of the hierarchical structure of the document.
    -->

    <!ELEMENT navStruct (descr+, (navObject|navTarget)+)>
    <!ATTLIST navStruct
        id            ID              #IMPLIED
    >


    <!-- Navigation Object - contains description(s) of target, as
    well as a pointer to entire content of target.
    "navType" attribute describes the kind of structural unit this object
    represents (e.g., "chapter", "section").  "levelNumber" attribute is
    the nesting level of this object
    (if within a hierarchy).  "value" attribute is a numerical
    representation of the text content of the
    label if this is a purely numerical (integer only) label (e.g., a
    page number).  "pageRef" is the id of the page navTarget on which
    this structure target begins.
    -->
    <!ELEMENT navObject (descr+, content)>
    <!ATTLIST navObject
        id            ID              #REQUIRED
        onFocus       %script;        #IMPLIED
        onBlur        %script;        #IMPLIED
        navType       CDATA           #IMPLIED
        levelNumber   CDATA           #IMPLIED
        value             CDATA           #IMPLIED
        pageRef       IDREF           #IMPLIED
    >


    <!-- Navigation Target - contains description(s) of target, as
    well as a pointer to entire content of target.
    navTargets are the equivalent of navObjects for the inclusion of
    non-hierarchical objects. "structRef" is the id of another navObject
    within this LWN that contains this navTarget.  "navType" attribute
    describes the kind of structure this target represents.  Acceptable
    values are:  "page", "note", "table", "figure", "prodnote",
    "sidebar", and "other.xxx" (where "xxx" is defined by the producer).
    -->

    <!ELEMENT navTarget  (descr+, content) >
    <!ATTLIST navTarget
        id            ID              #REQUIRED
        onFocus       %script;        #IMPLIED
        onBlur        %script;        #IMPLIED
        navType       CDATA           #IMPLIED
        value         CDATA           #IMPLIED
        structRef     IDREF           #REQUIRED
```

```
    >


    <!-- Description of this structure in the various media -->
    <!ELEMENT descr (text?, audio?, image?)>
    <!ATTLIST descr
            %i18n;
    >



    <!-- Content Element - pointer into SMIL to beginning of navObject. -->
    <!ELEMENT content EMPTY>
    <!ATTLIST content
       id              ID                #IMPLIED
       src             %uri;             #REQUIRED
    >


    <!-- Text Element - Contains text of navObject heading or docTitle. -->
    <!ELEMENT text (#PCDATA)>
    <!ATTLIST text
       id              ID                #IMPLIED
       class                  CDATA             #IMPLIED
    >


    <!-- Audio Element - audio clip of navObject heading. -->
    <!ELEMENT audio EMPTY>
    <!ATTLIST audio
       id              ID                #IMPLIED
       class                  CDATA             #IMPLIED
       src             %uri;             #REQUIRED
       clipBegin    CDATA             #IMPLIED
       clipEnd      CDATA             #IMPLIED
    >


    <!-- Image Element - image that may accompany heading. -->
    <!ELEMENT image EMPTY>
    <!ATTLIST image
       id              ID                #IMPLIED
       class                  CDATA             #IMPLIED
       src             %uri;             #REQUIRED
    >


    <!-- METADATA ELEMENTS  -->

    <!-- Uid Element - Globally unique identifier for DTB. -->
    <!ELEMENT uid (#PCDATA) >

    <!-- Charset Element - Character set utilized in LWN. -->
    <!ELEMENT charset (#PCDATA)>

    <!-- Meta Element - producer-defined metadata about this LWN -->
    <!ELEMENT meta EMPTY>
    <!ATTLIST meta
       name            CDATA                #REQUIRED
```

```
      content       CDATA           #REQUIRED
      scheme        CDATA           #IMPLIED
>
```

[Contents](#)

Appendix 2 - DTD for Portable Bookmarks/Highlights

(This section is normative)

```
<!-- bookmark_072.dtd
version: 0.72
                revised: 01/17/2001 - added label attrs to bookmark and hilite
        T. McLaughlin and M. Moodie -->
<!-- ******************** Entities ****************** -->
<!ENTITY % uri "CDATA">
<!-- ********************* Elements ******************** -->
<!-- BookmarkSet: The set of bookmarks for a book consists of the
title, a unique identifier of the book, the last place the reader
left off and zero or more bookmarks, highlights, and associated audio
or textual notes. This set is intended for export of bookmarks,
highlights and notes to another player; the markup is not required
for a player's internal representation of bookmarks.
   -->
<!ELEMENT bookmarkSet   (title, uid, lastmark, (bookmark |
hilite)*) >
<!-- Title: The book's title in text and an optional audio clip.
   -->
<!ELEMENT title (text, audio?) >

<!-- uid: A globally unique identifier for the book.
   -->
<!ELEMENT uid   (#PCDATA) >

<!-- Bookmark: Location and optional note. Location consists of a
uri pointing to the id attribute of the <par> element in the
SMIL file that contains the bookmark plus a time offset in seconds
(or character offset) to the exact place.  Player should by default
automatically number bookmarks in the order in which they fall in the
book.
   -->
<!ELEMENT bookmark      (ncxRef, uri, (timeoffset | charoffset), note?) >
<!ATTLIST bookmark
        label           CDATA           #IMPLIED
>

<!-- NcxRef: Captures current location in NCX (the id of the
current navObject)at time lastmark, bookmark, or highlight is set.
Ensures that current location in NCX and SMIL are synchronized after
```

moving to a lastmark, etc., so that any global navigation commands
issued by the user will start from the current location. -->
<!ELEMENT ncxRef   (#PCDATA)>

<!-- Lastmark: Location where reader left off and where player
will resume play when restarted.
   -->
<!ELEMENT lastmark   (ncxRef, uri, (timeoffset | charoffset)) >

<!-- Hilite: A block of text with an optional note attached.
   -->
<!ELEMENT hilite   (hilStart, hilEnd, note?) >
<!ATTLIST hilite
        label   CDATA   #IMPLIED
>

<!-- HilStart: Starting point of hilited block.
   -->
<!ELEMENT hilStart   (ncxRef, uri, (timeoffset | charoffset)) >

<!-- HilEnd: End point of hilited block.
   -->
<!ELEMENT hilEnd   (ncxRef, uri, (timeoffset | charoffset)) >

<!-- Uri: pointer to id of <par> or <seq> in SMIL, to
id in text-only file, or to audio file that contains the bookmark.
   -->
<!ELEMENT uri   (#PCDATA) >

<!-- Timeoffset: Exact position of bookmark in SMIL file or
audio-only file referenced by the uri; in seconds.
   -->
<!ELEMENT timeoffset   (#PCDATA) >

<!-- Charoffset: Exact position of bookmark in text-only file
referenced by the uri: in bytes.
   -->
<!ELEMENT charoffset   (#PCDATA) >

<!-- Note: The note is for the user's input, random thoughts,
musings, etc. It can be text or audio or both.
   -->
<!ELEMENT note   (text?, audio?) >

<!-- Text: Text of title or note.
   -->
<!ELEMENT text   (#PCDATA) >
<!-- Audio: Audio clip of title or note, in any format supported
by standard.
   -->
<!ELEMENT audio EMPTY >
<!ATTLIST audio

    src   %uri;   #REQUIRED

```
        clipBegin    CDATA     #IMPLIED
        clipEnd      CDATA     #IMPLIED
>
```

[Contents](#)

Appendix 3 - DTD for Resource File

(This section is normative)

<!-- Resource File DTD
file: resource.dtd
Authors: Tom McLaughlin, Michael Moodie, Thomas Kjellberg Christensen Created 07-27-00 Last Modified:
01-30-2001 MM Changed classResource to navTypeResource to reflect change in NCX from class attribute to navType
attribute. Changed class attribute on navTypeResource to navTypeRef Changed id attribute on idResource to idRef
01-17-2001 TKC Added idResource -->
<!-- ********** Attribute Types *********** -->
<!-- languagecode: An RFC1766 language code. -->
<!ENTITY % languagecode "NMTOKEN">
<!-- *************** Resource Elements ********** -->
<!-- navResource: Root element of DTD. -->
<!ELEMENT navResource (navTypeResource | elementResource | idResource)+ >
<!-- navTypeResource contains information about the alternative representations of an element present in the NCX. An
alternative representation can be used to convey navigational information, e.g., provide a descriptive name for the kind
of segment (part, chapter, section, etc.) the user is encountering. Text can be used for screen or braille display, audio for
digital talking book players, and image for screen display.
-->
<!ELEMENT navTypeResource (text?, audio?, image?) >
<!ATTLIST navTypeResource

```
    navTypeRef    CDATA        #REQUIRED
    lang        %languagecode;  #IMPLIED
>
```

<!-- elementResource contains information about the alternative representations of an XML element present in the
document text file. An alternative representation can be used to convey navigational information, e.g., provide a
descriptive name for the element (<p>, <a>, <li>, etc.) the user is encountering. Text can be used for screen or braille
display, audio for digital talking book players, and image for screen display.
-->
<!ELEMENT elementResource (text?, audio?, image?) >
<!ATTLIST elementResource

```
    elementRef   CDATA    #REQUIRED
    classRef     CDATA            #IMPLIED
    lang         %languagecode;  #IMPLIED
>
```

<!-- idResource contains information about alternative representations of "references" in the Guide and <customTest>
elements in SMIL. The player will use the "type" attribute of a given <reference> in the Guide or the id attribute on
<customTest> in the <head> of a SMIL file to locate the corresponding *idResource*. For example, a <customTest>
element (tagged <customTest id="pagenum" ...>) would call the *idResource* with *idRef* equal to "pagenum".
<!ELEMENT idResource (text?, audio?, image?) >
<!ATTLIST idResource

```
      idRef    CDATA          #REQUIRED
      lang     %languagecode; #IMPLIED
>
```
<!ELEMENT text (#PCDATA) >
<!ELEMENT audio EMPTY >
<!ATTLIST audio

```
      src              CDATA          #REQUIRED
      clipBegin    CDATA         #IMPLIED
      clipEnd      CDATA          #IMPLIED
>
```
<!-- If clipBegin is not used the media object is played from the beginning of source
If clipEnd is not used the media object is played to the end of the source. If the value of the clipEnd attribute exceeds the duration of the media object, the value is ignored, and the clip end is set equal to the effective end of the media object. Reference: SMIL 2.0, Module 7, The SMIL Media Object Module [SMILmedia]. Copyright © 2000 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. http://www.w3.org/Consortium/Legal/.
-->
<!ELEMENT image EMPTY >
<!ATTLIST image

```
src   CDATA   #REQUIRED
```
> Contents

Appendix 4 - DTB SMIL Profile and DTB-Specific DTD

(This section is normative)

4.1 DTB SMIL 2.0 DTD

```
<!--
==========================================================================
-->
<!-- DTB SMIL 2.0 DTD
===================================================== -->
<!-- file: DTB-SMIL20.dtd

        This is DTB SMIL 2.0, a proper subset of SMIL 2.0.

        Copyright 2000 World Wide Web Consortium
           (Massachusetts Institute of Technology, Institut National de
            Recherche en Informatique et en Automatique, Keio University).
            All Rights Reserved.

        Permission to use, copy, modify and distribute the SMIL Basic DTD and
        its accompanying documentation for any purpose and without fee is
        hereby granted in perpetuity, provided that the above copyright notice
```

```
          and this paragraph appear in all copies.  The copyright holders make
          no representation about the suitability of the DTD for any purpose.

          It is provided "as is" without expressed or implied warranty.

          Revision for DTB SMIL Authored by: Mark Hakkinen, Tom
   McLaughlin, Michael Moodie, James Pritchett; 2000/12/19


          Author:      Jacco van Ossenbruggen, Kenichi Kubota
          Revision:    $Id: SMIL20Basic.dtd,v 1.4 2000/10/17 09:52:17
   jvanoss Exp $


   -->
   <!-- This is the driver file for the DTB SMIL DTD.

