ANSI/NISO Z39.86-2002 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.

An American National Standard
Developed by the
National Information Standards Organization

Approved March 6, 2002
by the
American National Standards Institute

Published by the National Information Standards Organization
Bethesda, Maryland

NISO Press, Bethesda, Maryland, U.S.A.

Amazon v. Audio Pod
US Patent 8,738,740
Amazon EX-1003

Published by
NISO Press
4733 Bethesda Avenue, Suite 300
Bethesda, MD 20814
www.niso.org

Copyright ©2002 by the National Information Standards Organization.
All rights reserved under international and Pan-American Copyright Conventions. For noncommercial purposes only this publication may be reproduced or transmitted in any form or by any means without prior permission in writing from the publisher. All inquiries regarding commercial reproduction or distribution should be addressed to NISO Press, 4733 Bethesda Avenue, Suite 300, Bethesda, MD 20814 USA.

Printed in the United States of America

ISSN: 1041-5653 National Information Standards Series
ISBN: 1-880124-54-8

∞ This paper meets the requirements of ANSI/NISO Z39.48-1992 (R 1997) Permanence of Paper.

Contents

Foreword vii

1. General Information 1

1.1 Purpose and Scope of Standard 1
1.2 Definitions 1
1.3 Strategy 4
1.4 Accessibility Issues 4
1.5 Relationship to Other Specifications 5
1.5.1 Relationship to Unicode 5
1.6 Patent Rights 5
1.7 Maintenance Agency 5

2. Overview 6

3. The DTB Package File 8

3.1 Package Identity 9
3.2 Publication Metadata 9
3.2.1 Dublin Core Metadata 9
3.2.2 DTB ID Scheme 11
3.2.3 X-Metadata 12
3.3 Manifest 14
3.4 Spine 16
3.5 Tours 16
3.6 Guide 17

4. Content Format for Text 17

4.1 Introduction 17
4.2 Using the DTBook Element Set 17
4.2.1 DTBook Markup Related to SMIL 18
4.2.2 Modular Extension of the DTD 18

5. Audio File Formats 19

5.1 Distribution Formats 19
5.2 Formats for Audio Notes 20

6. Image File Formats 20

7. Synchronization of Media Files 21

7.1 Introduction 21
7.1.1 Background 21
7.1.2 SMIL Modules 22
7.2 Application of SMIL to DTBs 23
7.3 SMIL Elements 23
7.3.1 Core Attributes 27
7.4 SMIL Requirements for DTBs 28
7.4.1 "Escapable" Structures 28
7.4.2 Automatic Invocation of Special Navigation Modes 28
7.4.3 "Skippable" Structures 28
7.4.4 Packaging Files across Several Media Units 29
7.4.5 Links 29
7.4.6 Layout Syntax 29
7.4.7 Content of <par>s 29
7.4.8 Notes and Annotations in SMIL 30
7.4.9 Images in SMIL 30
7.4.10 Text-Only DTBs 30
7.5 SMIL Metadata 30
7.6 Examples 31
7.7 Clock Values 34

8. Navigation Control File (NCX) 35

8.1 Introduction 35
8.2 Key NCX Requirements 35
8.3 NCX Elements 35
8.4 Other File Requirements 39
8.4.1 Navigation Metadata 39
8.4.2 DTBs Spanning Multiple Media Units 40
8.4.3 mapRef Attribute 40
8.4.4 smilCustomTest Element 40
8.5 How the NCX Works 41
8.6 Example 42

9. Portable Bookmarks and Highlights 45

9.1 Introduction 45
9.2 Bookmark/Highlight Elements 46
9.3 Examples 49

10. Resource File 50

10.1 Introduction 50
10.2 Resource Elements 52
10.3 Resource File Requirements 53
10.4 Examples 53

 ©NISO

11. Packaging Files for Distribution 55
11.1 Introduction 55
11.2 Distribution Requirements 55
11.3 DistInfo Elements 56
11.4 Examples 58
12. Presentation Styles 59
12.1 Introduction 59
12.2 Implementing Style Sheets for DTBs 60
13. Types of DTB 60
13.1 Types 60
13.2 Required Files 62
13.3 Content Rendering 62
14. Digital Rights Management 62
15. Time-Scale Modification 63
16. Conformance 63
17. References to Other Specifications/Documents 64
17.1 Normative References 64
17.2 Informative References 65
Appendix 1 - DTBook DTD 66
Appendix 2 - DTB-Specific SMIL DTD 107
Appendix 3 - NCX DTD 110
Appendix 4 - DTD for Portable Bookmarks/Highlights 114
Appendix 5 - DTD for Resource File 116
Appendix 6 - Distribution Information DTD 118
Appendix 7 - Designation of Maintenance Agency 121

©NISO

Foreword

(This foreword is not a part of ANSI/NISO Z39.86-2002, 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.

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 a digital rights management system.

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

(*continued*)

This standard was processed and approved for submittal to ANSI by the National Information Standards Organization. It was balloted by the NISO Voting Members November 1, 2001- December 17, 2001. It will next be reviewed in 2007. Suggestions for improving this standard are welcome. They should be sent to the National Information Standards Organization, 4733 Bethesda Avenue, Suite 300, Bethesda, MD 20814. 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:

NISO Voting Members

3M
Jerry Karel
Susan Boettcher (Alt)

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

American Chemical Society
Robert S. Tannehill, Jr.

American Library Association
Paul J. Weiss

American Society for Information Science and Technology
Mark H. Needleman

American Society of Indexers
Judith Gibbs
Jacqueline Rodebaugh (Alt)

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
David L. Austin

AIIM International
Betsy A. Fanning

Association of Jewish Libraries
Caroline R. Miller
Elizabeth Vernon (Alt)

Association of Research Libraries
Duane E. Webster
Julia Blixrud (Alt)

BiblioMondo Inc.
Martin Sach

Book Industry Communication
Brian Green

Cambridge Scientific Abstracts
Michael Cairns
Matthew Dunie (Alt)

Checkpoint Systems, Inc.
Paul Simon

College Center for Library Automation
J. Richard Madaus
Ann Armbrister (Alt)

Congressional Information Service, Inc.
Robert Lester

devine, inc.
Robert Boissy

Elsevier Science
Anthony Ross
John Mancia (Alt)

Endeavor Information Systems, Inc.
Verne Coppi
Cindy Miller (Alt)

epixtech, Inc.
John Bodfish
Ricc Ferante (Alt)

Ex Libris
James Steenbergen
Carl Grant (Alt)

Follett Corporation
D. Jeffrey Blumenthal
Don Rose (Alt)

Fretwell-Downing Informatics
Robin Murray

Gale Group
Katherine Gruber
Justine Carson (Alt)

Gaylord Information Systems
William Schickling
Linda Zaleski (Alt)

GCA Research Institute
Jane Harnad

H.W. Wilson Company
Ann Case

Information Use Management & Policy Institute
Charles McClure
John Carlo Bertot (Alt)

Infotrieve
Jan Peterson

Innovative Interfaces, Inc.
Gerald M. Kline
Sandy Westall (Alt)

 ©NISO

NISO Voting Members *(continued)*

Institute for Scientific Information

The International DOI Foundation
Norman Paskin

Library Binding Institute
Donald Dunham

The Library Corporation
Mark Wilson
Nancy Capps (Alt)

Library of Congress
Winston Tabb
Sally H. McCallum (Alt)

Los Alamos National Laboratory
Richard E. Luce

Lucent Technologies
M.E. Brennan

Medical Library Association
Nadine P. Ellero
Carla J. Funk (Alt)

MINITEX
Cecelia Boone
William DeJohn (Alt)

Modern Language Association
Daniel Bokser
Cameron Bardrick (Alt)

Motion Picture Association of America
William M. Baker
Axel aus der Muhlen (Alt)

Music Library Association
Lenore Coral
Mark McKnight (Alt)

National Agricultural Library
Gary K. McCone

National Archives and Records Administration
Mary Ann Hadyka

National Federation of Abstracting and Information Services
Marion Harrell

National Library of Medicine
Betsy L. Humphreys

Nylink
Mary-Alice Lynch
Jane Neale (Alt)

OASIS

OCLC, Inc.
Donald J. Muccino

Openly Informatics
Eric Hellman (Alt)

Proquest Information and Learning
Todd Fegan
James Brei (Alt)

Recording Industry Association of America
Linda R. Bocchi
Michael Williams (Alt)

Research Libraries Group
Lennie Stovel
Joan Aliprand (Alt)

SIRS Mandarin, Inc.
Leonardo Lazo
Harry Kaplanian (Alt)

Sirsi Corporation
Greg Hathorn
Slavko Manojlovich (Alt)

Society for Technical Communication
Annette Reilly
Kevin Burns (Alt)

Society of American Archivists
Lisa Weber

Special Libraries Association
Marcia Lei Zeng

Triangle Research Libraries Network
Jordan M. Scepanski
Mona C. Couts (Alt)

U.S. Department of Commerce, National Institute of Standards and Technology, Office of Information Services

U.S. Department of Defense, Defense Technical Information Center
Gopalakrishnan Nair
Jane L. Cohen (Alt)

U.S. National Commission on Libraries and Information Science
Denise Davis

VTLS Inc.
Vinod Chachra

(continued)

NISO Voting Members *(continued)*

At the time NISO approved this standard, the following individuals served on its Board of Directors:

NISO Board of Directors

Beverly P. Lynch, Chair
University of California, Los Angeles

Jan Peterson, Vice Chair/Chair-Elect
Infotrieve, Inc.

Donald J. Muccino, Immediate Past Chair
OCLC, Inc.

Jan Peterson, Treasurer
Infotrieve, Inc.

Patricia R. Harris, Executive Director
National Information Standards Organization

Pieter S. H. Bolman
Elsevier Science

Priscilla Caplan
Florida Center for Library Automation

Carl Grant
Ex Libris (USA), Inc.

Brian Green
BIC/EDItEUR

Jose-Marie Griffiths
University of Pittsburgh

Richard E. Luce
Los Alamos National Laboratory

Sally McCallum
Library of Congress

Norman Paskin
The International DOI Foundation

Steven Puglia
U.S. National Archives and Records Administration

Albert Simmonds
OCLC, Inc.

Standards Committee AQ

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

Mr. Donald J. Breda
American Council of the Blind

Mr. George Brummell
Blinded Veterans Association

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

Mr. Glen Cavanaugh
Telex Communications, Inc.

Mr. Curtis Chong
World Blind Union

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

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

Mr. Keith Creasy
American Printing House for the Blind

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

Mr. Jack Decker
American Printing House for the Blind

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

Mr. Jim Dust
Telex Communications, Inc.

Dr. Michael Gosse
National Federation of the Blind

Mr. Luis Gutierrez
American Foundation for the Blind

Mr. Mark Hakkinen
Japanese Society for the Rehabilitation of Persons with Disabilities

Mr. John Hedges
American Printing House for the Blind

(continued)

 ©NISO

Standards Committee AQ *(continued)*

Ms. Vivian Seki
Hadley School for the Blind

Ms. Rosemary Kavanagh
Canadian National Institute for the Blind

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

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

Ms. Kathie Korpolinski
Recording for the Blind and Dyslexic

Mr. Dominic Labbé
VisuAide, Inc.

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

Mr. Thomas McLaughlin
National Library Service for the Blind and Physically Handicapped,
Library of Congress

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

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

Mr. Gilles Pepin
VisuAide, Inc.

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

Ms. Janina Sajka
American Foundation for the Blind

Mr. Rudy Savage
Talking Book Publishers, Inc.

Mr. Larry Skutchan
American Printing House for the Blind

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

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

Ms. Karen Taylor
Canadian National Institute for the Blind

Acknowledgements

Standards Committee AQ gratefully acknowledges the contributions made by the DAISY Consortium (www.daisy.org) to this work. The Consortium created a series of open international specifications (DAISY 2.0 ©1998, DAISY 2.01 ©1999, and DAISY 2.02 ©2001) that formed the foundation on which this standard is built. DAISY representatives served on Committee AQ since its inception and knowledge gained in their work on DAISY projects greatly informed the complex discussions and decisions leading to the creation of this document. In addition, they hosted several list-servs on which many issues critical to DTB work in general, and to this standard specifically, were discussed and resolved. It is no exaggeration to state that without their groundbreaking efforts and their ongoing contributions to Committee work, this standard would not exist in anything like its current level of sophistication.

In addition, the Committee wishes to thank the following individuals for their substantial assistance to the process of creating the standard: Robert Berkovitz, Sensimetrics Corporation; Harvey Bingham; Mike Brown; John Churchill, Recording for the Blind and Dyslexic; Manon Gaudet, VisuAide, Inc.; Al Gilman; Markus Gylling, Swedish Library of Talking Books and Braille; Steve Jacobs, NCR Corporation; Lynn Leith, Canadian National Institute for the Blind; Tatsu Nishizawa, Plextor Corporation; Dave Pawson, Royal National Institute for the Blind; James Pritchett, Recording for the Blind and Dyslexic; Dr. Gregg Vanderheiden, TRACE Research and Development Center, University of Wisconsin; Mr. Paul Vassallo, National Institute of Standards & Technology; with special thanks to members of the DAISY Consortium's Specifications and Guidelines Work Team and DTD Work Team. Thanks also to these members of the W3C Synchronized Multimedia (SYMM) Working Group: Dick Bulterman, Oratrix; Wo Chang, NIST; Lloyd Rutledge, CWI; Patrick Schmitz, Microsoft.

 ©NISO

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 (similar to fast-forward and rewind on an analog cassette) may also be implemented.

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

May. In 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 1.0.1* 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*.

Package File. The *Open eBook Forum Package File* (OPF) is an XML file conforming to the oebpkg101.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.

 ©NISO

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.

 ©NISO

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 7 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 shoud be sent to the maintenance agency.

A list of errata relating to this standard will be maintained at http://www.loc.gov/nls/z3986/v100/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. Section 13 describes these various types of DTBs.

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 1.0.1, 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, "DTBook DTD.") 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 play-

 ©NISO

back, 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 2, "DTB-Specific SMIL DTD," for the DTD that defines the DTB SMIL application.

Navigation Control File. The DTB system supports two modes of navigation, global and local. Global navigation — movement by structure (chapter, section, subsection) and by other selected points such as pages, 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 3 contains the XML 1.0 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 3, "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 4, "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 5, "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 6, "Distribution Information DTD," presents 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 1.0.1, 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 1.0.1 for complete details on the Package File.

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

PACKAGE IDENTITY - a unique identifier for the OEB publication as a whole

METADATA - Publication metadata (title, author, publisher, etc.)

MANIFEST - A list of files (documents, images, style sheets, etc.) that make up the publication. The manifest also includes fallback declarations for files of types not supported by this specification

SPINE - An arrangement of documents providing a linear reading order

TOURS - A set of alternate reading sequences through the publication, such as selective views for various reading purposes, reader expertise levels, etc

GUIDE - A set of references to fundamental structural features of the publication, such as table of contents, foreword, bibliography, etc.

Here is an informal outline of the package file:

```
<?xml version="1.0"?>
<!DOCTYPE package PUBLIC "+//ISBN 0-9673008-1-9//DTD OEB 1.0.1 Package//EN"
"http://openebook.org/dtds/oeb-1.0.1/oebpkg101.dtd">
<package>
     <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 1.0.1 package DTD (oebpkg101.dtd) and its associated entity reference (oeb1.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 1.0.1, with the following two exceptions:

1. Section 2.3.1 does not apply. Specifically, there is no requirement on DTB authors or

 ©NISO

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*.

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:

```
<package 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 1.0.1:

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

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
  - Content: The title of the DTB.
  - Occurrence: Required
- dc:Creator
  - Content: Names of primary author or creator of the intellectual content of the publication.
  - Occurrence: Optional (not all documents have known creators) - recommended.
  - Added attributes:
    - role — (optional) The function performed by the creator (e.g., author, editor). See Publication Structure for details on normative list of values.
    - file-as — (optional) A normalized form of the contents suitable for machine processing.
- dc:Subject
  - Content: The topic of the content of the publication.
  - Occurrence: Optional - recommended.
- dc:Description
  - Content: Plain text describing the publication's content.
  - Occurrence: Optional
- dc:Publisher
  - Content: The agency responsible for making the DTB available. (Compare dtb:sourcePublisher and dtb:producer.)
  - Occurrence: Required
- dc:Contributor
  - Content: A party whose contribution to the publication is secondary to those named in dc:Creator.
  - Occurrence: Optional
  - Added attributes:
    - role — (optional) The function performed by the contributor (e.g., translator, compiler). See Publication Structure for details on normative list of values.
    - file-as — (optional) A normalized form of the contents suitable for machine processing.
- dc:Date
  - Content: Date of publication of the DTB. (Compare dtb:sourceDate and

 ©NISO

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.
    - Occurrence: Required
    - Added attributes:
        - event — (optional) Significant occurrence related to publication of the DTB. Allows repetition of dc:Date to describe, for example, multiple revisions. Best practice is to use dtb:revision and dtb:revisionDate instead.

- dc:Type
    - 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].
    - Occurrence: Optional
- dc:Format
    - 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 Z39.86-2002".
    - Occurrence: Required
- dc:Identifier
    - 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.
    - Occurrence: Required
    - 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
    - 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.
    - Occurrence: Optional - recommended.
- dc:Language
    - Content: Language of the content of the publication. An [RFC 1766] language code. For Sweden: "sv" or "sv-SE"; for UK: "en" or "en-GB"; for US: "en" or "en-US"; etc.
    - Occurrence: Required
- dc:Relation
    - Content: A reference to a related resource.
    - Occurrence: Optional
- dc:Coverage
    - Content: The extent or scope of the content of the resource. Not expected to be used for DTBs.
    - Occurrence: Optional
- dc:Rights
    - Content: Information about rights held in and over the DTB. (Compare dtb:sourceRights.)
    - Occurrence: Optional

3.2.2 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
  - 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.
  - Occurrence: Optional - recommended. Not repeatable.
- dtb:sourceEdition
  - Content: A string describing the edition of the resource (e.g., a print original, ebook, etc.) from which the DTB is derived.
  - Occurrence: Optional - recommended. Not repeatable.
- dtb:sourcePublisher
  - 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.)
  - Occurrence: Optional - recommended. Not repeatable.
- dtb:sourceRights
  - 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.)
  - Occurrence: Optional - recommended. Not repeatable.
- dtb:sourceTitle
  - 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.
  - Occurrence: Optional. Not repeatable.
- dtb:multimediaType
  - Content: One of the six types of DTB described in section 13. Values are: audioOnly, audioNCX, audioPartText, audioFullText, textPartAudio, textNCX.
  - Occurrence: Required. Not repeatable.
- dtb:narrator
  - Content: Name of the person whose recorded voice is embodied in the DTB.
  - Occurrence: Optional - recommended. Repeatable.
- dtb:producer
  - Content: Name of the organization/production unit that created the DTB.

 ©NISO

(Compare dc:Publisher.)
    - Occurrence: Optional. Repeatable.
- dtb:producedDate
    - 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.
    - Occurrence: Optional. Not repeatable.
- dtb:revision
    - Content: Non-negative integer value of the specific version of the DTB. Incremented each time the DTB is revised.
    - Occurrence: Optional. Not repeatable.
- dtb:revisionDate
    - 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.
    - Occurrence: Optional. Not repeatable.
- dtb:revisionDescription
    - Content: A string describing the changes introduced in a specific dtb:revision.
    - Occurrence: Optional. Not repeatable.
- dtb:totalTime
    - Content: Total playing time of all SMIL files comprising the content of the DTB. Clock Values from SMIL 2.0 Timing and Synchronization Module [SMIL]. See section 7.7, "Clock Values."
    - Occurrence: Required. Not repeatable.
- dtb:audioFormat
    - 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.
    - Occurrence: Optional, recommended for audio DTBs. Repeatable.
        - Formats specified in section 5 are shown below, followed by the normative value for dtb:audioFormat:
          MPEG-4 AAC: MP4-AAC
          MPEG-1/2 Layer III: MP3
          Linear PCM,RIFF WAVE format: WAV
          (Values are not case-sensitive.)

(This example is informative.)

