NISO
How the information world
CONNECTS

ANSI/NISO Z39.86-2005 (R2012)   **ISSN: 1041-5653**

Specifications for the Digital Talking Book

Abstract: This standard defines the format and content of the electronic file set that comprises a digital talking book (DTB) and establishes a limited set of requirements for DTB playback devices. It uses established and new specifications to delineate the structure of DTBs whose content can range from XML text only, to text with corresponding spoken audio, to audio with little or no text. DTBs are designed to make print material accessible and navigable for blind or otherwise print-disabled persons.

About NISO Standards

NISO standards are developed by the Committees and Working Groups of the National Information Standards Organization. The development process is a strenuous one that includes a rigorous peer review of proposed standards open to each NISO Voting Member and any other interested party. Final approval of the standard involves verification by the American National Standards Institute that its requirements for due process, consensus, and other approval criteria have been met by NISO. Once verified and approved, NISO Standards also become American National Standards.

This standard may be revised or withdrawn at any time. For current information on the status of this standard contact the NISO office or visit the NISO website at:
http://www.niso.org

Contents

Preface to 2005 Edition

(This preface is not a part of ANSI/ Z39.86-2005, *Specifications for the Digital Talking Book*. It is included for information only.)

ANSI/NISO Z39.86 was first released in 2002. Preparation of this version of the standard was prompted by several factors: first, a desire to stay abreast of current practices in related standards and specifications; second, a need to clarify a few ambiguities present in the first release; and third, the wish to add enhancements requested by those using, or planning to use, the standard.

A variety of changes stemming from all of the above reasons were made. Some of the more significant enhancements are listed below:

- Producer-controlled pauses. These allow a DTB author to cause the presentation to pause at specified points and wait for a user action. This function could be used in workbooks when the user needs to, for example, explore a model before continuing.

- A more robust method of ensuring that players can maintain their location when navigation methods are mixed

- A cleaner mechanism for enabling direct access to pages

- New elements for marking up a textual content file

- A more efficient method of identifying individual books when more than one is present on a single piece of media

Individual changes made to the standard are described in comments that can be viewed by reading the HTML version in an HTML authoring tool or with a text editor.

Foreword

(This foreword is not a part of ANSI/ Z39.86-2005 (R2012), *Specifications for the Digital Talking Book.* It is included for information only.)

This standard presents specifications for digital talking books (DTBs) for blind, visually impaired, physically handicapped, learning-disabled, or otherwise print-disabled readers. For many years, "talking books" have been made available to print-disabled readers on analog media such as phonograph records and audiocassettes. These media serve 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 leave much to be desired when reading reference works, textbooks, magazines, and other materials that are often accessed randomly. In contrast, digital media offer readers the ability to move around in 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 by placing bookmarks or highlighting material. 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 do not have access to the spelling of the words they hear. 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 them difficult for interested readers to access. A DTB allows the user to easily skip over or read footnotes. 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.

The DTB goes far beyond the limits imposed on analog audio books because it can include not just the audio rendition of the work, but the full textual content and images as well. Because the textual content file is synchronized with the audio file, a DTB offers multiple sensory inputs to readers, a great benefit to, for example, learning-disabled readers. 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 device connected to their DTB player and accessing the textual content file. DTBs containing a textual content file but no audio material might be accessed via synthetic speech, screen-enlarging software, or a Braille device.

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 will normally be in the context of an intellectual property protection system.

Reaffirmation

At the time this standard was up for its periodic review, it was determined that a major revision was needed to update the standard for a modular, extensible XML format. Following a draft for trial use of the revised standard, feedback was received that the scope had been broadened so substantially and the revisions were so significant that implementers needed a transition period of several years. As a result, it was decided that the revision would receive a new standard number and title (ANS/NISO Z39.98-2012, *Authoring and Interchange Framework for Adaptive XML Publishing Specification*) and the existing Z39.86-2005 standard would be reaffirmed to remain active during the transition period.

The NISO consensus body approved the reaffirmation on March 18, 2012 and it was approved by ANSI on April 16, 2012.

Suggestions for improving this standard are welcome. They should be sent to the National Information Standards Organization, 3600 Clipper Mill Road, Suite 302, Baltimore, MD 21211, telephone (301) 654-2512.

NISO Voting Members

This Standard (2005 edition) was processed and approved for submittal to ANSI by the National Information Standards Organization. NISO approval of this Standard does not necessarily imply that all Voting Members voted for its approval. At the time it approved this Standard, NISO had the following members:

3M
Susan Boettcher
Roger D. Larson, Alt

American Association of Law Libraries
Robert L. Oakley
Mary Alice Baish, Alt

American Chemical Society
Matthew Toussant

American Library Association
Betty Landesman

American Society for Information Science and Technology (ASIS&T)
Gail Thornburg

American Society of Indexers
Judith Gibbs

American Theological Library Association
Myron Chace

ARMA International
Diane Carlisle

Armed Forces Medical Library
Diane Zehnpfennig
Emily Court, Alt

Art Libraries Society of North America (ARLIS/NA)
Mark Bresnan

Association for Information and Image Management (AIIM)
Betsy A. Fanning

Association of Information and Dissemination Centers (ASIDIC)
Margie Hlava

Association of Jewish Libraries
Caroline R. Miller
Elizabeth Vernon, Alt

Association of Research Libraries
Duane E. Webster
Julia Blixrud, Alt

Auto-Graphics, Inc.
Paul Cope

Barnes & Noble, Inc.
Douglas Cheney

Book Industry Communication
Brian Green

California Digital Library
Daniel Greenstein
John Kunze, Alt

Cambridge Information Group
Michael Cairns
Matthew Dunie, Alt

The Cherry Hill Company
Cary Gordon

College Center for Library Automation (CCLA)
Richard Madaus
Ann Armbrister, Alt

Colorado State Library
Brenda Bailey-Hainer
Steve Wrede, Alt

CrossRef
Edward Pentz
Amy Brand, Alt

Davandy, L.L.C.
Michael J. Mellinger

DYNIX Corporation
Ed Riding
Gail Wanner, Alt

EBSCO Information Services
Gary Coker
Oliver Pesch, Alt

Elsevier
Paul Mostert

Endeavor Information Systems, Inc.
Sara Randall
Shelley Hostetler, Alt

Entopia, Inc.
Igor Perisic

Ex Libris, Inc
James Steenbergen

Fretwell-Downing Informatics
Robin Murray

Gale Group
Katherine Gruber
Justine Carson, Alt

Geac Library Solutions
Eric Conderaerts
Eloise Sullivan, Alt

GIS Information Systems, Inc.
Candy Zemon
Paul Huf, Alt

H.W. Wilson Company
Ann Case
Patricia Kuhr, Alt

ANSI/NISO Z39.86-2005 Review Voting Pool

NISO Board of Directors

Advisory Committee Members

Acknowledgements

Specifications for the Digital Talking Book

1   General Information

1.1   Purpose and Scope of Standard

(This section is informative.)

This standard establishes specifications for digital talking books (DTBs) for blind, visually impaired, physically handicapped, learning-disabled, 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 primarily for DTB files and their interrelationships. It also includes specifications for DTB playback devices in two areas: player performance related to file requirements and player behavior in areas defined in user requirements.

1.2   Definitions

(This section is normative.)

The following abbreviations, acronyms, phrases, 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
Fully 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.

DTBook
An XML element set (dtbook.dtd) that defines the markup for the textual content of a DTB.

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

FIXED
When used in definitions of XML element attributes, means that the attribute has a single, fixed value specified in the DTD. See IMPLIED and REQUIRED.

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
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.

IMPLIED
When used in definitions of XML element attributes, means that the attribute is optional and that no default value is supplied. See FIXED and REQUIRED.

Informative
Supplying background or explanation. 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 can be controlled by the textual content file or the SMIL file(s); the granularity is limited by the degree to which the textual content file has been marked up or the level to which synchronization has been applied in the SMIL file(s). Time-based movement through a document (e.g., fast-forward and rewind as on an analog cassette, or time jumps by specified intervals) may also be implemented.

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

May
In normative sections, the word *may* means that a course of action is optional.

Media Unit
A single object on which a DTB is stored for distribution to the reader. For example, a single CD-ROM disk.

Must
In normative sections, 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*.

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.

Normative
Setting forth requirements that must be met to establish conformance with this standard; or providing recommendations or optional courses of action. For recommended or optional features, conformance is not dependent on the fact of implementation, but, if implemented, that implementation is as prescribed in this standard. Contrast with *Informative*. Notes within a normative section may be informative.

OEBF
The Open eBook Forum™ [OEBF] is an organization formed to create and maintain standards and promote the successful adoption of electronic books. The *Open eBook Publication Structure Version 1.2* provides a specification for representing the content of a book when it is converted from print to electronic form. This DTB standard uses a subset (the Package File) of that specification.

OPF
Open eBook Forum Package File. See *Package File*.

2

Package File
The *Open eBook Forum Package File* (OPF) is an XML file conforming to the oebpkg12.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.

Playback System
The hardware/software platform that renders the contents of a DTB to a reader. Synonymous with *Player*.

Player
See *Playback System*.

Reader
The person reading the digital talking book. Synonymous with *User*.

REQUIRED
When used in definitions of XML element attributes, means that the attribute must always be provided. See FIXED and IMPLIED.

Shall
See *Must*

Should
In normative sections, the word *should* means that a course of action is recommended but not required.

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

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

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

Textual Content File
The content of the subject document in a character set specified by ISO/IEC 10646 [ISO 10646] to which XML markup valid to the DTBook DTD has been applied.

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

URI
A *Uniform Resource Identifier* is a compact string of characters for identifying resources: documents, images, audio files, etc. Within a DTB, URIs are most likely to appear as attribute values for various XML elements, used as a way of identifying other documents or files either in whole or part. For the purposes of this specification, URIs must adhere to the syntax defined in RFC 2396 [RFC 2396]. A URI may include a fragment identifier suffix beginning with "#" that matches some named anchor in the target document. See *Fragment Identifier*.

User
See *Reader*.

XML
The Extensible Markup Language [XML] is a standardized language for marking up files containing structured information.

XSL
The Extensible Stylesheet Language is a series of recommendations by the Worldwide Web Consortium that describes how XML documents can be transformed and rearranged [XSLT], then formatted [XSL] for screen, handheld device, paper, or audio presentation.

XSLT
A language for transforming XML documents into other XML documents. [XSLT] is designed for use as part of XSL. See *XSL*.

1.3   Strategy

(This section is informative.)

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.

1.4   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, due to the inaccessibility of printed documents to these users. It is in the best interest of users that, to the greatest extent possible, files, streams, transformation processes, and players make information available in as many presentation modes as practical, including human-narrated audio, Braille, synthesized speech, large print with user-specifiable size and text re-wrapping for players with visual display, and text and audio synchronization and other enhancements for persons with learning disabilities. Users will also be greatly benefited if controls on players are readily usable by people with a wide range of manual dexterity.

During the development of this standard, an advisory document, DTB Playback Device Features List, was created. Although it is not a normative part of this standard, player developers will find useful accessibility concepts embodied in it.

In addition to the provisions of this standard, valuable supplemental information is available from the guidelines and techniques produced by the Worldwide Web Consortium's Web Accessibility Initiative. At this time, these documents include:

Web Content Accessibility Guidelines,

Authoring Tool Accessibility Guidelines, and

User Agent Accessibility Guidelines.

(This section is normative.)

Not 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 it is

advertised. It is also strongly recommended that DTB production tools and processes be made accessible to persons with disabilities.

1.5   Relationship to Other Specifications

(This section is normative.)

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.

1.5.1   Relationship to Unicode

(This section is normative.)

Playback systems must support at least UTF-8 and UTF-16 encodings. See section 2.2 of the XML specification [XML].

1.6   Patent Rights

(This section is informative.)

Implementation of this standard may involve 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 as they hold on reasonable and nondiscriminatory 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 they use that is defined by the relevant standards and specifications referenced by this standard. There may be applicable patents of which this standards committee is unaware; it is the responsibility of the implementer to ensure that the implementation is non-infringing.

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

1.7   Maintenance Agency

(This section is informative.)

The maintenance agency designated in Appendix 2 will be responsible for reviewing and acting upon suggestions for modifications to this standard. Questions concerning the implementation of this standard and requests for information should be sent to the maintenance agency.

A list of errata, proposed changes, and maintenance activities related to this standard will be maintained at http://www.daisy.org/z3986/2005/errata.html.

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.). Such features enable readers with visual and physical disabilities to access the information in DTBs flexibly and efficiently, and allow sighted users with learning or reading disabilities to receive the information through multiple senses. For a full discussion of these capabilities, see the "Document Navigation Features List" [Navigation Features], the user requirements document on which this standard was based. A document written during the development of this standard, Theory Behind the DTBook DTD [DTBook Theory], 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 be produced with a variety of capabilities. 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 Playback Device Features List [Player Features] mentioned above presents the committee's priorities for a range of functions across three types of playback devices.

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

Package File
The Package File, drawn from the Open eBook Publication Structure 1.2, contains administrative information about the DTB and the files that comprise it. A valid XML version 1.0 file, it contains a set of metadata describing the DTB, a list (the *manifest*) of the files that make up the DTB, and a *spine* that defines the default reading order of the document. See section 3, "Package File."

Textual Content File
A DTB can contain part or all of the text of the document as an XML 1.0 file marked up in accordance with the document type definition (DTD) defined for this standard, dtbook.dtd. (See Appendix 1.) The textual content file enables properly-configured playback devices to spell words on demand, carry out keyword searches, and permit finely-grained navigation. It can 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 can 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 formats specified by this standard.

Image Files
In addition to text and audio, DTBs can include images that can be presented on players with visual displays. Section 6, "Image 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 use of 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 use the synchronization information to both access points in the audio presentation and to track, during audio playback, the corresponding position in the textual content file. This standard uses a subset of the full SMIL 2.0 specification. See section 7, "Synchronization of Media Files," for discussion of these issues, and Appendix 1 for the DTD information.

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, figures, or notes -- is effected through the Navigation Control file for XML applications (NCX). The NCX presents 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 information about the DTD for the NCX. Local, more finely-grained, navigation is not handled by the NCX but is enabled through the textual content file or SMIL file(s) or through time-based movement through the audio presentation, depending on the document and on the player. See section 8, "Navigation Control File (NCX)," and Appendix 1, "NCX DTD" for specifications related to the NCX.

Bookmark/Highlight File

This standard supports user-set, exportable bookmarks and highlights to which text and audio notes can be applied. Specifications for the XML 1.0 file for portable bookmarks and highlights are presented in section 9, "Portable Bookmarks and Highlights" and Appendix 1, "DTD for Portable Bookmarks/Highlights."

Resource File

The resource file contains or references 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. See section 10, "Resource File," and Appendix 1, "DTD for Resource File" for file specifications.

Distribution Information File

Given the great size of audio files even when heavily compressed, it will be common for large books to span several media units. Section 11, "Packaging Files for Distribution," describes how the "distInfo" file maps the location of each SMIL file to a specific media unit, e.g., disk 1 of 3. It also explains how, when several books are distributed on the same media unit, the distInfo file stores information about each book for presentation to the reader. Appendix 1, "Distribution Information DTD," provides information on the document type definition for "distInfo" files.

Presentation Styles

Section 12, "Presentation Styles," discusses how the presentation of a DTB in various media can be controlled through the use of optional style sheets.

3   The DTB Package File

(This section is informative.)

The Package File, drawn from the Open eBook Forum™ (OEBF) Publication Structure 1.2, contains administrative information about the DTB, the files that comprise it, and how these files interrelate. 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. See section 2 of the full OEBF Publication Structure 1.2 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 1.2
Package//EN"
    "oebpkg12.dtd">

<package xmlns="http://openebook.org/namespaces/oeb-package/1.0/"
    unique-identifier="foo">
        <metadata>...</metadata>
        <manifest>...</manifest>
        <spine>...</spine>
        <tours>...</tours>
        <guide>...</guide>
</package>
```

(This section is normative.)

A DTB conforming to this standard must include exactly one Package File which must be a valid XML 1.0 document conforming to the OEBF Publication Structure 1.2 package DTD (oebpkg12.dtd) and its associated entity reference (oeb12.ent). The full specification, DTD, and entity reference for the OEBF package file are available for download from the OEBF site [OEBF]. The Package File must be named with the extension ".opf". If a DTB spans multiple media units, the identical Package File must be present on each media unit.

A Package File conforming to this standard must comply with all aspects of section 2 of the OEBF Publication Structure 1.2, with the following two exceptions:

1) Section 2.3.1 does not apply. Specifically, there is no requirement on DTB authors or playback devices to implement the fallback mechanism for files that are not of the OEBF core MIME media types.

2) Section 2.4 of the Publication Structure states that the spine element may refer only to item elements of media type *text/x-oeb1-document*. In DTB applications, the spine must only reference items of media type *application/smil*.

Namespace (xmlns) attributes and their values, although declared as #FIXED, must be explicitly specified in the document instance. Entity declarations must occur in the internal DTD subset. See further section 16.1 "General File Conformance Requirements."

3.1   Package Identity

(This section is normative.)

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 primary identifier. The value of unique-identifier must match the id attribute of one and only one dc:Identifier element, which is a descendant of the package element.

The primary identifier of the DTB must be globally unique.

(This example is informative.)

Example 3.1:

```
<?xml version="1.0"?>
<!DOCTYPE package PUBLIC "+//ISBN 0-9673008-1-9//DTD OEB 1.2 Package//EN"
        "oebpkg12.dtd">
<package xmlns="http://openebook.org/namespaces/oeb-package/1.0/" unique-
identifier="uid">
    <metadata>
        <dc-metadata...>
            <dc:Identifier id="uid" scheme="DTB">uk-rnib-db02006
            </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 must contain supplemental metadata in an `x-metadata` element. The `x-metadata` element must contain at least one instance of the `meta` element, which uses `name` and `content` attributes to define its value. (See section 3.2.3, "X-Metadata.")

3.2.1 Dublin Core Metadata

(This section is normative.)

The use of Dublin Core metadata within a compliant DTB must conform to the following description from the OEBF Publication Structure 1.2:

The `dc-metadata` element contains specific publication-level metadata as defined by the Dublin Core initiative (http://purl.org/dc/). The descriptions below are included for convenience, and the Dublin Core's own definitions take precedence (see http://www.ietf.org/rfc/rfc2413.txt).