          This DTD module is identified by the PUBLIC and SYSTEM identifiers:

          PUBLIC "-//NISO//DTD DTB SMIL 2.0 //EN"
          SYSTEM "http://www.loc.gov/nls/niso/DTB-SMIL20.dtd"


   -->
   <!ENTITY % NS.prefixed "IGNORE">
   <!ENTITY % SMIL.prefix "">
   <!-- Define the Content Model -->
   <!ENTITY % smil-model.mod PUBLIC  "-//NISO//ENTITIES DTB SMIL 2.0
   Document Model 1.0//EN"
                "DTB-SMIL-model-1.mod">
   <!-- Modular Framework Module  ................................ -->
   <!ENTITY % smil-framework.module "INCLUDE">
   <![%smil-framework.module;[
   <!ENTITY % smil-framework.mod
          PUBLIC "-//W3C//ENTITIES SMIL 2.0 Modular Framework 1.0//EN"
                "smil-framework-1.mod" >
   %smil-framework.mod;]]>
   <!--     The SMIL 2.0 Basic Profile supports the lightweight multimedia
            features defined in SMIL language. This profile includes
            the following SMIL modules:

                   SMIL 2.0 BasicLayout Module
                   SMIL 2.0 BasicLinking Module
                   SMIL 2.0 BasicMedia and MediaClipping Modules
                   SMIL 2.0 Structure Module
                   SMIL 2.0 BasicInlinTiming, SyncbaseTiming, EventTiming,
                          MinMaxTiming and BasicTimeContainers Modules
                   SMIL 2.0 BasicContentControl and SkipContentControl Modules

            In addition, DTB SMIL has added the following SMIL module:
                   SMIL 2.0 Metainformation Module


   -->
   <!ENTITY % layout-mod PUBLIC "-//W3C//ELEMENTS SMIL 2.0 Layout//EN"
       "SMIL-layout.mod">
   %layout-mod;
   <!ENTITY % link-mod PUBLIC "-//W3C//ELEMENTS SMIL 2.0 Linking//EN"
```

```
    "SMIL-link.mod">
%link-mod;
<!ENTITY % BasicLinkingModule "INCLUDE">
<!ENTITY % media-mod PUBLIC "-//W3C//ELEMENTS SMIL 2.0 Media Objects//EN"
    "SMIL-media.mod">
%media-mod;
<!ENTITY % struct-mod PUBLIC "-//W3C//ELEMENTS SMIL 2.0 Document
Structure//EN"
    "SMIL-struct.mod">
%struct-mod;
<!ENTITY % timing-mod PUBLIC "-//W3C//ELEMENTS SMIL 2.0 Timing//EN"
    "SMIL-timing.mod">
%timing-mod;
<!ENTITY % control-mod PUBLIC "-//W3C//ELEMENTS SMIL 2.0 Content//EN"
    "SMIL-control.mod">
%control-mod;
<!-- DTB SMIL Additions -->
<!-- METAINFORMATION -->
<!ENTITY % meta-mod PUBLIC "-//W3C//ELEMENTS SMIL 2.0 Document
Metainformation//EN"
    "SMIL-metainformation.mod">
%meta-mod;
```

[Contents](#)

4.2 DTB SMIL Document Model

```
<!--
======================================================================
-->
<!-- DTB SMIL 2.0 Document Module
======================================= -->
<!-- file: DTB-SMIL-model-1.mod

        This is DTB SMIL 2.0 , a proper subset of SMIL 2.0.

        Copyright 2000 W3C (MIT, INRIA, Keio), All Rights Reserved.

        This DTD module is identified by the PUBLIC and SYSTEM identifiers:

        PUBLIC "-//NISO//ENTITIES DTB SMIL 2.0 Document Model 1.0//EN"
        SYSTEM "DTB-SMIL-model-1.mod"

        Revision for DTB SMIL Authored by: Mark Hakkinen, Tom
McLaughlin, Michael Moodie, James Pritchett; 2000/12/19

        Author: Kenichi Kubota, Warner ten Kate, Jacco van Ossenbruggen
        Revision: $Id: smilbasic-model-1.mod,v 1.32 2000/10/09 10:34:31
jvanoss Exp $

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

```
                This file defines the DTB SMIL 2.0 Document Model.
                All attributes and content models are defined in the second
                half of this file.  We first start with some utility definitions.
                These are mainly used to simplify the use of modules in the
                second part of the file.

-->
<!-- The following section added for DTB SMIL -->
<!-- ================== Util: Head
========================================= -->
<!ENTITY % head-meta.content "metadata">
<!ENTITY % head-layout.content "layout|switch">
<!ENTITY % head-control.content "customAttributes">
<!-- ================== Util: Body - Media
================================ -->
<!ENTITY % media-object "text|img|audio|video|animation|textstream|ref">
<!-- ================== Util: Body - Content Control
===================== -->
<!ENTITY % content-control "switch">
<!--================== Util: Body - Linking
============================== -->
<!ENTITY % BasicLink "a|anchor|area">
<!ENTITY % link "%BasicLink;">
<!--
=======================================================================
-->
<!--
=======================================================================
-->
<!--
=======================================================================
-->
<!--
      The actual content model and attribute definitions for each module
      section follow below.
-->
<!-- ================== Timing
=========================================== -->
<!ENTITY % BasicInlineTiming.module "INCLUDE">
<!ENTITY % MinMaxTiming.module "INCLUDE">
<!ENTITY % SyncbaseTiming.module "INCLUDE">
<!ENTITY % EventTiming.module "INCLUDE">
<!ENTITY % BasicTimeContainers.module "INCLUDE">
<![%BasicInlineTiming.module;[
   <!ENTITY % basicTimeContainers "par|seq">
   <!ENTITY % timeContainer "%basicTimeContainers;">
   <!ENTITY % basicTimeContainers.content
    "(%media-object;|%content-control;|%link;|%basicTimeContainers;)*">
   <!ENTITY % timeContainer.content "%basicTimeContainers.content;">
   <!ENTITY % basicInlineTiming.attrib "%BasicInlineTiming.attrib;">
   <!ENTITY % basicParTime.attrib "
         %Endsync.attrib;
         %Fill.attrib;
         %FillDefault.attrib;
```

```
      ">
      <!ENTITY % basicSeqTime.attrib "
          %Fill.attrib;
          %FillDefault.attrib;
      ">
   ]]>
   <!ENTITY % timeContainer "">
   <!ENTITY % timeContainer.content "">
   <!ENTITY % basicInlineTiming "">
   <!ENTITY % basicParTime.attrib "">
   <!ENTITY % basicSeqTime.attrib "">
   <![%MinMaxTiming.module;[
       <!ENTITY % minMaxTiming.attrib "%MinMaxTiming.attrib;">
   ]]>
   <!ENTITY % minMaxTiming.attrib "">
   <!ENTITY % smil-time.attrib "
      %basicInlineTiming.attrib;
      %minMaxTiming.attrib;
      %Fill.attrib;
   ">
   <!ENTITY % parTime.attrib "
      %basicInlineTiming.attrib;
      %basicParTime.attrib;
      %minMaxTiming.attrib;
   ">
   <!ENTITY % seqTime.attrib "
      %basicInlineTiming.attrib;
      %basicSeqTime.attrib;
      %minMaxTiming.attrib;
   ">
   <!ENTITY %  par.content "%timeContainer.content;">
   <!ENTITY %  seq.content "%timeContainer.content;">

   <!-- Added CustomTest.attrib to par and seq attribs for DTB SMIL -->
   <!ENTITY %  par.attrib "
      %parTime.attrib;
      %Test.attrib;
      %Region.attrib;
      %CustomTest.attrib;
   ">
   <!ENTITY %  seq.attrib "
      %seqTime.attrib;
      %Test.attrib;
      %Region.attrib;
      %CustomTest.attrib;">
   <!-- ================== Content Control
   ================================== -->
   <!ENTITY % BasicContentControl.module "INCLUDE">
   <!ENTITY % SkipContentControl.module "INCLUDE">
   <!-- Following Entity added for DTB SMIL -->
   <!ENTITY % CustomTestAttributes.module "INCLUDE">
   <!ENTITY % switch.content "(layout|%timeContainer;|%media-object;|
                             %content-control;|a|area|anchor)*">
   <!-- Following Entity changed for DTB SMIL by adding
```

```
     CustomTest.attrib -->
     <!ENTITY % switch.attrib "%Test.attrib; %CustomTest.attrib;">
     <!-- Following Entities added for DTB SMIL -->
     <!ENTITY % customAttributes.attrib "%Test.attrib; %SkipContent.attrib;">
     <!ENTITY % customTest.attrib "%SkipContent.attrib;">
     <!-- ================= Layout
     ========================================== -->
     <!ENTITY % BasicLayout.module "INCLUDE">
     <!ENTITY % layout.content "(root-layout?,(region)*)">
     <!ENTITY % region.content "EMPTY">
     <!ENTITY % rootlayout.content "EMPTY">
     <!ENTITY % region.attrib "%Test.attrib; %SkipContent.attrib;">
     <!ENTITY % rootlayout.attrib "%Test.attrib; %SkipContent.attrib;">
     <!ENTITY % layout.attrib "%Test.attrib;">
     <!-- ================= Linking
     ========================================== -->
     <!ENTITY % LinkingAttributes.module "INCLUDE">
     <!ENTITY % BasicLinking.module "INCLUDE">
     <!ENTITY % a.content "(%timeContainer;|%media-object;|
                             %content-control;)*">
     <!ENTITY % area.content "EMPTY">
     <!ENTITY % anchor.content "EMPTY">
     <!ENTITY % a.attrib "%smil-time.attrib; %Test.attrib;">
     <!ENTITY % area.attrib "
             %smil-time.attrib;
             %Test.attrib;
             %SkipContent.attrib;
     ">
     <!ENTITY % anchor.attrib "
             %smil-time.attrib;
             %Test.attrib;
             %SkipContent.attrib;
     ">
     <!-- ================= Media
     ========================================== -->
     <!ENTITY % BasicMedia.module "INCLUDE">
     <!ENTITY % MediaClipping.module "INCLUDE">
     <!ENTITY % media-object.content "(switch|area|anchor)*">
     <!ENTITY % media-object.attrib "
        %smil-time.attrib;
        %Test.attrib;
        %Region.attrib;
     ">
     <!-- Following Section, Metadata, added for DTB SMIL -->
     <!-- ================= Metadata
     ========================================== -->
     <!ENTITY % meta.content "EMPTY">
     <!ENTITY % meta.attrib "%SkipContent.attrib;">
     <!ENTITY % metadata.content "EMPTY">
     <!ENTITY % metadata.attrib "%SkipContent.attrib;">
     <!-- ================= Structure
     ========================================== -->
     <!ENTITY % smil.content "(head?,body?)">
     <!-- Head Content Modified for DTB SMIL -->
```

```
<!ENTITY % head.content "(
        meta*,
        ((%head-control.content;),   meta*)?,
        ((%head-meta.content;),      meta*)?,
        ((%head-layout.content;),    meta*)?

)">
<!ENTITY % body.content "(%timeContainer;|%media-object;|
                          %content-control;|%link;)*">
<!-- ================== End of DTB-SMIL-model-1.mod
====================== -->
```

[Contents]

4.3 SMIL 2.0 Modules Included in DTB SMIL Profile

In addition to the modules included in the two preceding appendices, the following SMIL 2.0 modules must be referenced by authoring tools when validating DTB SMIL files.