Example 3.2:

```
...
<metadata>
 <dc-metadata xmlns:dc="http://purl.org/dc/elements/1.0/"
 xmlns:oebpackage="http://openebook.org/namespaces/oeb-package/1.0/">
      <dc:Title>Revised Standards and Guidelines of
      Service for the Library of Congress Network of Libraries for the
      Blind and Physically Handicapped 1995</dc:Title>
      <dc:Subject>library information networks</dc:Subject>
      <dc:Subject>libraries and the physically
      handicapped--standards--U.S.</dc:Subject>
      <dc:Subject>libraries and the
      blind--standards--U.S.</dc:Subject>
      <dc:Identifier id="uid"
```

```
      scheme="DTB">us-nls-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 Z39.86-2002</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: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 distInfo file and any associated audio `changeMsgs` are not considered part of the DTB and thus shall not be listed (See section 11, "Packaging Files for Distribution.") 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` also includes fallback declarations for files of types not supported by this standard. (See OEBF Publication Structure for

 ©NISO

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 `item` elements listing SMIL files in the `manifest` must have a media-type attribute of "application/smil". The `item` elements for the NCX, textual content file(s), Package File, and Resource File must have `media-type` attribute values of "text/xml". The order of `item` elements within the `manifest` is not significant.

(This example is informative.)
A sample `manifest` for a DTB with audio, structure, and text (multimediaType=audioFullText) follows:

Example 3.3:

```
...

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

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

Example 3.4:

```
...

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

3.4 Spine

(This section is normative.)

The `spine`, 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 `itemrefs` in the `spine`. The `itemrefs` 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`.

 ©NISO

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 `guides`.

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 dtbook110.dtd, which can be found in Appendix 1, "DTBook DTD." The `version` attribute on the `dtbook` element must be present and contain the value drawn from the above-named DTD.

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."

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 SYSTEM "dtbook110.dtd"
>
```

For discussion of other ways of expressing the DOCTYPE, see section 2.3 of Appendix 1, "DTBook DTD."

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

 ©NISO

declaration. Such instructions in square brackets are called the "internal subset of declarations." For example:

```
<!DOCTYPE dtbook SYSTEM "dtbook110.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 (dtbook110.dtd). The third line declares the entity "% externalblock" and gives it the value "drama". Since dtbook110.dtd contains an entity of the same name, and the internal subset overrules the base (external) DTD (dtbook110.dtd) in areas of conflict, everywhere in dtbook110.dtd 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 dtbook110.dtd (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 Appendix 1, "DTBook DTD" 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. The file extensions shown for each format must be used in audio filenames in compliant DTBs. Values are not case-sensitive.

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. File extension: .aac
- MPEG-1/2 Layer III (MP3) [MPEG] - ISO/IEC 11172-3, ISO/IEC 13818-3. File extension: .mp3
- Linear PCM - RIFF WAVE format [RIFFWAV]. File extension: .wav
  File header contains information about sample rate, number of channels, etc. 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.

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] (extension: .jpg), PNG [RFC 2083] (extension: .png), or Scalable Vector Graphics [SVG] (extension: .svg). Filenames must use the file extensions specified; values are not case-sensitive. 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.

 ©NISO

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 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 `pars` 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 implementation, drawing from the following modules, which are grouped by functional area. Modules marked with asterisks are used in whole or in part in this application; the others are 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
  - *BasicInlineTiming
  - MinMaxTiming
  - SyncbaseTiming
  - EventTiming
  - *BasicTimeContainers
- Content Control
  - *BasicContentControl
  - *CustomTestAttributes
  - SkipContentControl
- Layout
  - *BasicLayout
- Linking
  - *BasicLinking
  - *LinkingAttributes
- Media Objects
  - *BasicMedia
  - *MediaClipping
- Metainformation
  - *Metainformation
- Structure
  - *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.

 ©NISO

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 in this standard in Appendix 2. 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 id attributes are often required in the DTB application when they are 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 dtbsmil110.dtd.

Time containers (`seqs` or `pars`) within SMIL files must contain `ids`. Media objects (`audio`, `text`, and `img`) may also contain `ids`, 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 shall 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 `timeOffsets`, 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.”

- xml:lang (NMTOKEN, IMPLIED)

Valid inside: None

- <head>

  Description: Contains information not directly related to the temporal presentation: metadata (using `meta` element), optional `layout`, and optional `customAttributes`, in that order.

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

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

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

  Valid inside: `<smil>`

- <meta>

  Description: Contains information describing the SMIL document.

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

  Declaration: `<!ELEMENT meta EMPTY >`

  Attributes:
  - content (CDATA, #IMPLIED)
  - name (CDATA, #REQUIRED)

  Valid inside: <head>

  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:
  - %Core.attrib;
  - 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:
  - id (ID, REQUIRED) Value of `region` attribute on media object references the id on appropriate `region` element.
  - bottom (CDATA, 'auto') Locates `region` in display space. See SMIL 2.0 for details.
  - left (CDATA, 'auto') Locates `region` display space. See SMIL 2.0 for details.
  - right (CDATA, 'auto') Locates `region` in display space. See SMIL 2.0 for details.
  - top (CDATA, 'auto') Locates `region` in display space. See SMIL 2.0 for details.
  - height (CDATA, 'auto') Locates `region` in display space. See SMIL 2.0 for details.
  - width (CDATA, 'auto') Locates `region` in display space. See SMIL 2.0 for details.
  - fit ((hidden|fill|meet|scroll|slice) 'hidden') Specifies behavior if the intrinsic height and width of a visual media object differ from those of the `region` in which it is displayed. See SMIL 2.0 for definitions of attribute values.
  - backgroundColor (CDATA, IMPLIED) Sets background color of the area of the `region` that is not covered by the media object(s) being displayed.

 ©NISO

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

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:

- %Core.attrib; See section 7.3.1, "Core Attributes."
- 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.) which the user can choose to have presented or skipped during normal playback of a DTB. See definition of customTest attribute for par and seq below.

Declaration: <!ELEMENT customTest EMPTY >

Syntax: <customTest ...attributes... />

Attributes:

- id (ID, REQUIRED) Id here serves as a unique identifier referenced by a customTest attribute on par or seq in body of SMIL.
- class (CDATA, IMPLIED)
- title (CDATA, IMPLIED)
- xml:lang (NMTOKEN, IMPLIED)
- 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.
- override ((visible|hidden) 'hidden') Specifies whether the playback device should present to (value= "visible") or hide from (value = "hidden") the reader the ability to override the setting of defaultState. 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:

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

Valid inside: <smil>

Comments: The body contains zero or more seqs or pars and may also directly contain zero or more media objects (text, audio, img), or links (a).

- <seq>
  Description: Container for a sequence of SMIL events, e.g., a series of pars and seqs.
  Declaration: `<!ELEMENT seq (par|seq|text|audio|img|a)+ >`
  Syntax: `<seq>...content...</seq>`
  Attributes:
  - id (ID, REQUIRED):
  - class (CDATA, IMPLIED): Used for several purposes: First, to identify structures such as tables, lists, and notes from which a reader can "escape" with a single action. Second, to identify structures such as tables and lists for which special navigation functions should be automatically invoked when entered. To support these functionalities, the class attribute will be valued with element names drawn from the DTBook DTD, e.g., "table," "list," and "note."
  - customTest (IDREF, IMPLIED): ID reference linking seq with matching customTest element in head.
  - dur (CDATA, IMPLIED) The duration of the seq. It uses the same attribute value syntax as begin.

  Valid inside: body, seq, par
  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): Used for several purposes: First, to identify structures such as tables, lists, and notes from which a reader can "escape" with a single action. Second, to identify structures such as tables and lists for which special navigation functions should be automatically invoked when entered. To support these functionalities, the class attribute will be valued with element names drawn from the DTBook DTD, e.g., "table," "list," and "note."
  - customTest (IDREF, IMPLIED): ID referencing matching customTest element in head.

  Valid inside: body, seq
  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: `<text ...attributes... />`
  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 whould 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").

 ©NISO

Valid inside: `body, par, seq`

- <audio>
  Description: Points to segment of audio content to be rendered.
  Declaration: `<!ELEMENT audio EMPTY >`
  Syntax: `<audio...`attributes`... />`
  Attributes:
  - id (ID, IMPLIED): Optional identifier.
  - src (CDATA, REQUIRED): URI of audio file containing clip to be rendered.
  - type (CDATA, IMPLIED): Type of media file.
  - clipBegin (CDATA, IMPLIED): 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, "Clock Values."
  - clipEnd (CDATA, IMPLIED): 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`.
  - 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`.

  Valid inside: `body, par, seq`

- <img>
  Description: Points to image to be rendered.
  Declaration: `<!ELEMENT img EMPTY >`
  Syntax: `<image...`attributes`... />`
  Attributes:
  - id (ID, IMPLIED): Optional identifier.
  - src (CDATA, REQUIRED): URI of image 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 image will be presented. References the id of the appropriate `region`.

  Valid inside: `body, par, seq`

- <a>
  Description: Defines a link. The default behavior is to be active for the duration of the media object it contains.
  Declaration: `<!ELEMENT a (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.

  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)

7.4 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 specific structures (at a minimum tables, lists, producer's notes, annotations, and notes) with a single action. To support this functionality, any such structure consisting of multiple time containers (i.e., `seqs` and `pars`) must be wrapped in a `seq`. In addition, a class attribute must be applied to the `seq` or `par` containing a table, list, producer's note, annotation, or note using element names drawn from the DTBook DTD (i.e., "table", "list", "prodnote", "annotation", and "note").

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` or `par` containing a table or list using element names drawn from the DTBook DTD (i.e., "table" and "list"). DTBs and players may also support this functionality for other structures using the same mechanism.

7.4.3 "Skippable" Structures

(This section is normative.)

Players should offer the user the option to "turn off" certain structures in a DTB, that is, select structures such as notes or line numbers that the player will then automatically skip over during sequential playback. To support this capability, compliant DTBs must include `customTest` attributes on `seqs` or `pars` containing those structures. In addition, `customAttributes`, 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 `linenum`, `note`, `noteref`, `annotation`, `pagenum`, optional `prodnote`, and `sidebar`. `Notes`, `annotations`, optional `prodnotes`, and `sidebars` containing multiple paragraphs must be represented as a series of `pars` wrapped in a `seq`, so that a `customTest` can be applied to the `seq`, permitting the user to skip the entire sequence. Attribute values (for customTest attributes on `seqs` or `pars` and for the `id` attribute on `customTest` elements) shall be the names of the "skippable" elements, drawn from the DTBook DTD (e.g., "linenum", "note", etc.) except as noted in the following paragraph.

Different `customTest` attributes may be applied to a single DTBook element, depending on the element's attributes. For example, `<prodnote render="optional">` might be assigned the `customTest "prodnote_opt"`, while `<prodnote render="required">` would not need to be assigned a `customTest` as the user should not have the option of turning it off.

In DTB applications, the element `customTest` will only be used when the

 ©NISO

producer wishes to allow the reader to turn a class of structures on or off, so the `override` attribute on the `customTest` element must always be set to "visible". See description of `<customTest>` above. The SMIL specification chose to make "hidden" the default value so it is critical that the `override="visible"` attribute always be present when the `customTest` element is used.

When a user navigates to a skippable element that has been turned off, the player must 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> tags 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 audio files, when audio is present. The default behavior of a link is to be active for at least the duration of the media 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).

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 (<a> tags) be applied to media objects (normally `audio`) for all `noterefs` and `annorefs`, with the corresponding `notes` and `annotations` as the targets. The presence of the links will enable key player functionality, such as easy access to `notes` when `noterefs` are turned on and `notes` turned off.

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

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).

7.5 SMIL Metadata

(This section is normative.)

`Metadata` is included in the <head> element using the `<meta>` tag. 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, "Clock Values."
  - Occurrence: Required

 ©NISO

    - 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
SYSTEM "dtbsmil110.dtd">
<smil>
     <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`. The initial `seq` includes the attribute "`dur`" which specifies that the entire SMIL file is one hour, three minutes, 24.9 seconds long. Each `par` (a page number, a heading, 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 `par`s.

The `par` for the second paragraph includes a link that "wraps" the `audio` element.

Example 7.2:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE smil SYSTEM "dtbsmil110.dtd">
<smil>

<head>
...

</head>
<body>
          <seq id="baseseq" dur="1:03:24.9">
              <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">
                       <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>
```

 ©NISO

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

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

Notes or sidebars containing multiple paragraphs will be represented as a series of `pars` 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` tag 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" dur="02:14:34.156">
                ... (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>
...
```

7.7 Clock Values

(This section is normative.)

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

Full-clock-val (hours, minutes, seconds, and fractions of seconds): 3:22:55.91

Partial-clock-val (minutes, seconds, and fractions of seconds): 43:15.044

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

If either of the first two formats is used, leading zeroes must be added to single-digit values for minutes and seconds to ensure mm:ss format.

 ©NISO

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 ncx110.dtd (see Appendix 3, "NCX DTD") and comply with the additional normative requirements of section 8.4. The `version` attribute on the `ncx` element of the NCX file must be present and contain the value specified in the above-named DTD. The NCX file itself must be named with the extension ".ncx".

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, navList*)>`
  Syntax: `<ncx ...attributes...>...content...</ncx>`
  Attributes:
  - version (CDATA, FIXED) "1.1.0": Specifies the version of the DTD used in this instance. Three digits with decimal point separators; digits one, two, and three will be updated to reflect major, moderate, and minor changes to the DTD, respectively. This attribute must be present, but parsers will not enforce its presence, just its value.
  - lang (NMTOKEN, IMPLIED): Specifies the [RFC 1766] language code of the language of the document.
  - dir ((ltr|rtl), IMPLIED): The dir attribute specifies the direction of the text, where ltr is left to right and rtl is right to left.

Valid inside: None

- <head>

  Description: Contains 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 will have its attributes duplicated in a smilCustomTest element in the NCX. The NCX thus gathers in one place all customTest elements used in the SMIL files, for presentation to the user.
  Declaration: `<!ELEMENT smilCustomTest EMPTY>`
  Syntax: `<smilCustomTest`...attributes...`/>`
  Attributes:
  - id (ID, REQUIRED)
  - defaultState (true | false) 'false'
  - override (visible | hidden) 'hidden'

  Valid inside: `<head>`

- <meta>

  Description: Contains metadata applicable to the NCX file.
  Declaration: `<!ELEMENT meta EMPTY>`
  Syntax: `<meta` ...attributes...`/>`
  Attributes:
  - name (CDATA, REQUIRED)
  - content (CDATA, REQUIRED)
  - 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. Contains text and an optional pointer to an audio rendering of the title for presentation to the reader.
  Declaration: `<!ELEMENT docTitle (text, audio?)>`
  Syntax: `<docTitle` ...attributes...> ...content... `</docTitle>`
  Attributes:
  - id (ID, IMPLIED): Optional unique identifier.

  Valid inside: `<ncx>`

- <docAuthor>

  Description: The author of the document. Contains text and an optional pointer to an audio rendering of the author for presentation to the reader.
  Declaration: `<!ELEMENT docAuthor (text, audio?)>`
  Syntax: `<docAuthor` ...attributes...>...content...`</docAuthor>`
  Attributes:
  - id (ID, IMPLIED): Optional unique identifier.

  Valid inside: `<ncx>`

- <text>

  Description: Contains the text of the heading or title referenced by the containing element.
  Declaration: `<!ELEMENT text (#PCDATA)>`
  Syntax: `<text` ...attributes...>...content...`</text>`
  Attributes:
  - id (ID, IMPLIED): Optional unique identifier.
  - class (CDATA, IMPLIED): Optional descriptor of this instance of the element.

 ©NISO

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

- `<audio>`

  Description: Contains a pointer to an audio version of the heading or title referenced by the containing element.
  Declaration: `<!ELEMENT audio EMPTY>`
  Syntax: `<audio...attributes.../>`
  Attributes:
  - id (ID, IMPLIED): Optional unique identifier.
  - class (CDATA, IMPLIED): Optional descriptor of this instance of the element.
  - src (CDATA, REQUIRED): The URI of the audio media object. 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."
  - 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, "Clock Values."
  - 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: `<navLabel>,<docTitle>, <docAuthor>`

- `<navMap>`

  Description: Container for primary navigation information.
  Declaration: `<!ELEMENT navMap (navLabel*, navPoint+)>`
  Syntax: `<navMap ...attributes...>...content...</navMap>`
  Attributes:
  - 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. Secondary navigation elements, such as page numbers, 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:
  - id (ID, REQUIRED): Optional unique identifier.
  - onFocus (%script, IMPLIED): Can be used by player to provide location feedback to the user, e.g., highlight the current navPoint on a visual display of the NCX.
  - onBlur (%script, IMPLIED): Can be used by player to provide location feedback to the user, e.g., remove highlighting when focus has shifted to another navPoint on a visual display.
  - class (CDATA, IMPLIED): Describes the kind of structural object represented by this navPoint, e.g. chapter, section; used to select a presentation from the resource file. Player will locate the resource whose `type` attribute equals "ncx", whose `elementRef` attribute value is navPoint, and whose `classRef` references the class of the current `navPoint`.
  - value (CDATA, IMPLIED): An integer representing the text content of a

numeric label, e.g. a page number. Useful for providing integer values for non-Arabic numbers to simplify sorting.
- pageRef (IDREF, IMPLIED): The id of the navTarget representing the page on which this navPoint begins.

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`. `navPoints` may be nested to represent the hierarchical structure of a document.

- `<navLabel>`

Description: Contains a description of a given `<navMap>`, `<navPoint>`, `<navList>`, or `<navTarget>` in various media for presentation to the user. Can be repeated so descriptions can be provided in multiple languages.
Declaration: `<!ELEMENT navLabel (((text, audio?) | audio), img?)>`
Syntax: `<navLabel` ...attributes...`>`...content...`</navLabel>`
Attributes:
- lang (NMTOKEN, IMPLIED): Specifies the [RFC 1766] language code of the language of the heading. Can be used to control presentation of headings in a specified language.
- dir ((rtl|ltr), IMPLIED): Specifies the direction of the text, where ltr is left to right and rtl is right to left.

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

- `<img>`

Description: Information about the location of an image file.
Declaration: `<!ELEMENT img EMPTY>`
Syntax: `<img` ...attributes... `/>`
Attributes:
- id (ID, IMPLIED): Optional unique identifier.
- class (CDATA, IMPLIED): Optional descriptor of this instance of the element.
- src (CDATA, REQUIRED): The URI of the media object.

Valid inside: `<navLabel>`

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

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

- `<navList>`

Description: Container for secondary navigational information.
Declaration: `<!ELEMENT navList (navLabel+, navTarget+)>`
Syntax: `<navList` ...attributes...`>`...content...`</navList>`
Attributes:
- id (ID, IMPLIED): Optional unique identifier.
- class (CDATA, IMPLIED): The kind of objects represented by this `navList`, described by their `dtbook` element name, e.g. `pagenum`, `note`.

Valid inside: `<ncx>`

 ©NISO

Comments: The navList element contains secondary navigation information within navTargets. It is similar to navMap except navTargets may not nest, whereas navPoints can. Used for lists of elements such as page numbers, footnotes, figures, tables, etc. that the user may want to access directly but would clutter up the primary navigation information.

- <navTarget>

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

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

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

  Attributes:
  - id (ID, REQUIRED): Unique identifier matching the id attribute of the corresponding element in the text and/or SMIL files.
  - onFocus (CDATA, IMPLIED): Can be used by player to provide location feedback to the user, e.g., highlight the current navTarget on a visual display of the NCX.
  - onBlur (CDATA, IMPLIED): Can be used by player to provide location feedback to the user, e.g., remove highlighting when focus has shifted to another navTarget on a visual display.
  - value (CDATA, IMPLIED): An integer representing the text content of a numeric label, e.g. a page number. Useful for providing integer values for non-Arabic numbers to simplify sorting.
  - class (CDATA, IMPLIED): Describes the kind of object represented by this navTarget, described by its DTBook element name, e.g., pagenum, note. It may be used to select a presentation from the resource file. Player will locate the resource whose type attribute equals "ncx", whose elementRef attribute value is navTarget, and whose classRef references the class of the current navTarget.
  - mapRef (IDREF, REQUIRED): Pointer to the innermost navPoint that contains this navTarget.

  Valid inside: <navList>

  Comments: The navTarget element contains one or more navLabels representing the referenced part of the document, e.g., a page or footnote, along with a pointer to content. The mapRef attribute is used to synchronize the navList and navMap.

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 tag. 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
  - 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."
  - Occurrence: Required
- dtb:depth
  - Content: Positive integer indicating depth of structure of the DTB as exposed by the NCX.
  - Occurrence: Required
- dtb:generator
  - Content: Name and version of software that generated the NCX.
  - Occurrence: Optional - recommended.
- dtb:maxPageNormal
  - Content: Non-negative integer indicating the content of the highest "normal" `pagenum` in the DTB.
  - Occurrence: Required
- dtb:pageFront
  - Content: Non-negative integer indicating the number of "front" pages in the DTB.
  - Occurrence: Required
- dtb:pageNormal
  - Content: Positive integer indicating the number of "normal" pages in the DTB.
  - Occurrence: Required
- dtb:pageSpecial
  - Content: Non-negative integer indicating the number of "special" pages in the DTB.
  - 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 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 mapRef Attribute

(This section is normative.)

The mapRef attribute on each `navTarget` must reference the innermost `navPoint` that contains the element referenced by the `navTarget`.

8.4.4 smilCustomTest Element

(This section is normative.)

Each unique `customTest` element that appears in one or more SMIL files must have its attributes duplicated in a `smilCustomTest` element in the `head` of the NCX.

 ©NISO

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 `navPoints` 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)
    - History.....(Level 2)
    - Development of Standards.....(Level 2)
- Standards.....(Level 1)
    - 1 Core Services.....(Level 2)
        - 1.1.....(Level 3)
            - a.....(Level 4)
        - 1.2.....(Level 3)

Foreword and Standards are at the same level, in this case the highest level, Level 1. The nesting of `navPoints` 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 local navigation. 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 `navLists`. Unlike `navMap`, `navLists` do not represent the structure of the book by nesting `navTargets`. In Example 8.1, there are two `navLists`: the first contains three `navTargets` representing page numbers, and the second contains three `navTargets` representing notes.

Each `navPoint` or `navTarget` provides navigation information about one piece of the document, e.g. a chapter heading, section number, page number, figure, etc. The `text` element contains the actual heading, page number, etc. for visual or text-to-speech presentation; the `audio` element uses SMIL 2.0 syntax to point to a clip containing the audio presentation of the same information. One or both 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 `mapRef` attribute of `navTarget` allows synchronization of `navLists` with the `navMap`. `mapRef` points to the innermost `navPoint` that contains the page number, note, or other element referenced by the `navTarget`. Similarly, the `pageRef` attribute of `navPoint` points to the `navTarget` representing the page on which the `navPoint` begins.

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 v1.1.0//EN"
    "http://www.loc.gov/nls/z3986/v100/ncx110.dtd"
>
<ncx version="1.1.0">
   <head>
     <smilCustomTest id="pagenum" defaultState="false"
     override="visible"/>
     <smilCustomTest id="note" defaultState="true" override="visible"/>
     <meta name="dtb:uid" content="us-nls-00001"/>
     <meta name="dtb:depth" content="6"/>
     <meta name="dtb:generator" content="NLSv001"/>
     <meta name="dtb:pageNormal" content="47"/>
     <meta name="dtb:pageSpecial" content="0"/>
     <meta name="dtb:pageFront" content="5"/>
     <meta name="dtb:maxPageNormal" 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" />
   </docTitle>
   <docAuthor>
       <text>Association of Specialized and Cooperative Library
       Agencies</text>
       <audio src="rs_title.mp3" />
   </docAuthor>

   <navMap>
       <navPoint class="chapter" id="lvl1_3" pageRef="p1">
          <navLabel>
           <text>Foreword</text>
           <audio src="rs_fwdx.mp3" clipBegin="00:01.5"
           clipEnd="00:02.0" />
          </navLabel>
          <content src="sample.smil#h1_3" />
          <navPoint class="section" id="lvl2_1" pageRef="p1">
           <navLabel>
              <text>History</text>
              <audio src="rs_fwdx.mp3" clipBegin="00:03.4"
              clipEnd="00:03.9" />
          </navLabel>
          <content src="sample.smil#h2_1" />
       </navPoint>
       <navPoint class="section" id="lvl2_2" pageRef="p2">
          <navLabel>
```