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 `dc-metadata` in an OEB package is required to have an attribute of

```
    xmlns:dc=http://purl.org/dc/elements/1.1/
```
and

```
    xmlns:oebpackage="http://openebook.org/namespaces/oeb-package/1.0/".
```

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. Note that all Dublin Core element types may be repeated (occur more than once) within `dc-metadata`.

- **dc:Title**

  o Content: The title of the DTB, including any subtitles.

  o Occurrence: Required

- **dc:Creator**

  o Content: Names of primary author or creator of the intellectual content of the publication.

  o Occurrence: Optional (not all documents have known creators) - recommended.

  o 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**

  o Content: The topic of the content of the publication.

  o Occurrence: Optional - recommended.

- **dc:Description**

  o Content: Plain text describing the publication's content.

  o Occurrence: Optional

- **dc:Publisher**

  o Content: The agency responsible for making the DTB available. (Compare `dtb:sourcePublisher` and `dtb:producer`.)

  o Occurrence: Required

- **dc:Contributor**

  o Content: A party whose contribution to the publication is secondary to those named in `dc:Creator`.

  o Occurrence: Optional

  o 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**

  o Content: Date of publication of the DTB. (Compare `dtb:sourceDate` and `dtb:producedDate`.) In format from [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.

  o Occurrence: Required

- o 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.

- **dc:Type**

    - o Content: The nature of the content of the DTB (i.e., sound, text, image). Best practice is to draw from the Dublin Core's enumerated list [DC-Type].

    - o Occurrence: Optional

- **dc:Format**

    - o Content: The standard or specification to which the DTB was produced. Values of `dc:Format` in a DTB conforming to this standard are valid only if they read "ANSI/NISO Z39.86-2005".

    - o Occurrence: Required

- **dc:Identifier**

    - o Content: A string or number identifying the DTB. One instance of this element, that which is referenced from the `package unique-identifier` attribute, must include an id.

    - o Occurrence: Required

    - o Added attributes:
        - ▪ scheme – (optional) The name of the system or authority that generated or assigned the identifier. For example, "DOI", "ISBN", or "DTB".

- **dc:Source**

    - o Content: A reference to a resource (e.g., a print original, ebook, etc.) from which the DTB is derived. Best practice is to use the ISBN when available.

    - o Occurrence: Optional - recommended.

- **dc:Language**

    - o Content: Language of the content of the publication. An [RFC 3066] language code. For Sweden: "sv" or "sv-SE"; for UK: "en" or "en-GB"; for US: "en" or "en-US"; etc.

    - o Occurrence: Required

- **dc:Relation**

    - o Content: A reference to a related resource.

    - o Occurrence: Optional

- **dc:Coverage**

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

    - o Occurrence: Optional

- **dc:Rights**

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

    - o Occurrence: Optional

3.2.2   DTB ID Scheme

(This section is informative.)

Various schemes are available for identifying digital publications. In the DTB domain, the requirements for an identifier are simply to identify the publication in a manner that is highly likely to be globally unique. A major purpose of the uniqueness requirement is to prevent filename collisions among bookmark files.

To meet this base requirement, a simple DTB id scheme might be used. A DTB identifier under this scheme consists of a hyphen-separated string consisting of a two-letter country code drawn from [ISO 3166], an agency code unique within its country, and an identifier unique within the agency. For example, us-afb-x12345.

This scheme will provide a simple solution to the uniqueness requirement that will serve DTB-publishers' needs in the short term. In the longer term, as the requirements of a global library of alternative format materials become more important, other more sophisticated mechanisms will doubtless be employed.

3.2.3   X-Metadata

(This section is normative.)

The following names were developed for the DTB application to supply information that the Dublin Core element set does not cover. These names may only appear within the `x-metadata` containing element, as values of the `name` attribute on the `meta` element. Each `x-metadata` name below is shown as either "Repeatable" (it may be used more than once) or "Not repeatable". Content producers may introduce other metadata within `x-metadata` besides those listed below, if needed. However, metadata names shall not begin with the prefix "dtb:" unless defined in this standard. Players must not fail when encountering unknown metadata but must, at a minimum, ignore it.

- **dtb:sourceDate**

  o   Content: Date of publication of the resource (e.g., a print original, ebook, etc.) from which the DTB is derived. In format from [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.

  o   Occurrence: Optional - recommended. Not repeatable.

- **dtb:sourceEdition**

  o   Content: A string describing the edition of the resource (e.g., a print original, ebook, etc.) from which the DTB is derived.

  o   Occurrence: Optional - recommended. Not repeatable.

- **dtb:sourcePublisher**

  o   Content: The agency responsible for making available the resource (e.g., a print original, ebook, etc.) from which the DTB is derived. (Compare `dc:Publisher`.)

  o   Occurrence: Optional - recommended. Not repeatable.

- **dtb:sourceRights**

  o   Content: Information about rights held in and over the resource (e.g., a print original, ebook, etc.) from which the DTB is derived. (Compare `dc:Rights`.)

  o   Occurrence: Optional - recommended. Not repeatable.

- **dtb:sourceTitle**

  o Content: The title of the resource (e.g., a print original, ebook, etc.) from which the DTB is derived. To be used only if different from `dc:Title`.

  o Occurrence: Optional. Not repeatable.

- **dtb:multimediaType**

  o Content: One of the six types of DTB defined in the Structure Guidelines [StructGuide]. Values are: audioOnly, audioNCX, audioPartText, audioFullText, textPartAudio, textNCX.

  o Occurrence: Required. Not repeatable.

- **dtb:multimediaContent**

  o Content: Summary of the general types of media used in the content of this DTB. Value is a comma-delimited list of the top-level media types defined in [RFC2046]. In the current version of this standard, the only applicable types will be `audio`, `text`, and `image`. Media types that are referenced only by NCX, Resource File, or distInfo must not be listed here.

  o Occurrence: Required. Not repeatable.

- **dtb:narrator**

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

  o Occurrence: Optional - recommended. Repeatable.

- **dtb:producer**

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

  o Occurrence: Optional. Repeatable.

- **dtb:producedDate**

  o Content: Date of first generation of the complete DTB, i.e. Production completion date. (Compare `dc:Date`.) In format from [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.

  o Occurrence: Optional. Not repeatable.

- **dtb:revision**

  o Content: Non-negative integer value of the specific version of the DTB. Incremented each time the DTB is revised.

  o Occurrence: Optional. Not repeatable.

- **dtb:revisionDate**

  o Content: Date associated with the specific `dtb:revision`. In format from [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.

  o Occurrence: Optional. Not repeatable.

- **dtb:revisionDescription**

  o Content: A string describing the changes introduced in a specific dtb:revision.

  o Occurrence: Optional. Not repeatable.

- **dtb:totalTime**

  o Content: Total playing time of all SMIL files comprising the content of the DTB. Value is a Clock Value from SMIL 2.0 Timing and Synchronization Module [SMIL]. See section 7.7, "Media Clipping and Clock Values."

  o Occurrence: Required. Not repeatable.

- **dtb:audioFormat**

  o Content: A string describing the format in which the audio files in the DTB file set are written. If more than one audio format is used, this element may be repeated.

  o Occurrence: Optional, recommended for audio DTBs. Repeatable.

  o Formats specified in section 5 are shown below, followed by the normative value for dtb:audioFormat:

  o MPEG-4 AAC: MP4-AAC

  o MPEG-1/2 Layer III: MP3

  o Linear PCM,RIFF WAVE format: WAV

  o (Values are not case-sensitive.)

(This example is informative.)

Example 3.2:

```
<metadata>
    <dc-metadata xmlns:dc="http://purl.org/dc/elements/1.1/"
    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-db00001</dc:Identifier>
        <dc:Identifier
        scheme="DOI">10.1000/DX44998</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>ANSI/NISO Z39.86-2005</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-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:multimediaContent" content="audio" />
        <meta name="dtb:multimediaType" content="audioNCX" />
        <meta name="dtb:totalTime" content="06:22:34.143" />
    </x-metadata>

</metadata>
....
```

3.3   Manifest

(This section is normative.)

The `manifest`, which is a child of the `package` element, must contain a complete list of all of the files (documents, audio files, images, style sheets, etc.) that make up a given DTB, including the package file itself. The `manifest` shall list only files of types permitted by this standard. The `manifest` shall list only files that are part of the DTB. The distInfo file and any associated audio `changeMsgs` (or any other files listed *only* in the distInfo file) are not considered part of the DTB and thus shall not be listed (See section 11, "Packaging Files for Distribution.") The Resource File and any associated media files are considered part of the DTB and thus shall be listed. Each file is referenced by an `item` element. Each `item` must have an `href` attribute that is the URI of the referenced file and is unique within the `manifest`. This URI must not include fragment identifiers; if relative, it is interpreted as relative to the package file itself. Further, any relative URIs contained within an XML file listed in the `manifest` are considered to be relative to the referring file.

In addition, each `item` must have a `media-type` attribute containing the MIME media type of the file, and an `id` attribute. The `id` is used primarily when a `manifest item` is referenced by the `spine`. The `manifest` may also include fallback declarations to allow players to choose among alternative presentation formats. (See OEBF Publication Structure for details.) Support for the fallback mechanism is not required by this standard. The NCX entry in the Package File manifest must have an id value equal to "ncx". The Resource File entry in the Package File manifest must have an id value equal to "resource". The order of `item` elements within the `manifest` is not significant.

All `media-type` attribute values must conform to RFC 2046. In addition, the `media-type` attribute must have the value of the IANA-registered MIME media type for that type of file, if one exists (see RFC 2048 for information on the registration process for MIME media types). If no MIME media type has been registered for the file type, then the MIME media type recommended in the applicable standard for that file type must be used. If there is no standard MIME media type for the file, then an appropriate "x-" MIME media type must be used, following the rules of RFC 2046. In the case of WAV files, this standard mandates the use of the media type value "audio/x-wav". For files defined by this standard that are XML documents, this standard defines mandatory, unique media types following RFC 3023: "application/x-dtbncx+xml" for the NCX, "application/x-dtbresource+xml" for the resource file, and "application/x-dtbook+xml" for textual content files.

In addition, producers must use particular file name extensions for all the different kinds of files mentioned in this standard (see table below). These values are case-sensitive. In cases where the file name extension and MIME media type values do not agree, players should consider the MIME media type to take precedence.

The following table summarizes the required file name extensions and MIME media type values for all the different kinds of files that may appear in the package manifest:

Kind of file	File name extension	MIME media type
MPEG-4 AAC audio	.mp4	audio/mpeg4-generic
MPEG-1/2 Layer III (MP3) audio	.mp3	audio/mpeg
Linear PCM - RIFF WAVE format audio	.wav	audio/x-wav
JPEG image	.jpg	image/jpeg
PNG image	.png	image/png
Scalable Vector Graphics (SVG) image	.svg	image/svg+xml
Cascading Style Sheets (CSS)	.css	text/css
SMIL files	.smil	application/smil
Package file	.opf	text/xml
DTD and DTD fragments (entities or modules)	[no requirement]	application/xml-dtd
Navigation Control File (NCX)	.ncx	application/x-dtbncx+xml
Textual content files (dtbook)	.xml	application/x-dtbook+xml
Resource file	.res	application/x-dtbresource+xml

(This example is informative.)

A sample manifest for a DTB with audio, structure, and text follows:

Example 3.3:

```
....
<manifest>

    <item id="opf" href="rs.opf" media-type="text/xml" />
    <item id="text" href="rs.xml" media-type="text/x-dtbook+xml" />
    <item id="text_style" href="dtbbase.css" media-type="text/css" />
    <item id="ncx" href="rs.ncx" media-type="application/x-dtbncx+xml"
/>
    <item id="ncx_style" href="ncx16.css" media-type="text/css" />
    <item id="SMIL" href="rs.smil" media-type="application/smil" />
    <item id="foreword" href="rs_fwdx.mp3" media-type="audio/mpeg" />
    <item id="standards" href="rs_stdx.mp3" media-type="audio/mpeg" />
    <item id="appendices" href="rs_app.mp3" media-type="audio/mpeg" />
    <item id="index" href="rs_index.mp3" media-type="audio/mpeg" />
    <item id="fig_01" href="fig1.png" media-type="image/png" />
    <item id="resource" href="rs.res" media-type="application/x-
dtbresource+xml" />
```

```
              <item id="resource_audio" href="res.mp3" media-type="audio/mpeg"
         />


         </manifest>
         ....
```

Here is a `manifest` for an audio-only version of the above DTB where separate SMIL files were created for each segment of the book.

Example 3.4:

```
         ....
         <manifest>
              <item id="opf" href="rs.opf" media-type="text/xml" />
              <item id="ncx" href="rs.ncx" media-type="application/x-dtbncx+xml"
         />
              <item id="foreword" href="rs_fwdx.mp3" media-type="audio/mpeg" />
              <item id="standards" href="rs_stdx.mp3" media-type="audio/mpeg" />
              <item id="appendices" href="rs_app.mp3" media-type="audio/mpeg" />
              <item id="index" href="rs_index.mp3" media-type="audio/mpeg" />
              <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>
         ....
```

3.3.1   Allowed Characters in File Names

(This section is normative.)

To assure interoperability when transporting DTBs between various locales and platforms, characters in file names must be restricted to the *alphanum* definition of [RFC2396], and a subset of the RFC2396 *mark* definition. This effectively reduces allowed characters to [A-Za-z0-9._-].

The following definition "filename" identifies in EBNF form which characters are allowed in file names. The definition "foldername" identifies allowed characters in folder names, when folders are used in path specifications of URIs that reference members of the DTB file set.

```
    digit ::=  [#x0030-#x0039]
    lowalpha ::= [#x0061-#x007A]
    upalpha ::=  [#x0041-#x005A]
    alphanum ::= digit|lowalpha|upalpha
    hyphen ::= [#x002D]
    underscore ::= [#x005F]
    period ::= [#x002E]
    filename ::= (alphanum|hyphen|underscore|period)+
    foldername ::= (alphanum|hyphen|underscore)+
```

(This section is informative.)

These restrictions should be regarded as an interim solution. As soon as a global recommendation or standard for internationalized resource identification is established, consideration for adoption will be a high priority.

3.3.2 Case Sensitivity of URIs

(This section is normative.)

All URIs referencing fileset members are case sensitive.

Fileset members must not be given names that result in multiple identical names following case normalization.

3.4 Spine

(This section is normative.)

The `spine`, a child of the `package` element, shall consist of a list of one or more `itemref` elements whose order defines the default linear reading order for the DTB. Each `itemref` must contain an `idref` which points to the `id` of a SMIL file listed in the `manifest`. Only SMIL files can be referenced by `itemref`s in the `spine`. The `itemref`s must be listed in the `spine` in the order in which the SMIL files are to be presented. A player must consult the `spine` when it reaches the end of a SMIL file to determine which file to render next.

(The following examples are informative.)

The first of the following examples shows the `spine` that corresponds to the first of the two `manifest` examples above.

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` element is an optional child of the `package` element. The [OEBF](#) Publication Structure describes `tours` 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 structures 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 the functionality described above, it may partially achieve it by producing customized `navLists` in the NCX.

(This section is normative.)

Compliant players are not required to support `tours`.

3.6   Guide

(This section is informative.)

As specified in the [OEBF](#) Publication Structure, the `guide`, a child of the `package` element, 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` is not expected to be used in DTBs.

(This section is normative.)

Compliant players are not required to support `guide`s.

4   Content Format for Text

4.1   Introduction

(This section is normative.)

This standard defines an [XML](#) 1.0 Document Type Definition -- DTBook -- for markup of the textual content files of books and other publications presented in digital talking book format. To be compliant with this standard, a textual content file of a DTB must be a valid [XML](#) file conforming to dtbook-2005-1.dtd (see [Appendix 1](#), "DTBook DTD"). See Section 3.3, "[Manifest](#)" for filename extension requirements. The `version` and `xmlns` attributes on the `dtbook` element must be explicitly specified in the document instance, using values drawn from the above-named DTD. Entity declarations must occur in the internal DTD subset. See further section 16.1 "[General File Conformance Requirements](#)."

A DTB that includes textual content will, in most cases, contain only one textual content file. However, when necessary (with a very large book, for example), a DTB can contain multiple textual content files, each of which must be valid to the DTBook DTD.

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 of the DTD must employ the mechanisms specified by [XML](#) 1.0. See section 4.2.2, "[Modular Extension of the DTD](#)."

Two metadata items must be present in the <head> of compliant textual content files, contained in a <meta> element: dtb:uid and dc:Title. dtb:uid is the globally unique identifier for the DTB. The value is the same as that of the `dc:Identifier` element referenced by the package file's unique-identifier element. See section 3.1, "Package Identity. dc:Title contains the title of the DTB. Inclusion of the full range of applicable Dublin Core elements is recommended, to make a DTBook document more useful as stand-alone content.

4.2   Using the DTBook Element Set

(This section is informative.)

A document developed during the creation of this standard, Theory Behind the DTBook DTD [DTBook Theory], discusses the rationale underlying the DTBook element set and the benefits it provides to digital talking book applications.

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 [DTBook 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 for applying DTBook markup is available from the DAISY Consortium. These Structure Guidelines [StructGuide] describe the correct application of the DTBook element set, emphasize the importance of capturing the structure of the text content, and provide detailed examples of the use of all DTBook elements.

The DTBook element set has considerable application outside of the digital talking book as well. It was designed to enable the production of documents in a variety of accessible formats. At least one U.S. Braille translation software package has implemented a facility that imports DTBook documents and automatically translates and formats them in Grade 2 Braille. It is expected that similar automated processes will be developed for converting properly marked-up documents into large print and for rendering DTBook documents in Braille, synthetic speech, and large print "on the fly." Finally, an attribute called "showin" is incorporated in the DTBook element set to control the display of selected segments of a DTBook document. For example, descriptions of a graph might vary between Braille and large print editions; "showin" could allow only the appropriate version to show in each edition, although both would be present in the DTBook document.

This standard does not mandate the degree of markup to be applied to a textual content file. However, the richer the markup, the greater the functionality available to the reader.

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

4.2.1   DTBook Markup Related to SMIL

(This section is normative.)

To ensure efficient player operation with DTBs containing textual content files, the `smilref` attribute must be present and non-empty for each element in the textual content file referenced by a SMIL file. The `smilref` value shall normally be the URI of the SMIL time container (`par` or `seq`) containing the media object that references a given element. However, in a text-only DTB consisting of a sequence of text media objects, `smilref` contains the URI of the media object that references the element. The `smilref` attribute permits the DTB player to resume SMIL-based playback following text-based navigation, full-text searches, etc.

4.2.2   Modular Extension of the DTD

(This section is informative.)