- smil-attribs-1.mod
- smil-datatypes-1.mod
- smil-framework-1.mod
- smil-qname-1.mod
- SMIL-control.mod
- SMIL-layout.mod
- SMIL-link.mod
- SMIL-media.mod
- SMIL-metainformation.mod
- SMIL-struct.mod
- SMIL-timing.mod

[Contents]

4.4 DTB-Specific SMIL DTD

```
<!--SMIL 2.0 DTB-specific DTD
      file:dtbsmil.dtd
      Last modified: 01/30/2001 M. Moodie
Changed content models of layout, body, seq, and par from ()* to ()+
to ensure that some content is present.
Modified comment on body to recommend use of base set of seqs to hold
total time of file.
      Created: 01/29/2001    M. Moodie, T. McLaughlin, L. Rasmussen
This DTD is intended for use only with DTB applications.  Documents
valid to this DTD will also be valid to the DTB SMIL Profile, but not
necessarily vice versa, as this DTD contains only a subset of the
elements and attributes present in the DTB SMIL Profile.  This DTD is
in some areas more restrictive than the Profile (e.g., requiring IDs
on some elements), to enforce structure critical to the DTB
application.
```

```
    -->

    <!ENTITY        % Core.attrib
            "id      ID        #IMPLIED
            class    CDATA     #IMPLIED
            title    CDATA     #IMPLIED"
    >


    <!ELEMENT       smil      (head?, body?) >
    <!ATTLIST       smil
            %Core.attrib;
            xml:lang          NMTOKEN         #IMPLIED
    >


    <!ELEMENT       head      ((meta)*, (layout)?, (customAttributes)? ) >
    <!ATTLIST       head
            %Core.attrib;
            xml:lang          NMTOKEN         #IMPLIED
    >



    <!ELEMENT       meta      EMPTY >
    <!ATTLIST       meta
            name              CDATA           #REQUIRED
            content           CDATA           #IMPLIED
    >

    <!-- only smil basic layout allowed; not CSS2.
          root-layout not included, is implementation dependent.
    -->
    <!ELEMENT       layout    (region)+ >
    <!ATTLIST       layout
            %Core.attrib;
            xml:lang          NMTOKEN         #IMPLIED
    >


    <!ELEMENT       region    EMPTY >
    <!ATTLIST       region
            id                ID              #REQUIRED
            height            CDATA           'auto'
            width             CDATA           'auto'
            bottom            CDATA           'auto'
            top               CDATA           'auto'
            left              CDATA           'auto'
            right             CDATA           'auto'
            fit               (hidden|fill|meet|scroll|slice) 'hidden'
            z-index           CDATA           #IMPLIED
            backgroundColor   CDATA           #IMPLIED
            showBackground    (always|whenActive)     'always'
    >


    <!ELEMENT       customAttributes        (customTest)+ >
    <!ATTLIST       customAttributes
            %Core.attrib;
```

```
                xml:lang          NMTOKEN          #IMPLIED
    >


    <!ELEMENT       customTest        EMPTY >
    <!ATTLIST       customTest
            id                        ID               #REQUIRED
            class                     CDATA            #IMPLIED
            defaultState              (true|false)     'false'
            title                     CDATA            #IMPLIED
            xml:lang                  NMTOKEN          #IMPLIED
            override                  (allowed|not-allowed)   'not-allowed'
    >


    <!-- Even though body functions as a seq, and you don't need a
    base set of seqs wrapping the whole presentation, for DTB
    applications a base set of seqs should be used.  The dur attribute on
    the first seq is used by the player to determine the length of the
    SMIL presentation. -->
    <!ELEMENT       body    (par|seq|text|audio|img|a)+ >
    <!ATTLIST       body
            %Core.attrib;
            xml:lang          NMTOKEN          #IMPLIED
    >


    <!ELEMENT       seq     (par|seq|text|audio|img|a)+ >
    <!ATTLIST       seq
            id              ID              #REQUIRED
            class           CDATA           #IMPLIED
            customTest      IDREF           #IMPLIED
          begin             CDATA           #IMPLIED
            end             CDATA           #IMPLIED
            dur             CDATA           #IMPLIED

    >


    <!-- pars are not allowed to nest.
    -->
    <!ELEMENT       par     (seq|text|audio|img|a)+ >
    <!ATTLIST       par
            id              ID              #REQUIRED
            class           CDATA           #IMPLIED
            customTest      IDREF           #IMPLIED
          begin             CDATA           #IMPLIED
            end             CDATA           #IMPLIED
            dur             CDATA           #IMPLIED

    >


    <!ELEMENT       text    EMPTY >
    <!ATTLIST       text
            region          CDATA           #IMPLIED
            src             CDATA           #REQUIRED
            begin           CDATA           #IMPLIED
            end             CDATA           #IMPLIED
```

```
                  dur              CDATA             #IMPLIED
>


<!ELEMENT        audio    EMPTY >
<!ATTLIST        audio
         src              CDATA             #REQUIRED
         clipBegin        CDATA             #IMPLIED
         clipEnd          CDATA             #IMPLIED
         region           CDATA             #IMPLIED
         begin            CDATA             #IMPLIED
         end              CDATA             #IMPLIED
         dur              CDATA             #IMPLIED
>
<!ELEMENT        img      EMPTY >
<!ATTLIST        img
         region           CDATA             #IMPLIED
         src              CDATA             #REQUIRED
         begin            CDATA             #IMPLIED
         end              CDATA             #IMPLIED
         dur              CDATA             #IMPLIED
>


<!ELEMENT        a        (text|audio|img)* >
<!ATTLIST        a
         href             CDATA             #REQUIRED
         xml:lang         NMTOKEN           #IMPLIED
         %Core.attrib;
         begin            CDATA             #IMPLIED
         end              CDATA             #IMPLIED
         dur              CDATA             #IMPLIED
>
```

[Contents](#)

Appendix 5 - DTBook3 DTD

(This section is normative)

```
<!-- dtbk3-07.dtd revision 3-07 2001-01-31 Harvey Bingham -->

    <!--dtbook3 XML Document Type Definition implementing the
        NISO Digital Talking Book Version 3-07.

    Harvey Bingham <hbingham@acm.org>
    George Kerscher <kerscher@montana.com>
    Michael Moodie <mmoo@loc.gov>
    DAISY Markup Specification Team <daisy-wt-must@svb.nl>

    Copyright (c) 1999, 2001 National Information Standards Organization
```

1. Purpose

The Digital Talking Book 3.0 Document Type Definition (DTD) provides
the means to markup the text of a published book to permit support for
the combination of professional narration, and navigation into that
narration. The markup tags in the book convey its content in structure,
and some metadata about the book content and its structures.

The Document Type Definition (DTD) names and defines the allowable element
types, their allowable content, and their attributes. Correct markup
of the text of the book permits the textual material to be synchronized
using SMIL files with the professionally narrated version of that book.
The synchronization can permit concurrent display of the text being
narrated. The textual content can be searched in context to locate material
desired for narration.

This application of XML is the next generation after several DAISY
versions of 2.X specifications. Those applications developed a
Navigation Control Center (NCC) for synchronizing document structure
with narration. The DAISY (Digital Audio-based Information SYstem)
Consortium contributed substantially to the development of this DTD.

The NCC evolved into an XML application called NCX derived from
the markup of documents tagged using the dtbook3 DTD. Richer
structuring capability is one of the objectives of this DTD. The
Synchronized Multimedia Integration Language (SMIL) 2.0 will be used.

Some HTML elements are omitted from dtbook3. HTML authoring tools may
include many of the additional tags that are left out of dtbook3. Endtag
markup, sometimes optional in HTML, are required for use with dtbook3,
as any XML application requires endtags. The benefit of including
endtags is that the tagged document has dependable structure.

The tools available for browsing HTML may be used with dtbook3
material, at the expense of discarding some specific tagging and
attributes that are not part of HTML 4.0. A CSS-based stylesheet
that identifies the presentation expectations for the non-HTML tags,
or a filter to map those tags onto suitable HTML tags can provide
appropriate visual presentation.

2. Document Tagging Content

A Digital Talking Book 3 document is an XML application. Therefore it
should begin with the XML processing instruction identifying the
version of XML, and the character set encoding:

    <?xml version="1.0" encoding="ISO-8859-1" ?>

Alternative encodings "UTF-8", "UTF-16", "ISO-10646-UCS-2", or
"ISO-10646-UCS-4" should be used for the various encodings and
transformations of Unicode/ISO/IEC 106466. XML applications are
expected to support Unicode. Other alternatives are also acceptable,
including "ISO-8859-1", "ISO-8859-2", ..."ISO-8859-9" for parts of

ISO 8859. Also the values "ISO-2022-JP", "Shift_JIS", and "EUC-JP"
should be used for the various encoded forms of JIS X-0208-1997.

Note the default encoding above has a restricted set of characters,
commonly known as "ISO-Latin 1". It may be inadequate for some books.
See the discussion in Section 3 below for ways to augment this encoding
with character entity sets.

This is followed by the document type declaration, the DOCTYPE:

```
<!DOCTYPE dtbook3 PUBLIC
    "-//NISO//DTD dtbook3.dtd//EN"
    "http://www.loc.gov/nls/niso/dtbook3.dtd"
```

The ExternalID begins with PUBLIC and is followed by the formal public
identifier, indicating

```
-//           unregistered
NISO//        owning organization
DTD           kind of external entity
dtbook3.dtd   current version
//EN          the public text language is English.
```

Then comes the PUBLIC URI for the DTD: where it may be found on the
internet.

```
"http://www.loc.gov/nls/niso/dtbook3.dtd"
```

The third string identifies the SYSTEM (local) version of the DTD.

```
"dtbook3.dtd"
```

In this form it is expected to be found rooted in the same directory as
this dtbook3 document. If it isn't there, find it from the PUBLIC URI.

This declaration identifies dtbook3 as the name of the root element.,
That name is the first to occur in any book that is tagged to this DTD.

Note that the reference above is to the latest version of the DTD.
When a document is tagged to a particular version of the DTD, the
following is the alternative, here referring to version 3-07.

```
<!DOCTYPE dtbook3 PUBLIC
    "-//NISO//DTD dtbook3.dtd Version 3-07 2001-01-31//EN"
    "http://www.loc.gov/nls/niso/dtbk3-07.dtd"
    "dtbk3-07.dtd">
```

3. Modular Extension to the DTD

The dtbook3 DTD has two parameter entities defined that provide hooks
to allow an individual book to modularly extend the content models
for its block and inline parameter entities.

```
<!ENTITY % externalblock "">
<!ENTITY % externalinline "">
```

These hooks appear in corresponding block and inline content models.
With this "" content they have no effect on books tagged to the
dtbook3 DTD. In a book that needs a modular extension, values are
given by redefinition in the internal subset of that book. This
extends the dtbook3 DTD without having to change it.

A book can augment the dtbook3 DTD by including other declarations
in the internal subset of declarations (in square brackets before the
concluding ">") of the initial DOCTYPE declaration that identifies
the dtbook3 DTD. Those additional declarations in the internal subset
take preference over any in the dtbook3 DTD. The effective DTD is
thereby augmented by the parameter entity values and any other
declarations of the book's internal subset. For example:

```
<!DOCTYPE dtbook3 PUBLIC
"-//NISO//DTD dtbook3.dtd//EN"
"http://www.loc.gov/nls/niso/dtbook3.dtd"
"dtbook3.dtd"
[
        <!ENTITY % dramaModule
"http://www.loc.gov/nls/niso/drama.dtd" >
        %dramaModule;
        <!ENTITY % externalblock "| drama">
        <!ENTITY % externalinline "| stagedir">
]>
```

The "%dramaModule;" invocation causes all declarations made within
dramaModule to become the initial part of the dtbook3 DTD. Within the
book, the empty entity declarations for both % externalblock and for
% externalinline are replaced by these new definitions. Thus the
block element drama can appear wherever block elements may occur in
dtbook3. Similarly any actual content needed for %externalinline;
can appear in that extension to wherever %inline; appears in the DTD.

More than one module may be needed and included in a book, for example
both poem and drama can appear in the internal subset of the book.
For example:

```
[
        <!ENTITY % poemModule
"http://www.loc.gov/nls/niso/poem.dtd" >
        %poemModule;
        <!ENTITY % dramaModule
"http://www.loc.gov/nls/niso/drama.dtd" >
        %dramaModule;
        <!ENTITY % externalblock "| poem | drama" >
        <!ENTITY % externalinline "| stagedir">
]>
```

Note that arbitrary external modules from other sources may not have
all the needed attributes. XML allows augmentation of ATTLISTs in the

internal subset. For each module, some accommodation to its use in dtbook3 may be required.

Also note that element name collisions may be possible, with names in those modules and associated content models overriding those in dtbook3. For modules under control of dtbook3 design, such collisions can be avoided. A more general solution uses namespace prefixes to element and attribute names to clearly indicate the module source.

The fully marked-up drama follows. Since declarations in the internal subset take precedence over like-named ones from the external entity containing the base DTD (that is, dtbook3), the module containing the drama tags is included along with the tags in the base DTD that are not duplicated in the drama module. So if a <p> tag is defined in the drama module, its definition overrides that of the <p> tag in dtbook3.

Note that tools and players processing any extended markup that affects navigation structure will need to know of modular extensions.