 ©NISO

```
            <text>Development of Standards</text>
            <audio src="rs_fwdx.mp3" clipBegin="00:56.3"
            clipEnd="00:57.7" />
         </navLabel>
         <content src="sample.smil#h2_2" />
      </navPoint>
    </navPoint>
    <navPoint class="chapter" id="lvl1_7" pageRef="p16">
      <navLabel>
         <text>Standards</text>
         <audio src="rs_stdx.mp3" clipBegin="00:01.3"
         clipEnd="00:02.1" />
      </navLabel>
      <content src="sample.smil#h1_7" />
      <navPoint class="section" id="lvl2_11" pageRef="p16">
        <navLabel>
         <text>1 Core Services</text>
         <audio src="rs_stdx.mp3" clipBegin="00:02.9"
         clipEnd="00:04.9" />
      </navLabel>
      <content src="sample.smil#h2_10" />
      <navPoint class="subsection" id="lvl3_1" pageRef="p16">
        <navLabel>
         <text>1.1</text>
         <audio src="rs_stdx.mp3" clipBegin="00:05.7"
         clipEnd="00:06.7" />
      </navLabel>
      <content src="sample.smil#h3_1" />
      <navPoint class="sub-subsection" id="lvl4_1" pageRef="p16">
        <navLabel>
         <text>a.</text>
         <audio src="rs_stdx.mp3" clipBegin="00:18.7"
         clipEnd="00:19.1" />
        </navLabel>
        <content src="sample.smil#h4_1" />
      </navPoint>
     </navPoint>
     <navPoint class="subsection" id="lvl3_2" pageRef="p16">
       <navLabel>
         <text>1.2</text>
         <audio src="rs_stdx.mp3" clipBegin="00:50.5"
         clipEnd="00:51.4" />
       </navLabel>
       <content src="sample.smil#h3_2" />
     </navPoint>
   </navPoint>
 </navPoint>
</navMap>

<navList id="pages" class="pagenum">
  <navLabel>
     <text>Pages</text>
     <audio src="navlabels.mp3" clipBegin="00:00" clipEnd="00:01.1" />
  </navLabel>
```

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

  <navList id="notes" class="note">
    <navLabel>
      <text>Notes</text>
      <audio src="navlabels.mp3" clipBegin="00:01.5" clipEnd="00:02.6" />
    </navLabel>
    <navTarget class="note" id="nref_1" mapRef="lvl2_2">
      <navLabel>
        <text>1</text>
        <audio src="rs_fwdx.mp3" clipBegin="01:22.6" clipEnd="01:23.5" />
      </navLabel>
       <content src="sample.smil#nref_1" />
     </navTarget>
     <navTarget class="note" id="nref_2" mapRef="lvl2_2">
       <navLabel>
         <text>2</text>
         <audio src="rs_fwdx.mp3" clipBegin="02:00.6" clipEnd="02:01.4" />
       </navLabel>
       <content src="sample.smil#nref_2" />
     </navTarget>
     <navTarget class="note" id="nref_3" mapRef="lvl2_2">
       <navLabel>
         <text>3</text>
         <audio src="rs_fwdx.mp3" clipBegin="03:13.3" clipEnd="03:14.1" />
       </navLabel>
       <content src="sample.smil#nref_3" />
     </navTarget>
   </navList>
</ncx>
```


©NISO

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 bookmark100.dtd, the DTD for Portable Bookmarks/Highlights found in Appendix 4. Similarly, a player with bookmark/highlight import capabilities must correctly process bookmarks and highlights and their associated notes that are formatted in accordance with boomark100.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 audio file 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 audio files shall be identified by `<timeOffset>` in fractional seconds (Seconds = DIGIT+, Fraction = 3DIGIT). Offsets from element boundaries in textual content files shall be identified by `<charOffset>`, measured in characters, counting from the nearest previous tag with an id; white space is normalized (collapsed to one character) and tags are not counted.

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.")

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: "setpb-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 found in Appendix 4, along with descriptions of any applicable attributes.

- <bookmarkSet>
  Description: The root element in the Bookmark/Highlight DTD. Contains all data pertaining to bookmarks, highlights, and lastmarks for a given DTB.
  Declaration: `<!ELEMENT bookmarkSet (title, uid, lastmark?, (bookmark | hilite)*) >`
  Syntax: `<bookmarkSet>`...content...`</bookmarkSet>`
  Attributes: None
  Valid inside: None
- <title>
  Description: Contains the title, in text and in an optional audio clip, of the DTB for which the bookmark set was created.
  Declaration: `<!ELEMENT title (text, audio?) >`
  Syntax: `<title>`...content...`</title>`
  Attributes: None
  Valid inside: `<bookmarkSet>`
  Comments: When bookmark sets are exported to other compliant playback devices, the title will allow users to identify and manage them.
- <text>
  Description: Text of title or note.
  Declaration: `<!ELEMENT text (#PCDATA)>`
  Syntax: `<text>`...content...`</text>`
  Attributes: None
  Valid inside: `<title>`,`<note>`
- <audio>
  Description: Audio clip of title 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:
  - src (%uri, #REQUIRED): The src attribute holds the URI of the audio file that contains the referenced clip.
  - clipBegin (CDATA, IMPLIED): The `clipBegin` attribute specifies the beginning of a segment of a continuous media object as a time offset from the start of the media object. The value syntax is defined by the SMIL 2.0 Timing and Synchronization Module[SMIL].
  - clipEnd (CDATA, IMPLIED): The `clipEnd` attribute specifies the end of a segment of a continuous media object as a time offset from the start of the media object. It uses the same attribute value syntax as `clipBegin`.

  Valid inside: `<title>, <note>`
- <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>`

 ©NISO

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 or character offset to the exact point.
  Declaration: `<!ELEMENT lastmark (ncxRef, uri, (timeOffset | charOffset)) >`
  Syntax: `<lastmark>`...content...`</lastmark>`
  Attributes: None
  Valid inside: `<bookmarkSet>`
  Comments: The `<lastmark>` is set automatically by the playback device.
- <ncxRef>
  Description: Captures current location in NCX (the id of the current `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, to id in text-only file, or to audio file that contains the `<lastmark>, <bookmark>, <hiliteStart>`, or `<hiliteEnd>`.
  Declaration: `<!ELEMENT uri (#PCDATA)>`
  Syntax: `<uri>`...content...`</uri>`
  Attributes: None
  Valid inside: `<lastmark>,<bookmark>, <hiliteStart>, <hiliteEnd>`
  Comments: The `<uri>` points to the id of the `<par>` or `<seq>` in the SMIL file that contains the `<lastmark>, <bookmark>, <hiliteStart>`, or `<hiliteEnd>`.
- <timeOffset>
  Description: Exact position of `<lastmark>, <bookmark>, <hiliteStart>`, or `<hiliteEnd>` in audio file referenced (via the SMIL file) by the URI; in seconds, measured from beginning of audio file.
  Declaration: `<!ELEMENT timeOffset (#PCDATA) >`
  Syntax: `<timeOffset>`...seconds.fraction...`</timeOffset>`
    - Seconds = DIGIT+
    - Fraction = 3DIGIT

  Attributes: None
  Valid inside: `<lastmark>, <bookmark>, <hiliteStart>, <hiliteEnd>`
- <charOffset>
  Description: Exact position of `bookmark, lastmark, hiliteStart`, or `hiliteEnd` in textual content file referenced by the URI.
  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 or character offset to the exact point.

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

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

  Attributes:

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

  Valid inside: `<bookmarkSet>`

- `<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:

  - 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 or character offset to the exact point.

  Declaration: `<!ELEMENT hiliteStart (ncxRef, uri, (timeOffset | 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 or character offset to the exact point.

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

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

  Attributes: None

  Valid inside: `<hilite>`

 ©NISO

9.3 Examples

(This section is informative.)

In Example 9.1, the reader has set two bookmarks, one in chapter 1, 22 seconds from the start of paragraph 8, and the other in chapter 3, 88 seconds from the start of paragraph 12. The reader has added the text note "Atlanta burns" to the second bookmark. The user has also highlighted a passage in chapter 4 beginning at the start of paragraph 1 and ending 246 seconds after the start of paragraph 6, labeling it with a ten-second audio comment. The reader last stopped reading (as indicated by the `<lastmark>`) in chapter 5, paragraph 23. The default filename for this bookmark file would be "us-rfbd-JT065.bmk".

Example 9.1:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE bookmarkSet SYSTEM "bookmark100.dtd">

<bookmarkSet>

     <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>173</timeOffset>
     </lastmark>

     <bookmark>
       <ncxRef>gwtw.ncx#lvl1_1</ncxRef>
       <uri>gwtw_ch1.smil#para008</uri>
       <timeOffset>22</timeOffset>
     </bookmark>
     <bookmark>
       <ncxRef>gwtw.ncx#lvl1_3</ncxRef>
       <uri>gwtw_ch3.smil#para012</uri>
       <timeOffset>88</timeOffset>
       <note>
         <text>Atlanta burns.</text>
       </note>
     </bookmark>

     <hilite>
       <hiliteStart>
         <ncxRef>gwtw.ncx#lvl1_4</ncxRef>
         <uri>gwtw_ch4.smil#para001</uri>
         <timeOffset>0</timeOffset>
       </hiliteStart>
       <hiliteEnd>
         <ncxRef>gwtw.ncx#lvl1_4</ncxRef>
```

```
          <uri>gwtw_ch4.smil#para006</uri>
          <timeOffset>246</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 SYSTEM "bookmark100.dtd">
<bookmarkSet>

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

     <lastmark>
       <ncxRef>chemtd.ncx#lvl1_3</ncxRef>
       <uri>chemtd.xml#para297</uri>
       <charOffset>130</charOffset>
     </lastmark>

   </bookmarkSet>
```

10. Resource File

10.1 Introduction

(This section is informative.)

The optional Resource File supplies text segments and pointers to audio clips or images that can assist the reader in using a DTB. These media objects or "resources" provide information missing from a document or present only in a form inaccessible to the reader. Some examples of applications are:

1. Documents with definite structures but missing headings, such as books with multilevel structures but no headings on items below the level of sections. The resource file could contain the word "subsection" for presentation to the reader when stepping through the document via the NCX.
2. Player implementations that present generic labels such as "level 2" when the user changes levels in the NCX. The resource file could present the actual name of items found at that level in that specific context, e.g., "chapter".

 ©NISO

3. "Where Am I?" applications, especially in situations where headings are absent.
4. Detailed information about the marked-up 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 all of the element names in the DTBook DTD, alerting the user when crossing the boundaries of paragraphs, list items, table cells, etc.
5. Information about the types of text structures that the user can choose to "turn off" via the SMIL file. (See section 7.4.3, "'Skippable' Structures.") The Resource File could contain text segments or pointers to audio clips of the names of the structures affected; for example, page numbers, notes, or sidebars.

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

1. Resources tailored to a given document, for use primarily during global navigation. As the user navigates via the NCX, the player, when necessary, will look in the Resource File to locate the `resource` whose `type` attribute equals "ncx", whose `elementRef` attribute value is `navPoint` or `navTarget` as appropriate, and whose `classRef` references the class of the current NCX `navPoint` or `navTarget`.

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

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

3. Representations of skippable structures listed in the `head` of the NCX. The player will locate the `resource` whose type attribute equals "ncx", whose `elementRef` attribute value is `smilCustomTest` and whose idRef attribute references the id of the current `smilCustomTest` element. For example, the `smilCustomTest` element tagged `<smilCustomTest id="prodnote" />`) would call the `resource` with `idRef` equal to "prodnote".

The text, audio, and image alternatives allow a `resource` to be presented in a medium appropriate to the playback system's capabilities and the user's preferences. Images are conceived as holding iconic representations of heading types. The `lang` attribute on the `resource` element allows alternative representations to be supplied in multiple languages.

Resources would be called only when appropriate; that is, in response to clear user requirements and when needed. For example, a `resource` with `type="ncx"` and `classRef="chapter"` would not be called if a chapter heading with textual and audio content was already present.

(This section is normative.)

If a Resource File is implemented, it must meet the following requirements. The Resource File is a valid XML 1.0 file conforming to the Document Type Definition resource110.dtd. (See Appendix 5, "DTD for Resource File.") The `version` attribute on the `resources` element of any compliant Resource File must be present and

contain the value drawn from the above-named DTD. The Resource File shall be named with the extension ".res". Identical copies of the Resource File shall be distributed on each media unit of the DTB.

10.2 Resource Elements

(This section is informative.)

Brief descriptions of resource the 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?, resource+) >`
  Syntax: `<resources...attribute>...content...</resources>`
  Attributes:
    - version (CDATA, FIXED): Specifies the version of the DTD used in this instance. Three digits, with decimal point separators. Digits one, two, and three will change with major, moderate, and minor changes to the DTD, respectively. This attribute must be present but parsers will not enforce its presence, just its value.

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

  Valid inside: `<head>`
- <resource>
  Description: Contains text and pointers to audio clip or image serving as an alternative representation of elements or element content in textual content file and NCX.
  Declaration: `<!ELEMENT resource (((text, audio?) | audio), img?)>`
  Syntax: `<resource...attributes...>...content...</resource >`
  Attributes:
    - type (dtbook | ncx) REQUIRED: Specifies whether the resource applies to the textual content file (`dtbook`) or the NCX (ncx).
    - elementRef (CDATA, REQUIRED): Specifies the name of the element for which the `resource` is to be supplied.
    - classRef (CDATA, IMPLIED): Specifies the class attribute value of the element for which the resource is to be supplied. See section 10.3, "Resource File Requirements" for normative content.
    - idRef (CDATA, IMPLIED): Specifies the name of the id attribute on the

 ©NISO

smilCustomTest element in NCX for which the resource is to be supplied.
    - lang (NMTOKEN, IMPLIED): Specifies the language of the resource item, using an [RFC 1766] language code.

  Valid inside: <resources>

- <text>

  Description: Contains the text used for alternative representation.
  Declaration: <!ELEMENT text (#PCDATA) >
  Syntax: <text>...content...</text>
  Attributes: None
  Valid inside: <resource>

- <audio>

  Description: Points to file containing audio representation and provides time offsets of beginning and end of clip.
  Declaration: <!ELEMENT audio EMPTY >
  Syntax: <audio...attributes... />
  Attributes:
    - src (%URI, REQUIRED): URI of the audio file.
    - clipBegin (%SMILtimeVal, IMPLIED): 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, "Clock Values."
    - clipEnd (%SMILtimeVal, IMPLIED): 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: <resource>

- <img>

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

  Valid inside: <resource>

10.3 Resource File Requirements

(This section is normative.)

If a player implementing resource functionality for DTBook elements encounters an element in the textual content file that includes a class attribute, the player must present the associated resource with the corresponding classRef, if one exists. Otherwise, if the appropriate resource without a classRef exists, the player must present it.

10.4 Examples

(This section is informative.)

In Example 10.1, the Resource File contains a resource for the word "chapter" to be presented when encountering navPoints of this class in the NCX. Resources are supplied for four selected DTBook elements; the last of these resources uses the classRef attribute to specify a given class of the element code. Finally, a resource is provided for a smilCustomTest with an id of "prodnote".

Example 10.1:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE resources PUBLIC "-//NISO//DTD resource v1.1.0//EN"
"http://www.loc.gov/nls/z3986/v100/resource110.dtd">
<resources version="1.1.0">

     <resource type="ncx" elementRef="navPoint" classRef="chapter"
     lang="en">
        <text>Chapter</text>
        <audio src="chapter.mp3" />
        <img src="chapter.png" />
     </resource >
     <resource type="dtbook" elementRef="li" lang="en">
        <text>list item</text>
        <audio src="elemres.mp3" clipBegin="00:36"
        clipEnd="00:38.14" />
     </resource>
     <resource type="dtbook" elementRef="p" lang="en">
        <text>paragraph</text>
        <audio src="elemres.mp3" clipBegin="00:47.51"
        clipEnd="00:49.34" />
     </resource>
     <resource type="dtbook" elementRef="td" lang="en">
       <text>table cell</text>
       <audio src="elemres.mp3" clipBegin="01:22.12"
       clipEnd="01:24.01" />
     </resource>
     <resource type="dtbook" elementRef="code"
     classRef="javascript" lang="en">
       <text>javascript</text>
       <audio src="elemres.mp3" clipBegin="01:45.15"
       clipEnd="01:47.01" />
     </resource>
     <resource type="ncx" elementRef="smilCustomTest"
     idRef="prodnote" lang="en">
       <text>producer's note</text>
       <audio src="elemres.mp3" clipBegin="01:54.17"
       clipEnd="01:56.44" />
     </resource>
...