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

A DTBook document is an [XML](#) application. Therefore it should begin with the XML declaration identifying the version of XML, and the optional character set encoding. (See [Appendix 1](#), "DTBook DTD" for more information.)

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

This is followed by the document type declaration:

```
<!DOCTYPE dtbook PUBLIC "-//NISO//DTD dtbook 2005-1//EN"
   "http://www.daisy.org/z3986/2005/dtbook-2005-1.dtd">
```

For discussion of other ways of expressing the DOCTYPE, see section 2.3 of the "DTBook DTD" listed in [Appendix 1](#).

A book can invoke other DTDs or modules to augment the DTBook 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 dtbook PUBLIC "-//NISO//DTD dtbook 2005-1//EN"
   "dtbook-2005-1.dtd"
         [
             <!ENTITY % dramaModule SYSTEM "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 this entity, that is "brings it into" the current document, just as the DOCTYPE declaration invoked the base DTD (dtbook-2005-1.dtd). The third line declares the entity "% externalblock" and gives it the value "drama". Since dtbook-2005-1.dtd contains an entity of the same name, and the internal subset overrules the base (external) DTD (dtbook-2005-1) in areas of conflict, everywhere in dtbook-2005-1 where "%externalblock;" appears (that is, wherever block elements are allowed), the value "drama" is added. 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 dtbook-2005-1 (that is, 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 (stagedir) from the drama module.

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

See section 3 of the "DTBook DTD" (see Appendix 1) for a more detailed discussion of this issue.

5   Audio File Formats

5.1   Distribution Formats

(This section is normative.)

A set of audio file formats is listed below. A compliant audio player must be capable of decoding at least one of the formats listed. It is strongly recommended that players be able to decode all listed formats. Content compliant with this standard must be delivered in one of the formats below, or any mixture of them.

It is permissible for parts of a single book to be encoded in different audio formats. For example, a producer may choose to encode a lengthy bibliography at a lower bit rate or with a different codec than the main body of the book. Players must support transitions between differently encoded sections smoothly. There is no restriction on the granularity of these parts, i.e. they may occur at any point in the SMIL presentation.

Support for multi-channel rendering is not required. Stereo signals must be recognized and rendered at least in monaural format.

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.

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

- Linear PCM - RIFF WAVE format [RIFFWAV].

- 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.

See Section 3.3, "Manifest" for filename extension requirements.

While the ISO standards for MP3 and AAC require support for variable bit rate playback, players compliant with this standard are only required to support constant bit rate playback.

Players must support sample rates of 44.1, 22.05, and 11.025 kHz at a depth of 16 bits per sample. Compressed audio must be encoded such that the output sampling rate is restricted to one of the above three rates.

5.2   Formats for Audio Notes

(This section is normative.)

Audio players capable of recording and exporting audio notes for bookmarks and highlights must support encoding in the following format or one of the formats specified in section 5.1. 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 or 16 kbps. Encoder and decoder are simple to implement. File extension shall be: .726

6   Image File Formats

(This section is normative.)

Images included in DTBs must be presented in one or more of the following formats: JPEG (JFIF V 1.02) [JPEG], PNG [RFC 2083], or Scalable Vector Graphics [SVG]. Compliant playback devices that support image display must be capable of displaying JPEG and PNG; support for SVG is recommended. Appendix 8 of the SVG specification addresses accessibility issues. See Section 3.3, "Manifest" for filename extension requirements.

7   Synchronization of Media Files

7.1   Introduction

7.1.1   Background

(This section is informative.)

The Synchronized Multimedia Integration Language (SMIL 2.0) [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 textual content 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 and textual rendering decoders are present and no system for intellectual property protection restricts access.

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 use 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 textual content file (referenced by a 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 `customTests`, which allows SMIL authors to identify by class selected elements of a document (e.g., notes, page numbers, line numbers). The playback device can then expose to the user the presence of these classes and allow the user to select whether a given class of elements is to be read or skipped over during sequential playback.

The duration of time containers in SMIL presentations may be determined by the duration of child media objects and time containers, or by recognized events that signal the end of a time container. DTB producers may elect to create content that incorporates pauses in the presentation to allow for users to examine related materials or to work on a problem, for example. Users, once they complete the related task, will signal that playback can continue by an appropriate mechanism in the player user interface.

Pauses in sequential presentation are authored by specifying a time container with an end attribute value that indicates that the container will remain active until the occurrence of a user event, such as the pressing of a resume or play button, or until the end of a specified duration.

Escapable structures are implemented by a user initiated escape event. Structures that are escapable are wrapped by a time container that will play until either the normal end time of the child time container(s) or the occurrence of a user escape event, whichever comes first. In this case, the DTB player that supports escapability may provide a user interface mechanism for indicating that the user has requested to escape from the current structure.

The DTB producer determines granularity of the synchronization events. Synchronization events can be limited to the primary structural elements (those indicated in the NCX) or can 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 includes mark-up tags for the desired elements and that those elements include unique identifiers that can be referenced in the SMIL files.

The SMIL file for a DTB typically will consist of a sequence of 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 media objects) and `par` (parallel time grouping in which multiple media objects play back at the same time). A simple form of DTB SMIL file would be as follows, where the three `par`s shown are played one after the other, and the text and audio content referenced in each `par` are rendered simultaneously:

```
<smil>
 ...
<seq>
  <par><text.../><audio.../></par>
  <par><text.../><audio.../></par>
  <par><text.../><audio.../><img.../></par>
</seq>
....
</smil>
```

7.1.2   SMIL Modules

(This section is informative.)

Synchronization of media objects in this standard is based on the [SMIL 2.0](#) specification. Developers are requested to reference [SMIL 2.0](#) [SMIL] for complete background and details. Only a small subset of the SMIL specification is used in this application, 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 not used but are included because they are part of a core set of modules required for *host language conformance* under W3C modularization guidelines.

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

The modules mentioned above can be combined, using W3C modularization guidelines, to form a *profile* specific to DTB applications. Section 2 of the SMIL specification, "The SMIL 2.0 Modules," describes this process in detail.

7.2   Application of SMIL to DTBs

(This section is normative.)

To simplify validation using commonly available parsers and to lessen the complexity of determining content models and applicable attribute lists, a DTB-Specific SMIL DTD is included with this standard (see Appendix 1). This DTD includes only those elements and attributes from the modules listed above that are required for the DTB application. In addition, it is more restrictive than the SMIL modules in that some attributes are required in the DTB application when they are only implied in the SMIL modules.

A compliant DTB must contain at least one SMIL file. All SMIL files included in a DTB must be valid XML documents conforming to dtbsmil-2005-1.dtd. See Section 3.3, "Manifest" for filename extension requirements. The xmlns attribute on the smil element must be explicitly specified in the document instance, using the value drawn from the above-named DTD. Entity

declarations must occur in the internal DTD subset. See further section 16.1 "General File Conformance Requirements."

Time containers (`seq`s or `par`s) within SMIL files must contain ids. Media objects (`audio`, `text`, and `img`) may also contain `id`s, although this practice will generally be limited to single-medium DTBs. See section 7.4.10, "Text-Only DTBs."

In the textual content file, each segment to be synchronized (e.g., heading, paragraph, list item, etc.) must be contained within an element carrying a unique id to which the corresponding SMIL segment points. In addition, any textual content file element referenced by a SMIL file must include a `smilref` attribute specifying the URI of the time container or media object that references it. The `smilref` value will normally be the URI of the SMIL time container containing the media object that references a given element. However, in a text-only DTB consisting of a sequence of text media objects, `smilref` shall contain the URI of the referencing media object itself. See section 4.2.1, "DTBook Markup Related to SMIL."

It is strongly recommended that the SMIL file(s) have a level of granularity matching that of the textual content file. That is, if the textual content file is marked up to the paragraph level, the SMIL file(s) should include synchronization to the paragraph level.

All time offsets in SMIL files (and all other applicable DTB files, e.g., 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.

7.3   SMIL Elements

(This section is informative.)

As mentioned above, the DTB application uses 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, which make up the DTB-Specific SMIL DTD.

- **<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, "Core Attributes."
  - **xmlns (CDATA, FIXED) "http://www.w3.org/2001/SMIL20/":** Specifies the default XML namespace for all elements in SMIL. See [XML-Namespaces] for details on namespaces. This attribute and its value (given in DTD) must be explicitly specified in the document instance.
  - **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.

  Declaration: `<!ELEMENT head (meta*, (layout, meta*)?, (customAttributes,meta*)?) >`

  Syntax: `<head>`...content...`</head>`

  Attributes:

    o **%Core.attrib;** See section 7.3.1, "[Core Attributes.](#)"

    o **xml:lang (NMTOKEN, IMPLIED)**

  Valid inside: `<smil>`

- **<meta>**

  Description: Contains information describing the SMIL document.

  Declaration: `<!ELEMENT meta EMPTY >`

  Syntax: `<meta ...attributes... />`

  Attributes:

    o **content (CDATA, #IMPLIED)**

    o **name (CDATA, #[REQUIRED])**

  Valid inside: `<head>`

  Comments: See section 7.5, "[SMIL Metadata](#)" for normative content.

- **<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:

    o **%Core.attrib;** See section 7.3.1, "[Core Attributes.](#)"

    o **xml:lang (NMTOKEN, IMPLIED)**

  Valid inside: `<head>`

  Comments: Syntax is restricted. See section 7.4.6, "[Layout Syntax](#)" for normative content.

- **<region>**

  Description: Controls the position, size, and scaling of media objects (e.g., text, img).

  Declaration: `<!ELEMENT region EMPTY >`

  Syntax: `<region ...attributes... />`

Attributes:

- o **id (ID, REQUIRED)** Value of `region` attribute on media object references the id on appropriate `region` element.

- o **bottom (CDATA, 'auto')** Locates `region` in display space. See [SMIL 2.0](#) for details.

- o **left (CDATA, 'auto')** Locates `region` display space. See [SMIL 2.0](#) for details.

- o **right (CDATA, 'auto')** Locates `region` in display space. See [SMIL 2.0](#) for details.

- o **top (CDATA, 'auto')** Locates `region` in display space. See [SMIL 2.0](#) for details.

- o **height (CDATA, 'auto')** Locates `region` in display space. See [SMIL 2.0](#) for details.

- o **width (CDATA, 'auto')** Locates `region` in display space. See [SMIL 2.0](#) for details.

- o **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` in which it is displayed. See [SMIL 2.0](#) for definitions of attribute values.

- o **backgroundColor (CDATA, IMPLIED)** Sets background color of the area of the `region` that is not covered by the media object(s) being displayed.

- o **showBackground ((always|whenActive) 'always')** Controls whether the `backgroundColor` of a `region` is shown when no media is being rendered to the `region`. See [SMIL 2.0](#) for definitions of attribute values.

- o **z-index (CDATA, IMPLIED)** Used for control of multilayered displays.

- o **xml:lang (NMTOKEN, [IMPLIED](#))**

Valid inside: `<layout>`

Comments: All media objects whose `region` attribute references the `id` on a given `region` element will be displayed in that `region`.

- **<customAttributes>**

  Description: Contains one or more `customTests` that 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:

  - o **%Core.attrib;** See section 7.3.1, "[Core Attributes.](#)"

  - o **xml:lang (NMTOKEN, IMPLIED)**

  Valid inside: `<head>`

  Comments: See section 7.4.3, "['Skippable' Structures](#)" for normative content.

- **<customTest>**

  Description: Defines the kinds of structures (e.g., page numbers, notes, line numbers, etc.) that 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: `<customTest ...attributes... />`

Attributes:

- o **id (ID, REQUIRED)** Id here serves as a unique identifier referenced by a `customTest` attribute on `par` or `seq` in body of SMIL.

- o **class (CDATA, IMPLIED)**

- o **title (CDATA, IMPLIED)**

- o **xml:lang (NMTOKEN, IMPLIED)**

- o **defaultState ((true|false) 'false')** Specifies whether player will render (`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.

- o **override ((visible|hidden) 'hidden')** Specifies whether runtime resetting of `defaultState` should be encouraged (`value= "visible"`) or discouraged (`value = "hidden"`). See section 7.4.3, "'Skippable' Structures" for normative content.

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:

  - o **%Core.attrib;** See section 7.3.1, "Core Attributes.."

  - o **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 `par`s and `seq`s.

  Declaration: `<!ELEMENT seq (par|seq|text|audio|img|a)+ >`

  Syntax: `<seq>...content...</seq>`

  Attributes:

  - o **id (ID, REQUIRED):**

  - o **class (CDATA, IMPLIED):** Optional descriptor of this instance of the element. Can be used to identify structures such as tables and lists for which special navigation functions should be automatically invoked when entered. See section 7.4.2, "Automatic Invocation of Special Navigation Modes." Can also be used to select a presentation from the resource file. See section 10.3, "Resource File Requirements."

  - o **customTest (IDREF, IMPLIED):** ID reference linking `seq` with matching `customTest` element in `head`.

- **dur (CDATA, IMPLIED)** The duration of the seq. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module [SMIL].

- **xml:lang (NMTOKEN, IMPLIED)**

- **end (CDATA, IMPLIED)** Determines the active duration of Escapable or Producer Pause content. See section 7.8, "End Attribute Values", for end value syntax. See also sections 7.4.1, "'Escapable' Structures" and 7.4.11, "Producer Pauses."

- **fill ((freeze | remove) 'remove')** Determines whether a visual element is frozen at its final state or is no longer presented. See section 7.4.11, "Producer Pauses" and section 10.3.1 of "SMIL 2.0."

Valid inside: body, seq, par, a

Comments: It is permissible 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):**

  - **class (CDATA, IMPLIED):** Optional descriptor of this instance of the element. Can be used to identify structures such as tables and lists for which special navigation functions should be automatically invoked when entered. See section 7.4.2, "Automatic Invocation of Special Navigation Modes." Can also be used to select a presentation from the resource file. See section 10.3, "Resource File Requirements."

  - **customTest (IDREF, IMPLIED):** ID referencing matching customTest element in head.

  - **xml:lang (NMTOKEN, IMPLIED)**

  Valid inside: body, seq, a

  Comments: See section 7.4.7, "Content of pars" for normative content.

- **<text>**

  Description: Points to segment of textual content to be rendered.

  Declaration: <!ELEMENT text EMPTY >

  Syntax:

  Attributes:

  - **id (ID, IMPLIED):** Optional identifier.

  - **src (CDATA, REQUIRED):** URI of fragment of textual content file to be rendered.

  - **type (CDATA, IMPLIED):** Type of media file.

  - **region (CDATA, IMPLIED):** Specifies the region (defined in layout in document head) in which the text will be presented. References the id of the appropriate region. All types of text objects that are to appear in the same

rendering space would 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"`).

- o **xml:lang (NMTOKEN,** [IMPLIED]**)**

Valid inside: `body, par, seq, a`

- **<audio>**

  Description: Points to segment of audio content to be rendered.

  Declaration: `<!ELEMENT audio EMPTY >`

  Syntax: `<audio...attributes... />`

  Attributes:

  - o **id (ID, IMPLIED):** Optional identifier.
  - o **src (CDATA, REQUIRED):** URI of audio file containing clip to be rendered.
  - o **type (CDATA, IMPLIED):** Type of media file.
  - o **clipBegin (CDATA, REQUIRED):** 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 [SMIL]. See section 7.7, "Media Clipping and Clock Values."
  - o **clipEnd (CDATA, REQUIRED):** 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`.
  - o **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 appropriate `region`.
  - o **xml:lang (NMTOKEN,** [IMPLIED]**)**

  Valid inside: `body, par, seq, a.`

- **<img>**

  Description: Points to image to be rendered.

  Declaration: `<!ELEMENT img EMPTY >`

  Syntax: `<image...attributes... />`

  Attributes:

  - o **id (ID, IMPLIED):** Optional identifier.
  - o **src (CDATA, REQUIRED):** URI of image file to be rendered.
  - o **type (CDATA, IMPLIED):** Type of media file.
  - o **region (CDATA, IMPLIED):** Specifies the `region` (defined in `layout` in document `head`) in which the image will be presented. References the id of the appropriate `region`.
  - o **xml:lang (NMTOKEN,** [IMPLIED]**)**

  Valid inside: `body, par, seq, a`

- **<a>**

    Description: Defines a link. The default behavior is to be active for the duration of the media object it contains.

    Declaration: `<!ELEMENT a (par|seq|text|audio|img)* >`

    Syntax: `<a>...content...</a>`

    Attributes:

    - **%Core.attrib;** See section 7.3.1, "Core Attributes."
    - **xml:lang (NMTOKEN, IMPLIED)**
    - **href (%URI;, REQUIRED)** Specifies the URI of the target of the link. The URI may include a fragment identifier.
    - **external ((true|false) 'false')** An external link points to media content that is not part of the DTB. The external media content must be rendered by an external application, whether or not that content is renderable by the DTB player.

    Valid inside: `body, par, seq`

    Comments: See section 7.4.5, "Links" for normative content.

7.3.1 Core 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)**

7.3.2 xml:lang Attribute

(This section is normative)

The xml:lang attribute specifies the [RFC 3066] language code of SMIL elements that have, or reference, content.

7.4 SMIL Requirements for DTBs

7.4.1 "Escapable" Structures

(This section is normative.)

DTB players should provide the functionality to allow readers to "escape" from the DTB rendition of the following structures, with a single action: tables, lists, producer's notes, annotations, sidebars and notes. Escape means to move local navigation to the point directly following the current structure (table, list, etc.).

To support this functionality, the SMIL entry for any such structure must consist of a seq element containing at least one child time container (seq or par). In addition, a class attribute identifying the type of structure must be applied to the parent seq containing just the escapable structure. Class attribute values are unrestricted, except for tables and lists, where the values must be "table" and "list," per section 7.4.2.

The parent `seq` element containing just the escapable structure must have an `end` attribute with the value of `"DTBuserEscape;childId-value.end"`, where `childID-value` is the `id` value of the last child time container of the parent `seq`. The value, as specified, defines to the

player that the escapable structure will be terminated by either a user-initiated escape action (DTBuserEscape) OR the normal end of the last child time container within the escapable structure, whichever occurs first. See section 7.8, "End Attribute Values" and Example 7.4.

A Resource File entry must be supplied for each type of escapable structure identified within a DTB. Typically, resources are associated with escapable structures via the `class` attribute on SMIL time containers. See section 10, "Resource File."

The above rules only apply to those DTBs where these structures can be identified in SMIL. For example, in a DTB where time containers hold complete chapters, "escapability" is not possible for the structures described in this section.

7.4.2 Automatic Invocation of Special Navigation Modes

(This section is normative.)