The list of URLs for DTDs of compatible modules may be found at:

        "http://www.loc.gov/nls/niso/dtbook3modules.htm"

Note that the default encoding in the initial processing instruction,

        encoding="ISO-8859-1"

may be inadequate for some books or some modules, such as math. One of the ISO 10646-based encodings should be used, such as UTF-8 or UTF-16, which all XML processors must be able to read. An older alternative is to identify and activate needed character entity sets. The character entity names suggest their appearance, and may be usefully spelled out, in place of their visual rendering.

4. References

The latest version of the dtbook3 DTD is available at

        http://www.loc.gov/nls/niso/dtbook3.dtd

Earlier individual versions are available, differentiated by suffix: This version is dtbk3-07.dtd.

        http://www.loc.gov/nls/niso/dtbk3-07.dtd

The history of changes prior to this dtbk3-07.dtd is in:

        http://www.loc.gov/nls/niso/dtbk3-dtd-changes.txt

Expanded DTD documentation of the current DTD is available as an HTM 4.0 document at:

        http://www.loc.gov/nls/niso/dtbook3doc.htm

A specific version documentation is available, of the form

    http://www.loc.gov/nls/niso/dtbook3-07doc.htm

The SMIL 1.0 specification is available at:

    http://www.w3.org/TR/REC-smil

The SMIL 2.0 specification has not yet been approved as of this
version 7 of dtbook3. It is expected to be adopted, and used.
The latest working draft, subject to change, is available at:

    http://www.w3.org/TR/smil20

This dtbook3.dtd is an application of the Extensible Markup Language
(XML) 1.0. The XML specification is available at:

    http://www.w3.org/TR/REC-xml

Dtbook3 is based on an XML version of the HTML 4.0 Strict DTD,i
with design adaptation for dtbook3. That HTML DTD and more detailed
documentation of the element types that come from that DTD is
available at:

    http://www.w3.org/TR/REC-html40

-->

<!--hb: change record:
    1998-10-08 original by Harvey Bingham
    1999-07-20 revision 3-04
    1999-09-16 revision 3-05
    1999-09-24 revision 3-06 See revisions in change history:
                          chgs0106.txt
                  from the start thru version 6
    2001-01-31 revision 3-07

            1. Updated version number to 3-07.

            2. Moved prior changes to change history file:

                "http://www.loc.gov/nls/niso/chgs0106.txt"

            3. externalblock may gain content from the book
               internal subset, to enable addition of block-level
               element types:
               for examples,
                   <!ENTITY % externalblock "| drama" >
               or if both drama and play were needed:
                   <!ENTITY % externalblock "| drama | poem" >

            4. externalinline may augment the inlines used in the
               book, if in the book internal subset a different value

is assigned to it, representing the inlines needed by
the module or modules included in that internal subset.
For example a book needing action concurrent with a
speech, with content model %inline; might include in
the parameter entity of the internal subset of the book:
    `<!ENTITY % externalinline "| action" >`

5. % list content replaced ul and ol by list. Remains
   a parameter entity, so can be extended in book
   internal subset if desired.

6. % dtbook3block added %externalblock; to allow extension
   of other block element types as needed by external
   module or modules.

7. % inlineinblock Added <imgcaption> to contain and
   associate with <img> its caption.

8. events attribute from HTML 4.0 is left empty,
   but could be added in book internal subset.

9. showin attribute for text elements added to permit
   identification of the kinds of display appropriate
   for the element, so choice of presentation by the
   reader among alternative readings can be provided,
   when appropriate. Values of showin are:

code	Braille	Large print	Print	
xxx				hide
xxp			p	print only
xls		l		large print only
xlp		l	p	largeprint and print
bxx	b			braille only
bxp	b		p	braille and print
blx	b	l		braille and largeprint
blp	b	l	p	braille, largeprint, and print

The default value is blp.

10. prodnote added imgref attribute to associate
    the prodnote with one or more images.

11. sidebar added hd to content model.

12. poem eliminated, and its sub-elements author,
    linegroup, lin. If poem is needed, bring in
    poem module through book internal subset.

13. drama eliminated and its sub-elements act,
    scene, stage, speaker, speech. If drama is
    needed, bring in through book internal subset.

14. line changed name of <lin> to line, made

it generally

available, such as in law text, not limited to poem.

15. img Added to Use: mention of description using
    <imgcaption>. Also that id is needed when

referenced

by <imgcaption imgref="#yyy">.

16. imggroup element added to contain <img> and its
    associated <imgcaption> and

<prodnote>. Content

model allows multiple <img> if they share

a caption,

multiple <imgcaption> if several captions refer to
a single <img>, and multiple

<prodnote> if different

versions are needed for different media.

17. imgcaption contains any printed caption that
    identifies one or more <img>. Associate

<imgcaption>

with each particular <img> by the imgref attribute
value referencing each of the id values of the

<img>s.

The <imgcaption> may amplify or repeat the alt
attribute string content on the <img>.

18. hd Use is revised to remove drama and poem references.

19. hd eliminated attribute level (or depth), as the
    hd list container indicates that information if needed.

20. blockquote Repaired content model to put #PCDATA first.
    Eliminated %inlineinquote; Note a <blockquote> may
    contain many subsidiary blocks.

21. list Replaced ol and ul, added attributes, and
    extended content model to allow heading <hd> and
    <prodnote>.