</resources >
```

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

Example 10.2:

```
   ...

     <resource type="ncx" elementRef="navPoint" classRef="chapter"
     lang="da">
```

 ©NISO

```
      <text>Kapitel</text>
      <audio src="kapitel.mp3" clipBegin="00:00" clipEnd="00:02.23" />
      <img src="Kapitel.png" />
    </resource>
    <resource type="ncx" elementRef="navPoint" classRef="chapter"
    lang="en">
      <text>Chapter</text>
      <audio src="chapter.mp3" clipBegin="00:00" clipEnd="00:02.01" />
      <img src="chapter.png" />
    </resource>
...
```

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. Retracing path back to point of origin after following a link that required inserting a new disk.
5. 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.
6. 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.
7. 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 distInfo110.dtd (see Appendix 6, "Distribution Information DTD"), and shall be named "distInfo.dinf". The `version` attribute on the `distInfo` element must be present and contain the value drawn from the above-named DTD.

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").

Values for the attribute `media` on the element `<book>` and for the attribute `mediaRef` on the elements `smilref` and `changeMsg` 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+) >`
  Syntax:`<distInfo...attribute...>...content...</distInfo>`
  Attributes:
    - version (CDATA, FIXED) "1.1.0": Specifies the version of the DTD used in this instance. Three digits with decimal point separators; digits one, two, and three will be updated to reflect major, moderate, and minor changes to the DTD, respectively. This attribute must be present, but parsers will not enforce its presence, just its value.

  Valid inside: None
- `<book>`
  Description: Identifies a DTB that is present, in part or whole, on this piece of distribution media.
  Declaration: `<!ELEMENT book (distMap?, changeMsg*)>`
  Syntax: `<book ...attributes...>...content...</book>`
  Attributes:
    - 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

 ©NISO

section 3.1, "Package Identity."

- 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.
- 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 sequence 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 `distMaps` and zero or more `changeMsgs`.

- <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 `smilRefs`.

- <smilRef>

Description: A reference to a DTB SMIL file.

Declaration: `<!ELEMENT smilRef EMPTY >`

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

Attributes:

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

- <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:

- 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.
  - lang (NMTOKEN, IMPLIED): Specifies the [RFC 1766] language code of the language in which the message is presented.

Valid inside: `<book>`

- <text>

Description: Contains text of media change message.

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

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

Attributes: None

Valid inside: `<changeMsg>`

- <audio>

Description: Pointer to audio clip of media change message.

Declaration: `<!ELEMENT audio EMPTY>`
Syntax: `<audio...attributes.../>`
Attributes:

- src (%URI, IMPLIED): URI of audio content of media change message.
- clipBegin (CDATA, IMPLIED): 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, "Clock Values."
- clipEnd (CDATA, IMPLIED): 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>`

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 SYSTEM "distInfo110.dtd">

<distInfo version="1.1.0">

          <book uid="us-rfbd-tbfz284" pkgRef="./FZ284.opf" media="1:3">

              <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>
```

 ©NISO

```
            <changeMsg mediaRef="1:3">
               <text>Insert disc one.</text>
               <audio src="insert.wav" clipBegin="00:00"
               clipEnd="00:02.256"/>
            </changeMsg>
            <changeMsg mediaRef="2:3">
               <text>Insert disc two.</text>
               <audio src="insert.wav" clipBegin="00:03.002"
               clipEnd="00:05.881"/>
            </changeMsg>
            <changeMsg mediaRef="3:3">
               <text>Insert disc three.</text>
               <audio src="insert.wav" clipBegin="00:06.901"
               clipEnd="00:10.003"/>
            </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. Players would parse the package files to obtain book metadata, etc.

Example 11.2:

```
<?xml version="1.0"  encoding="UTF-8"?>
<!DOCTYPE distInfo SYSTEM "distInfo110.dtd">
<distInfo version="1.1.0">

     <book uid="us-nls-db00001" pkgRef="./book1/AllAboutDogs.opf" />
     <book uid="us-nls-db98765" pkgRef="./book2/AllAboutCats.opf" />

</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. Types of DTB

13.1 Types

(This section is informative.)

Digital talking books produced in compliance with this standard fall into six types representing the proportions in which six key files are present. In all six types, the Package File `spine` defines the linear reading order of the DTB. A DTB that incorporates audio and textual content files for the full contents of the document, as well as a structured navigation control file (NCX) for efficient navigation (type 4 - audioFullText), offers the most features to a reader.

 ©NISO

Type 1 — Full audio only (dtb:multimediaType = audioOnly)
The full contents of the document are present in recorded speech. The DTB contains no structure. The navigation control file (NCX) contains only the title of the book and a pointer to the first SMIL file. The SMIL files contain only `<audio>` elements in sequence. This Type of DTB will be represented primarily by "legacy" titles transferred from analog to digital form. Direct navigation to points within the DTB is not possible for the reader.

Type 2 — Full audio with structure (dtb:multimediaType = audioNCX)
The full contents of the document are present in recorded speech. The NCX contains (via SMIL) links to the structural elements of the book and can contain links to features such as page numbers, narrated footnotes, etc. The SMIL files contain only `<audio>` elements in sequence. The reader can navigate directly only to items included in the NCX.

Type 3 — Full or partial audio with structure and partial text (dtb:multimediaType = audioPartText)
The full or partial contents of the document are present in recorded speech. The NCX contains (via SMIL) links to the structural elements of the book and can contain links to features such as page numbers, narrated footnotes, etc. In addition, part of the document is present as a textual content file. The segments included in the textual content file might be chosen either because keyword searching, spelling, and direct access to the text would be beneficial in those segments (e.g., glossaries), or because a synthetic speech rendering would be as useful as a human speech version (e.g., indexes). Images can also be included. The reader can navigate directly to items in the NCX and to tagged items in the textual content file. Where text renditions of a segment are present, they are synchronized with the corresponding audio (and any associated images) in the SMIL files; elsewhere, SMIL contains only `<audio>` elements in sequence.

Type 4 — Full audio with structure and full text (dtb:multimediaType = audioFullText)
The full contents of the document are present in recorded speech. The NCX contains (via SMIL) links to the structural elements of the book and can contain links to features such as page numbers, narrated footnotes, etc. The full text of the document is present as a textual content file. Images can also be included. Text, audio, and any images are synchronized in the SMIL files. The reader can navigate directly to items in the NCX and to tagged items in the textual content file.

Type 5 — Full text with structure and partial audio (dtb:multimediaType = textPartAudio)
The full text of the document is present as a textual content file but only part of the document is present as recorded speech. Images can also be included. The NCX contains (via SMIL) links to the structural elements of the book and can contain links to features such as page numbers, narrated footnotes, etc. The reader can navigate directly to items in the NCX and to tagged items in the textual content file. Where audio renditions of a segment are present, they are synchronized with the corresponding text (and any associated images) in the SMIL files; elsewhere, SMIL contains `<text>` (and any synchronized `<image>`) elements in sequence. This type includes books such as dictionaries, where the full text is present but the only audio contains human speech recordings of word pronunciations.

Type 6 — Full text with structure but no audio (dtb:multimediaType = textNCX)
The full text of the document is present as a textual content file. The NCX contains (via SMIL) links to the structural elements of the book and can contain links to features such as page numbers. The reader can navigate directly to items in the NCX and to tagged items in the textual content file. Images can be included. The SMIL files contain `<text>` elements in sequence, synchronized with any images present. There are no audio files.

13.2 Required Files

(This section is normative.)

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

DTB Type	OPF	NCX	Audio	Text	SMIL	Image
audioOnly (Full audio only)	R	R	R	N/A	R	N/A
audioNCX (Full audio+structure)	R	R	R	O	R	N/A
audioPartText (Audio+structure+partial text)	R	R	R	R	R	O
audioFullText (Audio+structure+full text)	R	R	R	R	R	O
textPartAudio (Full text+structure+partial audio)	R	R	R	R	R	O
textNCX (Full text+structure, no audio)	R	R	N/A	R	R	O

13.3 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 screen may be presented, according to the user's preferences and the features provided on the playback system. If only an audio file is present, straight audio playback 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 the DTB in any way, based on the value of `dtb:multimediaType` in the package file metadata, it must report this fact to the user. Further, a playback system should inform the user when unable to render in any way specific content it encounters in the DTB.

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

 ©NISO

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 `timeOffsets`, 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 the standard can be enforced through the DTDs included in this standard, others cannot, and must be enforced through other means.

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.din.de/gremien/nas/nabd/iso3166ma/codlstp1/

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 1.0.1: http://www.openebook.org

RFC 1766
: RFC 1766 - Tags for the Identification of Languages: http://www.ietf.org/rfc/rfc1766.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

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

 ©NISO

SMIL
SMIL 2.0 W3C Recommendation 07 August 2001: http://www.w3.org/TR/2001/REC- smil20-20010807/
SVG
Scalable Vector Graphics http://www.w3.org/TR/2001/REC-SVG-20010904/
XML
XML Version 1.0: http://www.w3.org/TR/2000/REC-xml-20001006.html
XML-Style
Associating Style Sheets with XML Documents 1.0: http://www.w3.org/TR/xml-stylesheet/
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.loc.gov/nls/z3986/v100/dtbook110doc.htm
DTBook Theory
Theory behind the DTBook DTD: http://www.daisy.org/publications/specifications/theory/dtbook/
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
StructGuide
Structure Guidelines: http://www.daisy.org/products/menupps.htm
UAAG
User Agent Accessibility Guidelines: http://www.w3.org/TR/UAAG10/
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

Appendix 1 - DTBook DTD

(This section is normative.)

The following DTD is available in plain-text form from the maintenance agency at http://www.loc.gov/nls/z3986/v100/.

```
<!-- DTBook DTD V1.1.0 2002-02-27 -->
<!-- file: dtbook110.dtd  (Note: Update version attribute on dtbook element
     when version changes.)-->
<!--dtbook110 Digital Talking Book XML Document Type Definition
     Implementing the ANSI/NISO Digital Talking Book V1.1.0 Document
     Tagging Requirements

     Harvey Bingham <hbingham@acm.org>
     George Kerscher <kerscher@montana.com>
     Michael Moodie <mmoo@loc.gov>
     David Pawson <dpawson@rnib.org.uk>

     Assisted by DAISY Consortium and NISO DTB Committee work teams.

     1. Purpose.

     The Digital Talking Book Document Type Definition (DTD) provides
     the means to mark up the text of a document to permit support for
     the combination of professional narration and navigation into that
     narration. It also facilitates the output of a document's content in
     a variety of accessible formats. The markup tags in the book convey
     its content in structure, and contain some metadata about the book
     content and its structure.

     The Document Type Definition names and defines the allowable element
     types, their allowable content, and their attributes. Correct markup
     of the text of the book permits the textual material to be synchronized
     using SMIL [SMIL2.0] files with the professionally narrated version of
     that book. The synchronization can permit concurrent display of the
     text being narrated. The textual content can be searched in context to
     locate material desired for narration.

     More detailed documentation of this dtbook dtd [DTBOOKV110DTD] is
     available as an html document. See [DTBOOKV110DOC].

     1.1. Prior Related Work

     The DAISY (Digital Audio-based Information SYstem) Consortium
     contributed substantially to the development of this DTD.

     This application of XML is the next generation after several DAISY
     versions of 2.X specifications, see [DAISY202].

     The DAISY Statement of Principles for the Creation and Production
     of Accessible Books and Materials [DAISY-2-GUIDELINES] represents
     the minimum standard to be met by Libraries of the Blind and producers
     of alternative format materials.

     Its Navigation Control Center (NCC) provided for synchronizing
     document structure with narration.
```

 ©NISO

The NCC evolved into an XML application called the "Navigation Control File for XML applications" (NCX). Its content is derived from the markup of documents tagged using the dtbook DTD. Richer structuring capability is one of the objectives of that DTD. The Synchronized Multimedia Integration Language [SMIL2.0] is used to provide synchronized narrations and text. The NCX provides navigation using the identified elements of documents tagged to this DTD.

The dtbook DTD includes many, but not all, of the element types found in both the [HTML401STRICT] and [XHTML11STRICT] strict DTDs. HTML authoring tools permit those additional element tags, and may ignore the additional tags that are dtbook-specific. The lowercase names from XHTML are used, rather than the uppercase names from HTML.

1.2. Evolution from HTML and XHTML

Dtbook110 has 79 element types. It shares 47 element types with the HTML4.0 Strict DTD [HTML401STRICT] (as adjusted to use the lower-case names consonant with the XHTML Strict DTD [XHTML11STRICT]). It omits 30 element types from them, and has 32 unique element types.

Endtag markup is sometimes optional in HTML. It is required for use with xhtml and dtbook. Any XML application [XML12] requires endtags, or their abbreviated form for empty elements, such as "<br />". The benefit of including endtags is that the tagged document has dependable structure that can be validated against the dtbook dtd.

Some tools available for browsing HTML may be used with dtbook material, at the expense of their discarding or ignoring some specific tagging and attributes that are not part of HTML 4.0. A CSS-based stylesheet [CSS1] or [CSS2] that identifies the presentation expectations for the HTML and non-HTML tags, or a filter to map those tags onto suitable HTML tags can provide appropriate visual presentation.

2. Document Tagging Content

A Digital Talking Book document is an XML application. Therefore, it must begin with the XML processing instruction, followed by the DOCTYPE declaration.

2.1. XML Processing Instruction

The XML Processing Instruction identifies the version of XML, and the optional character set encoding for the document:

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

2.2. Character Set Encodings

The character set in which the document is encoded is identified by one of a number of strings. All XML applications are expected to be able to recognize the UNICODE/ISO/IEC 10646 encodings "UTF-8" and "UTF-16" [ISO10646].

Some alternative encodings to "UTF-8" (or "ISO-10646-UCS-2") or "UTF-16" (or "ISO-10646-UCS-4") may be used. These include "ISO-8859-1", "ISO-8859-2", ... "ISO-8859-9" for parts of ISO 8859.

See [ISO8859]. Note that US-ASCII (i.e. encoding all characters over decimal 127, e.g. from 128 to 255, as &#nnn;) is conformant with UTF-8 (and ISO-8859-1, HTTP's default header encoding.)

Also, the values "ISO-2022-JP", "Shift_JIS", or "EUC-JP" can be used for various Japanese encoded forms of JIS X-0208-1997. See [JIS].

The Unicode characters may be represented as their code points, using the form &#hHHHH; where HHHH is a hexadecimal value formed from the digits 0-9 and letters A-F. Any initial H with value "0" may be elided.

2.3. DOCTYPE Declaration

The document type declaration, the DOCTYPE, follows. It has several forms.

The simpler form assumes that the proper version of the dtbook DTD is in the same directory as the dtbook file itself.

```
<!DOCTYPE dtbook SYSTEM
    "dtbook110.dtd">
```

A more general form provides the PUBLIC URI from which the SYSTEM filename can be substituted, should that system copy be missing:

```
<!DOCTYPE dtbook PUBLIC
    "http://www.loc.gov/nls/z3986/v100/dtbook110.dtd"
    "dtbook110.dtd">
```

That assumes the URI can be reached, which may not be true for portable dtbook players.

The still more general form recommended for xml applications [XML12] is:

```
<!DOCTYPE dtbook PUBLIC
    "-//NISO//DTD dtbook v1.1.0//EN"
    "dtbook110.dtd">
```

where the Formal Public Identifier (FPI) on the second line is converted to the URI where it may be publicly found:

```
http://www.loc.gov/nls/z3986/v100/dtbook110.dtd
```

The [OASIS-TR9401] Entity Management Catalog provides an indirect means to provide that mapping from FPI to the dtd.

That catalog is more generally useful to provide the mapping from any external entity names (such as modules) to URIs where they may be found.

Note that the reference above is to a particular version of the DTD, distinguished by the "v110".

2.4. Digital Talking Book File MIME Type

A Digital Talking Book document is tagged to the dtbook XML application. Its MIME media-type is "text/xml". The tagged book

 ©NISO

filename should have suffix ".xml". See [RFC2045].

3. Modular Extension to the DTD

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

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

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

A book can augment the dtbook DTD by including other declarations or parameter entity references in the internal subset of declarations. The internal subset may occur in square brackets following the ExternalID and before the concluding ">" of the initial DOCTYPE declaration that identifies the dtbook DTD.

Those additional markup declarations in the internal subset take preference over any in the dtbook DTD itself. The effective DTD is thereby augmented by the parameter entity values and any other declarations of the book's internal subset. When a given parameter entity declaration appears more than once in the external modules and the dtbook DTD, the first occurrence of that declaration is the one that takes effect, with modules in the internal subset being processed in order, before the DTD itself.

For example:

```
<!DOCTYPE dtbook SYSTEM
"dtbook.dtd"
[
    <!ENTITY % dramaModule SYSTEM "drama.dtd">
    %dramaModule;
    <!ENTITY % externalblock "| drama">
    <!ENTITY % externalinline "| stagedir">
]>
```

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

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

```
<!DOCTYPE dtbook SYSTEM
"dtbook.dtd"
[
    <!ENTITY % poemModule SYSTEM "poem.dtd">
    %poemModule;
    <!ENTITY % dramaModule SYSTEM "drama.dtd">
    %dramaModule;
    <!ENTITY % externalblock "| poem | stanza | verse | drama">
    <!ENTITY % externalinline "| stagedir">
]>
```

Such external modules need to include the definitions of any parameter entities that are used in the modules since their definitions are needed before they can be expanded in their references. They cannot depend on parameter entities in the SystemLiteral or PubidLiteral.

Note that arbitrary external modules from other sources may not have all the needed attributes. XML allows augmentation of ATTLISTs in the internal subset. Additional attribute names can be added to an associated element type. Any redefinitions of a particular named attribute resulting from presence in the internal subset have precedence.

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

Following the document type declaration, the fully marked-up document appears, including tags from the external modules found in the internal subset. Declarations in the internal subset or in external entity references (such as %dramaModule;) referenced therein take precedence over like-named ones from the external entity containing the base DTD (that is, dtbook110.dtd). Thus the declarations from the module containing the drama and poem tags are included along with the tags in the base DTD (that is dtbook110.dtd) that are not duplicated or redefined in the drama module. So if a <p> tag is defined in the drama module, its definition overrides that of the <p> tag in dtbook. There is an exception: an ATTLIST for elementname that adds attributes from the internal subset augments the ATTLIST attributes with different attribute names in the ATTLIST of the same elementname in the dtbook110.dtd.

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

The form above for augmenting the dtbook dtd through the document's internal subset does not require the XML namespace mechanism, with its namespace-specific prefix on element and attribute names to disambiguate any potential name collisions. However, use of XML namespaces [XML-NAMES] is not precluded.

4. References

These references are informative. The bracketed names here are targets for

 ©NISO

indirect reference from the corresponding bracketed names in other parts of this document or in descriptions within this section.

[CSS1] Cascading Style Sheets, Level 1. Rec-CSS1-1999011 Revised 11 Jan 1999

http://www.w3.org/TR/REC-CSS1

[CSS2] Cascading Style Sheets, Level 2 CSS2 Specification REC-CSS2-19980512

http://www.w3.org/TR/REC-CSS2

[DAISY202] The DAISY 2.02 Specification for the DAISY Digital Talking Book (DTB) format, which enables navigation within a sequential and hierarchical structure consisting of (marked-up) text synchronized with audio.

http://www.daisy.org/dtbook/spec/2/final/d202/daisy_202.html

[DAISY-2-GUIDELINES] The DAISY 2.02 Specification for the Creation and Production of Accessible Books and Materials, Version 0.99 1999-09-23 represents minimum standard to be met by Libraries for the Blind and producers of alternative format materials:

http://www.daisy.org/dtbook/guidelines/draft/principles.htm

[DTBOOKV110DTD] The dtbook DTD v1.1.0 (this DTD) is available at:

http://www.loc.gov/nls/z3986/v100/dtbook110.dtd

Note that some browsers do not permit downloading a file with suffix dtd.

[DTBOOKV110DOC] Digital Talking Book Expanded Document Type Definition Documentation for Version V110 of this DTD is available as an HTML 4.0 document:

http://www.loc.gov/nls/z3986/v100/dtbook110doc.htm

Should revisions occur, a new directory with node named "vxxx" (rather than v110) that indicates the revision level will contain the revisions. Any prior specific version of the dtbook dtd and its documentation will persist.