DTB player developers may choose to automatically invoke special player navigation modes when the reader enters a table or list. (See "Document Navigation Features List [Navigation Features].") To support this functionality, a class attribute must be included on the seq containing a table or list, using the values "table" or "list," as appropriate).

7.4.3 "Skippable" Structures

(This section is normative.)

Players should offer the user the option to "turn off" certain structures in a DTB, that is, select structures that the player will then automatically skip over during sequential playback. To support this capability, a `seq` or `par` that exactly contains any of the structures specified below must have a `customTest` attribute applied to it. In addition, the `customAttributes` element, as well as a `customTest` element for each "skippable" structure, must be present in the `head` of each SMIL file and contain content.

At a minimum, `customTest` attributes must be applied to time containers for line numbers, notes, note references, annotations, page numbers, optional producer's notes, and optional sidebars. Notes, annotations, optional producer's notes, and optional sidebars containing multiple segments must be represented as a series of `par`s wrapped in a `seq`, so that a `customTest` can be applied to the `seq`, permitting the user to skip the entire sequence.

The rules in the above paragraphs only apply to those DTBs where SMIL time containers exactly contain the "skippable" structures. In a DTB where time containers hold complete chapters, for example, "skippability" is not possible for the structures described in this section.

Attribute values (for customTest attributes on `seq`s or `par`s and for the `id` attribute on `customTest` elements) are left to the producer. Producers who utilize skippable elements must supply a resource file entry for each such element, (via the corresponding `smilCustomTest` element in the NCX) so that a player may expose those skippable elements in its user interface.

The value of `defaultState` for a given `customTest` element must be the same in all SMIL files of a DTB.

The recommended value of the `override` attribute is 'visible'. Note that this is contrary to the default value in the SMIL 2 specification. See description of <customTest> above.

When a user moves via global navigation directly to a skippable element that has been turned off, the player should ignore the current state and render the content of that element.

Section 8.5, "How the NCX Works," describes how information on skippable structures can be gathered in the NCX for efficient presentation to the user.

7.4.4 Packaging Files across Several Media Units

(This section is normative.)

When a DTB spans several media units (e.g., CD-ROM discs), all files required to render any given SMIL file must be present on the same media unit as that SMIL file. This requirement ensures that players need only track the location of SMIL files in order to provide a complete DTB presentation.

7.4.5 Links

(This section is normative.)

If links (i.e., `<a>` (anchor) elements with `href` attributes) are present in the textual content file of a DTB, they must also be included in the corresponding SMIL file(s). Related links in textual content and SMIL files must point to the same information in the textual content and SMIL files, respectively. Specifically, the id of the SMIL time container that is the destination of the SMIL link must be equal to the smilref attribute on the text element that is the destination of the text link. Nesting of anchors is prohibited. The default behavior of a link is to be active for at least the duration of the object it contains. Players may establish other behaviors (e.g., maintaining links in the active state for a preset period of time -- possibly modifiable by the user -- or until the next link is encountered).

An external link points to media content that is not part of the DTB. Such a link must be identified as external by the `external` attribute on the `<a>` element. The default value of the `external` attribute is false. The external media content must be rendered by an external application, whether or not that content is renderable by the DTB player.

7.4.6 Layout Syntax

(This section is normative.)

This standard allows only SMIL 2.0 Basic Layout syntax (i.e., CSS2 syntax and others are not permitted).

7.4.7 Content of <par>s

(This section is normative.)

Each par can contain no more than one each of `text`, `audio`, `image`, and `seq`. See section 7.4.10, "Text-Only DTBs" for further discussion of this issue.

When both textual content and audio files are present, `text` and `audio` objects within the same `par` must both represent the same body of material (e.g., the same paragraph).

Because of resource limitations on portable DTB players, SMIL presentations must not be created such that multiple audio media objects are rendered simultaneously. Reading systems are not required to support simultaneous rendering of multiple audio files.

7.4.8 Notes and Annotations in SMIL

(This section is normative.)

It is strongly recommended that links be applied to media objects (normally `audio`) for all `noteref`s and `annoref`s, with the corresponding `note`s and `annotation`s as the targets. The presence of the links will enable key player functionality, such as easy access to `note`s when `noteref`s are turned on and `note`s turned off.

It is recommended that `noteref`s and `note`s be implemented in SMIL such that the default, linear presentation (on a simple player) of the `noteref`s and `note`s is in the order and location appropriate to the producing agency's policy for rendering `note` references and `note`s.

7.4.9 Images in SMIL

(This section is informative.)

Duration of image display will be equal to that of the longest media object or time container contained within the same `par`. Example 7.2 below shows a sample implementation of SMIL for an image and its associated caption and producer's note.

7.4.10 Text-Only DTBs

(This section is normative.)

Text-only DTBs must include SMIL files. This will ensure user access to the many features enabled by SMIL. As mentioned above, it is strongly recommended that the SMIL file(s) have a level of granularity matching that of the textual content file.

In a DTB that contains no audio material, the duration of `text` media objects is controlled either by the user (i.e., the player renders the next `text` object on command) or the player (e.g., a text-to-speech engine or a pacing algorithm for a large-print or Braille display triggers the next media object).

7.4.11 Producer Pauses

(This section is normative.)

Producers may create content that allows for the presentation of the DTB to be paused until the user takes an action that will resume playback. These pauses may be used by the producer, for example, to allow time for a user to perform an exercise or examine a physical model related to the content.

Producers may optionally specify a maximum active duration for the time container. This active duration is the simple duration of the content plus any additional time the producer wishes to allow before the DTB resumes playback if the user does not initiate the resume directly.

Content created for this purpose will consist of a `seq` element containing at least one child time container (`seq` or `par`); the parent `seq` has an `end` attribute value of "DTBuserResume", and, optionally, a time duration. The time container with this `end` attribute value will have an indefinite time duration, unless a duration is specified, and only terminate upon user invocation of the DTBuserResume, or when the end of any specified duration is reached. The player will present the content of any child media objects and then wait for the occurrence of the DTBuserResume or the end of the specified duration, if present. See section 7.8, "End Attribute Values", for `end` value syntax.

Players that do not support Producer Pauses will ignore the `end` event and continue playing.

See Examples 7.5 and 7.6.

7.5   SMIL Metadata

(This section is normative.)

Metadata is included in the `<head>` element using the `<meta>` element. Content producers may introduce other metadata besides those listed below, if needed. However, metadata names shall not begin with the prefix "dtb:" unless defined in this standard. Players must not fail when encountering unknown metadata but must, at a minimum, ignore it.

- dtb:generator
    - Content: Name and version of software that generated the SMIL file.
    - Occurrence: Optional - recommended.

- dtb:totalElapsedTime

  - Content: The total time elapsed up to the beginning of this SMIL file. Clock Values from SMIL 2.0 Timing and Synchronization Module [SMIL]. See section 7.7, "Media Clipping and Clock Values."

  - Occurrence: Required

  - Comments: Set to zero for DTBs of type textNCX.

- dtb:uid

  - Content: The globally unique identifier for the DTB. The value is the same as that of the `dc:Identifier` element referenced by the package file's unique-identifier element. See section 3.1, "Package Identity."

  - Occurrence: Required

7.6   Examples

(This section is informative.)

The following example illustrates the use of `head` and its contents. Three instances of the `meta` element contain the unique id of the DTB, the tool that generated this SMIL file, and the elapsed time to the start of the file. 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 time containers with `customTest="pagenum"` will be skipped by default, while time containers with `customTest="notes"` or `customTest="prodnotes"` will automatically be played. If the user interface of the playback device supports it, the user can change these settings.

Example 7.1:

```
<?xml version="1.0"  encoding="UTF-8"?>
<!DOCTYPE smil PUBLIC "-//NISO//DTD dtbsmil 2005-1//EN"
  "http://www.daisy.org/z3986/2005/dtbsmil-2005-1.dtd">
<smil xmlns="http://www.w3.org/2001/SMIL20/">
    <head>
        <meta name="dtb:uid" content="dk-dbb-4z0065" />
        <meta name="dtb:generator" content="smilgen2.4" />
        <meta name="dtb:totalElapsedTime" content="01:33:56.233" />
        <layout>
            <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"
override="visible"/>
            <customTest id="note" defaultState="true"
override="visible"/>
            <customTest id="prodnote" defaultState="true"
override="visible"/>
        </customAttributes>
    </head>
    <body>
```

```
          ...
        </body>

    </smil>
```

Example 7.2 shows the use of SMIL elements within body. Each par (a page number, a heading, two paragraphs, and a figure are shown) includes the segment of text, the image (if applicable), and the corresponding audio clip that are to be rendered simultaneously. The figure falls between the two paragraphs.

The image file is presented in parallel with text and audio versions of the figure caption and a producer's note describing the figure. The entire group is wrapped in a par, with the image file rendered simultaneously with a sequence of two pars.

A link "wraps" the par for the second paragraph.

Example 7.2:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE smil PUBLIC "-//NISO//DTD dtbsmil 2005-1//EN"
  "http://www.daisy.org/z3986/2005/dtbsmil-2005-1.dtd">
<smil xmlns="http://www.w3.org/2001/SMIL20/">
<head>
....
</head>
<body>
    <seq id="baseseq" >
        <par id="p1" customTest="pagenum">
            <text region="text" src="rs.xml#pg_1" />
            <audio src="rs_fwdx.mp3" clipBegin="00:00:00"
            clipEnd="00:00:00.91" />
        </par>

        <par id="h1">
            <text region="text" src="rs.xml#h1_1" />
            <audio src="rs_fwdx.mp3" clipBegin="00:00:01.62"
            clipEnd="00:00:02.53" />
        </par>

        <par id="para1">
            <text region="text" src="rs.xml#para_1" />
            <audio src="rs_fwdx.mp3" clipBegin="00:00:03.51"
            clipEnd="00:01:45.36" />
        </par>

        <par id="img1">
            <img region="image" src="fig1.png" />
            <seq id="icap1">
                <par id="cap1">
                    <text region="caption" src="rs.xml#caption_1" />
                    <audio src="rs_fwdx.mp3" clipBegin="00:01:45.98"
                    clipEnd="00:01:52.66" />
                </par>

                <par id="pnote1" customTest="prodnote" class="prodnote">
                    <text region="text" src="rs.xml#prodnote_1" />
                    <audio src="rs_fwdx.mp3" clipBegin="00:01:53.08"
                    clipEnd="00:02:55.34" />
```

```
                        </par>
                    </seq>
                </par>

                <a href="rs12.smil#h2_9">
                    <par id="para2">
                        <text region="text" src="rs.xml#para_2" />
                        <audio src="rs_fwdx.mp3" clipBegin="00:02:56.21"
                        clipEnd="00:04:03.75" />
                    </par>
                </a>
                ...
            </seq>
        </body>
    </smil>
```

Notes or sidebars containing multiple segments will be represented as a series of `par`s wrapped in a `seq`, so that the user can skip the entire sequence. The first part of Example 7.3 illustrates this situation. In addition, note references occurring in the middle of a paragraph will require this special syntax so that the playback device can properly render the content with or without either the note reference or the note.

In the second half of Example 7.3, the first `par` contains the portion of paragraph 120 preceding a note reference (identified with a `span` element in the textual content file). The second `par` holds the note reference itself (i.e., "footnote 1"). The third `par` contains the contents of footnote 1 and the last holds the remainder of paragraph 120. Note that the `seq` and each `par` contain 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 id="baseseq" >
            ... (a series of pars)
            <seq id="sidebar_1" customTest="sidebar">
                <par id="para_9">
                    <text region="text" src="rs.xml#para_9" />
                    <audio src="rs_fwdx.mp3" clipBegin="02:02.711"
                    clipEnd="02:14.678" />
                </par>

                <par id="para_10">
                    <text region="text" src="rs.xml#para_10" />
                    <audio src="rs_fwdx.mp3" clipBegin="02:15.545"
                    clipEnd="02:44.612" />
                </par>
            </seq>
            ... (a series of pars)
            <seq id="para_120">
                <par id="span_3">
                    <text region="text" src="rs.xml#span_3" />
                    <audio src="rs_fwdx.mp3" clipBegin="46:58.744"
                    clipEnd="47:21.659" />
                </par>
```

```
        <par id="nref_1" customTest="noteref">
           <text region="text" src="rs.xml#nref_1" />
           <audio src="rs_fwdx.mp3" clipBegin="47:22.610"
           clipEnd="47:23.555" />
        </par>

        <par id="ftn_1" customTest="note" class="note">
           <text region="notes" src="rs.xml#ftn_1" />
           <audio src="rs_notes.mp3" clipBegin="00:00.091"
           clipEnd="00:34.754" />
        </par>

        <par id="span_4">
           <text region="text" src="rs.xml#span_4" />
           <audio src="rs_fwdx.mp3" clipBegin="47:24.057"
           clipEnd="47:582" />
        </par>
     </seq>
     ... (a series of pars)

   </seq>

</body>
....
```

Example 7.4:

The following example shows the use of DTBuserEscape and childID.end for escapability.
The escapable content is contained in the `par` with `id` sf0003. The `seq` which wraps the `par`
has an `end` attribute which specifies that the `seq` will end when either the DTBuserEscape
event occurs, or the child time container (the `par`) with `id` sf0003 ends after the normal play
time.

```
<seq id="fr003" end="DTBuserEscape;sf0003.end"  customTest="prodnote"
class="prodnote">
 <par id="sf0003">
  <text src="book02.xml#sf0003" />
  <audio src="book02.mp3" clipBegin="npt=00:00:00"
clipEnd="npt=00:00:11.5" />
 </par>
</seq>
```

Example 7.5:

The following example shows the use of DTBuserResume for producer-controlled pauses. In
the first part of the example, a `prodnote` indicates to the reader that a supplemental, physical
model is to be examined at this point of the book presentation. Instructions are given to the
user to press a resume key when they are ready to resume reading. After the instruction is
presented, the playback pauses until the users presses the key to resume playback.

```
<p id="sf0002">The water molecule is composed of one Oxygen and two
Hydrogen atoms.</p>

<prodnote id="sf0003">Please examine the model of the water molecule.
Notice how the two Hydrogen atoms are a attached to the Oxygen atom.
Press
the resume key when you have finished to continue reading.</prodnote>
```

```
<p id="sf0004">Heavy water is chemically the same as regular (light)
water,
but with the two hydrogen atoms replaced with deuterium atoms.</p>
```

In the second part of the example, the corresponding SMIL time containers are shown. In the seq with `id="sf0003"`, the `end` attribute has the value `DTBuserResume`, to indicate that this time container will not end until the user activates the DTB player's resume key. The `fill="freeze"` attribute is specified to force the DTB Player with text display (either visual or Braille) to maintain the text of the `prodnote` until the user resumes playback.

```
<par id="sf0002">
 <text region="textRegion" src="chemtext.xml#sf0002" />
 <audio src="chem002.mp3" clipBegin="00:00:00.00" clipEnd="00:00:04.00"
/>
</par>
<seq id="sf0003" end="DTBuserResume" fill="freeze"
customTest="prodnote" >
 <par id="sf0003a" >
    <text region="textRegion" src="chemtext.xml#sf0003" />
    <audio src="chem002.mp3" clipBegin="00:00:04.08"
clipEnd="00:00:16.00" />
 </par>
</seq>
<par id="sf0004">
 <text region="textRegion" src="chemtext.xml#sf0002" />
 <audio src="chem002.mp3" clipBegin="00:00:16.08" clipEnd="00:00:25.00"
/>
</par>
```

Example 7.6:

In the next example, the producer has applied a maximum time to be allowed for the producer-supplied pause. If the user does not activate the DTBuserResume button within the allowed time, the playback will resume once the specified time interval has passed. In the `seq` with `id="sf0003"`, the `end` attribute has the value `"DTBuserResume;sf0003a.end+00:01:00"`, which defines that the `seq` will end when either the DTBuserResume is activated or 1 minute has elapsed since the end of the child time container (sf0003a), whichever comes first. This allows the user one minute following the end of the `par` with `id="sf0003a"` to view the model, before playback resumes automatically.

```
<seq id="sf0003" end="DTBuserResume;sf0003a.end+00:01:00" fill="freeze"
customTest="prodnote" >
  <par id="sf0003a" >
   <text region="textRegion" src="chemtext.xml#sf0003" />
   <audio src="chem002.mp3" clipBegin="00:00:04.08"
clipEnd="00:00:16.00" />
  </par>
 </seq>
```

7.7   Media Clipping and Clock Values

Section 7.5.1, "MediaClipping Attributes" of SMIL 2.0 [SMIL] defines SMIL clipping attributes and their value syntax. In addition, the SMIL 2.0 [SMIL] Timing and Synchronization Module

describes several different formats in which "clock values" (timing) may be represented. See *Clock Values* in Section 10.3.1 of that module. Playback devices must support the formats and syntaxes described in the above-mentioned sections, as restricted below. The three formats are:

Full-clock-value (hours, minutes, seconds, and fractions of seconds): e.g., 3:22:55.91 (or npt=3:22:55.91)

Partial-clock-value (minutes, seconds, and fractions of seconds): e.g., 43:15.044 (or npt=43:15.044)

Timecount-value (one or more digits, plus an optional fraction and unit of measurement -- h=hours, min=minutes, s=seconds, ms=milliseconds): e.g., 34.6s (or npt=34.6s), 356ms (or npt=356ms), 58.214 (or npt=58.214). (For Timecount values, if no unit is shown, the default is "s" for seconds.) No embedded white space is allowed in clock values, although leading and trailing white space characters will be ignored.

Clipping and Clock values have the following syntax:

```
Clip-value-MediaClipping ::= [ "npt=" ] Clock-value
Clock-value              ::= ( Full-clock-value | Partial-clock-value |
Timecount-value )
Full-clock-value         ::= Hours ":" Minutes ":" Seconds ("."
Fraction)?
Partial-clock-value      ::= Minutes ":" Seconds ("." Fraction)?
Timecount-value          ::= Timecount ("." Fraction)? (Metric)?
Metric                   ::= "h" | "min" | "s" | "ms"
Hours                    ::= DIGIT+; any positive number
Minutes                  ::= 2DIGIT; range from 00 to 59
Seconds                  ::= 2DIGIT; range from 00 to 59
Fraction                 ::= DIGIT+
Timecount                ::= DIGIT+
2DIGIT                   ::= DIGIT DIGIT
DIGIT                    ::= [0-9]
```

The SMPTE syntax from SMIL 2.0 section 7.5.1, "MediaClipping Attributes", is not allowed.