22. List attributes may indicate the list kind.
    Presence of these attributes can help alternative
    assistive technology reconstruct the list original use.
        class (part of %attrs;) can describe unusual lists,
            such as table of contents, list of figures,
            or list of tables. For these, special
            stylesheet treatment is appropriate.
        type
            "ul" unordered list,
            "ol" ordered list for which enum indicates form
        enum   enumeration in ordered list: enum values:
            "1" numeric
            "a" lower case alphabetic

```
                                    "A" upper case alphabetic
                                    "i" lower case roman
                                    "I" upper case roman
                              bullet  character or description of image used
                                      for unordered list item prefix

                23. lic is a list item component within a <li>.

                24. link moved from elsewhere in DTD to section on
                    Document Head where it is only used.

                25. meta removed reference to Dublin Core.
                    Most metadata about a book is available in
                    the package file.

                26. Added comment section 3 on Modular Extension to DTD.

                27. Edited text of many Use: comments

        -->


        <!-- Comment Classification Conventions

            Some comments start with a pattern followed by a colon:

                Use: element type and its use.
                hb: Explanation by Harvey Bingham.
                HB: yyyy-mm-dd is dated revision comment by Harvey Bingham.

            Other comments without such a pattern are details about the
            DTD structure, or about dtbook3 objects.
        -->

    <!--==================== Character Entities
    ==============================-->

        <!-- Characters for interoperability.
            A few characters may have special markup meaning, so are expressed
            as character entities in text, preceded by "&" and followed by ";".
                "lt" &lt;
                "gt" &gt;
                "amp" &amp;
                "apos" neutral single quote
            The notation below #decimal-number indicates the character code
            position, for unicode or ascii. Note that the < and & characters
            in the declarations of "lt" and "amp" are doubly escaped to meet the
            requirement that entity replacement be well-formed.
        -->

    <!ENTITY lt "&#38;#60;">
    <!ENTITY gt "&#62;">
    <!ENTITY amp "&#38;#38;">
    <!ENTITY apos "&#39;">
```

```
<!ENTITY quot "&#34;">

     <!-- Three larger character sets included in HTML 4.0 are omitted here:
         HTMLlat1.ent, HTMLsymbol.ent, and HTMLspecial.ent.
         Unicode is available to XML applications, so these characters are
         available. The initial processing instruction that identifies
         dtbook3 as an XML application should use a more inclusive encoding,
         as described at the start of section 2, for an even more extensive
         set is needed.
     -->


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


     <!--These parameter entities are used in HTML. Note that in HTML 4.0
         and XHTML, all variants (loose, strict, frame, transitional)
         these names are lowercase.
             ContentType, ContentTypes, Charset, Charsets, LanguageCode,
             LinkTypes, MediaDesc, URI, Datetime, Script, StyleSheet, Text
         But no stylesheetcontent or scriptcm appears in any HTML4 or XHTML DTD.

     -->

<!ENTITY % contenttype "CDATA">

<!ENTITY % contenttypes "CDATA">

<!ENTITY % charset "CDATA">

<!ENTITY % charsets "CDATA">

<!ENTITY % languagecode "NMTOKEN">

<!ENTITY % character "CDATA">

<!ENTITY % linktypes "CDATA">

<!ENTITY % mediadesc "CDATA">

<!ENTITY % uri "CDATA">

<!ENTITY % datetime "CDATA">

<!ENTITY % script "CDATA">

<!ENTITY % scriptcm "#PCDATA">

<!ENTITY % stylesheet "CDATA">

<!ENTITY % stylesheetcontent "#PCDATA">

<!ENTITY % text "CDATA">
```

```
<!--================= dtbook3 External Module Inclusion
==================-->

    <!--HB: 2001-01-31 externalblock
        may gain content from the book
        internal subset, to enable addition of block-level
        element types: for examples,
            <!ENTITY % externalblock "| drama" >
        or if both drama and play were needed:
            <!ENTITY % externalblock "| drama | poem" >
    -->


<!ENTITY % externalblock "">

    <!--HB: 2001-01-31 externalinline
        may augment the inlines used in the
        book, if in the book internal subset a different value
        is assigned to it, representing the inlines needed by
        the module or modules included in that internal subset.
        For example a book needing action concurrent with a
        speech, with content model %inline; might include
        in the parameter entity of the internal subset of the book:
            <!ENTITY % externalinline "| action" >
    -->


<!ENTITY % externalinline "">



<!--================= dtbook3 Content Models
=============================-->

    <!--HB: 2001-01-31 % list
        content replaced ul and ol by list. Remains
        a parameter entity, so can be extended in book
        internal subset if desired. -->
<!ENTITY % list "list">

    <!--HB: 2001-01-31 % dtbook3block
        added %externalblock; to allow extension
        of other block element types as needed by external
        module or modules.
    -->
<!ENTITY % dtbook3block
    "author | notice | prodnote | sidebar | note %externalblock;">

    <!--HB: 2001-01-31 % inlineinblock
        Added <imgcaption> to contain and
        associate with <img> its caption.
    -->
<!ENTITY % inlineinblock
    "a | citation | img | imgcaption | code | samp | kbd | var | pagenum">

<!ENTITY % block
    "p | %list; | dl | div | blockquote | hr
```

```
                 | table | address | noscript | %dtbook3block;">

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

       <!-- Omitted as XML uses Unicode, so doesn't need them. May need
           character entities if the encoding is the minimal -->

   <!--================== Generic Attributes
   ===============================-->

   <!ENTITY % coreattrs
     "id           ID            #IMPLIED
      class        CDATA         #IMPLIED
      style        %stylesheet;  #IMPLIED
      title        %text;        #IMPLIED   ">

   <!ENTITY % i18n
     "lang         %languagecode; #IMPLIED
      dir          (ltr|rtl)      #IMPLIED   ">

       <!--HB: 2001-01-31 events
           attribute from HTML 4.0 is left empty,
           but could be added in book internal subset.

       -->
   <!ENTITY % events
     "">

       <!--HB: 2001-01-31 showin
       attribute for text elements added to permit
       identification of the kinds of display appropriate
       for the element, so choice of presentation by the
       reader among alternative readings can be provided,
       when appropriate. Values of showin are:

             Braille Large Print
         code         print        Interpretation

         xxx                       hide
         xxp                    p  print only
         xls          l            large print only
         xlp          l         p  largeprint and print
         bxx     b                 braille only
         bxp     b              p  braille and print
         blx     b    l            braille and largeprint
         blp     b    l         p  braille, largeprint, and print

       The default value, blp , may be omitted.
       -->

   <!ENTITY % attrs
     "%coreattrs;
      %i18n;
```

```
      showin         (xxx|xxp|xlx|xlp|bxx|bxp|blx|blp) #IMPLIED
      %events;">


   <!ENTITY % attrsrqd
     "id            ID              #REQUIRED
      class         CDATA           #IMPLIED
      style         %stylesheet;    #IMPLIED
      title         %text;          #IMPLIED
      %i18n;
      showin         (xxx|xxp|xlx|xlp|bxx|bxp|blx|blp) #IMPLIED
      %events;">


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


   <!ENTITY % dtbook3.content "head, book">

        <!--Use: dtbook3 is the root element in the Digital Talking Book
            3.0 DTD. Contains metadata in <head> and the document itself in
            <book>.
        -->

   <!ELEMENT dtbook3 (%dtbook3.content;) >
   <!ATTLIST dtbook3
      %i18n;
      >


   <!--=================== Book Content
   ======================================-->

        <!--Use: book surrounds the actual content of the document, which
            is divided into <frontmatter>, <bodymatter>, and
   <rearmatter>.
            <head>, which contains metadata, precedes <book>.
        -->

   <!ELEMENT book (frontmatter, bodymatter?, rearmatter?) >
   <!ATTLIST book
      %attrs;
      onload            %script;   #IMPLIED
      onunload          %script;   #IMPLIED
      >


   <!--======================= Book Major Structures
   ========================-->

        <!--Use: frontmatter contains preliminary material such as the
            copyright notice, foreword, acknowledgments, table of contents, etc.
            which serves as a guide to the contents and nature of a book.
        -->

   <!ELEMENT frontmatter (doctitle, (level | level1 | %block; | script)+) >
   <!ATTLIST frontmatter
      %attrs;
```

```
        >

            <!--Use: bodymatter consists of the text proper of a book, as
                opposed to preliminary material <frontmatter> or supplementary
                information <rearmatter>.
            -->

    <!ELEMENT bodymatter  (level | level1 | %block; | script)+ >
    <!ATTLIST bodymatter
        %attrs;
        >

            <!--Use: rearmatter contains supplementary material such as
                appendices, glossaries, bibliographies, and indices following the
                <bodymatter> of the book.
            -->

    <!ELEMENT rearmatter  (level | level1 | %block; | script)+ >
    <!ATTLIST rearmatter
        %attrs;
        >

    <!--================= dtbook3 Recursive Structure level
    ================-->

            <!--Use: level is an alternative tag for marking the major
                structures in a book. It may be used recursively, i.e., repeated
                indefinitely with each successive occurrence nesting within the
                previous. It may also be included in a subsequent higher level.
                The class attribute identifies the actual name (e.g., part,
                chapter, section, subsection) of the structure it marks.
                the depth attribute indicates the nesting depth, starting at 1.
                Subordinate levels have greater depth.
            -->

    <!ELEMENT level ((levelhd | %block; | %inlineinblock; | level)*) >
    <!ATTLIST level
        %attrs;
        depth           CDATA               #IMPLIED
        >

    <!--============ dtbook3 Hierarchic Structure level1 ... level6
    ==========-->


            <!--Use: level1 is the highest level container of major divisions of
                a book. Used in <frontmatter>, <bodymatter>, and
    <rearmatter> to
                mark the largest divisions of the book (usually parts or chapters),
                inside which level2 subdivisions (often sections) may nest.
                The class attribute identifies the actual name (e.g., part, chapter)
                of the structure it marks.
            -->
```

```
<!ELEMENT level1 ((h1 | level2 | %block; | %inlineinblock;)*) >

    <!--Use: level2 contains subdivisions that nest within <level1>
        divisions. The class attribute identifies the actual name (e.g.,
        subpart, chapter, subsection) of the structure it marks.
    -->

<!ELEMENT level2 ((h2 | level3 | %block; | %inlineinblock;)*) >

    <!--Use: level3 contains subdivisions that nest within <level2>
        subdivisions (e.g., subsections within subsections). The class
        attribute identifies the actual name (e.g., section, subpart,
        subsubsection) of the subordinate structure it marks.
    -->

<!ELEMENT level3 ((h3 | level4 | %block; | %inlineinblock;)*) >

    <!--Use: level4 contains subdivisions that nest within <level3>
        subdivisions. The class attribute identifies the actual name
        of the subordinate structure it marks.
    -->

<!ELEMENT level4 ((h4 | level5 | %block; | %inlineinblock;)*) >

    <!--Use: level5 contains subdivisions that nest within <level4>
        subdivisions. The class attribute identifies the actual name
        of the subordinate structure it marks.
    -->

<!ELEMENT level5 ((h5 | level6 | %block; | %inlineinblock;)*) >

    <!--Use: level6 contains subdivisions that nest within <level5>
        subdivisions. The class attribute identifies the actual name
        of the subordinate structure it marks.
    -->

<!ELEMENT level6 ((h6 | %block; | %inlineinblock;)*) >

<!ATTLIST level1
   %attrs;
   >

<!ATTLIST level2
   %attrs;
   >

<!ATTLIST level3
   %attrs;
   >

<!ATTLIST level4
   %attrs;
   >
```

```
<!ATTLIST level5
    %attrs;
    >

<!ATTLIST level6
    %attrs;
    >

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

<!ENTITY % phrase
      "em | strong | dfn | code | samp | kbd | var | citation | abbr |
acronym">

<!ENTITY % special
      "a | img | object | br | script | q | sub | sup | span | bdo | linenum">

<!ENTITY % speciala
      "img | object | br | script | q | sub | sup | span | bdo | linenum">


<!--========================= Inline Entities
=============================-->


<!ENTITY % dtbook3inline
       "sent | w | pagenum | prodnote | noteref %externalinline;">

<!ENTITY % inline "#PCDATA | %phrase; | %special; | %dtbook3inline;">

      <!-- %inlinea; for a excludes nested a -->
<!ENTITY % inlinea "#PCDATA | %phrase; | %speciala; | %dtbook3inline;">

      <!-- %inlines; excludes direct nesting of sentences sent. -->
<!ENTITY % inlines "#PCDATA | %phrase; | %special; | pagenum | w |
          prodnote | noteref">

      <!-- %inlinew; for word w excludes any of the %dtbook3inline;. -->
<!ENTITY % inlinew "#PCDATA | %phrase; | %special;">

      <!-- %inlinenopagenum; excludes pagenum in table th and td
directly. -->
<!ENTITY % inlinenopagenum
      "#PCDATA | %phrase; | %special; | sent | w | noteref" >

<!ENTITY % inlinenoprodnote
      "#PCDATA | %phrase; | %special; | sent | w | pagenum | noteref" >

<!--==================== Flow (Block or Inline) Entities
==================-->

<!ENTITY % flow "%inlinenoprodnote; | %block;" >
```

```
        <!-- ideally excludes pagenum though can get in through %block; -->
<!ENTITY % flownopagenum "%inlinenopagenum; | %block;" >


<!--============ Br, Linenum, Address, and Div Content Models
============-->



        <!--Use: br marks a forced line break.-->

<!ELEMENT br EMPTY >

<!ATTLIST br
   %coreattrs;
   >
        <!--Use: linenum contains a line number in, for example, legal text.
        -->

<!ELEMENT linenum (#PCDATA) >

<!ATTLIST linenum
   %attrs;
   >



        <!--Use: address contains a location at which a person or agency
           may be contacted.
        -->

<!ELEMENT address (%inline;)* >