[DTBOOK3] The last public beta version was dtbook3-07.dtd (2001-01-31).

http://www.loc.gov/nls/z3986/background/dtbk3_old_dtds/dtbk3-07.dtd

and its expanded documentation:

http://www.loc.gov/nls/z3986/background/dtbk3_old_dtds/dtbk3-07doc.htm

Those and prior versions are available at:

http://www.loc.gov/nls/z3986/background/dtbk3_old_dtds/index.html

The history of changes prior to this version, including those in internal drafts through dtbk3-12.dtd and before is in:

http://www.loc.gov/nls/z3986/background/dtbk3-dtd-changes.txt

In that directory also are the old dtdbk3 dtds, some of which have been used for test markup, and their documentation. See its index.html for the list. (Caution: some browsers may not permit downloading DTDs.)

http://www.loc.gov/nls/z3986/background/index.html

[HTML401STRICT] "HTML 4.0 Strict DTD," 1999-12-24, Dave Raggett, Arnaud Le hors, and Ian Jacobs. Dtbook110 was originally based on the HTML 4.0 Strict DTD with design adaptation for dtbook110. A principal adaptation is to use lower-case names for element types and attribute names. For expanded discussion, see [HTML401].

http://www.w3.org/TR/1999/REC-html401-19991224/strict.dtd

[HTML401] "HTML 4.01 Specification" W3C Recommendation 24 December 1999 Documentation of the element types that come from the HTML 4.0 Strict DTD [HTML401STRICT] is available at:

http://www.w3.org/TR/1999/REC-html401-19991224/

Dtbook110 is partially harmonized with the [XHTML11STRICT] DTD. The XHTML camelCase parameter entity names are retained, and comments and references following those parameter entities explain them. The lower-case element and attribute names are used. The simplified table content model of just table rows is included.

[ISO10646] "Information Technology - Universal Multiple-Octet Coded Character Set (UCS) - Part 1: Architecture and Basic Multilingual Plane," ISO/IEC 10646-1:1993. The current specification also takes into consideration the first five amendments to ISO/IEC 10646-1:1993.

[ISO8859] "Information Processing - 8-bit single-byte coded graphic character sets - Part 1: Latin alphabet No. 1," ISO 8859-1:1987. Other suffixes "-2 through -9" correspond to other character sets in the family.

[JIS] "JIS Character Sets" describes the history of JIS, and the several character sets for KANJI, KANA and other characters.

http://www.io.com/~kazushi/encoding/jis.html

[ANSINISOZ39-86-2002] Specifications for the Digital Talking Book.

http://www.niso.org

[NLS-Z3986] Development of ANSI/NISO Z39.86-2002 Contains links to the DTDs developed for Z39.86-2002, Specifications for the Digital Talking Book

http://www.loc.gov/nls/z3986/v100/index.html

[OASIS-TR9401] Entity Management, OASIS Technical Resolution 9401:1997 (Amendment 2 to TR 9401). Paul Grosso, 1997 September 10.

 ©NISO

http://www.oasis-open.org/specs/tr9401.html

[RFC1556] "Handling of Bi-directional Texts in MIME," H. Nussbacher, December 1993.

http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1556.html

[RFC1766] "Tags for the Identification of Languages," H. Alvestrand, March 1995.

http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1766.html

[RFC1942] "HTML Tables," D. Raggett, May 1996

http://www.ietf.org/rfc/rfc1942.txt

Contains detailed descriptions of table elements and their inheritance of attribute values. Adjustment for XML application is required: end-tags are necessary, not optional, attribute values must be quoted.

[RFC2045] "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies," N. Freed and N. Borenstein, November 1996. Note that this RFC obsoletes RFC1521, RFC1522, and RFC1590. The %ContentType; and %ContentTypes; media types and the %Charset; and %Charsets; character encoding values are from [RFC2045].

http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2045.html

[RFC2046] "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types," N. Freed, November 1996. Source for %ContentType; and %ContentTypes; permitted values:

http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2046.html

[RFC2396] "Uniform Resource Identifiers (URI): Generic Syntax," T. Berners-Lee, R. Fielding, L. Masinter, August 1998. Note that this RFC revises and replaces the generic definitions in RFC 1738 and RFC 1808.

http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2396.html

[SMIL2.0] The Synchronized Multimedia Integration Language SMIL 2.0 W3C Recommendation 07 August 2001 is available at:

http://www.w3.org/TR/2001/REC-smil20-20010807/smil20.html

[XHTML11] "XHTML (tm) 1.0: The Extensible HyperText Markup Language," W3C Recommendation 26 January 2000, A reformulation of HTML4 in XML 1.0 includes case-sensitive names, lower-case for elements and their attributes (but not parameter entity names) and in some cases equivalent content models that do not require SGML inclusions and exclusion exceptions (as occurred in the HTML4.0 strict DTD [HTML401STRICT]) is available at:

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

```
     [XML-NAMES] "Namespaces in XML" World Wide Web Consortium
     14-January-1999, REC-xml-names-19990114.

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

     [XHTML11STRICT] Expanded documentation of the element types that come
     from the XHTML11 strict.dtd and its other DTDs is available within
     the zip file:

         http://www.w3.org/TR/xhtml1/DTD/xhtml1/xhtml1.zip

     Note: some browsers cannot download a dtd directly.

     [XML12] This dtbook110.dtd is an application of the Extensible Markup
     Language XML 1.0 (Second Edition) W3C Recommendation 6 October 2000.
     It is available at:

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

     -->

<!-- change record:
         1998-10-08 original by Harvey Bingham
         1999-01-23 revision 3-01
         1999-06-25 revision 3-02
         1999-07-20 revision 3-03
         1999-09-16 revision 3-04
         1999-09-24 revision 3-05
         1999-11-05 revision 3-06
         2001-01-31 revision 3-07
         2001-03-08 revision 3-08
         2001-03-30 revision 3-09 basis for dtbook110.dtd
         2001-09-07 revision 3-10 version 1.0.0 first draft
         2001-09-21 revision 3-11 version 1.0.0 second draft
         2001-09-26 revision 3-12 version 1.0.0 third draft
         2001-09-30 dtbook100 version 1.0.0 initial public release
         2002-01-22 dtbook110 version 1.1.0

     The record of evolution of this dtd may be found in the archives.
     See [DTBOOK3].

         2001-12-20 revision 1.1.0 has syntactic changes, so
         increased version secondary revision as content models
         are extended compatibly and new capability is added.
         Added annoref to parameter entities where noteref occurs:
             %dtbookinline; %inlines; %inlinenopagenum;
             %inlinenoprodnote; %inlinenoanoprodnote;
         Made some additions to references, including JIS and HTML Tables,
             XML Namespaces, OASIS Catalog.
         Made editorial clarifications to many Use: and Attuse: comments.
         Added subsections.
         Removed xhtml parameter entities from xhtml not used here:
              %ContentTypes; %Charsets; %Datetime;
         For processing purposes, split dtd into part1.dtd and part2.dtd
         split processing into four pieces.
```

 ©NISO

```
DTD Changes for dtbook110.dtd   Harvey Bingham 2001-12-20

    Character entities amplified comments.
    Removed %Datetime; %ContentTypes; and %Charsets;
         not used herein.
    % Charsets removed as unused.
    % ContentTypes removed as unused.
    % Datetime removed as unused.
    % dtbookblock added imggroup.
    % dtbookblocknoimggroup added.
    % inlineinblock removed img.
    % block added img and imggroup.
    % blocknoimggroup added using %dtbookblocknoimggroup.
    % headmisc added comment
    % special added imggroup
    % specialnoa added imggroup
    % dtbookinline added annoref
    % inlines added annoref
    % inlinenopagenum added annoref
    % inlinenoprodnote added annoref
    % flow changed %block; to %blocknoimggroup;
    % flownopagenum changed %block; to %blocknoimggroup;
    img Use: changed idref to imgref
    % Trules added default meaning from xhtml strict dtd.
    % cellhalign clarified interitance.
    % cellvalign clarified interitance.
    table updated model to xhtml strict dtd, adding
        simple table of just rows (tr). Added attribute inheritance
        information from RFC1942.
    caption added imgref extension for dtbook comment.
    thead updated Use to xhtml strict dtd.
    tfoot updated Use to xhtml strict dtd.
    tbody updated Use to xhtml strict dtd.
    colgroup updated Attuse to xhtml strict dtd.

    Michael Moodie: 2002-02-14

    % block replaced %dtbookblock with %dtbookblocknoimggroup
    to eliminate duplication of imggroup in content model.
    level updated content model to include doctitle and docauthor
    to allow them to be contained in a level marking the cover
    or title page of a document.
    level1 updated content model to include doctitle and docauthor
    to allow them to be contained in a level1 marking the cover
    or title page of a document.
    div updated content model to include doctitle and docauthor
    to allow them to be contained in a div marking the cover
    or title page of a document.

    Michael Moodie: 2002-02-19

    level, level1-level6, div: Changed occurrence indicator on
    content model from * to +.

-->
```

```
<!-- Comment Classification Conventions

            Some comments start with a pattern followed by a colon:

                Use: element type and its use.

                Attuse: attribute use for associated element type.

                HB: date object comment on change by Harvey Bingham.

                MM: date object comment on change by Michael Moodie.

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

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

<!--HB: 2001-12-20 Character entities amplified comments. -->

<!-- Character entities for interoperability.
         The five following characters may have special markup meaning,
         so are expressed as character entities in text. They can be
         recognized since they are preceded by "&" and followed by ";".
         The notation below, #xHHHH (or #xHH) where H is a hexadecimal-number
         (formed from 0-9, A-F), indicates the character code position
         in Unicode/ISO-10646 [ISO10646]. Note that the "<" and "&"
         characters in the declarations of "lt" and "amp" are doubly escaped
         to meet the requirement that entity replacement be well-formed.
         As these entities occur in the first plane of Unicode, with encodings
         the same as ASCII, the "00" prefix can be implied, so omitted.
     -->

<!ENTITY lt   "&#x0026;#x003C;" >
     <!-- "&#38;#60;"  < Less than, normally starts a tag. -->

<!ENTITY gt   "&#x003E;" >
     <!-- "&#62;"      > Greater than, normally ends a tag. -->

<!ENTITY amp  "&#x0026;#x0026;" >
     <!-- "&#38;#38;"  & Ampersand, normally begins acharacter entity
          reference. -->

<!ENTITY apos "&#x0027;" >
     <!-- "&#39;"      ' Neutral Quote, Apostrophe, if needed within an
          attribute string so quoted. -->

<!ENTITY quot "&#x0022;" >
     <!-- "&#34;"      " Quotation mark, if needed within an
attribute string
          so quoted. -->

<!-- Three larger character sets included in HTML 4.0 are omitted here:
         HTMLlat1.ent, HTMLsymbol.ent, and HTMLspecial.ent.
         Unicode [ISO10646] is available to XML applications, so these
         characters are available. The initial processing instruction that
         identifies dtbook as an XML application should use a more inclusive
         encoding, as described at the start of section 2.
     -->
```

 ©NISO

```
<!--=================== Imported Parameter Entity Names ===================-->

<!-- Many parameter entities come from the [XHTML11STRICT] strict DTD.-->

<!--HB: 2001-12-20 Removed %Datetime; %ContentTypes; and %Charsets;
          not used herein.-->

<!ENTITY % Character "CDATA" >
     <!-- a single character from [ISO10646]. -->

<!ENTITY % Charset "CDATA" >
     <!-- a character encoding, as per [RFC2045]. -->

<!--HB: 2001-12-20 % Charsets removed as unused. -->

<!ENTITY % ContentType "CDATA" >
     <!-- media type, as per [RFC2046]. -->

<!--HB: 2001-12-20 % ContentTypes removed as unused. -->

<!--HB: 2001-12-20 % Datetime removed as unused. -->

<!ENTITY % LanguageCode "NMTOKEN" >
     <!-- a language code, per [RFC1766]. -->

<!ENTITY % Number "CDATA" >
     <!-- one or more digits. -->

<!ENTITY % LinkTypes "CDATA" >
     <!-- space-separated list of link types. -->

<!ENTITY % MediaDesc "CDATA" >
     <!-- single or comma-separated list of media descriptors;
         possible values include BRAILLE, PRINT, PROJECTION, SPEECH, ALL,
         or the default SCREEN. -->

<!ENTITY % StyleSheet "CDATA" >
     <!-- style sheet data. -->

<!ENTITY % Text "CDATA" >
     <!-- used for titles etc. -->

<!ENTITY % URI "CDATA" >
     <!-- a Uniform Resource Identifier, see [RFC2396]. -->

<!--================== dtbook External Module Inclusion ===================-->

<!ENTITY % externalblock "" >
     <!-- placeholder for block element expansion from external modules;
         if changed, string in external subset begins " |
blockelementname". -->

<!ENTITY % externalinline "" >
     <!-- placeholder for inline element expansion from external modules;
         if changed, string in external subset begins " |
inlineelementname". -->
```

```
<!--====================== dtbook110 Content Models =======================-->

<!ENTITY % list "list" >
     <!-- list container for ordered or unordered lists (distinguished
         by attribute value, not element types ol or ul). -->

<!--HB: 2001-12-20 % dtbookblock added imggroup. -->

<!ENTITY % dtbookblock
     "author | notice | prodnote | sidebar | note | imggroup |
         annotation %externalblock;" >
     <!-- block elements unique to dtbook. -->

<!--HB: 2001-12-20 % dtbookblocknoimggroup added. -->

<!ENTITY % dtbookblocknoimggroup
     "author | notice | prodnote | sidebar | note |
         annotation %externalblock;" >
     <!-- block elements unique to dtbook without imggroup. -->

<!--HB: 2001-12-20 % inlineinblock removed img. -->

<!ENTITY % inlineinblock
     "a | cite | caption | samp | kbd | pagenum" >
     <!-- inlines that may alternatively be in block elements. -->

<!--HB: 2001-12-20 % block added img and imggroup. -->
<!--MM: 2002-02-14 replaced %dtbookblock with %dtbookblocknoimggroup in
      % block to eliminate duplication of imggroup in content model. -->

<!ENTITY % block
     "p | %list; | dl | div | blockquote | hr | img | imggroup |
          table | address | line | %dtbookblocknoimggroup;" >
     <!-- block elements from [HTML401STRICT] dtd augmented by dtbook-unique
         elements (%list; differs.). -->

<!--HB: 2001-12-20 % blocknoimggroup added using
%dtbookblocknoimggroup. -->

<!ENTITY % blocknoimggroup
     "p | %list; | dl | div | blockquote | hr |
          table | address | line | %dtbookblocknoimggroup;" >
     <!-- block elements from [HTML401STRICT] dtd augmented by dtbook-unique
         elements. -->

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

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

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

<!ENTITY % coreattrs
     "id          ID             #IMPLIED
      class       CDATA          #IMPLIED
      style       %StyleSheet;   #IMPLIED
      title       %Text;         #IMPLIED" >
```

 ©NISO

```
     <!-- coreattrs are attributes permissible for most elements
         id       document-wide unique id
         class    space separated list of classes used for rendering
         style    associated style info
         title    advisory title/amplification
     -->

<!ENTITY % i18n
     "lang        %LanguageCode; #IMPLIED
      xml:lang    %LanguageCode; #IMPLIED
      dir         (ltr|rtl)      #IMPLIED" >
     <!-- i18n internationalization attributes

         lang        language code (backwards compatible)
         xml:lang    language code (as per XML 1.0 spec)

         dir         direction for weak/neutral text
                     ltr=left to right
                     rtl=right to left

     xhtml recommendation: use both lang and xml:lang, with same value,
     such as "en-US", on the major containing block, to provide source for
     the #IMPLIED values of its descendent elements. See [RFC1556].
     should the values differ, the xml:lang takes precedence.
     See [RFC1556] for handling bi-directional text in MIME.
     -->

<!ENTITY % showin
     "showin (xxx|xxp|xlx|xlp|bxx|bxp|blx|blp) #IMPLIED" >
     <!--showin attribute applies for text elements to permit identification
         of the kinds of display appropriate for the element, so presentation
         choice by the reader among alternative readings can be provided, when
         appropriate. Values of showin are coded with three letters in order:
         "b"=Braille, "l"=Largeprint, and "p"=Print; or "x"=inappropriate:

             Value  Braille  Largeprint Print   Interpretation

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

         There is no default value; this attribute value is implied
         from the most immediate ancestor that specifies a value.
         The usual default for showin is 'blp'. If only one showin
         value is needed it should be included with <book>.

         Different content for the same element (usually <prodnote>) meeting
         different needs is possible, with showin serving as a switch to
         differentiate among them. Both largeprint and print are appropriate
         for screen rendering as well as printing. Different corresponding
         styles may be appropriate.
```

```
        It is possible to include equivalent content from any major structure
        below <book> to provide the different content suitable for different
        media. These would be independent, sharing no direct content, possibly
        having common references to images, with different accompanying text
        descriptions.
    -->

<!ENTITY % attrs
    "%coreattrs;
     %i18n;
     smilref      CDATA       #IMPLIED
     %showin;" >
    <!-- %attrs; is part of most attribute lists. It includes

    %coreattrs; from which come the four #IMPLIED attributes:
        id, class, style, and title.

    %i18n; from which come the implied attributes: lang, xml:lang, and dir

    smilref is a pointer to a [SMIL2.0] file, normally to the time container
        (SMIL <par> or <seq>) containing the media object that references this
        element. However, in a text-only DTB consisting of a sequence of
        text media objects, <smilref> points to the media object that
        references this element. <smilref> allows resumption of SMIL
        presentation at the proper location after navigation via dtbook file.
        All <smilref> values are expected to be added to an augmented
        version of the <dtbook> during production.

    %showin; (See entity declaration.)
    -->

<!ENTITY % attrsrqd
    "id          ID             #REQUIRED
     class       CDATA          #IMPLIED
     style       %StyleSheet;   #IMPLIED
     title       %Text;         #IMPLIED
     smilref     CDATA          #IMPLIED
     %i18n;
     %showin; " >
    <!-- %attrsrqd; includes required id and implied class, style, and
        title.

    %i18n; from which come the implied attributes: lang, xml:lang, and dir

    smilref is a pointer to a [SMIL2.0] file, normally to the time container
        (SMIL <par> or <seq>) containing the media object that references this
        element. However, in a text-only DTB consisting of a sequence of
        text media objects, <smilref> points to the media object that
        references this element. <smilref> allows resumption of SMIL
        presentation at the proper location after navigation via dtbook file.
        All <smilref> values are expected to be added to an augmented
        version of the <dtbook> during production.

    %showin; (See entity declaration.)
    -->
```

 ©NISO

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

<!ENTITY % dtbookcontent
     "head, book" >
     <!-- dtbookContent designates that each dtbook has a <head> of
         metainformation preceding the <book> content.
     -->
<!--Use: dtbook is the root element in the Digital Talking Book DTD.
     <dtbook> contains metadata in <head> and the contents itself
     in <book>.
     -->

<!ELEMENT dtbook (%dtbookcontent;) >

<!--Attuse: dtbook

     "version" is required, and contains the specific
     version of the dtd, so that the dtd version for any dtbook can
     be recognized.

     "%i18n;" internationalization attributes characterize the <book>.
     Those values may be adjusted for language changes within it.
     -->

<!ATTLIST dtbook
   version CDATA #FIXED '1.1.0'

   %i18n;
   >

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

<!--HB: 2001-12-20 % headmisc added comment -->

<!ENTITY % headmisc
     "style | meta | link" >
     <!-- optional head elements -->

<!--Use: head contains metainformation about the book but no
     actual content of the book itself, which is placed in <book>.
     This information is consonant with the <head> information
     in xhtml, see [XHTML11STRICT]. Other miscellaneous elements can
     occur before and after the required <title>. By convention
     <title> should occur first.
     -->

<!ELEMENT head ((%headmisc;)*, title, (%headmisc;)*) >

<!--Attuse: head

     "profile" gives one or more
     whitespace-separated profile URI targets that may provide
     additional information about the current document.
     -->
```

```
<!ATTLIST head
   %i18n;
   profile %URI; #IMPLIED
   >

<!--Use: title contains the title of the book but is used only as
     metainformation in <head>. Use <doctitle> within
<book> for
     the actual book title, which will usually be the same.
     -->

<!ELEMENT title (#PCDATA) >

<!ATTLIST title
   %i18n;
   >

<!--Use: link is an empty element appearing in the <head> section
     of a document that establishes a connection between the current
     document and another document. The <link> element conveys
     relationship information (for example, "next" and "previous") that
     may be rendered by user agents in a variety of ways.
     -->

<!ELEMENT link EMPTY >

<!--Attuse: link

     Each attribute use indicated by a parameter entity is
     defined in the comment following its definition.
     -->

<!ATTLIST link
   %attrs;
   charset %Charset; #IMPLIED
   href %URI; #IMPLIED
   hreflang %LanguageCode; #IMPLIED
   type %ContentType; #IMPLIED
   rel %LinkTypes; #IMPLIED
   rev %LinkTypes; #IMPLIED
   media %MediaDesc; #IMPLIED
   >

<!--Use: meta indicates metadata about the book. It is an empty
     element that may appear repeatedly only in <head>.
     -->

<!ELEMENT meta EMPTY >

<!--Attuse: meta

     "http-equiv" connects the content attribute
     value to an http header field.