Clip durations must be positive, i.e. a clipEnd clock value must be greater than its corresponding clipBegin clock value. No clipEnd shall contain a clock value greater than the duration of the referenced media file.

7.8   End Attribute Values

(This section is normative)

This section defines the set of allowed end attribute values that are used on the seq element in a DTB for the specific purpose of supporting Escapability and Producer Pauses.

The end attribute provides additional control over the active duration of a time container. The allowed end values are specific to the described uses only and are not defined for any other use in a DTB.

DTB Players that support escapability and Producer Pauses must support the following values.

7.8.1   Allowed End Values

(This section is normative)

Section 10.3.1, "Attributes", of the SMIL 2.0 Recommendation [SMIL], defines values and value syntax for the `end` attribute. The following subset of those values is allowed in this standard:

Values for Escapable content

```
DTBuserEscape;childId-value.end
```

Values for Producer Pause

```
DTBuserResume(;childId-value.end + Clock-value)?
```

Values for Producer Pause that is also Escapable

```
DTBuserEscape; DTBuserResume(;childId-value.end + Clock-value)?
```

The `childId-value` is from the `id` of the last child time container of the `seq` to which the `end` attribute is applied.

`Clock-value` describes the length of the producer-specified pause (if present), and allowed values are defined in Section 7.7, "Media Clipping and Clock Values". The pause interval begins when the time container identified by `childId-value` ends.

The interpretation of a list of end values is detailed in the following section, "Computing the Active Duration."

7.8.2   Computing the Active Duration

(This section is normative)

The active duration of a time container is defined as the simple duration of all child time containers and media objects, modified by any `end` values. A DTB Player will compute the active duration, which will be indefinite if only DTBuserResume is specified in the `end` value. In the case of a list of end values, the first occurrence of any condition defined by a value in the list will signal the end of the active duration for the time container.

7.8.3   Processing Nested Structures

(This section is normative)

In the case of nested escapable structures, default SMIL event processing behavior would cause any escape event occurring in a child of a nested structure to bubble up the nested time container hierarchy and terminate the top level escapable time container. In some cases, a user may only wish to escape from the current time container, such as a list within a table cell or within another list, and move to the next sequential element in the nested time container hierarchy. DTB players which implement escapability should allow users the option of specifying whether escaping from a nested escapable time container exits the complete nested hierarchy, or escapes only from the immediate escapable structure inside which the current position is located.

8   Navigation Control File (NCX)

8.1   Introduction

(This section is informative.)

The Navigation Control file for XML applications (NCX) exposes the hierarchical structure of a DTB 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 PC users. 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, nonhierarchical lists and can be accessed by the user as well.

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. The alternative of a simple linear playback of the book will be available for those users not requiring the navigation features of the NCX.

8.2   Key NCX Requirements

(This section is normative.)

Every DTB must contain exactly one NCX file. The NCX file must be a valid XML document conforming to ncx-2005-1.dtd (see Appendix 1, "NCX DTD") and comply with the additional normative requirements of section 8.4. See Section 3.3, "Manifest" for filename extension requirements. The version and xmlns attributes on the ncx element must be explicitly specified in the document instance, using values drawn from the above-named DTD. Entity declarations must occur in the internal DTD subset. See further section 16.1 "General File Conformance Requirements."

8.3   NCX Elements

(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, docAuthor*, navMap, pageList?, navList*)>`

  Syntax: `<ncx ...attributes...>...content...</ncx>`

  Attributes:

    o **version (CDATA, FIXED) "2005-1":** Specifies the version of the DTD used in this instance. This attribute and its value (given in DTD) must be explicitly specified in the document instance.

    o **xmlns (CDATA, FIXED) "http://www.daisy.org/z3986/2005/ncx/":** Specifies the default XML namespace for all elements in the NCX. See [XML-

Namespaces] for details on namespaces. This attribute and its value (given in DTD) must be explicitly specified in the document instance.

- o **xml:lang (NMTOKEN, IMPLIED):** Specifies the [RFC 3066] language code of the language of the document.

- o **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 smilCustomTest data and metadata.

  Declaration: `<!ELEMENT head (smilCustomTest | meta)+>`

  Syntax: `<head>`...content...`</head>`

  Attributes: None

  Valid inside: `<ncx>`

- **<smilCustomTest>**

  Description: Duplicates customTest data found in SMIL files. Each unique customTest element that appears in one or more SMIL files and has been referenced at least once by a customTest attribute will have three of its attributes (`id`, `override`, and `defaultState`) duplicated in a smilCustomTest element in the NCX. The NCX thus gathers in one place all customTest elements used in the SMIL files, along with their `defaultState` setting, for presentation to the user. When a customTest element in SMIL has been applied to a time container holding one of the book structures defined in section 8.4.4 "smilCustomTest Element", the bookStruct attribute must be applied to the corresponding smilCustomTest and contain the appropriate value from the enumerated list.

  Declaration: `<!ELEMENT smilCustomTest EMPTY>`

  Syntax: `<smilCustomTest...attributes.../>`

  Attributes:

  - o **id (ID, REQUIRED)**

  - o **defaultState (true | false) 'false'**

  - o **override (visible | hidden) 'hidden'**

  - o **bookStruct (PAGE_NUMBER | NOTE | NOTE_REFERENCE | ANNOTATION | LINE_NUMBER | OPTIONAL_SIDEBAR | OPTIONAL_PRODUCER_NOTE) #IMPLIED**

  Valid inside: `<head>`

- **<meta>**

  Description: Contains metadata applicable to the NCX file.

  Declaration: `<!ELEMENT meta EMPTY>`

  Syntax: `<meta ...attributes... />`

Attributes:

- o **name (CDATA, REQUIRED)**

- o **content (CDATA, REQUIRED)**

- o **scheme (CDATA, IMPLIED)**

Valid inside: <head>

Comments: Required and optional metadata are listed in section 8.4.1.

- **<docTitle>**

  Description: The title of the document, presented as text and, optionally, in audio or image renderings, for presentation to the reader.

  Declaration: <!ELEMENT docTitle (text, audio?, img?)>

  Syntax: <docTitle ...attributes... > ...content... </docTitle>

  Attributes:

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

  Valid inside: <ncx>

  Comments: There is no required relationship between the content of docTitle in the NCX and the content of the doctitle element in the textual content file, if it exists.

- **<docAuthor>**

  Description: The author of the document, presented as text and, optionally, in audio or image renderings, for presentation to the reader.

  Declaration: <!ELEMENT docAuthor (text, audio?, img?)>

  Syntax: <docAuthor ...attributes... >...content...</docAuthor>

  Attributes:

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

  Valid inside: <ncx>

  Comments: There is no required relationship between the content of docAuthor in the NCX and the content of the docauthor element(s) in the textual content file, if it exists.

- **<text>**

  Description: Contains the text of a <docTitle> or <docAuthor> or text content of a <navLabel> or <navInfo>.

  Declaration: <!ELEMENT text (#PCDATA)>

  Syntax: <text ...attributes...>...content...</text>

  Attributes:

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

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

  Valid inside: <navLabel>, <docTitle>, <docAuthor>, <navInfo>

- **<audio>**

  Description: Contains a pointer to an audio clip of a <docTitle> or <docAuthor>, or of the audio content of a <navLabel> or <navInfo>.

  Declaration: <!ELEMENT audio EMPTY>

  Syntax: <audio...attributes... />

  Attributes:

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

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

    o **src (CDATA, REQUIRED):** The URI of the audio media object. Ordinarily, this will point to an audio file containing the content of the DTB. However, when a DTB spans several media units, the URI can point to a file containing a clip of the heading or title referenced. See section 8.4.2, "DTBs Spanning Multiple Media Units."

    o **clipBegin (%SMILtimeVal, REQUIRED):** 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 [SMIL]. See section 7.7, "Media Clipping and Clock Values."

    o **clipEnd (%SMILtimeVal, REQUIRED):** 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: <navLabel>, <docTitle>, <docAuthor>, <navInfo>

- **<img>**

  Description: Contains a pointer to graphical content associated with a <docTitle> or <docAuthor>, or of a <navLabel> or <navInfo>.

  Declaration: <!ELEMENT img EMPTY>

  Syntax: <img...attributes... />

  Attributes:

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

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

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

  Valid inside: <docTitle>, <docAuthor>, <navLabel>, <navInfo>

- **<navMap>**

  Description: Container for primary navigation information.

  Declaration: <!ELEMENT navMap (navInfo*, navLabel*, navPoint+)>

  Syntax: <navMap ...attributes...>...content...</navMap>

  Attributes:

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

  Valid inside: <ncx>

Comments: The `navMap` element contains the primary navigation information, pointing to each of the major structural elements of the document. Page numbers are contained in `pageList`. Other secondary navigation elements, such as footnotes, are not included in `navMap`, but are contained in `navLists`.

- **<navPoint>**

  Description: Contains description(s) of target and pointer to content.

  Declaration: `<!ELEMENT navPoint (navLabel+, content, navPoint*)>`

  Syntax: `<navPoint ...attributes...>...content...</navPoint>`

  Attributes:
  - o **id (ID, REQUIRED):** Unique identifier.
  - o **class (CDATA, IMPLIED):** Optional descriptor of this instance of the element. Can be used to select a presentation from the resource file. See section 10.3, "Resource File Requirements."
  - o **playOrder (CDATA, REQUIRED):** Positive integer denoting the location of the content of this `navPoint` in the default playing sequence. See section 8.4.3, "playOrder Attribute."

  Valid inside: `<navMap>, <navPoint>`

  Comments: The `navPoint` element contains one or more `navLabels`, representing the referenced part of the document, e.g. chapter title or section number, along with a pointer to content. `<navPoint>`s may be nested to represent the hierarchical structure of a document.

- **<navLabel>**

  Description: Contains a label identifying a given `<navMap>, <navPoint>, <pageList>, <pageTarget>, <navList>,` or `<navTarget>` in various media for presentation to the user. When applied to `<navPoint>`s, it generally contains the heading of the referenced section of the document. Can be repeated so labels can be provided in multiple languages.

  Declaration: `<!ELEMENT navLabel (((text, audio?) | audio), img?)>`

  Syntax: `<navLabel ...attributes...>...content...</navLabel>`

  Attributes:
  - o **xml:lang (NMTOKEN, IMPLIED):** Specifies the [RFC 3066] language code of the language of the heading. Can be used to control presentation of headings in a specified language.
  - o **dir ((rtl|ltr), IMPLIED):** Specifies the direction of the text, where ltr is left to right and rtl is right to left.

  Valid inside: `<navMap>, <navPoint>, <pageList>, <pageTarget>, <navList>, <navTarget>`

- **<navInfo>**

  Description: Contains an informative comment about a `<navMap>, <pageList>,` or `<navList>` in various media for presentation to the user. Can be repeated so comments can be provided in multiple languages.

  Declaration: `<!ELEMENT navInfo (((text, audio?) | audio), img?)>`

Syntax: `<navInfo ...attributes...>...content...</navInfo>`

Attributes:

- o **xml:lang (NMTOKEN, IMPLIED):** Specifies the [RFC 3066] language code of the language of the comment. Can be used to control presentation of comments in a specified language.

- o **dir ((rtl|ltr), IMPLIED):** Specifies the direction of the text, where ltr is left to right and rtl is right to left.

Valid inside: `<navMap>`, `<pageList>`, `<navList>`

Comment: While `navLabel` contains a brief identifying label for a `navMap`, `pageList`, or `navList`, `navInfo` is used to present longer, explanatory or informative text regarding the structure or content of these navigation features.

- **<content>**

  Description: Pointer into SMIL file to beginning of the item referenced by the `navPoint` or `navTarget`.

  Declaration: `<!ELEMENT content EMPTY>`

  Syntax: `<content...attributes... />`

  Attributes:

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

  - o **src (CDATA, REQUIRED):** The URI of the SMIL time container corresponding to the start of the referenced part of the document.

  Valid inside: `<navPoint>, <navTarget>`

- **<pageList>**

  Description: Container for pagination information.

  Declaration: `<!ELEMENT pageList (navInfo*, navLabel*, pageTarget+)>`

  Syntax: `<pageList ...attributes...>...content...</pageList>`

  Attributes:

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

  - o **class (CDATA, IMPLIED):** Optional descriptor of the element. Can be used to select a presentation from the resource file. See section 10.3, "Resource File Requirements."

  Valid inside: `<ncx>`

  Comments: The `pageList` element contains navigation information for pages within `pageTarget`s. Each navigable page within the book will be represented by a pageTarget within the pageList.

- **<pageTarget>**

  Description: Container for text, audio, image, and content elements containing navigational information for pages.

  Declaration: `<!ELEMENT pageTarget (navLabel+, content)>`

  Syntax: `<pageTarget ...attributes...>...content...</pageTarget>`

Attributes:

- o **id (ID, REQUIRED):** Unique identifier.
- o **value (CDATA, IMPLIED):** A positive integer representing the numeric value associated with a page. Combination of values of `type` and `value` attributes must be unique, when value attribute is present. See section 8.4.5 "Enabling Page Navigation."
- o **type (front | normal | special) REQUIRED:** Describes the kind of page represented by this `pageTarget`. Value must be one of: "front" (for roman-numeral pages at the start of a book), "normal" (for pages identified by arabic numerals), or "special" (for all other kinds of pages).
- o **class (CDATA, IMPLIED):** Optional descriptor of this instance of the element. Can be used to select a presentation from the resource file. See section 10.3, "Resource File Requirements."
- o **playOrder (CDATA, REQUIRED):** Positive integer denoting the location of the content of this `pageTarget` in the default playing sequence. See section 8.4.3, "playOrder Attribute."

Valid inside: `<pageList>`

- **<navList>**

  Description: Container for secondary navigational information.

  Declaration: `<!ELEMENT navList (navInfo*, navLabel+, navTarget+)>`

  Syntax: `<navList ...attributes...>...content...</navList>`

  Attributes:

  - o **id (ID, IMPLIED):** Optional unique identifier.
  - o **class (CDATA, IMPLIED):** Optional descriptor of this instance of the element. Can be used to select a presentation from the resource file. See section 10.3, "Resource File Requirements."

  Valid inside: `<ncx>`

  Comments: The `navList` element contains secondary navigation information within `navTarget`s. It is similar to `navMap` except `navTarget`s may not nest, whereas `navPoint`s can. Used for lists of elements such as 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 (navLabel+, content)>`

  Syntax: `<navTarget ...attributes...>...content...</navTarget>`

  Attributes:

  - o **id (ID, REQUIRED):** Unique identifier.
  - o **value (CDATA, IMPLIED):** A positive integer representing the numeric value associated with a `navTarget`. Useful for providing integer values for items within the `navList`.

- o **class (CDATA, IMPLIED):** Optional descriptor of this instance of the element. Can be used to select a presentation from the resource file. See section 10.3, "Resource File Requirements."

- o **playOrder (CDATA, REQUIRED):** Positive integer denoting the location of the content of this `navTarget` in the default playing sequence. See section 8.4.3, "playOrder Attribute."

Valid inside: `<navList>`

Comments: The `navTarget` element contains one or more `navLabel`s representing the referenced part of the document, e.g., a footnote, along with a pointer to `content`.

8.4　Other File Requirements

This section collects other normative requirements for the NCX file that cannot be enforced by the DTD.

8.4.1　Navigation Metadata

(This section is normative.)

Metadata shall be included in the head element of the NCX using the `meta` element. Content producers may introduce other metadata besides those listed below, if needed. However, metadata names shall not begin with the prefix "dtb:" unless defined in this standard. Players must not fail when encountering unknown metadata but must, at a minimum, ignore it.

- **dtb:uid**

  - o Content: The globally unique identifier for the DTB. The value is the same as that of the `dc:Identifier` element referenced by the `unique-identifier` attribute on the package file's `package` element. See section 3.1, "Package Identity."

  - o Occurrence: Required

- **dtb:depth**

  - o Content: Positive integer indicating depth of structure of the DTB as exposed by the NCX.

  - o Occurrence: Required

  - o dtb:generator

  - o Content: Name and version of software that generated the NCX.

  - o Occurrence: Optional - recommended.

- **dtb:totalPageCount**

  - o Content: Non-negative integer indicating the number of `pageTarget`s in the `pageList`. If there are no navigable pages, then dtb:totalPageCount must have a value of zero.

  - o Occurrence: Required

- **dtb:maxPageNumber**

  - o Content: Non-negative integer indicating the largest `value` attribute on `pageTarget` in the `pageList`. If there are no navigable pages, then dtb:maxPageNumber must have a value of zero.

  - o Occurrence: Required

8.4.2 DTBs Spanning Multiple Media Units

(This section is normative.)

When a DTB spans several distribution media (e.g., multiple CD-ROMs), the full NCX along with all audio clips and images directly referenced by it must be included on every media unit. This will ensure that the entire NCX will function properly on each piece of media.

8.4.3 playOrder Attribute

(This section is normative.)

The `playOrder` attribute is required on each `pageTarget`, `navTarget` and `navPoint`. It provides a means to collate all `pageTarget`s, `navTarget`s, and `navPoint`s into a single ordered sequence that reflects their order in the normal playback sequence of the book as presented in the `spine` and SMIL file(s). `playOrder` is a positive integer; the first `playOrder` value in a document shall be 1. When the `content` elements of any `pageTarget`s, `navTarget`s, or `navPoint`s reference the same SMIL time container, they must have the same `playOrder` value. `playOrder` must increase by one for each unique SMIL time container referenced by any `pageTarget`, `navTarget` or `navPoint`.

8.4.4 smilCustomTest Element

(This section is normative.)

Each unique `customTest` element that appears in one or more SMIL files and has been referenced at least once by a `customTest` attribute must have three of its attributes (`id`, `override`, and `defaultState`) duplicated in a `smilCustomTest` element in the `head` of the NCX. In addition, when the `customTest` element in SMIL has been applied to a time container holding one of the book structures defined below, the `bookStruct` attribute must be applied to the corresponding `smilCustomTest` and contain the appropriate value from the list below.

The special vocabulary below is provided to identify the semantics of the skippable elements listed. Producers are required to provide Resource File entries for all skippable elements. See section 7.4.3, "'Skippable' Structures" and Example 10.1. Players that are not capable of using the Resource File must provide text (if supported) and audio labels for each of the items listed here. Players capable of utilizing the Resource File must render resources provided with the DTB in preference to labels embedded in the player.

LINE_NUMBER
The number of a line in a poem, legal work, or other document.

NOTE
A comment, explanation, or reference placed apart from the text of a document (called "footnote" if placed at the bottom of the page, or "endnote" or "note" if placed at the end of the chapter or book.)