<!ATTLIST address
   %attrs;
   >

        <!--Use: div is a generic container for subdivisions of a book. The
           <level1> ... <level6> hierarchy, or the
<level> tag used recursively,
           should mark the major hierarchical structures of a book,
while <div>
           is used in less formal circumstances or when for production purposes
           it is desired that a structure should be treated differently. The
           class attribute identifies the actual name (e.g., part, chapter,i
           letter) of the structure it marks. Compare with <span> which is
           used in inline settings.
        -->

<!ELEMENT div (%block; | %inlineinblock;)* >

        <!--div level attribute that may extend or augment explicit levels,
           to indicate nesting level, with 1 corresponding to h1, and
           values 1, 2, ... . -->
<!ATTLIST div
   %attrs;
   level       CDATA      #IMPLIED
   >
```

```
<!--====== dtbook3 Block Elements Author, Notice, Prodnote,
Sidebar =======-->


    <!--Use: author identifies the writer of a given work.
    -->

<!ELEMENT author (%inline;)* >

<!ATTLIST author
   %attrs;
   >

    <!--Use: notice contains a warning, caution, or other type of
        admonition normally found in the margin of a book. Differs from a
        sidebar in that a notice must be presented at a specific location
        within the text and its presentation is not optional.
    -->

<!ELEMENT notice (%inline;)* >

<!ATTLIST notice
   %attrs;
   >

    <!--HB: 2001-01-31 prodnote added imgref attribute to associate
        the prodnote with one or more images.
    -->
    <!--Use: prodnote contains language added to the alternative-format
        version by the producers; commonly used to provide verbal
        descriptions of visual elements such as charts, graphs, etc.;
        to supply operating instructions; or to describe
        differences between the print book and the audio version.
    -->

<!ELEMENT prodnote (%flow;)* >

    <!--hb: prodnote may be specified to be required or optional for the
        user. If optional, some user preference may allow skipping over the
        content. but prodnote render="required" is essential for user to hear,
        An audible cue could announce the presence of the prodnote.
    -->
<!ATTLIST prodnote
   %attrs;
   imgref        IDREFS              #IMPLIED
   render (required | optional)    #IMPLIED
   >

    <!--HB: 2000-01-30 sidebar added hd to content model.-->
    <!--Use: sidebar contains information supplementary to the main
        text and/or narrative flow and is often boxed and printed apart
        from the main text block on a page.
    -->
```

```
<!ELEMENT sidebar (%flow;| hd)* >

    <!-- sidebar with id="xxx" is referenced in text via an a
HREF="#xxx" -->
<!ATTLIST sidebar
    %attrs;
    >

    <!--Use: note marks a footnote, endnote, annotation, etc. The
        reference to the note within the text is marked with a <noteref>.
    -->

<!ELEMENT note (%block; | %inlineinblock;)+ >
    <!--hb: note with id="yyy" is referenced in text by noteref
idref="#yyy" -->

<!ATTLIST note
    %attrsrqd;
    >
    <!--HB: 2001-01-31 poem eliminated, and its subelements author,
        linegroup, lin. If poem is needed, bring in poem module
        through book internal subset.
    -->

    <!--HB: 2001-01-31 drama eliminated and its subelements act,
        scene, stage, speaker, speech.  If drama is
        needed, bring in through book internal subset.
    -->

    <!--HB: 2001-01-31 line changed name of lin to line, made it generally
        available, such as in law text, not limited to poem.
    -->
    <!--Use: line marks a single logical line of text. Often used in
        conjunction with <linenum> in documents with numbered lines.
    -->

<!ELEMENT line (%inline;)* >

<!ATTLIST line
    %attrs;
    >

<!--================== The Anchor Element
==================================-->


<!ENTITY % shape "(rect | circle | poly | default)">

<!ENTITY % coords "CDATA">

    <!--Use: a contains an anchor, which is used in two ways: A name
        anchor identifies an exact position in a given document. A link
        anchor, when activated, "links to" (repositions the focus to)
```

```
                another location within that document or another document.
          -->


    <!ELEMENT a (%inlinea;)* >


    <!ATTLIST a
       %attrs;
       charset      %charset;       #IMPLIED
       type         %contenttype;   #IMPLIED
       name         CDATA           #IMPLIED
       href         %uri;           #IMPLIED
       hreflang     %languagecode;  #IMPLIED
       rel          %linktypes;     #IMPLIED
       rev          %linktypes;     #IMPLIED
       accesskey    %character;     #IMPLIED
       shape        %shape;         "rect"
       coords       %coords;        #IMPLIED
       tabindex     NMTOKEN         #IMPLIED
       onfocus      %script;        #IMPLIED
       onblur       %script;        #IMPLIED
       >


    <!--========================= Inline Elements
    =============================-->



         <!--Use: em indicates emphasis. Compare with <strong>.
         -->


    <!ELEMENT em (%inline;)* >

         <!--Use: strong marks stronger emphasis than <em>.
         -->


    <!ELEMENT strong (%inline;)* >

         <!--Use: dfn marks the first occurrence of a word or term that is
             defined or explained elsewhere in a book.
         -->


    <!ELEMENT dfn (%inline;)* >

         <!--Use: code designates a fragment of computer code.
         -->


    <!ELEMENT code (%inline;)* >

         <!--Use: samp contains a sample of work created by the author for
             use as an example or template. For example, a sample business letter,
             resume, computer program output, or form.
         -->


    <!ELEMENT samp (%inline;)* >
```

```
        <!--Use: kbd designates information that the reader is to input
            directly into a computer using the keyboard.
        -->

<!ELEMENT kbd (%inline;)* >

        <!--Use: var indicates an instance of a variable or program
            argument. Commonly used as a placeholder for text to be entered by
            the user.
        -->

<!ELEMENT var (%inline;)* >

        <!--Use: citation marks a reference to another document.
        -->




<!ELEMENT citation (%inline;)* >

        <!--Use: abbr designates an abbreviation, a shortened form of a
            word. For example: Mr., approx ., lbs., rec'd.
        -->

<!ELEMENT abbr (%inline;)* >

        <!--Use: acronym marks a word formed from key letters (usually
            initials) of a group of words. For example: UNESCO, NATO, XML.
        -->

<!ELEMENT acronym (%inline;)* >

<!ATTLIST em
    %attrs;
    >
<!ATTLIST strong
    %attrs;
    >
<!ATTLIST dfn
    %attrs;
    >
<!ATTLIST code
    %attrs;
    >
<!ATTLIST samp
    %attrs;
    >
<!ATTLIST kbd
    %attrs;
    >
<!ATTLIST var
    %attrs;
    >
```

```
<!ATTLIST citation
    %attrs;
    >


<!ATTLIST abbr
    %attrs;
    >



<!ATTLIST acronym
    %attrs;
    pronounce    (yes | no)    #IMPLIED
    >
```

```
        <!--Use: sub indicates a subscript character (printed below a
            character's normal baseline). Can be used recursively and/or
            intermixed with <sup>.
        -->
```

```
<!ELEMENT sub (%inline;)* >
```

```
        <!--Use: sup marks a superscript character (printed above a
            character's normal baseline). Can be used recursively and/or
            intermixed with <sub>.
        -->
<!ATTLIST sub

    %attrs;
    >
```

```
<!ELEMENT sup (%inline;)* >

<!ATTLIST sup
    %attrs;
    >
```

```
      <!--Use: span is a generic container for use in inline settings
          when no specific tag exists for a given situation. The class
          attribute may describe the nature of the text it marks (e.g., a
          typographical error). May be used to mark a class of items to which
          styles are to be applied. Compare with <div> which is
used in block
          settings.
      -->
```

```
<!ELEMENT span (%inline;)* >

<!ATTLIST span
    %attrs;
    >
```

```
        <!--Use: bdo is used in special cases where the automatic actions
            of the bidirectional algorithm would result in incorrect display.
```

```
      -->

<!ELEMENT bdo (%inline;)* >

<!ATTLIST bdo
   %coreattrs;
   lang         %languagecode; #IMPLIED
   dir          (ltr|rtl)      #REQUIRED
   >

<!--==================== dtbook3 Inline Sentence and Word
==================-->


      <!--Use: sent marks a sentence.
      -->

<!ELEMENT sent (%inlines;)* >

<!ATTLIST sent
   %attrs;
   >

      <!--Use: w marks a word.
      -->

<!ELEMENT w (%inlinew;)* >

<!ATTLIST w
   %attrs;
   >

<!--========= dtbook3 Inline Page Number, and Footnote Reference
=========-->

      <!--Use: pagenum contains a page number from the print document,
         recorded as the first text object on a page. The"page" attribute
         allows three types of page numbering schemes to be identified:
         "normal" arabic numbering in the body of the book, "front" pages
         (from the frontmatter), and "special" pagination schemes such as
         hyphenated numbers in appendices.
      -->


<!ELEMENT pagenum (#PCDATA) >

      <!--pagenum unique id value is expected, by convention derived
         from the actual pagenumber.
         The value of the page attribute indicates kind of pagenumber. -->

<!ATTLIST pagenum
   %attrs;
   page   (front | normal | special)  #IMPLIED
   >
```

```
      <!--Use: noteref marks one or more characters that reference a footnote,
          endnote, or annotation <note>.
      -->

  <!ELEMENT noteref (#PCDATA) >


  <!ATTLIST noteref
     %attrs;
     idref        CDATA            #REQUIRED
     type         %contenttype;  #IMPLIED
     >

  <!--===================== Inline Quotes
===================================-->

      <!--Use: q contains a short, inline quotation. Compare with
          <blockquote> which marks a longer quotation set off from the
          surrounding text.
      -->

  <!ELEMENT q (%inline;)* >

  <!ATTLIST q
     %attrs;
     cite         %uri;            #IMPLIED
     >

  <!--================== Client-side image maps
=============================-->

      <!-- Omitted as not needed. -->

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

     <!-- Image <img> comes from HTML. An <img> may be grouped
          using <imggroup> with its caption <imgcaption>, and special
          usage instructions with <prodnote>. The <imggroup>
          element may contain one or more <img> and any associated
          <imgcaption> and <prodnote>. Multiple <img>
may share a single
          caption, or multiple <imgcaption> may apply if several captions
          refer to a single <img>. Multiple <prodnote> may
apply if different
          versions are needed for different media.
     -->

  <!ENTITY % length "CDATA">

  <!ENTITY % multilength "CDATA">

  <!ENTITY % multilengths "CDATA">
```

```
<!ENTITY % pixels "CDATA">

      <!--HB: 2001-01-31 img Added to Use: mention of description using
          <imgcaption>. Also that id is needed when referenced by
          <imgcaption imgref="#yyy">.
      -->
      <!--Use: img marks a visual image. The "src" attribute specifies
          the location of the image file. The "alt" and "longdesc" attributes
          may be used to supply short and long descriptions, respectively.
          may be used to supply short and long descriptions, respectively,
          although prodnote will generally contain the latter.  Longdesc
          may contain a pointer to the prodnote. The referencing is typically
          of the form <imgcaption imgref="#yyy">The
Caption</imgcaption>
          containing the printed caption for the <img id="yyy">.
      -->


<!ELEMENT img EMPTY >

<!ATTLIST img
   %attrs;
   src          %uri;           #REQUIRED
   alt          %text;          #REQUIRED
   longdesc     %uri;           #IMPLIED
   height       %length;        #IMPLIED
   width        %length;        #IMPLIED
   usemap       %uri;           #IMPLIED
   ismap        (ismap)         #IMPLIED
   >

      <!--HB: 2001-01-31 imggroup element added to contain <img> and its
          associated <imgcaption> and <prodnote>. Content
          model allows multiple <img> if they share a caption,
          multiple <imgcaption> if several captions refer to
          a single <img>, and multiple <prodnote> if different
          versions are needed for different media.
      -->
      <!--Use: imggroup provides a container for <img> and its
associated
          <imgcaption> and <prodnote>. <prodnote>
will contain a
          description of the image.  Content model allows multiple <img>
          if they share a caption, multiple <imgcaption> if several
          captions refer to a single <img>, and multiple
<prodnote> if
          different versions are needed for different media (e.g.,
          large print, braille, etc.)
      -->


<!ELEMENT imggroup (prodnote | img | imgcaption)+ >

<!ATTLIST imggroup
```

```
      %attrs;
      >

         <!--HB: 2001-01-31 imgcaption contains any printed caption that
             identifies one or more <img>. Associate
   <imgcaption> with each
             particular <img> by the imgref attribute value referencing
             each of the id values of the <img>s. The
   <imgcaption> may amplify
             or repeat the alt attribute string content on the <img>.
         -->
         <!--Use: imgcaption contains the caption for one or more <img>.
             If the caption applies to more than one <img>, each idref
             in the list of IDs in the imgref is separated by whitespace.
         -->


   <!ELEMENT imgcaption (%inline;)+ >

   <!ATTLIST imgcaption
      %attrs;
      imgref          IDREFS          #IMPLIED
      >


   <!--======================== Object
   ========================================-->

         <!--
         An object is used to embed objects as part of HTML pages
         <param> elements should precede other content. XML mixed content
         model technicality precludes specifying this formally ...
         -->

         <!--Use: object marks an embedded object, which may consist of
             scripts, applets, images, etc.
         -->

   <!ELEMENT object (%flow; | param)* >

   <!ATTLIST object
      %attrs;
      declare     (declare)       #IMPLIED
      classid     %uri;           #IMPLIED
      codebase    %uri;           #IMPLIED
      data        %uri;           #IMPLIED
      type        %contenttype;   #IMPLIED
      codetype    %contenttype;   #IMPLIED
      archive     %uri;           #IMPLIED
      standby     %text;          #IMPLIED
      height      %length;        #IMPLIED
      width       %length;        #IMPLIED
      usemap      %uri;           #IMPLIED
      name        CDATA           #IMPLIED
      tabindex    NMTOKEN         #IMPLIED
      >
```

```
    <!--Use: param provides a named property for <object>.
    -->

<!ELEMENT param EMPTY >

<!ATTLIST param
   id            ID                #IMPLIED
   name          CDATA             #REQUIRED
   value         CDATA             #IMPLIED
   valuetype     (data|ref|object) "data"