     "name" value identifies the specific kind of
     content value.
```

 ©NISO

```
     "content" indicates the value for that "name",
     possibly constrained by the semantics for the individual names.

     "scheme" indicates a predetermined format for interpreting
     the content value, such as the Dublin Core.
     -->

<!ATTLIST meta
   %i18n;
   http-equiv NMTOKEN #IMPLIED
   name NMTOKEN #IMPLIED
   content CDATA #REQUIRED
   scheme CDATA #IMPLIED
   >

<!--Use: style provides the means to include styling information
     that applies to the book. It may appear only in <head>.
     It may include CDATA sections.
     -->

<!ELEMENT style (#PCDATA) >

<!--Attuse: style

     "type" indicates the MIME-Type [RFC2045].
     Type value should be 'text/css', rather than 'text/javascript'.

     "media" value indicates the media for stylesheet
     definition(s); if multiple, separated by commas.

     "title" can provide menu choice among alternative
     stylesheets.

     "xml:space" value='preserve' indicates that whitespace in
     the <style> content is preserved without need to include its
     value in each <style>. (xml:space='default' accepts system style
     adjustment, such as adding its own indenting.)
     -->

<!ATTLIST style
   %i18n;
   type %ContentType; #REQUIRED
   media %MediaDesc; #IMPLIED
   title %Text; #IMPLIED
   xml:space (default | preserve) 'preserve'
   >

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

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

<!ELEMENT book (frontmatter?, bodymatter?, rearmatter?) >
```

```
<!ATTLIST book
   %attrs;
   >

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

<!--Use: frontmatter usually contains <doctitle> and <docauthor>, as
     well as preliminary material that is often enclosed in appropriate
     <level> or <level1>. Content may include copyrightnotice, foreword,
     acknowledgments, table of contents, etc. <frontmatter> serves as a
     guide to the content and nature of a <book>.
     -->

<!ELEMENT frontmatter (doctitle | docauthor | level | level1 | %block;)+ >

<!ATTLIST frontmatter
   %attrs;
   >

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

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

<!ATTLIST bodymatter
   %attrs;
   >

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

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

<!ATTLIST rearmatter
   %attrs;
   >

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

<!--MM: 2002-02-14 Added doctitle and docauthor to content model of level
     to allow them to be contained in a level marking the cover or title page of
     a document. -->
<!--MM: 2002-02-19 Changed occurrence indicator on content model of level
     from * to +.  -->

<!--Use: level is an alternative tag for marking the major
     structures in a book. It may be used recursively, i.e., repeated
     indefinitely with each successive occurrence nesting within the
     previous. It may also be included in a subsequent higher level.
     Subordinate levels have greater depth. Contrast with the explicit
     <level1>...<level6> elements, which may not be intermixed with
     <level>.
     -->
```

 ©NISO

```
<!ELEMENT level (levelhd | %block; | %inlineinblock; | level | doctitle | docauthor)+ >

<!--Attuse: level

     "class" identifies the actual name (e.g., part,
     chapter, section, subsection) of the structure it marks.

     "depth" indicates the nesting depth, starting at 1.
     -->

<!ATTLIST level
   %attrs;
   depth CDATA #IMPLIED
   >

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

<!--MM: 2002-02-14 Added doctitle and docauthor to content model of level1
     to allow them to be contained in a level1 marking the cover or title
     page of a document. -->
<!--MM: 2002-02-19 Changed occurrence indicator on content model of level1
     from * to +.  -->

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

<!ELEMENT level1 (h1 | level2 | %block; | %inlineinblock; | doctitle | docauthor)+ >
<!ATTLIST level1
   %attrs;
   >

<!--MM: 2002-02-19 Changed occurrence indicator on content model of level2
     from * to +.  -->

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

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

<!ATTLIST level2
   %attrs;
   >

<!--MM: 2002-02-19 Changed occurrence indicator on content model of level3
     from * to +.  -->

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

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

<!ATTLIST level3
   %attrs;
   >

<!--MM: 2002-02-19 Changed occurrence indicator on content model of level4
     from * to +.  -->

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

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

<!ATTLIST level4
   %attrs;
   >

<!--MM: 2002-02-19 Changed occurrence indicator on content model of level5
     from * to +.  -->

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

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

<!ATTLIST level5
   %attrs;
   >

<!--MM: 2002-02-19 Changed occurrence indicator on content model of level6
     from * to +.  -->

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

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

<!ATTLIST level6
   %attrs;
   >

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

<!ENTITY % phrase
     "em | strong | dfn | code | samp | kbd | cite | abbr | acronym" >
     <!-- inline text elements -->
```

 ©NISO

```
<!--HB: 2001-12-20 % special added imggroup -->

<!ENTITY % special
     "a | img | imggroup | br | q | sub | sup | span | bdo | linenum" >
     <!-- special inline text elements -->

<!--HB: 2001-12-20 % specialnoa added imggroup -->

<!ENTITY % specialnoa
     "img | imggroup | br | q | sub | sup | span | bdo | linenum" >
     <!-- specialnoa inline text elements for anchor <a> -->

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

<!--HB: 2001-12-20 % dtbookinline added annoref. -->

<!ENTITY % dtbookinline
      "sent | w | pagenum | prodnote | annoref | noteref %externalinline;" >
     <!-- dtbook added inline text elements -->

<!ENTITY % inline
     "#PCDATA | %phrase; | %special; | %dtbookinline;" >
     <!-- inline text elements -->

<!ENTITY % inlinenoa
     "#PCDATA | %phrase; | %specialnoa; %externalinline;" >
     <!-- inlinenoa excludes nested <a> -->

<!--HB: 2001-12-20 % inlines added annoref. -->

<!ENTITY % inlines
      "#PCDATA | %phrase; | %special; | pagenum | w | prodnote | annoref |
          noteref %externalinline;" >
     <!-- inlines excludes direct nesting of sentences <sent> -->

<!ENTITY % inlinew
      "#PCDATA | %phrase; | %special; %externalinline;" >
     <!-- inlinew for word <w> excludes any of the %dtbookinline; -->

<!--HB: 2001-12-20 % inlinenopagenum added annoref. -->

<!ENTITY % inlinenopagenum
      "#PCDATA | %phrase; | %special; |
           sent | w | annoref | noteref %externalinline;" >
     <!-- inlinenopagenum excludes direct <pagenum> in
<table> <th> and <td> -->

<!--HB: 2001-12-20 % inlinenoprodnote added annoref. -->

<!ENTITY % inlinenoprodnote
      "#PCDATA | %phrase; | %special; |
           sent | w | pagenum | annoref | noteref %externalinline;" >
     <!-- inlinenoprodnote excludes direct <prodnote>, as
they shouldn't nest -->
```

```
<!--=================== Flow (Block or Inline) Entities===================-->

<!--HB: 2001-12-20 % flow changed %block; to %blocknoimggroup; -->

<!ENTITY % flow
     "%inlinenoprodnote; | %blocknoimggroup;" >
     <!-- flow elements add inlinenoprodnote to block -->

<!--HB: 2001-12-20 % flownopagenum changed %block; to %blocknoimggroup; -->

<!ENTITY % flownopagenum
     "%inlinenopagenum; | %blocknoimggroup;" >
     <!-- flownopagenum ideally excludes pagenum though can get in
         indirectly through elements of %blocknoimggroup; -->

<!--============ Br, Linenum, Address, and Div Content Models =============-->
<!--Use: br marks a forced line break.
      -->

<!ELEMENT br EMPTY >

<!--Attuse: br
     The %coreattrs; only appear, as there is no content
     to which the more general %attrs; apply.
     -->

<!ATTLIST br
   %coreattrs;
   >

<!--Use: linenum contains a line number, for example in legal text.
     -->

<!ELEMENT linenum (#PCDATA) >

<!ATTLIST linenum
   %attrs;
   >
<!--Use: address contains a location at which a person or agency
     may be contacted. By use of <line> to contain content of the
     individual lines, the class attribute can be used to identify
     the content of that <line>. For example, class values might include:
     name, address, region (state. province, etc.), country, location
     code (such as zipcode, provincial code), phone, fax, email, etc.
     -->

<!ELEMENT address (%inline; | line)* >
<!ATTLIST address
   %attrs;
   >

<!--MM: 2002-02-14 Added doctitle and docauthor to content model of div to
     allow them to be contained in a div marking the cover or title page of
     a document. -->
```

 ©NISO

```
<!--MM: 2002-02-19 Changed occurrence indicator on content model of div
     from * to +.  -->

<!--Use: div is a generic container for subdivisions of a book. The
     <level1> ... <level6> hierarchy, or the <level> tag used recursively,
     should mark the major hierarchical structures of a book, while <div>
     is used in less formal circumstances or when for production purposes
     it is desired that a structure should be treated differently.
     Compare with <span>, which is used in inline settings.
     -->

<!ELEMENT div (%block; | %inlineinblock; | doctitle | docauthor)+ >

<!--Attuse: div

     "level" may extend or augment explicit levels,
     to indicate nesting level, with values the positive integers, with
     '1' corresponding to <level1>, and value generally one larger than
     the enclosing level.

     "class" value can identify the
     actual name (e.g., part, chapter, letter) of the structure it marks.
     -->

<!ATTLIST div
   %attrs;
   level CDATA #IMPLIED
   >

<!--======= dtbook Block Elements Author, Notice, Prodnote, Sidebar =======-->

<!--Use: author identifies the writer of a work other than this one.
     Contrast with <docauthor>,  which identifies the author of this work.
     <author> typically occurs within <blockquote> or <cite>.
     -->

<!ELEMENT author (%inline;)* >

<!ATTLIST author
   %attrs;
   >

<!--Use: notice contains a warning, caution, or other type of admonition
     normally found in the margin of a book. In contrast with <sidebar>
     a <notice> must be presented at a specific location within the
     text. Its presentation is not optional.
     -->

<!ELEMENT notice (%inline;)* >

<!ATTLIST notice
   %attrs;
   >
```

```
<!--Use: prodnote contains language added to the alternative-format
     version by the producer; commonly used to:
     1) provide descriptions of one or more visual elements such
         as charts, graphs, etc.
     2) supply operating instructions
     3) describe differences between the print book and the audio
     version.
     -->

<!ELEMENT prodnote (%flow;)* >

<!--Attuse: prodnote

     "imgref" identifies the space-separated id value(s)
     on pertinent images <img>.

     "render" indicates that the content is
     'required or 'optional' for the user. If optional, some user
     preference may allow skipping over the content. But <prodnote
     render='required'> is essential content for the user. An
     audible cue could announce the presence of the <prodnote>.
     -->

<!ATTLIST prodnote
   %attrs;
   imgref IDREFS #IMPLIED
   render (required | optional) #IMPLIED
   >

<!--Use: sidebar contains information supplementary to the main
     text and/or narrative flow and is often boxed and printed apart
     from the main text block on a page. It may have a heading <hd>.
     -->

<!ELEMENT sidebar (%flow; | hd)* >

<!ATTLIST sidebar
   %attrs;
   >

<!--Use: note marks a footnote, endnote, etc. Any local reference to
<note id="yyy"> is by <noteref idref="#yyy">.
     -->

<!ELEMENT note (%block; | %inlineinblock;)+ >

<!ATTLIST note
   %attrsrqd;
   >

<!--Use: annotation is a comment on or explanation of a portion of
     a printed book. It differs from <note> in that an <annotation>
     is usually set in the margin or on a facing page, often with
     no explicit reference to it inserted in the text. Any local
     reference to <annotation id="xxx"> is by <annoref idref="#xxx">.
     -->
```

 ©NISO

```
<!ELEMENT annotation (%block; | %inlineinblock;)+ >

<!ATTLIST annotation
   %attrsrqd;
   >

<!--Use: line marks a single logical line of text. Often used in
     conjunction with <linenum> in documents with numbered lines.
     -->

<!ELEMENT line (%inline;)* >

<!ATTLIST line
   %attrs;
   >

<!--========================= The Anchor Element ==========================-->

<!--Use: a contains an anchor, which is used to reference another
     location, within the same or another <dtbook>.
     -->

<!ELEMENT a (%inlinenoa;)* >

<!--Attuse: a

     "href" value may have three forms:

     1) '#idref', in the <dtbook>, to the element type having the
         referenced id value;

     2) 'uri', a uniform resource identifier to a resource, typically a
         document, see [RFC2396], possibly restricted to work with only
         the <dtbook> document content, as referenced content is
         expected to be available on the same media, and off-media
         references may not be available;

     3) 'uri#xxx', in the resource uri, the element with id='xxx'.

     Uses of the remaining attributes other than %attrs; are:

         "type" is advisory content MIME type of the target, see [RFC1556];

         "hreflang" is the language code of the href target, see [RFC1766];

         "rel" is a list of forward link type(s), the relationship(s)
             expressed by the href value to the target, space-separated
             if multiple;

         "rev" is a list of reverse link types, the relationship(s)
             to this location from the href target, space-separated
             if multiple;

         "accesskey"=accessibility key character shortcut;

         "tabindex"=tabbing order.
     -->
```

```
<!ATTLIST a
   %attrs;
   type %ContentType; #IMPLIED
   href %URI; #IMPLIED
   hreflang %LanguageCode; #IMPLIED
   rel %LinkTypes; #IMPLIED
   rev %LinkTypes; #IMPLIED
   accesskey %Character; #IMPLIED
   tabindex %Number; #IMPLIED
   >

<!--========================== Inline Elements ==========================-->

<!--Use: em indicates emphasis. Usually <em> is rendered in italics.
Compare with <strong>.
     -->

<!ELEMENT em (%inline;)* >

<!ATTLIST em
   %attrs;
   >

<!--Use: strong marks stronger emphasis than <em>. Visually <strong> is
      usually rendered bold. -->

<!ELEMENT strong (%inline;)* >

<!ATTLIST strong
   %attrs;
   >

<!--Use: dfn marks the first occurrence of a word or term that is
     defined or explained there or elsewhere in <book>. Often
     <dfn> is rendered in italics, sometimes in parentheses.
     -->

<!ELEMENT dfn (%inline;)* >

<!ATTLIST dfn
   %attrs;
   >

<!--Use: kbd designates information that the reader is to input
     directly into a computer using the keyboard.
     -->

<!ELEMENT kbd (%inline;)* >

<!ATTLIST kbd
   %attrs;
   >

<!--Use: code designates a fragment of computer code.
     -->
```

 ©NISO

```
<!ELEMENT code (%inline;)* >

<!--Attuse: code
     "xml:space" value='preserve' preserves
     whitespace therein (except that an XML parser strips leading
     and trailing whitespace before passing the internal content
     including its original whitespace to the application.) The value
     'default' leaves the whitespace handling to the application.
     -->

<!ATTLIST code
   %attrs;
   xml:space (default | preserve) 'preserve'
   >

<!--Use: samp contains a sample of work created by the author for
     use as an example or template. For example, a sample business
     letter, resume, computer program output, or form.
     -->

<!ELEMENT samp (%inline;)* >

<!--Attuse: samp
     "xml:space" value 'preserve' preserves
     whitespace therein (except that an XML parser strips leading
     and trailing whitespace before passing the internal content
     including its original whitespace to the application.) The value
     'default' leaves the whitespace handling to the application.
     -->

<!ATTLIST samp
   %attrs;
   xml:space (default | preserve) 'preserve'
   >

<!--Use: cite marks a reference (or citation) to another document.
     -->

<!ELEMENT cite (%inline;)* >

<!ATTLIST cite
   %attrs;
   >

<!--Use: abbr designates an abbreviation, a shortened form of a
     word. For examples: Mr., approx., lbs., rec'd.
     Contrast with <acronym>.
     -->

<!ELEMENT abbr (%inline;)* >

<!--Attuse: abbr
     "title" value may expand that abbreviation.
     -->
```

```
<!ATTLIST abbr
   %attrs;
   >

<!--Use: acronym marks a word formed from key letters (usually
     initials) of a group of words. For examples: UNESCO, NATO, XML, US.
     Contrast with <abbr>.
     -->

<!ELEMENT acronym (%inline;)* >

<!--Attuse: acronym
     "title" value may expand that acronym.
     "pronounce" value 'yes' indicates that the
     acronym is pronounceable as a word (for example, NATO);
     'no' that the acronym is best presented as a sequence
     of letters (for examples, "XML" or  "US").
     -->

<!ATTLIST acronym
   %attrs;
   pronounce (yes | no) #IMPLIED
   >

<!--Use: sub indicates a subscript character (printed below a
     character's normal baseline). Can be used recursively and/or
     intermixed with <sup>.
     -->

<!ELEMENT sub (%inline;)* >

<!ATTLIST sub
   %attrs;
   >

<!--Use: sup marks a superscript character (printed above a
     character's normal baseline). Can be used recursively and/or
     intermixed with <sub>.
     -->

<!ELEMENT sup (%inline;)* >

<!ATTLIST sup
   %attrs;
   >

<!--Use: span is a generic container for use in inline settings
     when no specific tag exists for a given situation. The class
     attribute may describe the nature of the text it marks (e.g.,
     a typographical error). May be used to mark a class of items
     to which styles are to be applied. Compare with <div>, which
     is used in a block settings.
     -->

<!ELEMENT span (%inline;)* >
```


©NISO

```
<!ATTLIST span
   %attrs;
   >

<!--Use: bdo is used in special cases where the automatic actions
     of the bi-directional algorithm would result in incorrect display.
     -->

<!ELEMENT bdo (%inline;)* >

<!--Attuse: bdo
     "lang" indicates the language of the content.
     "dir" indicates the writing direction: 'ltr' is
     left-to-right, 'rtl' is right-to-left.
     -->

<!ATTLIST bdo
   %coreattrs;
   lang %LanguageCode; #IMPLIED
   dir (ltr | rtl) #REQUIRED
   >

<!--=================== dtbook Inline Sentence and Word===================-->

<!--Use: sent marks a sentence.
     -->

<!ELEMENT sent (%inlines;)* >

<!ATTLIST sent
   %attrs;
   >

<!--Use: w marks a word.
     -->

<!ELEMENT w (%inlinew;)* >

<!ATTLIST w
   %attrs;
   >

<!--======== Inline Page Number, Footnote and Annotation Reference ========-->

<!--Use: pagenum contains one page number as it appears from the print
     document, usually inserted at the point within the file immediately
     preceding the first item of content on a new page.     -->

<!ELEMENT pagenum (#PCDATA) >

<!--Attuse: pagenum

     "page" allows three kinds of page numbering
     schemes to be identified:
         'normal' Arabic numbering in the body of the book is
             the default,
         'front' pages (from the <frontmatter>, often roman numbering),
```

```
         'special' pagination schemes such as letter prefix hyphen Arabic
             number in appendices.
     Each pagenum needs a unique id value, by convention it is derived
     from the actual pagenumber. For multi-page continuous content,
     such as large <img> or <table>, put the sequence of <pagenum> on
     the page where that content starts.
     -->

<!ATTLIST pagenum
   %attrsrqd;
   page (front | normal | special) 'normal'
   >

<!--Use: noteref marks one or more characters that reference a footnote
     or endnote <note>. Contrast with <annoref>. <noteref> and <note>
     are independently skippable.
     -->

<!ELEMENT noteref (#PCDATA) >

<!--Attuse: noteref

     "idref" relates to the note, for example:
     <noteref idref='yyy'> refers to <note id='yyy'>.