NOTE_REFERENCE
A mark or character identifying a specific note.

ANNOTATION
A comment or explanation that differs from NOTE in that it is usually set in the margin or on a facing page, often with no explicit reference to it inserted in the text.

PAGE_NUMBER
The number of a page.

OPTIONAL_PRODUCER_NOTE
A comment or explanation added by the DTB producer that is commonly used to provide descriptions of visual elements or describe differences between the print book and the audio

version. OPTIONAL_PRODUCER_NOTE must not contain warnings or cautions about hazards.

OPTIONAL_SIDEBAR
Information supplementary to the main text and/or narrative flow that is often boxed and printed apart from the main text block on a page. OPTIONAL_SIDEBAR must not contain warnings or cautions about hazards.

8.4.5 Enabling Page Navigation

(This section is normative.)

In order to allow direct numeric access to pages, it is strongly recommended that the `value` attribute be applied to all `pageTarget`s. `value` is a positive integer. The combination of values for the `type` and `value` attributes must be unique, when the `value` attribute is present.

When a page number exists in the original document, it is strongly recommended that the `text` element in the `navLabel` on the corresponding `pageTarget` contain the textual representation of the page number. In documents where alphanumeric page numbers are present, players may use the content of the `text` element of the `navLabel` associated with the `pageTarget` for direct access to the page. The `navLabel` should be treated as the authoritative rendering of the page identifier.

8.5   How the NCX Works

(This section is informative.)

Upon opening a DTB, a player will ordinarily use the NCX `navMap` to define the user's choices for navigation. The `navMap` contains nested `navPoint`s that represent the major divisions of the document. For example, the structure of the book whose NCX is shown in section 8.6, Example 8.1, would look like this:

- Foreword..(Level 1)
    - o   History......................................(Level 2)
    - o   Development of Standards.........(Level 2)
- Standards..(Level 1)
    - o   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 `navPoint`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 Level 3 under Foreword, no smaller divisions can be accessed from the NCX. Such smaller divisions may be present, but they can only be reached through . The division of Standards marked "a." is at Level 4, 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 `pageList` (for pages) and `navList` (for all other non-hierarchical objects). Unlike

the `navMap`, `pageList` and `navList` do not allow nesting. Example 8.1 shows a `pageList` containing three `pageTarget`s representing page numbers, and a `navList` containing three `navTarget`s representing notes.

The `navInfo` element allows producers to describe in some detail the contents, purpose, and use of the `navMap`, `pageList`, and any `navList`s. See the `navList` in example 8.1 below.

Each `navPoint`, `pageTarget`, or `navTarget` provides navigation information about one piece of the document, e.g., a chapter heading, section number, page number, figure, etc. The `navLabel` element provides the content of the heading, page number, figure title, etc. in multiple media. Within a `navLabel`, the `text` element contains the actual heading, page number, etc. for visual, braille 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 are used to give location feedback to the user. The `content` element provides a pointer to an ID within a SMIL file that marks the beginning of the referenced portion of the DTB.

The required `playOrder` attribute on `pageTarget`, `navTarget` and `navPoint` allows synchronization of the `pageList` and `navList`s with the `navMap`. In determining what to say in response to "Where am I?" requests by the user, a player can find the current `navMap` location by finding the `navPoint` with the largest `playOrder` attribute value less than or equal to the value of the `playOrder` attribute for the most recently traversed `navPoint`, `pageTarget` or `navTarget`. Similarly, a player can determine the current page by searching for `pageTarget`s having equal or lower `playOrder` than that of the most recently traversed `navPoint`, `pageTarget` or `navTarget`.

This standard offers producers the ability to gather in the `head` of the NCX information on all skippable elements from the SMIL file(s). (See section 7.4.3, "'Skippable' Structures.") The `smilCustomTest` element may be repeated to list all skippable elements and their defaultStates. Playback systems may use this information to inform readers of their options and current settings for skippable structures.

8.6   Example

(This example is informative.)

Example 8.1:

```
<?xml version="1.0"  encoding="UTF-8"?>
<!DOCTYPE ncx PUBLIC "-//NISO//DTD ncx 2005-1//EN"
  "http://www.daisy.org/z3986/2005/ncx-2005-1.dtd">
<ncx version="2005-1" xml:lang="en"
xmlns="http://www.daisy.org/z3986/2005/ncx/">
  <head>
    <smilCustomTest id="pagenum" defaultState="false"
    override="visible" bookStruct="PAGE_NUMBER"/>
    <smilCustomTest id="note" defaultState="true"
    override="visible" bookStruct="NOTE"/>
    <meta name="dtb:uid" content="us-nls-00001"/>
    <meta name="dtb:depth" content="6"/>
    <meta name="dtb:generator" content="NLSv001"/>
    <meta name="dtb:totalPageCount" content="53"/>
    <meta name="dtb:maxPageNumber" content="49"/>
  </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" clipBegin="00:00.00" clipEnd="00:09.04"/>
        <img src="rs_title.png" />
</docTitle>
<docAuthor>
        <text>Association of Specialized and Cooperative
        Library Agencies</text>
        <audio src="rs_title.mp3" clipBegin="00:09.50" clipEnd="00:14.70"/>
</docAuthor>
<navMap>
    <navPoint class="chapter" id="lvl1_3" playOrder="2">
      <navLabel>
        <text>Foreword</text>
        <audio src="rs_fwdx.mp3" clipBegin="00:01.50"
        clipEnd="00:02.00" />
      </navLabel>
      <content src="sample.smil#h1_3" />
      <navPoint class="section" id="lvl2_1" playOrder="3">
        <navLabel>
          <text>History</text>
          <audio src="rs_fwdx.mp3" clipBegin="00:03.40"
          clipEnd="00:03.90" />
        </navLabel>
        <content src="sample.smil#h2_1" />
      </navPoint>
      <navPoint class="section" id="lvl2_2" playOrder="5">
        <navLabel>
          <text>Development of Standards</text>
          <audio src="rs_fwdx.mp3" clipBegin="00:56.30"
          clipEnd="00:57.70" />
        </navLabel>
        <content src="sample.smil#h2_2" />
      </navPoint>
    </navPoint>
    <navPoint class="chapter" id="lvl1_7" playOrder="10">
      <navLabel>
        <text>Standards</text>
        <audio src="rs_stdx.mp3" clipBegin="00:01.30"
        clipEnd="00:02.10" />
      </navLabel>
      <content src="sample.smil#h1_7" />
      <navPoint class="section" id="lvl2_11" playOrder="11">
        <navLabel>
          <text>1 Core Services</text>
          <audio src="rs_stdx.mp3" clipBegin="00:02.90"
          clipEnd="00:04.90" />
        </navLabel>
        <content src="sample.smil#h2_10" />
        <navPoint class="subsection" id="lvl3_1" playOrder="12">
          <navLabel>
            <text>1.1</text>
            <audio src="rs_stdx.mp3" clipBegin="00:05.70"
            clipEnd="00:06.70" />
          </navLabel>
          <content src="sample.smil#h3_1" />
          <navPoint class="sub-subsection" id="lvl4_1" playOrder="13">
            <navLabel>
                <text>a.</text>
```

```
                      <audio src="rs_stdx.mp3" clipBegin="00:18.70"
                        clipEnd="00:19.10" />
                    </navLabel>
                    <content src="sample.smil#h4_1" />
                  </navPoint>
                </navPoint>
                <navPoint class="subsection" id="lvl3_2" playOrder="14">
                  <navLabel>
                    <text>1.2</text>
                    <audio src="rs_stdx.mp3" clipBegin="00:50.50"
                    clipEnd="00:51.40" />
                  </navLabel>
                  <content src="sample.smil#h3_2" />
                </navPoint>
              </navPoint>
            </navPoint>
          . . .
      </navMap>
      <pageList id="pages">
        <navLabel>
            <text>Pages</text>
            <audio src="navlabels.mp3" clipBegin="00:00.00"
            clipEnd="00:01.10" />
        </navLabel>
        <pageTarget class="pagenum" type="normal" id="p1" value="1"
playOrder="1">
            <navLabel>
              <text>1</text>
              <audio src="rs_fwdx.mp3" clipBegin="00:00.00"
              clipEnd="00:00.90" />
            </navLabel>
            <content src="sample.smil#p1" />
        </pageTarget>
        <pageTarget class="pagenum" type="normal" id="p2" value="2"
playOrder="4">
            <navLabel>
              <text>2</text>
              <audio src="rs_fwdx.mp3" clipBegin="00:53.90"
              clipEnd="00:54.60" />
            </navLabel>
            <content src="sample.smil#p2" />
        </pageTarget>
        <pageTarget class="pagenum" type="normal" id="p3" value="3"
playOrder="9">
            <navLabel>
              <text>3</text>
              <audio src="rs_stdx.mp3" clipBegin="00:00.00"
              clipEnd="00:00.70" />
            </navLabel>
            <content src="sample.smil#p3" />
        </pageTarget>
        . . .
      </pageList>
      <navList id="notes" class="note">
        <navInfo>
            <text>This list contains the three notes found in this book.
                  Each entry in the list, numbered 1 through 3, points to a
```

```
note reference.
      </text>
      <audio src="rs_info.mp3" clipBegin="00:00.00"
      clipEnd="00:05.592" />
   </navInfo>
   <navLabel>
      <text>Notes</text>
      <audio src="navlabels.mp3" clipBegin="00:01.50"
      clipEnd="00:02.60" />
   </navLabel>
   <navTarget class="note" id="nref_1" playOrder="6">
     <navLabel>
      <text>1</text>
      <audio src="rs_fwdx.mp3" clipBegin="01:22.60"
      clipEnd="01:23.50" />
     </navLabel>
     <content src="sample.smil#nref_1" />
   </navTarget>
   <navTarget class="note" id="nref_2" playOrder="7">
     <navLabel>
      <text>2</text>
      <audio src="rs_fwdx.mp3" clipBegin="02:00.60"
      clipEnd="02:01.40" />
     </navLabel>
     <content src="sample.smil#nref_2" />
   </navTarget>
   <navTarget class="note" id="nref_3" playOrder="8">
     <navLabel>
      <text>3</text>
      <audio src="rs_fwdx.mp3" clipBegin="03:13.30"
      clipEnd="03:14.10" />
     </navLabel>
     <content src="sample.smil#nref_3" />
   </navTarget>
  </navList>
</ncx>
```

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.

This standard does not require that 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 notes, the player must format the information as a valid XML file conforming to bookmark-2005-1.dtd.(See Appendix 1, "DTD for Portable Bookmarks/Highlights.") Similarly, a player with

bookmark/highlight import capabilities must correctly process bookmarks and highlights and their associated notes that are formatted in accordance with bookmark-2005-1.dtd.

Export-capable players must be able to set bookmarks and highlight starts and ends at any point in a DTB, whether based on the position in the audio presentation or the textual content file. That is, players shall not be limited to capturing location information only at element boundaries. Offsets from element boundaries in the audio presentation shall be identified by `<timeOffset>`, with a clock value relative to the start of the SMIL time container referenced in the URI. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module [SMIL]. See section 7.7, "Media Clipping and Clock Values." Offsets from element boundaries in textual content files shall be identified by `<charOffset>`, measured in characters, counting from the start of the content of the referenced element; start- and end-tags are not counted, white space is then normalized (collapsed to one character).

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. When generating the filename for a note, the playback device must generate a filename extension appropriate to the recording format. (See Section 5, "Audio File Formats" for supported formats and Section 3.3, "Manifest" for filename extension requirements.)

Bookmark files (which may include highlights) shall be named, by default, with the value from the bookmark element `uid` and the extension ".bmk". For example: "se-tpb-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 a DTB. It is recommended that if more than one bookmark file is present for a given DTB, players allow the user to choose among them.

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. However, when preparing a bookmark file for export, players must sort the bookmarks and highlights into document order and write them in that order.

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 (see Appendix 1), 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:

  o **xmlns (CDATA, FIXED) "http://www.daisy.org/z3986/2005/bookmark/":** Specifies the default XML namespace for all elements in the bookmark file. See [XML-Namespaces] for details on namespaces. This attribute and its value (given in DTD) must be explicitly specified in the document instance.

  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 of DTB or of user-recorded note, in any format supported by standard. Title clip enables user to identify desired bookmark file if several are present.

  Declaration: `<!ELEMENT audio EMPTY >`

  Syntax: `<audio`...attributes... `/>`

  Attributes:

  - o **src (%URI, #REQUIRED):** The src attribute holds the URI of the audio file that contains the referenced clip.

  - o **clipBegin (%SMILtimeVal, 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 [SMIL]. See section 7.7, "Media Clipping and Clock Values."

  - o **clipEnd (%SMILtimeVal, 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>`

- **<uid>**

  Description: Globally unique identifier for the book, drawn from the package file. Matches the `dc:Identifier` referenced by the "unique-identifier" attribute on the package file's `package` element. See section 3.1, "Package Identity."

  Declaration: `<!ELEMENT uid (#PCDATA) >`

  Syntax: `<uid>`...content...`</uid>`

Attributes: None

Valid inside: `<bookmarkSet>`

- **<lastmark>**

  Description: Location where user most recently ceased reading 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 and/or character offset to the exact point.

  Declaration: `<!ELEMENT lastmark (ncxRef, URI, ((timeOffset, charOffset?)| 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 navPoint) 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>, <hiliteStart>, <hiliteEnd>`

- **<URI>**

  Description: Pointer to id of `<par>` or `<seq>` in SMIL that contains the `<lastmark>`, `<bookmark>, <hiliteStart>,` or `<hiliteEnd>`.

  Declaration: `<!ELEMENT URI (#PCDATA)>`

  Syntax: `<URI>`...content...`</URI>`

  Attributes: None

  Valid inside: `<lastmark>, <bookmark>, <hiliteStart>, <hiliteEnd>`

- **<timeOffset>**

  Description: Exact position of `<lastmark>, <bookmark>, <hiliteStart>,` or `<hiliteEnd>` in the sequential audio presentation; a non-negative clock value, relative to the start of the SMIL time container referenced in the URI. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module [SMIL]. See section 7.7, "Clock Values." .

  Declaration: `<!ELEMENT timeOffset (#PCDATA) >`

  Syntax: `<timeOffset>`...content (see section 7.7, "Clock Values for syntax)...`</timeOffset>`

  Attributes: None

Valid inside: `<lastmark>`, `<bookmark>`, `<hiliteStart>`, `<hiliteEnd>`

- **<charOffset>**

  Description: Exact position of `bookmark`, `lastmark`, `hiliteStart`, or `hiliteEnd` in textual content file referenced (via SMIL) by the URI. See section 9.1, "Introduction" for information on calculating `<charOffset>` values.

  Declaration: `<!ELEMENT charOffset (#PCDATA) >`

  Syntax: `<charOffset>`...content...`</charOffset>`

  Attributes: None

  Valid inside: `<lastmark>`, `<bookmark>`, `<hiliteStart>`, `<hiliteEnd>`

- **<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 and/or character offset to the exact point.

  Declaration: `<!ELEMENT bookmark (ncxRef, URI, ((timeOffset, charOffset?)| charOffset)), note?) >`

  Syntax: `<bookmark>`...content...`</bookmark>`

  Attributes:

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

  - o **xml:lang (%languagecode, #IMPLIED):** optional attribute for use in identifying the language of label and note for this bookmark, using an [RFC 3066] language code.

  Valid inside: `<bookmarkSet>`

- **<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 (hiliteStart, hiliteEnd, note?) >`

  Syntax: `<hilite>`...content...`</hilite>`

Attributes:

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

Valid inside: `<bookmarkSet>`

- **<hiliteStart>**

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 and/or character offset to the exact point.

Declaration: `<!ELEMENT hiliteStart (ncxRef, URI, ((timeOffset, charOffset?)| charOffset)) >`

Syntax: `<hiliteStart>`...content...`</hiliteStart>`

Attributes: None

Valid inside: `<hilite>`

- **<hiliteEnd>**

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 and/or character offset to the exact point.

Declaration: `<!ELEMENT hiliteEnd (ncxRef, URI, ((timeOffset, charOffset?)| charOffset)) >`

Syntax: `<hiliteEnd>`...content...`</hiliteEnd>`

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, 1 minute and 28 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 4 minutes and 6 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-rfbd-JT065.bmk".

Example 9.1:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE bookmarkSet PUBLIC "-//NISO//DTD bookmark 2005-1//EN"
  "http://www.daisy.org/z3986/2005/bookmark-2005-1.dtd">
 <bookmarkSet xmlns="http://www.daisy.org/z3986/2005/bookmark/">
    <title>
      <text>Gone with the Wind</text>
      <audio src="gwtw_title.mp3" />
    </title>
    <uid>us-rfbd-JT065</uid>
```

```
      <lastmark>
        <ncxRef>gwtw.ncx#lvl1_5</ncxRef>
        <URI>gwtw_ch5.smil#para023</URI>
        <timeOffset>03:52.00</timeOffset>
      </lastmark>

      <bookmark>
        <ncxRef>gwtw.ncx#lvl1_1</ncxRef>
        <URI>gwtw_ch1.smil#para008</URI>
        <timeOffset>00:22.00</timeOffset>
      </bookmark>

      <bookmark>
        <ncxRef>gwtw.ncx#lvl1_3</ncxRef>
        <URI>gwtw_ch3.smil#para012</URI>
        <timeOffset>01:28.00</timeOffset>
        <note>
          <text>Atlanta burns.</text>
        </note>
      </bookmark>

      <hilite>
        <hiliteStart>
          <ncxRef>gwtw.ncx#lvl1_4</ncxRef>
          <URI>gwtw_ch4.smil#para001</URI>
          <timeOffset>00:00.00</timeOffset>
        </hiliteStart>
        <hiliteEnd>
          <ncxRef>gwtw.ncx#lvl1_4</ncxRef>
          <URI>gwtw_ch4.smil#para006</URI>
          <timeOffset>04:06.00</timeOffset>
        </hiliteEnd>
        <note>
          <audio src="us-rfbd-JT065.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:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE bookmarkSet PUBLIC "-//NISO//DTD bookmark 2005-1//EN"
  "http://www.daisy.org/z3986/2005/bookmark-2005-1.dtd">
<bookmarkSet xmlns="http://www.daisy.org/z3986/2005/bookmark/">

  <title>
    <text>Chemistry Today</text>
  </title>
  <uid>uk-rnib-MM499</uid>

  <lastmark>
    <ncxRef>chemtd.ncx#lvl1_3</ncxRef>
```

```
            <URI>chemtd.smil#para297</URI>
            <charOffset>130</charOffset>
         </lastmark>

      </bookmarkSet>
```

Example 9.3 shows a lastmark for a text and audio book with both timeOffset and charOffset.

Example 9.3:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE bookmarkSet PUBLIC "-//NISO//DTD bookmark 2005-1//EN"
  "http://www.daisy.org/z3986/2005/bookmark-2005-1.dtd">
<bookmarkSet  xmlns="http://www.daisy.org/z3986/2005/bookmark/">
   <title>
      <text>Physics Yesterday</text>
   </title>
   <uid>uk-rnib-MM498</uid>

   <lastmark>
     <ncxRef>physyd.ncx#lvl1_5</ncxRef>
     <URI>physyd.smil#para_573</URI>
     <timeOffset>02:01.00</timeOffset>
     <charOffset>250</charOffset>
   </lastmark>

</bookmarkSet>
```

10   Resource File

10.1   Introduction

(This section is informative.)