   type          %contenttype;     #IMPLIED
   >

<!--=================== Horizontal Rule
===================================-->


    <!--Use: hr is an empty element indicating a horizontal rule. May be
        used to indicate a break in the text where only blank lines, a row
        of asterisks, a horizontal line, etc. are used in the print book.
    -->

<!ELEMENT hr EMPTY >

<!ATTLIST hr
   %coreattrs;
   >

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


    <!--Use: p contains a paragraph.
    -->

<!ELEMENT p (%inline; | %list; | dl)* >

<!ATTLIST p
   %attrs;
   >

<!--==================== Doctitle and Headings
===========================-->

    <!--hb: Reordered to move attribute lists under their element
types. -->

    <!--Use: doctitle marks the title of the book, as the first tag
        within <frontmatter>. It is used to quickly identify the book.
    -->

<!ELEMENT doctitle (%inline;)* >
```

```
<!ATTLIST doctitle
   %attrs;
   >

      <!--Use: levelhd contains the text of a heading within <level>.
         Corresponds to <h1> through <h6> used in
<level1> through  <level6>.
      -->

<!ELEMENT levelhd (%inline;)* >

<!ATTLIST levelhd
   %attrs;
   depth          CDATA              #IMPLIED
   >

      <!--Use: h1 contains the text of the heading for a <level1>
         structure.
      -->

<!ELEMENT h1 (%inline;)* >

<!ATTLIST h1
   %attrs;
   >

      <!--Use: h2 contains the text of the heading for a <level2>
         structure.
      -->

<!ELEMENT h2 (%inline;)* >

<!ATTLIST h2
   %attrs;
   >

      <!--Use: h3 contains the text of the heading for a <level3>
         structure.
      -->

<!ELEMENT h3 (%inline;)* >

<!ATTLIST h3
   %attrs;
   >

      <!--Use: h4 contains the text of the heading for a <level4>
         structure.
      -->

<!ELEMENT h4 (%inline;)* >

<!ATTLIST h4
   %attrs;
```

```
    >

        <!--Use: h5 contains the text of the heading for a <level5>
            structure.
        -->

<!ELEMENT h5 (%inline;)* >

<!ATTLIST h5
    %attrs;
    >

        <!--Use: h6 contains the text of the heading for a <level6>
            structure.
        -->

<!ELEMENT h6 (%inline;)* >


<!ATTLIST h6
    %attrs;
    >

        <!--HB: 2001-01-31 hd use is revised to remove drama and poem
references.
        -->
        <!--Use: hd marks the text of a heading in a <list> or
<sidebar>.
        -->

<!ELEMENT hd (%inline;)* >

        <!--HB: 2001-01-31 hd eliminated attribute level (or depth), as the
            hd list container indicates that information if needed.
        -->

<!ATTLIST hd
    %attrs;
    >

<!--=================== Preformatted Text
=================================-->


        <!-- HTML or XHTML preformatted text is omitted, as inappropriate for
            narrated material. -->

<!--=================== Block-like Quotes
=================================-->


        <!--HB: 2001-01-31 blockquote
            Repaired content model to put #PCDATA first.
            eliminated %inlineinquote; Note a <blockquote> may
```

```
                contain many subsidiary blocks.
        -->
        <!--Use: blockquote indicates a block of quoted content that is set
            off from the surrounding text by paragraph breaks. Compare with
            <q> which marks short, inline quotations.
        -->

    <!ELEMENT blockquote (%inline; | %block; | script)+ >

    <!ATTLIST blockquote
        %attrs;
        cite          %uri;              #IMPLIED
        >


    <!--================== Inserted/Deleted Text
    ============================-->

        <!-- Removed as unnecessary for narrated material. -->

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

        <!--Use: dl contains a definition list, usually consisting of
            pairs of terms <dt> and definitions <dd>. Any
definition can contain
            another definition list.
        -->

    <!ELEMENT dl (dt | dd | pagenum)+ >

    <!ATTLIST dl
        %attrs;
        >

        <!--Use: dt marks a term in a definition list.
        -->

    <!ELEMENT dt (%inline;)* >

    <!ATTLIST dt
        %attrs;
        >

        <!--Use: dd marks a definition of a term within a definition list.
        -->

    <!ELEMENT dd (%flow;)* >

    <!ATTLIST dd
        %attrs;
        >

        <!--HB: 2001-01-31 list Replaced ol and ul, added attributes, and
            extended content model to allow heading <hd>, <prodnote>,
```

```
                  and <pagenum>.
          -->

          <!--Use: list contains a list. The "type" attribute can indicate
              whether a list is ordered or unordered.
          -->

   <!ELEMENT list (prodnote*, hd?, (prodnote | li | pagenum)+) >

          <!--HB: 2001-01-31 List attributes may indicate the list kind.
              Presence of these attributes can help alternative assistive
              technology reconstruct the list original use.
                   class (part of %attrs;) can describe unusual lists,
                       such as table of contents, list of figures,
                       or list of tables. For these special
                       stylesheet treatment is appropriate.
                   type
                       "ul" unordered list,
                       "ol" ordered list for which enum indicates form
                   enum  enumeration in ordered list: enum values:
                       "1"  numeric
                       "a"  lower case alphabetic
                       "A"  upper case alphabetic
                       "i"  lower case roman
                       "I"  upper case roman
                   bullet   character or description of image used
                            for unordered list item prefix
          -->
   <!ATTLIST list
       %attrs;
       type    (ol | ul)             #IMPLIED
       depth   CDATA                 #IMPLIED
       enum    (1 | a | A | i | I)   #IMPLIED
       bullet  CDATA                 #IMPLIED
       >

          <!--Use: li marks each list item in a <list>. <li>
   content may be
              either inline or block and may include other nested
              lists. Alternatively it may contain a sequence of list item
              components, <lic>, that identify regularly occuring content,
              such as the heading and page number of each entry in a
              table of contents.
          -->

   <!ELEMENT li ((%flow;)* | lic+) >

   <!ATTLIST li
       %attrs;
       >

          <!--HB: 2001-01-31 lic is list item component within a <li>.
          -->
          <!--Use: lic ("list item component") allows ordered substructure
```

```
            within a list item <li>.  Used when a list item is made up of
            two or more components, as in a table of contents entry.
        -->

    <!ELEMENT lic (%inline;)+ >

        <!--lic class attribute may be used to identify the particular
            component of a list item <li>. For example, in a table
of contents
            class values might include "section", and "pagenumber". Nested
            in another <list> therein "subsection", and its "pagenumber".
        -->
    <!ATTLIST lic
        %attrs;
        >



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

        <!-- HTML or XHTML forms are not needed or supported. -->


    <!--======================= Tables
========================================-->

        <!--hb: HTML table model is used including the presentational
            attributes that have little meaning in Digital Talking Books,
            but may be useful for concurrent display in different media.
        -->

    <!ENTITY % tframe "(void | above | below | hsides | lhs | rhs | vsides |
        box | border)">

    <!ENTITY % trules "(none | groups | rows | cols | all)">

    <!ENTITY % talign "(left | center | right)">

    <!ENTITY % cellhalign
        "align       (left|center|right|justify|char) #IMPLIED
         char        %character;     #IMPLIED
         charoff     %length;        #IMPLIED  ">

    <!ENTITY % cellvalign
        "valign      (top|middle|bottom|baseline) #IMPLIED">

        <!--Use: table contains a table data arranged in rows and columns.
        -->

    <!ELEMENT table (caption?, (col*|colgroup*), thead?, tfoot?, tbody+) >

        <!--Use: caption describes a table. If used, it must follow
            immediately after the table start tag.
        -->
```

```
<!ELEMENT caption (%inline;)* >

     <!--Use: thead marks header information in a table, consisting of
         one or more rows (each marked with the tr tag) of <th> cells.
     -->

<!ELEMENT thead (tr)+ >

     <!--Use: tfoot marks table footer information, consisting of one
         or more rows (each marked with the tr tag).
     -->

<!ELEMENT tfoot (tr)+ >

     <!--Use: tbody marks a group of rows in the main body of a table.
         If the table is divided into several sections, each consisting of a
         number of rows, each section would be separately tagged with tbody.
     -->

<!ELEMENT tbody (tr)+ >

     <!--Use: colgroup is a group of columns that may share attribute
         values within a table.
     -->

<!ELEMENT colgroup (col)* >

     <!--Use: col is a means to apply attribute values to table
         columns.
     -->

<!ELEMENT col EMPTY >

     <!--Use: tr marks one row of a table containing <th> or
<td> cells.
     -->

<!ELEMENT tr (th|td)+ >

     <!--Use: th indicates a table cell containing header information.
     -->

<!ELEMENT th (%flownopagenum;)* >

     <!--Use: td indicates an individual data cell in the body of a table.
     -->

<!ELEMENT td (%flownopagenum;)* >

<!ATTLIST table
    %attrs;
    summary        %text;          #IMPLIED
    width          %length;        #IMPLIED
```

```
        border          %pixels;          #IMPLIED
        frame           %tframe;          #IMPLIED
        rules           %trules;          #IMPLIED
        cellspacing  %length;          #IMPLIED
        cellpadding  %length;          #IMPLIED
        datapagesize CDATA             #IMPLIED
        >


    <!ENTITY % calign "(top | bottom | left | right)">


    <!ATTLIST caption
        %attrs;
        >


    <!ATTLIST colgroup
        %attrs;
        span            NMTOKEN          "1"
        width           %multilength;  #IMPLIED
        %cellhalign;
        %cellvalign;
        >


    <!ATTLIST col
        %attrs;
        span            NMTOKEN          "1"
        width           %multilength;  #IMPLIED
        %cellhalign;
        %cellvalign;
        >


    <!ATTLIST thead
        %attrs;
        %cellhalign;
        %cellvalign;
        >


    <!ATTLIST tbody
        %attrs;
        %cellhalign;
        %cellvalign;
        >


    <!ATTLIST tfoot
        %attrs;
        %cellhalign;
        %cellvalign;
        >


    <!ATTLIST tr
        %attrs;
        %cellhalign;
        %cellvalign;
        >
```

```
<!ENTITY % scope "(row | col | rowgroup | colgroup)">

<!ATTLIST th
   %attrs;
   abbr          %text;          #IMPLIED
   axis          CDATA           #IMPLIED
   headers       IDREFS          #IMPLIED
   scope         %scope;         #IMPLIED
   rowspan       NMTOKEN         "1"
   colspan       NMTOKEN         "1"
   %cellhalign;
   %cellvalign;
   >


<!ATTLIST td
   %attrs;
   abbr          %text;          #IMPLIED
   axis          CDATA           #IMPLIED
   headers       IDREFS          #IMPLIED
   scope         %scope;         #IMPLIED
   rowspan       NMTOKEN         "1"
   colspan       NMTOKEN         "1"
   %cellhalign;
   %cellvalign;
   >


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

    <!--Use: head contains metainformation about the book but no
        actual content of the book itself, which is placed in <book>.
        This information is consonant with the head information
        in xhtml. See: http://www.w3.org/
    -->

<!ENTITY % head.misc "meta | link | style | script | object">

<!ELEMENT head (title, (%head.misc; | base)*) >
<!ATTLIST head
   %i18n;
   profile     %uri;          #IMPLIED
   >

    <!--Use: title contains the title of the book but is used only as
        metainformation in <head>. Use <doctitle> within
<book> for
        the actual book title, which will usually be the same.
    -->

<!ELEMENT title (#PCDATA) >
<!ATTLIST title
   %i18n;
   >
```

```
<!--HB: 2001-01-31 link moved from elsewhere in DTD to section on
    Document Head where it is only used.
-->
<!--Use: link is an empty element appearing in the <head> section
    of a document that establishes a connection between the current
    document and another document(s). The <link> element conveys
    relationship information (for example, "next" and "previous") that
    may be rendered by user agents in a variety of ways.
-->
```

```
<!ELEMENT link EMPTY >

<!ATTLIST link
   %attrs;
   charset      %charset;        #IMPLIED
   href         %uri;            #IMPLIED
   hreflang     %languagecode;   #IMPLIED
   type         %contenttype;    #IMPLIED
   rel          %linktypes;      #IMPLIED
   rev          %linktypes;      #IMPLIED
   media        %mediadesc;      #IMPLIED
   >
```

```
<!--Use: base contains the base URI from which local references
    start. It acts as an absolute URI that serves as the base URI for
    resolving relative URIs found within the document. It is an empty
    element that may appear only in <head>.
-->
```

```
<!ELEMENT base EMPTY >
<!ATTLIST base
   href         %uri;            #REQUIRED
   >
```

```
<!--Use: meta indicates metadata about the book. It is an empty
    element that may appear only in <head>.
-->
```

```
<!ELEMENT meta EMPTY >
```

```
<!--HB: 2001-01-31 meta removed reference to Dublin Core.
    Most metadata about a book is available in the package file.
-->
```

```
<!ATTLIST meta
   %i18n;
   http-equiv   NMTOKEN          #IMPLIED
   name         NMTOKEN          #IMPLIED
   content      CDATA            #REQUIRED
   scheme       CDATA            #IMPLIED
   >
```

```
<!--Use: style is means to include styling information that
    applies to the book. It may appear only in <head>.
```

```
    -->

<!ELEMENT style (%stylesheetcontent;) >

<!ATTLIST style
   %i18n;
   type          %contenttype;  #REQUIRED
   media         %mediadesc;    #IMPLIED
   title         %text;         #IMPLIED
   >

     <!--Use: script contains a script, a program that may accompany
         a document or be embedded directly in it. The program executes on the
         client's machine when the document loads, or at some other time such
         as when a link is activated. See <noscript> for an alternative
         in case the <script> cannot be executed.
     -->

<!ELEMENT script (%scriptcm;) >
<!ATTLIST script
   charset       %charset;      #IMPLIED
   type          %contenttype;  #REQUIRED
   language      CDATA          #IMPLIED
   src           %uri;          #IMPLIED
   defer         (defer)        #IMPLIED
   event         CDATA          #IMPLIED
   for           %uri;          #IMPLIED
   >

     <!--Use: noscript identifies an alternate method for carrying out
         a function when a playback device cannot execute a <script>. See
         <script>.
     -->

<!ELEMENT noscript (%block; | %inlineinblock;)+ >
<!ATTLIST noscript
   %attrs;
   >
```