     "type" provides advisory content MIME type of
     the target, see [RFC1556].
     -->

<!ATTLIST noteref
   %attrs;
   idref CDATA #REQUIRED
   type %ContentType; #IMPLIED
   >

<!--Use: annoref marks a text segment that references an <annotation>.
     Each <annoref> is usually a word, phrase, or whole line that
     is part of the surrounding text (identified in the original
     print book by bolding, italics, etc.). It should not normally
     be allowed to be turned off in a DTB application.
     -->

<!ELEMENT annoref (#PCDATA) >

<!--Attuse: annoref

     "idref" refers to the target id of an
     <annotation>.

     "type" provides advisory content MIME
     type of the targeted id, see [RFC1556].
     -->

<!ATTLIST annoref
   %attrs;
   idref CDATA #REQUIRED
```

 ©NISO

```
   type %ContentType; #IMPLIED
   >

<!--============================ Inline Quotes ============================-->

<!--Use: q contains a short, inline quotation. Compare with
     <blockquote>, which marks a longer quotation set off from the
     surrounding text.
     -->

<!ELEMENT q (%inline;)* >

<!--Attuse: q

     "cite" may provide a URI reference.
     -->

<!ATTLIST q
   %attrs;
   cite %URI; #IMPLIED
   >

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

<!-- Image <img> comes from HTML. An <img> may be grouped
        using <imggroup>, with <caption>, and special
        usage instructions or description with <prodnote>. The <imggroup>
        element may contain one or more <img> and any associated
        <caption> and <prodnote>. Multiple <img> may share a single
        caption, or multiple <caption> may apply if several captions
        refer to a single <img>. Multiple <prodnote> may apply if
        different versions are needed for different media.
     -->

<!ENTITY % Length "CDATA" >
     <!-- measured in pixels, percent (nn%) -->

<!ENTITY % MultiLength "CDATA" >
     <!-- measured in integer pixels "n", percent "nn%" of display width,
          "0*" indicating minimum appropriate width based on column
          content, or "nn*" the relative proportional width (".5*" is
          half the available width after any explicit widths have been
          consumed). The lengths are separated by commas or whitespace. -->

<!ENTITY % Pixels "CDATA" >
     <!-- 0 for no <table> border, positive integer for<table> border width
         in pixels. -->

<!--HB: 2001-12-20 img Use: changed idref to imgref. -->

<!--Use: img marks a visual image. An <img> will alwayscontain an alt and
     generally contain a longdesc, a pointer to a related <prodnote>. The
     <img> may be referenced by a <caption> or <prodnote>, using, for
     example, the form <caption imgref="#yyy">the Caption</caption> for
     the <img id="yyy">.
     -->
```

```
<!ELEMENT img EMPTY >

<!--Attuse: img

     "src" specifies by URI the location of the image file.

     "alt" is used to supply a short description of the <img>.

     "longdesc" generally contains a pointer to a related
     <prodnote> that contains a detailed description of the <img>.

     The attributes "height" and "width" provide visual sizing
     information, measured in pixels.
     -->

<!ATTLIST img
   %attrs;
   src %URI; #REQUIRED
   alt %Text; #REQUIRED
   longdesc %URI; #IMPLIED
   height %Length; #IMPLIED
   width %Length; #IMPLIED
   >

<!--Use: imggroup provides a container for one or more <img> and associated
     <caption>(s) and <prodnote>(s). A <prodnote> may contain a description
     of the image. The content model allows:

     1) multiple <img> if they share a caption, with the ids of
     each <img> in the <caption imgref="id1 id2 ...">,
     2) multiple <caption> if several captions refer to a single
     <img id="xxx"> where each caption has the same
     <caption imgref="xxx">,

     3) multiple <prodnote> if different versions are needed for different
     media (e.g., large print, braille, or print).  If several <prodnote>
     refer to a single <img id="xxx">, each prodnote has the same <prodnote
     imgref="xxx">.
     -->

<!ELEMENT imggroup (prodnote | img | caption)+ >

<!ATTLIST imggroup
   %attrs;
   >

<!--============================ Horizontal Rule ============================-->

<!--Use: hr is an empty element, minimally <hr />, indicating a horizontal
     rule. It may be used to indicate a break in the text where only
     blank lines, a row of asterisks, a horizontal line, etc. are used
     in the print book.
     -->

<!ELEMENT hr EMPTY >
```

 ©NISO

```
<!--Attuse: hr

     "title" may be used to suggest the reason for the break.
     -->

<!ATTLIST hr
   %coreattrs;
   >

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

<!--Use: p contains a paragraph, which may contain subsidiary <list> or <dl>.
     -->

<!ELEMENT p (%inline; | %list; | dl)* >

<!ATTLIST p
   %attrs;
   >

<!--================== Doctitle, Docauthor, and Headings ==================-->

<!--Use: doctitle marks the title of the book within <frontmatter>.
     By convention <doctitle> should appear only once. Contrast with
     <title>, which occurs as metadata in <head> and whose content
     is generally the same.
     -->

<!ELEMENT doctitle (%inline;)* >

<!ATTLIST doctitle
   %attrs;
   >

<!--Use: docauthor marks each author or editor of this work. Compare with
     <author>, used to mark the author of another work, within <blockquote>
     or <cite>.
     -->

<!ELEMENT docauthor (%inline;)* >

<!ATTLIST docauthor
   %attrs;
   >

<!--Use: levelhd contains the text of a heading within <level>.
     Corresponds to <h1> through <h6> used in <level1> through <level6>.
     -->

<!--Attuse: levelhd

     "depth" is a positive integer, corresponding to the <h1>...<h6>
     levelN, though not limited to just six levels. Any depth value,
     "n", should match that on the enclosing <level depth="n">.
     -->
```

```
<!ELEMENT levelhd (%inline;)* >

<!ATTLIST levelhd
   %attrs;
   depth CDATA #IMPLIED
   >

<!--Use: h1 contains the text of the heading for a <level1> structure.
     -->

<!ELEMENT h1 (%inline;)* >

<!ATTLIST h1
   %attrs;
   >

<!--Use: h2 contains the text of the heading for a <level2> structure.
     -->

<!ELEMENT h2 (%inline;)* >

<!ATTLIST h2
   %attrs;
   >

<!--Use: h3 contains the text of the heading for a <level3> structure.
     -->

<!ELEMENT h3 (%inline;)* >

<!ATTLIST h3
   %attrs;
   >

<!--Use: h4 contains the text of the heading for a <level4> structure.
     -->

<!ELEMENT h4 (%inline;)* >

<!ATTLIST h4
   %attrs;
   >

<!--Use: h5 contains the text of the heading for a <level5> structure.
     -->

<!ELEMENT h5 (%inline;)* >

<!ATTLIST h5
   %attrs;
   >

<!--Use: h6 contains the text of the heading for a <level6> structure.
     -->
```

 ©NISO

```
<!ELEMENT h6 (%inline;)* >

<!ATTLIST h6
   %attrs;
   >

<!--Use: hd marks the text of a heading in a <list> or <sidebar>.
     -->

<!ELEMENT hd (%inline;)* >

<!ATTLIST hd
   %attrs;
   >

<!--========================== Preformatted Text ==========================-->

<!-- HTML or XHTML preformatted text is omitted, as inappropriate for
         narrated material. -->

<!--========================== Block-like Quotes ==========================-->

<!--Use: blockquote indicates a block of quoted content that is set
     off from the surrounding text by paragraph breaks. Compare with
     <q>, which marks short, inline quotations.
     -->

<!ELEMENT blockquote (%block;)* >

<!--Attuse: blockquote

     "cite" permits inclusion of the
     URI from which the <blockquote> came.
     -->

<!ATTLIST blockquote
   %attrs;
   cite %URI; #IMPLIED
   >

<!--================== Definition List, and Other Lists====================-->

<!--Use: dl contains a definition list, usually consisting of pairs of
     terms <dt> and definitions <dd>. Any definition can contain another
     definition list.
     -->

<!ELEMENT dl (dt | dd | pagenum)+ >

<!ATTLIST dl
  %attrs;
   >
```

```
<!--Use: dt marks a term in a definition list <dl> for which a
     definition <dd> follows.
     -->

<!ELEMENT dt (%inline;)* >

<!ATTLIST dt
   %attrs;
   >

<!--Use: dd marks a definition of the preceding term <dt> within a
     definition list <dl>. A definition without a preceding <dt> has
     no semantic interpretation, but is visually presented aligned
     with other <dd>.
     -->

<!ELEMENT dd (%flow;)* >

<!ATTLIST dd
   %attrs;
   >

<!--Use: list contains some form of list, ordered or unordered.
     The list may have intermixed heading <hd> (generally only one,
     possibly with <prodnote>) and an intermixture of list items <li>
     and <pagenum>. If bullets and outline enumerations are part of
     the print content, they are expected to prefix those list items
     in content, rather than be implicitly generated. Note: XHTML
     has explicitly distinguished list element types: ol for ordered,
     and ul for unordered.
     -->

<!ELEMENT list (hd | prodnote | li | pagenum)+ >

<!--Attuse: list

     "type" indicates whether the list items
     <li> are ordered 'ol' or unordered 'ul'.

     "depth" indicates nesting depth of lists within list items <li>
     of ancestor lists, starting at 1.

     "enum" indicates the kind of enumeration:
         '1'=integer,
         'a'=lowercase,
         'U'=uppercase,
         'i'=lowercase Roman, or
          X'=uppercase Roman.

     "bullet" can come from Unicode, using the entity     reference form
'&xdddd;'.
     -->

<!ATTLIST list
   %attrs;
   type (ol | ul) #IMPLIED
```

 ©NISO

```
   depth CDATA #IMPLIED
   enum (1 | a | U | i | X) #IMPLIED
   bullet CDATA #IMPLIED
   >

<!--Use: li marks each list item in a <list>. <li> content may be
     either inline or block and may include other nested
     lists. Alternatively it may contain a sequence of list item
     components, <lic>, that identify regularly occurring content,
     such as the heading and page number of each entry in a
     table of contents.
     -->

<!ELEMENT li (%flow; | lic)* >

<!ATTLIST li
   %attrs;
   >

<!--Use: lic ("list item component") allows ordered substructure
     within a list item <li>. Used when a list item is made up of
     two or more components, as in a table of contents entry.
     The same number of <lic> should occur in each <li>. If not,
     correspondence of <lic> in different <li> is in order of
     occurrence for the current writing direction of the <li>.
     -->

<!ELEMENT lic (%inline;)* >

<!--Attuse: lic
     class attribute may be used to identify the particular
     component of a list item <li>. For example, in a table of contents
     class values might include "section", and "pagenumber".
     -->

<!ATTLIST lic
   %attrs;
   >

<!--=============================== Tables ================================-->

<!-- The XHTML <table> model is used, including the presentational
         attributes that have little meaning in Digital Talking Books,
         but may be useful for concurrent display in different media.
         That model is derived from IETF HTML table standard, see [RFC1942].
         Note: The XHTML <table> model has been enhanced from HTML to allow
         a simple <table> of one or more rows <tr>.
     -->


<!ENTITY % Scope
     "(row | col | rowgroup | colgroup)" >
     <!-- Scope specifies a set of data cells for which the <th> provides
         header information. -->
```

```
<!ENTITY % TFrame
     "(void | above | below | hsides | lhs | rhs | vsides | box | border)" >
     <!-- TFrame identifies the sides that are visually framed. -->

<!--HB: 2001-12-20 % Trules added default meaning from xhtml strict dtd.-->

<!ENTITY % TRules
     "(none | groups | rows | cols | all)" >
     <!-- %TRules identifies where visual rulings appear.
         If no Trules is present then assume:
         'none' if border is absent or border='0' otherwise 'all'. -->

<!--HB: 2001-12-20 % cellhalign clarified interitance. -->

<!ENTITY % cellhalign
     "align      (left|center|right|justify|char) #IMPLIED
      char       %Character;    #IMPLIED
      charoff    %Length;       #IMPLIED" >
     <!-- % cellhalign
         cellhalign sets horizontal alignment of content in
         a table cell.

         char indicates a character expected in each table
         cell of a column that text should align on. The default is
         the decimal point for the current language.

         charoff sets the alignment offset of the first character
         to align on, as specified with char.

         Inheritance order for horizontal alignment is <th>|<td>,
         <tr>, <thead>|<tfoot>|<tbody>, <col>, <colgroup>, default. The
         default value is from the user agent, and may be affected
         by user preference. The recommended default alignment for
         <th> is center, and for <td> is left.
          -->

<!--HB: 2001-12-20 % cellvalign clarified interitance. -->

<!ENTITY % cellvalign
     "valign     (top|middle|bottom|baseline) #IMPLIED" >
     <!-- % cellvalign
         valign sets vertical alignment of content in a table cell.
         Inheritance order for vertical alignment is <th>|<td>, <col>,
         <colgroup>, <tr>, <thead>|<tfoot>|<tbody>, default. The
         default value is from the user agent, and may be affected
         by user preference. The recommended default is middle.
         -->

<!--Use: table contains cells of tabular data arranged in rows and
     columns. A <table> may have a <caption>. It may have descriptions of
     the columns in <col>s or groupings of several <col> in <colgroup>.
     A simple <table> may be made up of just rows <tr>. A long table
     crossing several pages of the print book should have separate
     <pagenum> values for each of the pages containing that <table>
     indicated on the page where it starts. Note the logical order of
     optional <thead>, optional <tfoot>, then one or more of either
```

 ©NISO

```
     <tbody> or just rows <tr>. This order accommodatessimple or large,
     complex tables. The <thead> and <tfoot> information usually helps
     identify content of the <tbody> rows, For a multiple-page print
     <table> the <thead> and <tfoot> are repeated on each page, but
     not redundantly tagged.
     -->

<!--HB: 2001-12-20 table updated model to xhtml strict dtd, adding
         simple table of just rows (tr). Added attribute inheritance
         information from RFC1942. -->

<!ELEMENT table (caption?, (col* | colgroup*), thead?, tfoot?, (tbody+| tr+)) >

<!--Attuse: table

     "summary" value provides a textual summary.

     The attributes: "width", "border", "frame", "rules", "cellspacing",
     and "cellpadding" provide visual presentation guidance. See their
     explanation in the comment following those parameter entity
     declarations.
     -->

<!ATTLIST table
   %attrs;
   summary %Text; #IMPLIED
   width %Length; #IMPLIED
   border %Pixels; #IMPLIED
   frame %TFrame; #IMPLIED
   rules %TRules; #IMPLIED
   cellspacing %Length; #IMPLIED
   cellpadding %Length; #IMPLIED
   >

<!--Use: caption describes a <table> or <img>. If used with <table> it
     must follow immediately after the <table> start tag. If used
     with <img> or <imggroup> it is not so constrained.     -->

<!ELEMENT caption (%inline;)* >

<!--HB: 2001-12-20 caption added imgref extension for dtbook comment. -->

<!--Attuse: caption

     "imgref" value (or space-separated
     id values) identifies the id values of the <img>(s) to which the
     caption applies. Any internal id reference is preceded by '#id'.
     Note: imgref is an extension for dtbook.
     -->

<!ATTLIST caption
   %attrs;
   imgref IDREFS #IMPLIED
   >

<!--HB: 2001-12-20 thead updated Use to xhtml strict dtd. -->
```

```
<!--Use: thead marks header information in a <table>, consisting of
     one or more rows <tr> of <th> cells. Use <thead> to duplicate
     headers when breaking table across page boundaries, or for static
     headers when <tbody> sections are rendered in scrolling panel.
     -->

<!ELEMENT thead (tr)+ >

<!ATTLIST thead
   %attrs;
   %cellhalign;
   %cellvalign;
   >

<!--HB: 2001-12-20 tfoot updated Use to xhtml strict dtd. -->

<!--Use: tfoot marks footer information in a <table>, consisting of
     one or more rows <tr>, usually of <th> cells. Use <tfoot> to
     duplicate footers when breaking table across page boundaries,
     or for static footers when <tbody> sections are rendered in
     scrolling panel.
     -->

<!ELEMENT tfoot (tr)+ >

<!ATTLIST tfoot
   %attrs;
   %cellhalign;
   %cellvalign;
   >

<!--HB: 2001-12-20 tbody updated Use to xhtml strict dtd. -->

<!--Use: tbody marks a group of rows in the main body of a <table>. If
     the <table> is divided into several sections, each consisting of a
     number of rows, each section would be separately tagged with <tbody>.
     The same <thead> and <tfoot> apply to every <tbody> section.
     Use multiple <tbody> sections when rules are needed between groups
     of table rows.
     -->

<!ELEMENT tbody (tr)+ >

<!ATTLIST tbody
   %attrs;
   %cellhalign;
   %cellvalign;
   >

<!--Use: colgroup groups adjacent columns <col> that are semantically
     related.
     -->

<!ELEMENT colgroup (col)* >

<!--HB: 2001-12-20 colgroup updated Attuse to xhtml strict dtd. -->

<!--Attuse: colgroup
```

 ©NISO

```
     "span" causes the attributes of one
     col element to apply to more than one column, extending in the
     writing direction for the language. Span is ignored if
     any <col> are present.
     "width" specifies the width of the columns, e.g.

         width='64'        width in screen pixels
         width='0.5*'      relative width of 0.5

     "width" may contain a space- or comma-separated list of widths
     for each <col>, or percentages if values end in '%', or relative
     'nn*' to indicate a fractional portion of the remaining
     widths after more explicit forms have been specified for other
     columns in the colgroup (presumably the sum encompasses all
     the columns). '0*' indicates minimal acceptable width based
     on column content.

     The <col> in a <colgroup> may inherit attribute values
     from it, or the closest enclosing ancestor element:
     <thead>|<tfoot>|<tbody>. If none, then the values are up to the
     user agent, possibly by user preference.
     -->

<!ATTLIST colgroup
   %attrs;
   span NMTOKEN '1'
   width %MultiLength; #IMPLIED
   %cellhalign;
   %cellvalign;
   >

<!--Use: col elements define the alignment properties for cells in
     one or more columns.
     -->

<!ELEMENT col EMPTY >

<!--Attuse: col
     "span" indicates how many columns the <col>
     extends, in the writing direction of the <table>. The
     attribute values apply to <th> and <td> that start in the
     column, even if they extend into the next column(s), by
     span value more than 1, and that next <col> may have different
     attribute values.

     "width" may contain a space- or comma-separated list of widths
     for each <col> of the span, or percentages if values end in '%',
     or relative 'nn*' to indicate a fractional portion of the remaining
     widths after more explicit forms have been specified for other
     columns in the span (presumably the sum encompasses all
     the columns of this span, and of other <colgroup>s). '0*' indicates
     minimal acceptable width based on individual <col> content.

     <col> may inherit attribute values from the most immediate
     enclosing ancestor element: <colgroup>, <thead>|<tbody>|<tfoot>.
     If none, it is up to the user agent, possibly by user preference.
     -->
```

```
<!ATTLIST col
   %attrs;
   span NMTOKEN '1'
   width %MultiLength; #IMPLIED
   %cellhalign;
   %cellvalign;
   >

<!--Use: tr marks one row of a <table> containing <th> or <td> cells.
     -->

<!ELEMENT tr (th | td)+ >

<!--Attuse: tr

     %cellvalign; values provide default vertical alignment
     values for <th> and <td> in the row, overriding any from
         <thead>|<tbody>|<tfoot>, <col>, <colgroup>
     If none, it is up to the user agent, possibly by user preference.

     %cellhalign; values provide default horizontal alignment from
         <thead>|<tbody>|<tfoot>
     If none, it is up to the user agent, possibly by user preference.
     -->

<!ATTLIST tr
   %attrs;
   %cellhalign;
   %cellvalign;
   >

<!--Use: th indicates a table cell containing header information.
     -->

<!ELEMENT th (%flownopagenum;)* >

<!--Attuse: th
     The uses of attributes other than %attrs; %cellvalign; and %cellhalign;
     are shown below. See [HTML401STRICT] for details and examples.

     "abbr" provides an abbreviated name for a <th> cell that can be used
     when referring to that <th> cell. Its default value is the cell content.

     "axis" is used to place cells into conceptual categories in order to
     provide improved access to information.

     "headers" provides the id value(s), used with <td> cells, to reference
     one or more cells with <th id="xxx"> that contain headings that
     collectively describe or qualify the content of the cell, for example
     <td headers="id1 id2">.

     "scope" identifies one of
         (row | rowgroup | column | colgroup)
     to which the header information applies.