The Resource File supplies text and/or audio segments and optional images, that can assist the reader in using a DTB. Its use is optional, unless escapable or skippable structures are enabled in the DTB (see sections 7.4.1 and 7.4.3). These media objects or "resources" provide information missing from a document or present only in a form inaccessible to the reader. A resource typically consists of meaningful semantically rich information. A resource can be supplied in multiple languages simultaneously, allowing a player or user to determine in which language the resource should be rendered. Some examples of applications are:

1) Documents with definite structures but missing headings, such as books with multilevel structures but no navigation labels 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 be used to present the actual name of items found at that level in that specific context, e.g., "chapter".

3) "Where Am I?" applications.

4) Detailed information about the semantics of the current position in the textual content file; useful when exact knowledge of a document's finest structure is essential. The resource

file could provide text or audio clips of element names in the textual content file grammar, 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 can choose to "turn off" (see section 7.4.3, "'Skippable' Structures") or "escape" (see section 7.4.1, "'Escapable' Structures") via the SMIL file. The Resource File would contain text segments or pointers to audio clips of the names of the structures affected; for example, page numbers, notes, or tables.

The resource file uses a subset of W3C XPath 1.0 (see [XPath10]) to associate resources with arbitrary sets of elements in the DTB file set. For players that do not include a dedicated XPath processor, the subset is strict enough to allow the use of concatenated string matching to establish an association.

Resources would be called only when appropriate; that is, in response to explicit user requirements/settings, and when needed.

10.2   Resource Elements

(This section is informative.)

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

- **<resources>**

  Description: The root element in the Resource DTD.

  Declaration: `<!ELEMENT resources (head?, scope+) >`

  Syntax: `<resources...attributes...>...content...</resources>`

  Attributes:

    o **version (CDATA, FIXED): "2005-1"** Specifies the version of the DTD used in this instance. This attribute and its value (given in DTD) must be explicitly specified in the document instance.

    o **xmlns (CDATA, FIXED) "http://www.daisy.org/z3986/2005/resource/":** Specifies the default XML namespace for all elements in the Resource File. See [XML-Namespaces] for details on namespaces. This attribute and its value (given in DTD) must be explicitly specified in the document instance.

    o **id (ID, IMPLIED):** Optional identifier.

  Valid inside: None

- **<head>**

  Description: Optional container for metadata.

  Declaration: `<!ELEMENT head (meta*) >`

  Syntax: `<head>...content...</head>`

  Attributes: None

  Valid inside: `<resources>`

- **<meta>**

  Description: Producer-defined metadata.

  Declaration: `<!ELEMENT meta EMPTY >`

  Syntax: `<meta...attributes.../>`

Attributes:

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

Valid inside: `<head>`

- **<scope>**

  Description: Container for resources associated with a specific vocabulary, identified by its namespace.

  Declaration: `<!ELEMENT scope (nodeSet+)>`

  Syntax: `<scope...attributes...>...content...</scope>`

  Attributes:

  - o **nsuri (%URI;, REQUIRED):** A namespace URI defining to which namespace the `<resource>` descendants of this `<scope>` element apply.
  - o **id (ID, IMPLIED):** Optional identifier.

  Valid inside: `<resources>`

- **<nodeSet>**

  Description: Selector of a set of elements within the namespace defined by the `scope` parent.

  Declaration: `<!ELEMENT nodeSet (resource+)>`

  Syntax: `<nodeSet...attributes...>...content...</nodeSet>`

  Attributes:

  - o **select (%XPathSubset, REQUIRED):** W3C XPath 1.0 string, expressing the element set selection. XPath subset is defined in section 10.3.
  - o **id (ID, REQUIRED):** Required identifier.

  Valid inside: `<scope>`

- **<resource>**

  Description: Contains text and pointers to audio clip and/or image serving as labels to be applied to elements in the containing nodeSet.

  Declaration: `<!ELEMENT resource (((text, audio?) | audio), img?)>`

  Syntax: `<resource...attributes...>...content...</resource>`

  Attributes:

  - o **xml:lang (%languagecode;, REQUIRED):** Specifies the language of the resource item, using an [RFC 3066] language code.
  - o **id (ID, REQUIRED):** Required identifier.

  Valid inside: `<nodeSet>`

- **<text>**

  Description: Contains the text used for label.

  Declaration: `<!ELEMENT text (#PCDATA) >`

Syntax: `<text>`...content...`</text>`

Attributes:

- o **id (ID, IMPLIED):** Optional identifier.

- o **dir ((ltr|rtl), IMPLIED):** Text directionality.

Valid inside: `<resource>`

- **<audio>**

    Description: Points to file containing audio label and provides time offsets of beginning and end of clip.

    Declaration: `<!ELEMENT audio EMPTY >`

    Syntax: `<audio`...attributes... `/>`

    Attributes:

    - o **src (%URI, REQUIRED):** URI of the audio file.

    - o **clipBegin (%SMILtimeVal, REQUIRED):** 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 [SMIL]. See section 7.7, "Media Clipping and Clock Values."

    - o **clipEnd (%SMILtimeVal, REQUIRED):** 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`.

    - o **id (ID, IMPLIED):** Optional identifier.

    Valid inside: `<resource>`

- **<img>**

    Description: Points to file containing the image.

    Declaration: `<!ELEMENT img EMPTY >`

    Syntax: `<img`...attributes...`/>`

    Attributes:

    - o **src (%URI, REQUIRED):** URI of the image file.

    - o **id (ID, IMPLIED):** Optional identifier.

    Valid inside: `<resource>`

10.3 Resource File Requirements

(This section is normative.)

If a Resource File is implemented, it must meet the following requirements. The Resource File is a valid XML1.0 file conforming to the Document Type Definition resource-2005-1.dtd (see Appendix 1, "DTD for Resource File"). The `version` and `xmlns` attributes on the `resources` element must be explicitly specified in the document instance, using values drawn from the above-named DTD. Entity declarations must occur in the internal DTD subset. See further section 16.1 "General File Conformance Requirements." If a DTB spans multiple media units, identical copies of the Resource File and copies of all audio and image files directly referenced by it shall be distributed on each media unit of the DTB.

The varying media children of the `<resource>` element must be informational equivalents. `<resource>` children of a certain `<nodeSet>` element must be informational equivalents but in different languages. Within any given `<nodeSet>`, there must not be two resources with the same language. The namespace URI (the value of the `nsuri` attribute) must be unique for each <scope> element.

The W3C XPath 1.0 (see [XPath10]) subset strings, used in the `select` attribute, must conform to the following rules. The strings must be valid XPath 1.0 strings and must use the XPath 1.0 abbreviated syntax only. The only location path allowed is `/descendant-or-self::node()/`, in abbreviated syntax expressed as `//`. The node test may only select element names. Wildcard is allowed as an element selector in node test, but not in predicates. Zero or several predicates are allowed. Predicates can only select attributes, and must use explicit attribute `name` and attribute value. Attribute values in predicates are delimited by double or single quotes (" or '). Attribute values containing double quotes must be delimited by single quotes, and attribute values containing single quotes must be delimited by double quotes. Attribute values containing both double and single quotes are not allowed. Predicates must be singleton, that is, neither and nor or statements are allowed in predicates. Selection of elements and attributes must use the qname localpart only (see [XML-Namespaces]).

Below is the XPath1.0 subset defined in EBNF form:

```
DTBLocationPath   ::=   '//' Step
Step              ::=   NodeTest Predicate*
NodeTest          ::=   '*' | QNameLocalPart
Predicate         ::=   '[' PredicateExpr ']'
PredicateExpr     ::=   Expr
Expr              ::=   EqualityExpr
EqualityExpr      ::=   AttrSpecifier '=' Literal
AttrSpecifier     ::=   '@' QNameLocalPart
Literal           ::=   '"' [^"]* '"'  | "'" [^']* "'"
QNameLocalPart    ::=   NCName
NCName            ::=   (Letter | '_') (NCNameChar)
NCNameChar        ::=   Letter | Digit | '.' | '-' | '_' |
CombiningChar | Extender
```

where Letter, CombiningChar, Digit and Extender are defined in the XML 1.0 specification (see http://www.w3.org/TR/REC-xml#NT-Letter).

Within a particular vocabulary (as defined by the scope element), any element may have zero or more resources associated with it. However, in the case of multiple simultaneous associations, players implementing support for the resource file are required to select only one of these resources; *the first (in document order) matching* `<nodeSet>` child of the relevant <scope> element.

Document authors are expected to make use of this player requirement to sequentially order the `<nodeSet>` elements in a way that explicitly expresses resource priority. An informative example of such an ordering algorithm, implemented by an authoring tool, is as follows.

1) First, all `<nodeSet>` elements matching on a specified element name would occur before `<nodeSet>` elements matching on wildcards.

2) Second, within the two groups resulting from the above sort, `<nodeSet>` elements with a higher number of predicates (attribute matches) would occur before `<nodeSet>` elements with fewer predicates.

3) Third, resources matching on `id` attributes would be moved to the top of the sequence in order to take precedence over any other potential match.

From a player perspective, the resource selection process is typically as follows:

1) Given an element instance for which a resource may be associated (the "active element"), search for the resource file `<scope>` element whose nsuri attribute value matches the namespace uri of the active element.

2) If a matching `<scope>` element is found, search for the first `<nodeSet>` element whose XPath statement as given by the `select` attribute matches the active element. (Note that some XPath processors require addition of the qname prefix to the element selection clause before execution.)

3) If a matching `<nodeSet>` element is found, select one resource child of this set based on language preferences. (A player would typically select the resource whose language matches the language of the active element. Players may also implement default language and/or user language preference configuration, that when selecting resources for presentation would override the language of the active element. Players may or may not implement fallback behavior for the case where a matching `<nodeSet>` is found, but none of its resource children uses the desired language.)

10.4   Examples

(This section is informative.)

In Example 10.1, the Resource File contains resources in English only for the NCX and DTBOOK namespaces.

Within the NCX namespace (first `<scope>` element), two resources are supplied for skippable elements via selection of the NCX `<smilCustomTest>` element. The first `<nodeSet>` maps to a `<smilCustomTest>` element which has an attribute `bookStruct` with value "PAGE_NUMBER". The second `<nodeSet>` maps to a `<smilCustomTest>` element which has an attribute `id` with value "foo". A third `<nodeSet>` is mapped to a `<navPoint>` element with a `class` attribute valued "chapter".

Within the DTBOOK namespace (second `<scope>` element), two resources are supplied: one for the element `<td>`, and one for the element `<prodnote>` with a render attribute valued "optional".

Example 10.1:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE resources PUBLIC "-//NISO//DTD resource 2005-1//EN"
  "http://www.daisy.org/z3986/2005/resource-2005-1.dtd">
<resources xmlns="http://www.daisy.org/z3986/2005/resource/"
version="2005-1">
  <scope nsuri="http://www.daisy.org/z3986/2005/ncx/">
    <nodeSet id="ns001"
select="//smilCustomTest[@bookStruct='PAGE_NUMBER']">
      <resource xml:lang="en" id="r001">
        <text>page</text>
        <audio src="res_en.mp3" clipBegin="0.2s" clipEnd="0.4s"/>
      </resource>
    </nodeSet>
    <nodeSet id="ns002" select="//smilCustomTest[@id='foo']">
      <resource xml:lang="en" id="r002">
        <text>verse</text>
        <audio src="res_en.mp3" clipBegin="0.8s" clipEnd="1.0s"/>
      </resource>
    </nodeSet>
```

```
        <nodeSet id="ns003" select="//navPoint[@class='chapter']">
          <resource xml:lang="en" id="r003">
            <text>chapter</text>
            <audio src="chapter.mp3" clipBegin="2.8s" clipEnd="3.0s"/>
            <img src="chapter.png" />
          </resource >
        </nodeSet>
    </scope>

    <scope nsuri="http://www.daisy.org/z3986/2005/dtbook/">
      <nodeSet id="ns004" select="//prodnote[@render='optional']">
        <resource xml:lang="en" id="r004">
          <text>optional producers note</text>
          <audio src="res_en.mp3" clipBegin="2.8s" clipEnd="3.0s"/>
        </resource>
      </nodeSet>
      <nodeSet id="ns005" select="//td">
        <resource xml:lang="en" id="r005">
          <text>table cell</text>
          <audio src="res_en.mp3" clipBegin="1.2s" clipEnd="1.4s"/>
        </resource>
      </nodeSet>
    </scope>
  </resources>
```

In Example 10.2, three multiple languages resources are supplied for the DTBOOK namespace, and one resource is supplied for the SMIL namespace.

Within the DTBOOK namespace (first `<scope>` element), resources are supplied for a `<prodnote>` element with a class attribute valued image, a `<prodnote>` element with no attribute specification, and lastly for any element with a `class` attribute valued `image`.

Within the SMIL namespace (second `<scope>` element), a resource is supplied for an "escapable" element. The resource in this case maps via an arbitrary `class` attribute value.

Example 10.2:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE resources PUBLIC "-//NISO//DTD resource 2005-1//EN"
  "http://www.daisy.org/z3986/2005/resource-2005-1.dtd">
<resources xmlns="http://www.daisy.org/z3986/2005/resource/"
version="2005-1">
  <scope nsuri="http://www.daisy.org/z3986/2005/dtbook/">
    <nodeSet id="ns001" select="//prodnote[@class='image']">
      <resource xml:lang="en" id="r001">[...]</resource>
      <resource xml:lang="da" id="r002">[...]</resource>
      <resource xml:lang="ja" id="r003">[...]</resource>
    </nodeSet>
    <nodeSet id="ns002" select="//prodnote">
      <resource xml:lang="en" id="r004">[...]</resource>
      <resource xml:lang="da" id="r005">[...]</resource>
      <resource xml:lang="ja" id="r006">[...]</resource>
    </nodeSet>
    <nodeSet id="ns003" select="//*[@class='image']">
      <resource xml:lang="en" id="r007">[...]</resource>
      <resource xml:lang="da" id="r008">[...]</resource>
      <resource xml:lang="ja" id="r009">[...]</resource>
    </nodeSet>
```

```
        </scope>
        <scope nsuri="http://www.w3.org/2001/SMIL20/">
          <nodeSet id="ns004" select="//*[@class='table']">
            <resource xml:lang="en" id="r010">[...]</resource>
            <resource xml:lang="da" id="r011">[...]</resource>
            <resource xml:lang="ja" id="r012">[...]</resource>
          </nodeSet>
        </scope>
      </resources>
```

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 or sometimes use more than one disk to hold a single book. When multiple DTBs are included on a single distribution medium ("media unit"), 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 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) Retracing path back to point of origin after following a link that required inserting a new 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 might 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   Distribution Requirements

(This section is normative.)

When distributing one DTB per media unit, the Package File must be placed in the root of the media unit's file system. When distributing multiple DTBs per media unit, the distInfo File alone must be placed in the root of the media unit's file system. These restrictions do not apply when a DTB is contained on a non-removable storage medium such as a hard drive.

The distInfo File is required on all media units for a given DTB when that DTB spans more than one distribution media or when multiple DTBs are contained on one media unit. Otherwise, a distInfo File is optional. There shall be no more than one distInfo File per media unit (e.g., CD-ROM disk).

The distInfo File, if present, must be a valid XML 1.0 file conforming to distInfo-2005-1.dtd (see Appendix 1, "Distribution Information DTD"), and shall be named "distInfo.dinf". The `version` and `xmlns` attributes on the `distInfo` element must be explicitly specified in the document instance, using values drawn from the above-named DTD. Entity declarations must occur in the internal DTD subset. See further section 16.1 "General File Conformance Requirements."

Distribution on multiple media units has implications for the production of the NCX and SMIL. For the NCX, see section 8.4.2, "DTBs Spanning Multiple Media Units." For SMIL, see section 7.4.4, "Packaging Files across Several Media Units."

Optional `changeMsgs` may be used to supply customized messages instructing users on how to proceed when another media unit is needed to continue reading. Such `changeMsgs` enable presentation of messages in either text or audio. If no `changeMsg` is present when required, the player must render a default audio or text message (e.g., "please insert disk 2").

The distInfo file may include a list of all files in the distribution via the `<fileSet>` element. This is optional, and playback systems are not expected to process this in any particular way. If used, `<fileSet>` must list all files on all pieces of media in the distribution, including items not normally listed in the DTB package file, non-DTB files, the distInfo file itself, etc.

Values for the attribute `media` on the element `<book>` and for the attribute `mediaRef` on the elements `smilRef`, `changeMsg`, and `file` shall be in the format "x:y", where x is the sequence number of this media unit, and y is the total number of media units in the distribution of this book. If the book spans two or more media units, the `media` attribute on `<book>` must be present and contain a value.

11.3   DistInfo Elements

(This section is informative.)

- **<distInfo>**

  Description: The root element of a distInfo File.

  Declaration: `<!ELEMENT distInfo (book+, fileSet?) >`

  Syntax: `<distInfo...attribute...>...content...</distInfo>`

  Attributes:

  - **version (CDATA, FIXED) "2005-1":** Specifies the version of the DTD used in this instance. This attribute and its value (given in DTD) must be explicitly specified in the document instance.

  - **xmlns (CDATA, FIXED) "http://www.daisy.org/z3986/2005/distInfo/":** Specifies the default XML namespace for all elements in the distInfo File. See [XML-Namespaces] for details on namespaces. This attribute and its value (given in DTD) must be explicitly specified in the document instance.

  Valid inside: None

- **<book>**

  Description: Identifies a DTB that is present, in part or whole, on this piece of distribution media.