[Contents](#)

Appendix 6 - Accessibility Issues

(This section is informative)

Digital Talking Book files, streams, transformation processes and players have been designed to present their content to people with a wide range of abilities and disabilities. They are designed to allow presentation in forms other than conventional print. To the greatest extent possible, files, streams, transformation processes, and players should make information available in as many presentation modes as practical, including human-narrated audio, Braille, synthesized speech and, for players with visual display, large print with user-specifyable size and text re-wrapping, as well as text and audio synchronization and other enhancements for persons with learning disabilities. The controls of players should be easily used by people with a wide range of manual dexterity. Further, tools for producing DTBs should be designed from the outset to be usable by blind and visually impaired persons.

In addition to the provisions of this standard, valuable supplemental information is available from the guidelines produced by the Worldwide Web Consortium's Web Accessibility Initiative. At this time, these documents include:

- Web Content Accessibility Guidelines [WCAG],
- Authoring Tool Accessibility Guidelines[ATAG], and
- User Agent Accessibility Guidelines[UAAG].

(This section is normative)

It is not expected that all modes of presentation will be available in all players and documents, but it is strongly recommended that multiple equivalent presentations be made available to users whenever possible. Historically, products marketed to specific user groups with disabilities have sometimes proven unusable. Not all players need to be accessible to all target groups, but any device compliant with this standard must be accessible to the target group for which is it advertised.

Contents

Appendix 7 - Theory Behind the DTBook3 DTD

(This section is informative)

Overview

This standard defines an XML element set (DTBook3) to represent the content and structure of books and other publications presented in digital talking book (DTB) format. This element set borrows heavily from the W3C's HTML 4.0 Specification, and adds specific structural tags required to accurately and unambiguously represent the content. Because XML is used, the tagged text files of DTBs can be validated for conformance with the document type definition (dtbook3.dtd) that defines the DTB element set. The use of HTML core tags in the DTD also has several benefits, including the ability to take existing HTML-based content and add DTB-conformant structure elements, and to provide content that is easily rendered on visual or synthetic speech-based devices.

The DTBook3 DTD is unique in that it provides several classes of critical information within the structure of XML. The content is provided within semantically rich elements. The semantics are known to the system as a whole and provide essential functionality required by persons who are blind and print disabled. The semantics of the DTD as a whole and the individual elements include:

- hierarchical global navigation targets
- sequential reading order
- reading choices (e.g., to read or skip all footnotes)
- book component identification
- local reading methods tailored to type of book component (e.g., tables are read differently from paragraphs)

Inherent in the DTD are the concepts of navigation, reading order, and reading options. While the DTD provides traditional semantics about the various block and inline elements, it is the pervasive notion of overall reading functionality that sets the DTD apart from other approaches. The theory behind the DTD will be discussed in this appendix.

Theory

The defining features of the DTBook3 DTD are based on the global and local navigation requirements of the end-user. These requirements were gathered directly from users and are laid out in the Document Navigation Features List [Navigation Features].

Global Navigation

Global navigation is efficient movement by a user to a portion of a book the reader wishes to read (e.g., section 8.3 or Appendix 7), with that movement enabled by the Navigation Control File (NCX) . Two categories of destinations are found in the elements of the DTD. The first type is the hierarchical arrangement of headings, e.g., nested chapters, sections, sub-sections, etc., down to six levels denoted by the *level1...level6* tags or by the unlimited recursive use of the *level* element. The levels are containers only and ensure the strict hierarchical arrangement of their contents and headings. The headings themselves are the targets for navigation, because the container elements have no meaning to the reader. Global navigation can make it easy for the reader to move to a heading by expanding or collapsing the hierarchical view of the document. Efficient global navigation is accomplished through the NCX described elsewhere in this standard, but it is the strict hierarchy of headings within corresponding "level" containers that provides the ability to automatically create the hierarchical view in the NCX.

The heading element names themselves (*h1...h6*) were derived from the familiar HTML 4.0 DTD but their use is more tightly controlled in DTBook3. Where HTML allows a document to apply the heading tags in any order, DTBook3 requires that an *h1* be used only within *level1*, *h2* only immediately within *level2*, etc. The containers that wrap the headings are unique to this DTD and are not found in HTML. Use of the recursive *level* container that does not have a numeric identifier is perhaps more elegant, but requires a view of the document's tree structure during authoring or editing. Either approach can be used, but using both within a single document is discouraged.

The second type of global navigation destination provided in the DTD is comprised of text elements that a reader may want to skip on a first reading, such as footnotes. This is easily done by a sighted person reading a printed book; the reader simply chooses not to move to the bottom of the page to read the footnote. However, the reader might choose to later revisit a book and read the footnotes of interest. This functionality can be duplicated in a DTB. Anything that the user can turn off (or merely be notified of) in an automatic rendering can later be directly accessible by the reader. This principle therefore requires that certain types of elements be identified for automatic extraction into the NCX.

We can identify certain types of text elements where this principle can be applied. There are page numbers, which most people do not read, but certainly use for navigation. Footnotes, endnotes at the end of a chapter or rear notes found at the end of a book all share the same function and are only differentiated by their physical location in the printed book. Annotations share the same functionality, but only differ in that their position may be muchcloser (in the margin) to the item referencing the annotation.

Other items found in books are the interesting, but not essential, asides that occur throughout many types of books. Normally presented as sidebars these are items that the reader may choose to read on the first pass through a book or at a later time. While discriminating between sidebars and the main text is easy in a print book, it can be more difficult in an audio version, where these asides can be distracting to the reader. Print-disabled readers have therefore asked for the capability to disable the automatic presentation of sidebars.

In the version of books produced specifically for persons who are blind and print-disabled, producers add descriptions of visual elements such as photographs, illustrations, graphs, and charts. While this is extremely important information, it can be very time-consuming to read. Readers have requested the ability to selectively turn off the automatic rendering of these "producer notes" and return to them later to hear the detailed descriptions. All of the above elements are defined in DTBook3.

Specifically, the elements defined in the DTD for the functionality described above are:

pagenum

>this identifies page numbers.

sidebar

>The sidebars or asides in books are normally optional reading and can be identified as such so the reader can disable the automatic rendering of these items. It is important to note that another element, *notice*, identifies text elements also normally found in margins but essential to the reader's safety or understanding. Warnings, cautions, and other essential information found in floating boxes are tagged as *notice*, rather than *sidebar*, so that players can discriminate between the two types and prevent readers from disabling critical information. *Sidebar*s should be referenced in the text so that the reader is aware of their presence if they are disabled.

prodnote

> Producer notes are information added by the producing organizations that makes versions of printed books accessible to the blind and print-disabled. The *prodnote* element is provided with an attribute to distinguish if this is a producer's note which provides essential or supplemental information. The nonessential information can be turned off by the reader and accessed later. The reader should be notified of disabled *prodnote*s.

noteref and note

> Footnotes, endnotes, rear notes, and annotations fall into this category of paired items. The *noteref* is the reference in the text to further information found elsewhere. The *note* is the target content of the reference. For global navigation purposes, the *noteref* is included in the NCX and is presented to the reader in context (i.e., with adjacent text; implementation is player-dependent) to aid the reader in deciding whether or not to read the note itself.

Local Navigation

Local navigation (one could also call this function simply "reading") comprises movement within a single text element such as a list or table, or within a narrow range of text elements such as a group of words, sentences or paragraphs. Reading is an interaction between the reader and the content which involves the reader absorbing letters, words, sentences, paragraphs, list items, table cells, etc. while controlling the rate and making instant adjustments to reread a word or sentence, or choosing to jump to the next paragraph, capture the paragraph's topic and move on. The reading system for a DTB can not only enable this functionality, but make the entire process completely natural and fluid.

The global navigation described earlier is enabled through the functionality of the NCX. Local navigation/reading is made possible by the XML encoding of the text. Whether a DTB contains a full multimedia presentation through SMIL or only a text file, the XML-encoded text provides the local reading control.This is accomplished in a SMIL DTB by matching the *id* of each text element with the *id* of the parallel construct (*par*) in the SMIL multimedia presentation. While the SMIL is controlling the output of the simultaneous presentation of audio, text, and images (others are possible), decisions by the reader can change the position of the SMIL presentation down to the level of granularity the presentation allows. In other words, if the reader chooses to move the current focus to the next element at the same level, the SMIL presentation transitions to the *par* whose *id* matches that of the target element. If the current element is a block level element such as a paragraph, and the reader moves to the next paragraph, the *id* in the tag of the target paragraph is selected and passed to the SMIL player for synchronization.

If the DTB incorporates sentence- or word-level synchronization and the reader elects to move to the next element, then the system moves to the next inline element and passes its *id* to the SMIL player. The *id* in the XML-encoded text is the controlling mechanism for local navigation through the SMIL presentation.

In a DTB that consists only of an XML-tagged text file, local navigation does not utilize the *ids* of the elements, but relies principally on the hierarchical relationship among the elements described above. In addition, the unique *id* attribute values on targets for *noteref* cross-references, and the page numbers that may occur in the table of contents or index, provide navigation comparable to that in a print book.

The DTBook3 DTD contains all of the elements necessary for complete user control of reading. The block level elements are mostly borrowed from HTML 4.0. For inline applications, the common elements needed for a rich reading experience were assembled. Some elements such as *word* were added to the DTD to provide finely-grained synchronization with the SMIL presentation. Similarly, the element *sent* was introduced for sentence level synchronization. Word-level synchronization will normally be accomplished through software that examines the text file and recognizes the corresponding words in the audio stream. The *ids* and the positions in audio space will be set by the software.

Further details on navigation in tables, lists, etc. can be found in the Document Navigation Features List [Navigation Features].

Transition Between Global and Local Navigation

The combination of global and local navigation allows the reader to move efficiently through a digital talking book, adjusting the granularity of individual jumps appropriately. These two functions are defined separately because their functionality is accomplished in distinctly different ways -- global navigation through the NCX and the local through the structure of the XML source document. Both navigation capabilities exist simultaneously. One can always invoke global navigation and move to a different destination in the NCX; and because the destination is itself a single component of the text being read, local reading control or navigation is also present once one has arrived at the target. For example while reading mid-paragraph, the reader can elect to go to the next heading (global navigation). On the other hand, the reader may elect to move to the next paragraph or list item (local navigation). Both local and global navigation are concurrently possible in a DTB. So the NCX is ""omnipresent". This is a very powerful concept and it also distinguishes a reading system that is simply a SMIL player using the NCX, from a comprehensive reading system that complies strictly with this standard.

[Contents]

Appendix 8 - Distribution Information DTD

(This section is normative)

```
<!-- distinfo DTD
file: distinfo.dtd
Draft - 19 January 2001
        - Added <changemsg> to support customized "please insert disk
..." messages

Draft - 21 December 2000
        - Changed mediaseq attribute names
        - Changed itemref to smilref
        - Changed mapping system to use SMIL file names instead of
manifest idrefs

First draft - 10 August 2000
James Pritchett

An XML application to describe the contents of a single piece of DTB
distribution media.  It consists of a list of books to be found on the
media.  For each book, distinfo identifies the location of each book
within the media filesystem.  If the book is being distributed on multiple
pieces of media, the distinfo book element also includes:
1) the sequence id of this piece of media
2) a distribution map for the book, telling where to find all the
SMIL files for a book

-->
<!-- * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * -->

<!-- distinfo:  Document element, consists of one or more books.
-->
<!ELEMENT distinfo (book+)>
```

```
<!-- book:  a DTB that is present, in part or whole, on this piece of
distribution media.  The uid and href attributes are required.  "uid"
matches the package unique-identifier.  "href" is a URI that locates the
book's package file on this piece of media.

If this is a book fragment, then the "media" attribute identifies which
fragment is stored on this piece of media, and a single distmap element
is present to describe which SMIL files are present on which pieces of media.
The media attribute is in the format "x:y", where x is the sequence
number of this piece of media, and y is the total number of media pieces
in the distribution of this book.

In the case of a book fragment, <book> should contain exactly
one <distmap> and optionally 1
or more <changemsg> elements.
-->
<!ELEMENT book (distmap?, changemsg*)>
<!ATTLIST book
        uid     CDATA   #REQUIRED
        href    CDATA   #REQUIRED
        media   CDATA   #IMPLIED
>


<!-- distmap:  a map identifying which media the various SMIL files
reside upon.  This consists of one or more smilref elements.  The
distmap smilref's should match one-to-one those of the book package spine.
-->
<!ELEMENT distmap (smilref+)>

<!-- smilref:  a reference to a DTB SMIL file.  These are referenced
by file name. The mediaref attribute of each smilref identifies the piece of
media that the file resides upon, and is in the format "x:y" (see above).
-->
<!ELEMENT smilref EMPTY>
<!ATTLIST smilref
        file            CDATA   #REQUIRED
        mediaref        CDATA   #REQUIRED
>


<!-- changemsg:  A pointer to a custom message to be read when a new disk is
requested by the reading system.  "href" is a URI pointing to a SMIL file or
file fragment that represents the message.  "mediaref" identifies the disk for
which this message is intended and matches the "media" attribute of
<book> and
the "mediaref" attribute of <distmap>.
-->
<!ELEMENT changemsg EMPTY>
<!ATTLIST changemsg
        href            CDATA   #REQUIRED
        mediaref        CDATA   #REQUIRED
>
```