     "rowspan" indicates the total number of rows that the cell extends, by
     default 1. All spanned cells share these attributes.

     "colspan" indicates the total number of columns the cell extends,
     by default 1, in the writing direction of the table. All spanned
     cells share these attributes.
     -->
```

 ©NISO

```
<!ATTLIST th
   %attrs;
   abbr %Text; #IMPLIED
   axis CDATA #IMPLIED
   headers IDREFS #IMPLIED
   scope %Scope; #IMPLIED
   rowspan NMTOKEN '1'
   colspan NMTOKEN '1'
   %cellhalign;
   %cellvalign;
   >

<!--Use: td indicates a table cell containing data.
     -->

<!ELEMENT td (%flownopagenum;)* >

<!--Attuse: td

     The uses of attributes other than %attrs; %cellhalign; and %cellvalign;
     are shown below.  See [HTML401STRICT] for details and examples.

     "abbr" provides an abbreviated name for a <th> cell that can be used
     when referring to that <th> cell. Its default value is the cell content.

     "axis" is used to place cells into conceptual categories in order to
     provide improved access to information.

     "headers" provides the id value(s), used with <td> cells, to reference
     one or more cells with <th id="xxx"> that contain headings that
     collectively describe or qualify the content of the cell, for example
     <td headers="id1 id2">.

     "scope" identifies one of
         (row | rowgroup | column | colgroup)
     to which the header information applies.

     "rowspan" indicates the total number of rows that the cell extends, by
     default 1. All spanned cells share these attributes.

     "colspan" indicates the total number of columns the cell extends,
     by default 1, in the writing direction of the table. All spanned
     cells share these attributes.
     -->

<!ATTLIST td
   %attrs;
   abbr %Text; #IMPLIED
   axis CDATA #IMPLIED
   headers IDREFS #IMPLIED
   scope %Scope; #IMPLIED
   rowspan NMTOKEN '1'
   colspan NMTOKEN '1'
   %cellhalign;
   %cellvalign;
   >
```

Appendix 2 - DTB-Specific SMIL DTD

(This section is normative.)

The following DTD is available in plain-text form from the maintenance agency at http://www.loc.gov/nls/z3986/v100/.

```
<!--SMIL 2.0 DTB-specific DTD Version 1.1.0 2002-02-12
file: dtbsmil110.dtd

Authors: Michael Moodie, Tom McLaughlin, Lloyd Rasmussen
Change list:
2002-02-12  M. Moodie. Dropped version attribute on smil element, as this at-
tribute is not present in SMIL specification.

Description:
This DTD is intended for use only with DTB applications.  Documents valid to
this DTD will also be valid to the DTB SMIL Profile, but not necessarily vice
versa, as this DTD contains only a subset of the elements and attributes
present in the DTB SMIL Profile.  This DTD is in some areas more restrictive
than the Profile (e.g., requiring IDs on some elements), to enforce structure
critical to the DTB application.

The following identifiers apply to this DTD:
"-//NISO//DTD dtbsmil v1.1.0//EN"
"http://www.loc.gov/nls/z3986/v100/dtbsmil110.dtd"
-->

<!ENTITY   % Core.attrib
       "id          ID           #IMPLIED
       class        CDATA        #IMPLIED
       title        CDATA        #IMPLIED"
>
<!ENTITY % URI "CDATA">
    <!-- a Uniform Resource Identifier, see [RFC2396] -->

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

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

<!ELEMENT       meta     EMPTY >
<!ATTLIST       meta
       name         CDATA        #REQUIRED
       content      CDATA        #IMPLIED
>
<!-- only smil basic layout allowed; not CSS2.
     root-layout not included, is implementation dependent.
-->
```

 ©NISO

```
<!ELEMENT       layout   (region)+ >
<!ATTLIST       layout
       %Core.attrib;
       xml:lang     NMTOKEN      #IMPLIED
>

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

<!ELEMENT       customAttributes      (customTest)+ >
<!ATTLIST       customAttributes
       %Core.attrib;
       xml:lang     NMTOKEN      #IMPLIED
>

<!ELEMENT       customTest        EMPTY >
<!ATTLIST       customTest
       id                    ID                   #REQUIRED
       class                 CDATA                #IMPLIED
       defaultState          (true|false)         'false'
       title                 CDATA                #IMPLIED
       xml:lang              NMTOKEN              #IMPLIED
       override              (visible|hidden)     'hidden'
>
<!-- Even though body functions as a seq, and you don't need a base set of
seqs wrapping the whole presentation, for DTB applications a base set of
seqs should be used.
The dur attribute on the first seq is used by the player to determine the
length of the SMIL presentation. -->

<!ELEMENT       body     (par|seq|text|audio|img|a)+ >
<!ATTLIST       body
       %Core.attrib;
       xml:lang          NMTOKEN          #IMPLIED
>

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

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

<!ELEMENT      text     EMPTY >
<!ATTLIST      text
       id               ID           #IMPLIED
       region           CDATA        #IMPLIED
       src              CDATA        #REQUIRED
       type             CDATA        #IMPLIED
>


<!ELEMENT      audio    EMPTY >
<!ATTLIST      audio
       id               ID           #IMPLIED
       src              CDATA        #REQUIRED
       type             CDATA        #IMPLIED
       clipBegin        CDATA        #IMPLIED
       clipEnd          CDATA        #IMPLIED
       region           CDATA        #IMPLIED
>

<!ELEMENT      img  EMPTY >
<!ATTLIST      img
       id               ID           #IMPLIED
       region           CDATA        #IMPLIED
       src              CDATA        #REQUIRED
       type             CDATA        #IMPLIED
>

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

 ©NISO

Appendix 3 - NCX DTD

(This section is normative.)

The following DTD is available in plain-text form from the maintenance agency at http://www.loc.gov/nls/z3986/v100/.

```
<!-- NCX 1.1.0 DTD  2002-02-27
file: ncx110.dtd

  Authors: Mark Hakkinen, George Kerscher, Tom McLaughlin, James Pritchett, and
  Michael Moodie
  Change list:
  2002-02-12 M. Moodie. Changed content model of navLabel element to eliminate
  ambiguity.
  2002-02-27 M. Moodie. Grammatical changes suggested by editor.

  Description:

  NCX (Navigation Control for XML applications) is a generalized navigation
  definition DTD for application to Digital Talking Books, eBooks, and general
  web content models. This DTD is an XML application that layers navigation
  functionality on top of SMIL 2.0 content.

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

-->

<!-- The following identifiers apply to this DTD:
       "-//NISO//DTD ncx v1.1.0//EN"
       "http://www.loc.gov/nls/z3986/v100/ncx110.dtd"
-->

<!-- Basic Entities -->

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

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

<!-- ELEMENTS -->

<!-- Top Level NCX Container. -->
<!ELEMENT ncx (head, docTitle, docAuthor*, navMap, navList*)>
<!ATTLIST ncx
  version     CDATA     #FIXED "1.1.0"
  %i18n;
>
<!-- Document Head - Contains all NCX metadata.
-->
```

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

<!-- smilCustomTest - Duplicates customTest data found in SMIL files. Each unique
customTest element that appears in one or more SMIL files must have its attributes
duplicated in a smilCustomTest element in the NCX. The NCX thus gathers in one place
all customTest elements used in the SMIL files, for presentation to the user.
-->
<!ELEMENT smilCustomTest EMPTY>
<!ATTLIST smilCustomTest
id              ID                #REQUIRED
defaultState    (true|false)      'false'
override        (visible|hidden) 'hidden'>

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

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

<!ELEMENT docTitle (text, audio?)>
<!ATTLIST docTitle
  id            ID           #IMPLIED
  %i18n;
>

<!-- DocAuthor - the author of the document, immediately follows docTitle.
-->

<!ELEMENT docAuthor (text, audio?)>
<!ATTLIST docAuthor
  id            ID           #IMPLIED
  %i18n;
>

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

<!ELEMENT navMap (navLabel*, navPoint+)>
<!ATTLIST navMap
  id            ID           #IMPLIED
>

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

 ©NISO

```
<!ELEMENT navPoint (navLabel+, content, navPoint*)>
<!ATTLIST navPoint
  id            ID                #REQUIRED
  onFocus       %script;          #IMPLIED
  onBlur        %script;          #IMPLIED
  class         CDATA             #IMPLIED
  value         CDATA             #IMPLIED
  pageRef       IDREF             #IMPLIED
>

<!-- Navigation List - container for distinct, flat sets of navigable elements,
e.g.  page numbers, notes, figures, tables, etc.  Essentially a flat version of
navMap.  The "class" attribute describes the type of object contained in this
navList, using dtbook element names, e.g., pagenum, note.
-->

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

<!-- Navigation Target - contains description(s) of target, as well as a
pointer to entire content of target.  navTargets are the equivalent of
navPoints for use in navLists. "mapRef" is the id of another navPoint within
this NCX that contains this navTarget.  "class" attribute describes the kind of
structure this target represents, using its dtbook element name, e.g., pagenum,
note.
-->

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

<!-- Navigation Label - Contains a description of a given <navMap>, <navPoint>,
<navList>, or <navTarget> in various media for presentation to the user. Can be
repeated so descriptions can be provided in multiple languages. -->
<!ELEMENT navLabel (((text, audio?) | audio), img?)>
<!ATTLIST navLabel
       %i18n;
>

<!-- Content Element - pointer into SMIL to beginning of navPoint. -->
<!ELEMENT content EMPTY>
<!ATTLIST content
  id            ID                #IMPLIED
  src           %uri;             #REQUIRED
>
<!-- Text Element - Contains text of docTitle, navPoint heading, navTarget
```

```
(e.g., page number), or label for navMap or navList. -->
<!ELEMENT text (#PCDATA)>
<!ATTLIST text
  id            ID                    #IMPLIED
  class         CDATA                 #IMPLIED
>
<!-- Audio Element - audio clip of navPoint heading. -->
<!ELEMENT audio EMPTY>
<!ATTLIST audio
  id            ID                    #IMPLIED
  class         CDATA                 #IMPLIED
  src           %uri;                 #REQUIRED
  clipBegin     %SMILtimeVal;         #IMPLIED
  clipEnd       %SMILtimeVal;         #IMPLIED
>

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


©NISO

Appendix 4 - DTD for Portable Bookmarks/Highlights

(This section is normative.)

The following DTD is available in plain-text form from the maintenance agency at http://www.loc.gov/nls/z3986/v100/.

```
<!-- bookmark 1.0.0 DTD 2001-09-27
file: bookmark100.dtd

Authors: Tom McLaughlin and Michael Moodie

The following identifiers apply to this DTD:
"-//NISO//DTD bookmark v1.0.0//EN"
"http://www.loc.gov/nls/z3986/v100/bookmark100.dtd"
-->

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

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

<!-- Bookmark: Location and optional note. Location consists of a uri pointing
to the id attribute of the <par> element in the SMIL file that contains the
bookmark plus a time offset in seconds (or character offset) to the exact
place.  Player should by default automatically number bookmarks in the order in
which they fall in the book.
 -->
<!ELEMENT bookmark      (ncxRef, uri, (timeOffset | charOffset), note?) >
<!ATTLIST bookmark
       label            CDATA            #IMPLIED
>
<!-- NcxRef: Captures current location in NCX (the id of the current
navPoint)at time lastmark, bookmark, or highlight is set.  Ensures that current
location in NCX and SMIL are synchronized after moving to a lastmark, etc., so
that any global navigation commands issued by the user will start from the
current location. -->
<!ELEMENT ncxRef       (#PCDATA)>

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

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

<!-- HilStart: Starting point of highlighted block.
 -->
<!ELEMENT hiliteStart   (ncxRef, uri, (timeOffset | charOffset)) >

<!-- HilEnd: End point of highlighted block.
 -->
<!ELEMENT hiliteEnd          (ncxRef, uri, (timeOffset | charOffset)) >

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

<!-- Timeoffset: Exact position of bookmark in SMIL file or audio-only file
referenced by the uri; in seconds.fraction (seconds=DIGIT+, fraction=3DIGIT).
 -->
<!ELEMENT timeOffset    (#PCDATA) >

<!-- Charoffset: Exact position of bookmark in text-only file referenced by the
uri: in characters, counting from nearest previous tag with an id.  White space
is normalized (collapsed to one character) and tags are not counted.
 -->
<!ELEMENT charOffset     (#PCDATA) >

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

<!-- Text: Text of title or note.
 -->
<!ELEMENT text  (#PCDATA) >

<!-- Audio: Audio clip of user-recorded note, in any format supported by
standard.
 -->
<!ELEMENT audio EMPTY >
<!ATTLIST audio

   src         %uri;    #REQUIRED
   clipBegin   CDATA    #IMPLIED
   clipEnd     CDATA    #IMPLIED
>
```

 ©NISO

Appendix 5 - DTD for Resource File

(This section is normative.)

The following DTD is available in plain-text form from the maintenance agency at http://www.loc.gov/nls/z3986/v100/.

```
<!-- Resource File 1.1.O DTD 2002-02-27
 file: resource110.dtd

Authors: Tom McLaughlin, Michael Moodie, Thomas Kjellberg Christensen
Change list:
2001-12-06  M. Moodie. Changed content model of resource element to eliminate
   ambiguity.
2002-02-12  M. Moodie. Changed dtd version from 1.0.1 to 1.1.0 per changes in
   other DTDs.
2002-02-27  M. Moodie. Grammatical changes suggested by editor.

The following identifiers apply to this DTD:
"-//NISO//DTD resource v1.1.0//EN"
"http://www.loc.gov/nls/z3986/v100/resource110.dtd"
-->

<!-- ********** Attribute Types *********** -->
<!-- languagecode: An RFC1766 language code. -->
<!ENTITY % languagecode "NMTOKEN">

<!-- SMILtimeVal: SMIL 2.0 clock value. -->
<!ENTITY % SMILtimeVal "CDATA">

<!ENTITY % URI "CDATA">

<!-- **************** Resource Elements ********** -->

<!-- Resources: Root element of DTD.
-->
<!ELEMENT resources  (head?, (resource)+) >
<!ATTLIST resources
  version      CDATA        #FIXED "1.1.0"
>
<!-- Document Head - Contains metadata.
-->
<!ELEMENT head (meta*)>

<!-- Resource element contains information about the alternative representations
of an element present in the NCX or the textual content file. An alternative
representation can be used to convey navigational information, e.g., provide a
descriptive name for the kind of segment (part, chapter, section, etc.) the
user is encountering.  In addition, it can supply accessible versions of dtbook
element names and names of skippable structures listed in the head of the NCX.
Text can be used for screen or Braille display, audio for digital talking book
players,
and image for screen display.
```

```
Attribute use:
type - Specifies whether the resource applies to the textual content file
(dtbook) or the NCX (ncx).
elementRef - Specifies the name of the element for which the resource is to be
supplied.
classRef - Specifies the class attribute value of the element for which the
resource is to be supplied.
idRef - Specifies the name of the id attribute on the smilCustomTest element in
NCX for which the resource is to be supplied.
lang - Specifies the language of the resource item, using an RFC 1766 language
code.
-->

<!ELEMENT resource  (((text, audio?) | audio), img?) >
<!ATTLIST resource
  type          (ncx | dtbook)    #REQUIRED
  elementRef    CDATA             #REQUIRED
  classRef      CDATA             #IMPLIED
  idRef         CDATA             #IMPLIED
  lang          %languagecode;    #IMPLIED
>
<!ELEMENT text  (#PCDATA) >

<!ELEMENT audio  EMPTY >
<!ATTLIST audio
  src           %URI;             #REQUIRED
  clipBegin     %SMILtimeVal;     #IMPLIED
  clipEnd       %SMILtimeVal;     #IMPLIED
>

<!-- If the clipBegin attribute is not present in an instance of the
audio element, the audio file referenced must be played from its beginning.
If the clipEnd attribute is not present, the audio file must be played to
its end. If the value of the clipEnd attribute exceeds the duration of
the audio file, the value must be ignored, and the audio file played to
its end.
-->

<!ELEMENT img  EMPTY >
<!ATTLIST img
  src           %URI;             #REQUIRED
>

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

 ©NISO

Appendix 6 - Distribution Information DTD

(This section is normative.)

The following DTD is available in plain-text form from the maintenance agency at http://www.loc.gov/nls/z3986/v100/.

```
<!-- distInfo 1.1.0 DTD 2002-02-27
file: distInfo110.dtd

Author: James Pritchett
Change list:
2001-12-06  M. Moodie. Changed content model of changeMsg to eliminate
  ambiguity.
2002-02-12  M. Moodie. Changed dtd version from 1.0.1 to 1.1.0 per changes in
  other DTDs.
2002-02-27  M. Moodie. Grammatical changes suggested by editor.

Description:
An XML application to describe the contents of a single piece of DTB
distribution media.  It consists of a list of books to be found on the
media.  For each book, distInfo identifies the location of each book
within the media filesystem.  If the book is being distributed on multiple
distribution media (media units), the distInfo book element also includes:
1) the sequence id of this media unit
2) a distribution map for the book, telling where to find all the SMIL files
for a book

The following identifiers apply to this DTD:
"-//NISO//DTD distInfo v1.1.0//EN"
"http://www.loc.gov/nls/z3986/v100/distInfo110.dtd"

-->
<!-- * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * -->

<!ENTITY % URI      "CDATA">
<!ENTITY % SMILtimeVal  "CDATA">

<!-- distInfo:  Root element, consists of one or more books.
"version" specifies the version of this DTD used in this instance. Three
digits, with decimal point separators; digits one, two and three will
reflect major, moderate and minor changes, respectively. This attribute
must be present but parsers will not enforce its presence, just its value.
-->
<!ELEMENT distInfo (book+)>
<!ATTLIST distInfo
           version      CDATA        #FIXED "1.1.0"
>

<!-- book:  a DTB that is present, in part or whole, on this piece of
distribution media.  The uid and pkgRef attributes are required.  "uid"
matches the package unique-identifier.  "pkgRef" is a URI that locates the
book's package file on this media unit.

If this is a book fragment, then the "media" attribute identifies which
fragment is stored on this media unit, and a single distMap element
is present to describe which SMIL files are present on which media units.
```

```
The media attribute is 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.
In the case of a book fragment, <book> should contain exactly one
<distMap> and optionally one or more <changeMsg> elements.
-->
<!ELEMENT book (distMap?, changeMsg*)>
<!ATTLIST book
           uid          CDATA        #REQUIRED
           pkgRef       CDATA        #REQUIRED
           media        CDATA        #IMPLIED
>
<!-- distMap:  a map identifying which media the various SMIL files
reside upon.  This consists of one or more smilRef elements.  The
distMap smilRefs should match one-to-one those of the book package spine.
-->
<!ELEMENT distMap (smilRef+)>

<!-- smilRef:  a reference to a DTB SMIL file.  These are referenced
by file name. The mediaRef attribute of each smilRef identifies the piece of
media that the file resides upon, and is in the format "x:y" (see above).
-->
<!ELEMENT smilRef EMPTY>
<!ATTLIST smilRef
           file         CDATA        #REQUIRED
           mediaRef     CDATA        #REQUIRED
>

<!-- changeMsg:  A pointer to a custom message to be read when a new disk is
requested by the reading system.  "mediaRef" identifies the media unit which
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>.  "mediaRef" is 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.
-->
<!ELEMENT changeMsg ((text, audio?) | audio)>
<!ATTLIST changeMsg
           mediaRef     CDATA        #REQUIRED
           lang         NMTOKEN      #IMPLIED
>

<!-- text: Contains text of media change message.
-->
<!ELEMENT text (#PCDATA)>

<!-- audio: Pointer to audio content of media change message.
-->
<!ELEMENT audio EMPTY>
<!ATTLIST audio
           src          %URI;                #REQUIRED
           clipBegin    %SMILtimeVal;        #IMPLIED
           clipEnd      %SMILtimeVal;        #IMPLIED
>
```

 ©NISO

Appendix 7 - Designation of Maintenance Agency

(This Appendix is not part of American National Standard Z39.86-2002, Specifications for the Digital Talking Book. It is included for information only.)

The functions assigned to the maintenance agency as specified in section 1.7 will be administered by the National Library Service for the Blind and Physically Handicapped, Library of Congress. Questions concerning the implementation of this standard and requests for information should be sent to the Research and Development Officer, National Library Service for the Blind and Physically Handicapped, Library of Congress, Washington, DC 20542, or nls@loc.gov, including "Z3986" in the subject line.

©NISO