  Declaration: `<!ELEMENT book (docTitle, docAuthor*, distMap?, changeMsg*)>`

  Syntax: `<book ...attributes...>...content...</book>`

  Attributes:
  - o **uid (CDATA, REQUIRED):** The globally unique identifier for the DTB. The value is the same as that of the `dc:Identifier` element referenced by the unique-identifier attribute on the package file's `package` element. See section 3.1, "[Package Identity]."

  - o **pkgRef (CDATA, REQUIRED):** The URI of the book's package file. However, players must be able to locate a DTB's package file even if a distInfo File is not present.

  - o **media (CDATA, IMPLIED):** If the book spans two or more media units, the media attribute identifies the media unit in hand, in the format "x:y", where x is the sequen**ce** number of this media unit, 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 and zero or more `changeMsg`s.

- **<distMap>**

  Description: A map identifying which media unit 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. `distMap` is only necessary when a book spans multiple pieces of media.

- **<smilRef>**

  Description: A reference to a DTB SMIL file.

  Declaration: `<!ELEMENT smilRef EMPTY >`

  Syntax: `<smilRef...attributes.../>`

  Attributes:
  - o **file (CDATA, REQUIRED):** The filename of the given SMIL file.

  - o **mediaRef (CDATA, REQUIRED):** Identifies the media unit on which the given SMIL file resides, in the format "x:y", where x is the sequence number of that media unit, 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.

- **<docTitle>**

    Description: The title of the book, presented as text and, optionally, in audio or image renderings, for presentation to the reader.

    Declaration: `<!ELEMENT docTitle (text, audio?, img?)>`

    Syntax: `<docTitle ...attributes... >` ...content... `</docTitle>`

    Attributes:

    - o **xml:lang (NMTOKEN, IMPLIED):** Specifies the [RFC 3066] language code of the language of the title.

    Valid inside: `<book>`

- **<docAuthor>**

    Description: An author of the book, presented as text and, optionally, in audio or image renderings, for presentation to the reader.

    Declaration: `<!ELEMENT docAuthor (text, audio?, img?)>`

    Syntax: `<docAuthor ...attributes...>`...content...`</docAuthor>`

    Attributes:

    - o **xml:lang (NMTOKEN, IMPLIED):** Specifies the [RFC 3066] language code of the language of the title.

    Valid inside: `<book>`

- **<changeMsg>**

    Description: Contains text and/or audio versions of a custom message to be read when a new disk is requested by the reading system.

    Declaration: `<!ELEMENT changeMsg ((text, audio?) | audio)>`

    Syntax: `<changeMsg...attributes...>`…content...`</changeMsg>`

    Attributes:

    - o **mediaRef (CDATA, REQUIRED):** Identifies the media unit that this message (e.g., "Insert disc 2") specifies. Player invokes the correct `<changeMsg>` by matching its `mediaRef` attribute to the `mediaRef` attribute of the selected `<smilRef>`. In the format "x:y", where x is the sequence number of the specified media unit, and y is the total number of media pieces in the distribution of this book.
    - o **xml:lang (NMTOKEN, IMPLIED):** Specifies the [RFC 3066] language code of the language in which the message is presented.

    Valid inside: `<book>`

- **<fileSet>**

    Description: Contains a list of all files in the distribution.

    Declaration: `<!ELEMENT fileSet (file+) >`

    Syntax: `<fileSet>`...content...`</fileSet>`

    Attributes: None

    Valid inside: `<distInfo>`

- **<file>**

  Description: A file in the distribution.

  Declaration: `<!ELEMENT file EMPTY >`

  Syntax: `<file...attributes.../>`

  Attributes:

  - o **fileRef (CDATA, REQUIRED):** The URI of the file.

  - o **mediaRef (CDATA, IMPLIED):** Identifies the media unit on which the given file resides, in the format "x:y", where x is the sequence number of that media unit, and y is the total number of media pieces in the distribution of this book.

  Valid inside: `<fileSet>`

- **<text>**

  Description: Contains text of title, author, or media change message.

  Declaration: `<!ELEMENT text (#PCDATA) >`

  Syntax: `<text>...content...</text>`

  Attributes: None

  Valid inside: `<changeMsg>, <docTitle>, <docAuthor>`

- **<audio>**

  Description: Pointer to audio clip of title, author, or media change message.

  Declaration: `<!ELEMENT audio EMPTY>`

  Syntax: `<audio...attributes.../>`

  Attributes:

  - o **src (%URI, REQUIRED):** URI of audio content of media change message.

  - o **clipBegin (CDATA, REQUIRED):** 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 [SMIL]. See section 7.7, "Media Clipping and Clock Values."

  - o **clipEnd (CDATA, REQUIRED):** 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`.

  Valid inside: `<changeMsg>, <docTitle>, <docAuthor>`

- **<img>**

  Description: Contains a pointer to graphical content associated with a `<docTitle>` or `<docAuthor>`.

  Declaration: `<!ELEMENT img EMPTY>`

  Syntax: `<img...attributes... />`

  Attributes:

  - o **src (CDATA, REQUIRED):** The URI of the media object.

  Valid inside: `<docTitle>, <docAuthor>`

11.4   Examples

(This section is informative.)

Example 11.1 shows the distInfo File for the first disk of a book that spans three 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 three. 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.

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, the `changeMsg` whose mediaRef attribute matches that of the selected `smilRef` element would be played.

Example 11.1:

```
<?xml version="1.0"  encoding="UTF-8"?>
<!DOCTYPE distInfo PUBLIC "-//NISO//DTD distInfo 2005-1//EN"
   "http://www.daisy.org/z3986/2005/distInfo-2005-1.dtd">
<distInfo version="2005-1"
xmlns="http://www.daisy.org/z3986/2005/distInfo/">

  <book uid="us-rfbd-tbfz284" pkgRef="./FZ284.opf" media="1:3">
       <docTitle>
           <text>Development through the lifespan</text>
           <audio src="FZ284_1098.mp3" clipBegin="0s"
           clipEnd="3.087s" />
       </docTitle>
       <docAuthor>
           <text>Laura E. Berk</text>
           <audio src="FZ284_1098.mp3" clipBegin="3.5s"
           clipEnd="5.28s" />
       </docAuthor>

    <distMap>
      <smilRef file="FZ284_0001d.smil" mediaRef="1:3"/>
      <smilRef file="FZ284_0002d.smil" mediaRef="1:3"/>
      <smilRef file="FZ284_0003d.smil" mediaRef="1:3"/>
      <smilRef file="FZ284_0004d.smil" mediaRef="1:3"/>
      <smilRef file="FZ284_0005d.smil" mediaRef="2:3"/>
      <smilRef file="FZ284_0006d.smil" mediaRef="2:3"/>
      <smilRef file="FZ284_0007d.smil" mediaRef="2:3"/>
      <smilRef file="FZ284_0008d.smil" mediaRef="2:3"/>
      <smilRef file="FZ284_0009d.smil" mediaRef="2:3"/>
      <smilRef file="FZ284_0010d.smil" mediaRef="3:3"/>
    </distMap>

    <changeMsg mediaRef="1:3">
       <text>Insert disc one.</text>
       <audio src="insert.wav" clipBegin="0s"
       clipEnd="2.256s" />
    </changeMsg>
    <changeMsg mediaRef="2:3">
       <text>Insert disc two.</text>
       <audio src="insert.wav" clipBegin="3s"
       clipEnd="5.881s" />
```

```
            </changeMsg>
            <changeMsg mediaRef="3:3">
               <text>Insert disc three.</text>
               <audio src="insert.wav" clipBegin="6.901s"
               clipEnd="10s" />
            </changeMsg>
        </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 media unit so the media attribute is omitted. This distInfo file also includes the optional <fileSet> element, which lists all the files in the two books, plus some non-DTB "extras".

Example 11.2:

```
<?xml version="1.0"  encoding="UTF-8"?>
<!DOCTYPE distInfo PUBLIC "-//NISO//DTD distInfo 2005-1//EN"
   "http://www.daisy.org/z3986/2005/distInfo-2005-1.dtd">
<distInfo version="2005-1"
xmlns="http://www.daisy.org/z3986/2005/distInfo/">
    <book uid="us-nls-db00001" pkgRef="./book1/AllAboutDogs.opf" >
        <docTitle>
            <text>All About Dogs:  Everything you wanted to know about
                    canines, but were afraid to ask</text>
            <audio src="distAudioClips.mp3" clipBegin="00:00"
            clipEnd="00:09.632" />
            <img src="./book1/cover.jpg" />
        </docTitle>
        <docAuthor>
            <text>George Kerscher</text>
            <audio src="distAudioClips.mp3" clipBegin="00:10"
            clipEnd="00:13.109" />
        </docAuthor>
        <docAuthor>
            <text>Nesbit Kerscher</text>
            <audio src="distAudioClips.mp3" clipBegin="00:14"
            clipEnd="00:15.53" />
        </docAuthor>
    </book>

    <book uid="us-nls-db98765" pkgRef="./book2/AllAboutCats.opf" >
        <docTitle>
            <text>All About Cats:  Everything you wanted to know about
                    felines, but were afraid to ask</text>
            <audio src="distAudioClips.mp3" clipBegin="00:20"
            clipEnd="00:24.87" />
            <img src="./book2/cover.jpg" />
        </docTitle>
        <docAuthor>
            <text>Frances White</text>
            <audio src="distAudioClips.mp3" clipBegin="00:26"
            clipEnd="00:27.45" />
        </docAuthor>
        <docAuthor>
            <text>Isabel White</text>
```

```
                <audio src="distAudioClips.mp3" clipBegin="00:28"
                clipEnd="00:29.71" />
            </docAuthor>
        </book>

        <fileSet>
            <file fileRef="./book1/AllAboutDogs.opf" />
            <file fileRef="./book1/AllAboutDogs.ncx" />
            <file fileRef="./book1/AllAboutDogs.smil" />
            <file fileRef="./book1/db00001_1.mp4" />
            <file fileRef="./book1/db00001_2.mp4" />
            <file fileRef="./book1/db00001_3.mp4" />
            <file fileRef="./book1/cover.jpg" />
            <file fileRef="./book1/NLS.res" />
            <file fileRef="./book1/NLSres.mp4" />
            <file fileRef="./book2/AllAboutCats.opf" />
            <file fileRef="./book2/AllAboutCats.ncx" />
            <file fileRef="./book2/AllAboutCats.smil" />
            <file fileRef="./book1/db98765_1.mp4" />
            <file fileRef="./book1/db98765_2.mp4" />
            <file fileRef="./book1/db98765_3.mp4" />
            <file fileRef="./book1/db98765_4.mp4" />
            <file fileRef="./book2/cover.jpg" />
            <file fileRef="./book2/NLS.res" />
            <file fileRef="./book2/NLSres.mp4" />
            <file fileRef="./extras/Guide_Dogs.mov" />
            <file fileRef="./extras/Attack_Cats.mov" />
            <file fileRef="distAudioClips.mp3" />
            <file fileRef="distInfo.dinf" />
        </fileSet>
    </distInfo>
```

12   Presentation Styles

12.1 Introduction

(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 currently 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 can also be applied by the content producer as well as by the user. Producer-supplied style sheets are particularly important for complex documents with formatting or presentational requirements that would not be met by a player's or user's default styles.

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 and supports CSS or XSL.

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 can be indicated with a specific audio cue.

12.2   Implementing Style Sheets for DTBs

(This section is normative.)

Style sheets are optional components of DTBs and DTB distribution systems. DTB producers may choose to supply default visual, Braille, or audio style sheets.

Style sheets must not be written in such a way as to prevent users from overriding them. DTBs referencing style sheets must do so using standard W3C mechanisms to link an XML source to its style sheet (see [XML-Style]). All style sheet processing instructions must include the *media* attribute specifying which medium the style sheet applies to. Acceptable values are: all (for all media), aural (for audio presentations), braille (for refreshable Braille displays), embossed (for embossed Braille), handheld (for devices with small monochrome screens), print (for visual formatting of printed output), and screen (for color computer screens). For example:

```
<?xml-stylesheet href="brstyle.css" type="text/css" media="braille"?>
```

Playback systems that use common PC-based browsers should support presentation styles at least to the extent the browser itself does. However, it is strongly recommended that any DTB player incorporating a visual display implement at least CSS1. Portable players will not generally 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, thereby allowing the user to define styles that supersede default player- or producer-defined styles. It is strongly recommended that players implementing style sheets support user control of presentation styles.

When multiple style sheets are present for the content being rendered, user-defined styles, if present, shall take precedence, followed by producer-defined and player-defined styles, in that order.

13   Content Rendering

(This section is normative.)

Players must determine how to render content from the types of files present. If only a textual content file is found, a synthetic speech rendering and output to a Braille display and/or a visual display 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 shall be initiated. A player that supports only a subset of the media included in DTBs must, when encountering an unsupported medium, ignore the unsupported files and correctly render those it does support. In addition, if the playback system cannot render any of the media in the DTB, based on the value of `dtb:multimediaContent` in the package file metadata, it must report this fact to the user.

14 Digital Rights Management

(This section is informative.)

Protection of intellectual property will continue to be an important issue for national libraries and other agencies serving people with print disabilities. How this responsibility is met in Digital Talking Book distribution programs, however, will vary from country to country due to differences in the legal environment surrounding the distribution of alternative format materials. It will also vary by item depending on whether the material is under copyright or in the public domain. When applicable, however, it is critical that agencies use reasonable administrative and technical measures to protect copyright holders' rights. It is equally important, though, that agencies ensure access to alternative format materials by their target populations. Thus, DTB producers and distributors that implement DRM systems must do so in a manner that does not limit or prevent access to compliant DTBs by eligible users.

15 Time-Scale Modification

(This section is normative.)

It is strongly recommended that playback systems implement Time-Scale Modification (TSM) to enable user control of playback speed. Playback rates continuously variable from one-third to three times normal speed are recommended. It is also recommended that players allow users the option of disabling pitch correction during TSM operation.

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.

16 Conformance

(This section is normative.)

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. It should be noted that while many aspects of file conformance can be enforced through the DTDs included in this standard, others cannot, and must be enforced through other means.

16.1 General File Conformance Requirements

(This section is normative.)

This specification allows DTB playback devices to make use of validating and non-validating XML processors. For all XML documents in the DTB file set, the following grammatical rules and file conformance requirements are incorporated to allow this flexibility:

- all document grammars use an attribute named id as the singular ID token

- all attributes declared as #FIXED in the respective DTD are required to be explicitly specified in the document instance

- all entity declarations must occur in the internal DTD subset

17   References to Other Specifications/Documents

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

17.1   Normative References

(This section is normative.)

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/

Dublin Core
Dublin Core Metadata Initiative: http://dublincore.org/

DC-Type
Dublin Core Type Vocabulary: http://dublincore.org/documents/dcmi-type-vocabulary/

ISO 3166
ISO 3166 - Codes for the Representation of Names of Countries and their Subdivisions:
http://www.iso.ch/iso/en/ISOOnline.openerpage

ISO 8601
W3C Profile of ISO8601 - Representation of Dates and Times: http://www.w3.org/TR/NOTE-datetime.html

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/iso/en/ISOOnline.openerpage

ISO/IEC 10646
ISO/IEC 10646 - Universal Multiple-Octet Coded Character Set: http://www.unicode.org

JPEG
JPEG JFIF V1.02: http://www.jpeg.org/public/jfif.pdf

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/iso/en/ISOOnline.openerpage or from your national standards body. In the
United States, this is the American National Standards Institute: http://www.ansi.org

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

OEBF
The Open eBook Forum Publication Structure, version 1.2: http://www.openebook.org

RFC 2046
Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types:
http://www.ietf.org/rfc/rfc2046.txt

RFC 2083
Portable Network Graphics, Version 1: http://www.ietf.org/rfc/rfc2083.txt

RFC 2396
Uniform Resource Identifiers (URI): Generic Syntax: http://www.ietf.org/rfc/rfc2396.txt

RFC 3066
Tags for the Identification of Languages: http://www.ietf.org/rfc/rfc3066.txt

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

SMIL
SMIL 2.0 W3C Recommendation [Second Edition] 07 January 2005:
http://www.w3.org/TR/2005/REC-SMIL2-20050107/

SVG
Scalable Vector Graphics: http://www.w3.org/TR/2001/REC-SVG-20010904/

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

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

XML-Style
Associating Style Sheets with XML Documents 1.0: http://www.w3.org/TR/xml-stylesheet/

XPath10
XML Path Language (XPath) Version 1.0: http://www.w3.org/TR/xpath

XSL
Extensible Stylesheet Language (XSL) Version 1.0: http://www.w3.org/TR/xsl/

17.2   Informative References

(This section is informative.)

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

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

DAISY
The DAISY Consortium: http://www.daisy.org

DTBook HTML
HTML version of expanded DTBook DTD:
http://www.daisy.org/z3986/2005/dtbook/dtbookdoc.html

DTBook Theory
Theory behind the DTBook DTD:
http://www.daisy.org/publications/docs/theory_dtbook/theory_dtbook.html

Navigation Features
Document Navigation Features List: http://www.loc.gov/nls/z3986/background/navigation.htm

Player Features
Playback Device Features List: http://www.loc.gov/nls/z3986/background/features.htm

RFC 2048
Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures
http://www.ietf.org/rfc/rfc2048.txt

RFC 3023
XML Media Types http://www.ietf.org/rfc/rfc3023.txt

StructGuide
Structure Guidelines: http://www.daisy.org/publications/guidelines/sg-daisy3/structguide.htm

UAAG
User Agent Accessibility Guidelines: http://www.w3.org/TR/UAAG10/

Validating and Non-Validating Processors
XML 1.0 Specification (Third Edition), section 5.1: http://www.w3.org/TR/2004/REC-xml-20040204/#proc-types

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

XSLT
XSL Transformations (XSLT) Version 1.0: http://www.w3.org/TR/xslt

Appendices

Appendix 1: Document Type Definitions (DTDs)

(This section is normative.)

The following DTDs that must be used in conjunction with this standard have been tested and verified and are maintained on the Maintenance Agency website: http://www.daisy.org/z3986/

DTBook DTD

DTB-Specific SMIL DTD

NCX DTD

DTD for Portable Bookmarks/Highlights

DTD for Resource File

Distribution Information DTD

Appendix 2: Designation of Maintenance Agency

(This section is informative.)

The functions assigned to the maintenance agency as specified in section 1.7 will be administered by the DAISY Consortium. Questions concerning the implementation of this standard and requests for information should be sent to the staff of the DAISY Consortium at daisy-staff@daisy.org, including "Z3986" in the subject line.