

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Digital audiobook file format and player requirements

Exigences relatives au format de fichier de livre audio et aux lecteurs

IECNORM.COM : Click to view the full PDF of IEC 62571:2011



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2011 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester.

If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de la CEI ou du Comité national de la CEI du pays du demandeur.

Si vous avez des questions sur le copyright de la CEI ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de la CEI de votre pays de résidence.

IEC Central Office
3, rue de Varembe
CH-1211 Geneva 20
Switzerland
Email: inmail@iec.ch
Web: www.iec.ch

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

- Catalogue of IEC publications: www.iec.ch/searchpub

The IEC on-line Catalogue enables you to search by a variety of criteria (reference number, text, technical committee,...). It also gives information on projects, withdrawn and replaced publications.

- IEC Just Published: www.iec.ch/online_news/justpub

Stay up to date on all new IEC publications. Just Published details twice a month all new publications released. Available on-line and also by email.

- Electropedia: www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 20 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary online.

- Customer Service Centre: www.iec.ch/webstore/custserv

If you wish to give us your feedback on this publication or need further assistance, please visit the Customer Service Centre FAQ or contact us:

Email: csc@iec.ch

Tel.: +41 22 919 02 11

Fax: +41 22 919 03 00

A propos de la CEI

La Commission Electrotechnique Internationale (CEI) est la première organisation mondiale qui élabore et publie des normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

A propos des publications CEI

Le contenu technique des publications de la CEI est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

- Catalogue des publications de la CEI: www.iec.ch/searchpub/cur_fut-f.htm

Le Catalogue en-ligne de la CEI vous permet d'effectuer des recherches en utilisant différents critères (numéro de référence, texte, comité d'études,...). Il donne aussi des informations sur les projets et les publications retirées ou remplacées.

- Just Published CEI: www.iec.ch/online_news/justpub

Restez informé sur les nouvelles publications de la CEI. Just Published détaille deux fois par mois les nouvelles publications parues. Disponible en-ligne et aussi par email.

- Electropedia: www.electropedia.org

Le premier dictionnaire en ligne au monde de termes électroniques et électriques. Il contient plus de 20 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans les langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International en ligne.

- Service Clients: www.iec.ch/webstore/custserv/custserv_entry-f.htm

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions, visitez le FAQ du Service clients ou contactez-nous:

Email: csc@iec.ch

Tél.: +41 22 919 02 11

Fax: +41 22 919 03 00

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Digital audiobook file format and player requirements

Exigences relatives au format de fichier de livre audio et aux lecteurs

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX

XE

ICS 33.160.60; 35.240.30

ISBN 978-2-88912-603-3

CONTENTS

FOREWORD.....	7
1 Scope.....	9
2 Normative references	9
3 General	9
4 MultiAudio audiobook extensions.....	11
4.1 Table of contents.....	11
4.2 Description	14
4.2.1 General	14
4.2.2 Tag.Identifier	14
4.2.3 Tag.OrdinalNumber	14
4.2.4 Tag.Reserved.....	14
4.2.5 Tag.Length	14
4.2.6 Data	14
5 Audiobook extendable structure (AB_ExtStruct)	15
5.1 General.....	15
5.2 Tag	15
5.3 Offset to extra data.....	15
5.4 Fixed length chunk (AB_FixedChunk).....	15
5.4.1 General	15
5.4.2 Example	16
5.5 Variable length chunk (AB_VariableChunk)	16
5.5.1 General	16
5.5.2 Example	17
5.6 Extra data	18
6 Audiobook header (AB_Header).....	18
6.1 General.....	18
6.2 Tag	18
6.3 AB_Header fixed chunk	18
6.3.1 General	18
6.3.2 Tag.....	19
6.3.3 Audiobook specification major version	19
6.3.4 Audiobook specification minor version	19
6.3.5 Number of audiobooks (N_Bk).....	19
6.3.6 Offset to playlist indexes	19
6.4 AB_Header variable chunk	19
6.4.1 General	19
6.4.2 Tag.....	19
6.4.3 Playlist indexes	20
6.5 AB_Header extra data	20
7 Audiobook book (AB_Book).....	20
7.1 General.....	20
7.2 Tag	21
7.3 AB_Book fixed chunk	21
7.3.1 General	21
7.3.2 Tag.....	22

7.3.3	Text format	22
7.3.4	Flags	22
7.3.5	TOC_Playlist index	22
7.3.6	Offset to book title	22
7.3.7	Offset to author	22
7.3.8	Offset to narrator	22
7.3.9	Offset to copyright statement	22
7.3.10	Offset to audiobook UUID	22
7.3.11	Offset to book ISBN identifier	23
7.3.12	Offset to book ISSN identifier	23
7.3.13	Offset to book DOI identifier	23
7.3.14	Offset to book index number	23
7.3.15	Offset to book shelf number	23
7.3.16	Offset to book part number	23
7.3.17	Offset to other identifier	23
7.3.18	Offset to first audiobook level	23
7.3.19	Offset to playlist content location identifier	23
7.4	AB_Book variable chunk	24
7.4.1	General	24
7.4.2	Tag	24
7.4.3	Book title	24
7.4.4	Author	25
7.4.5	Narrator	25
7.4.6	Copyright Statement	25
7.4.7	Book UUID	25
7.4.8	Book ISBN Identifier	25
7.4.9	Book ISSN Identifier	25
7.4.10	Book DOI Identifier	26
7.4.11	Book index number	26
7.4.12	Book shelf number	26
7.4.13	Book part number	26
7.4.14	Other identifier	26
7.4.15	First audiobook level	26
7.4.16	Playlist content location identifier	27
7.5	AB_Book extra data	27
8	Audiobook level (AB_Level)	27
8.1	General	27
8.2	Tag	28
8.3	AB_Level fixed chunk	28
8.3.1	General	28
8.3.2	Tag	28
8.3.3	Text format	29
8.3.4	Flags	29
8.3.5	Number of sub-levels	29
8.3.6	Level number	29
8.3.7	TOC_TrackEntry index	29
8.3.8	Offset to sub-level chunk	29
8.3.9	Offset to level name	29
8.3.10	Offset to level description	29

8.3.11	Book time offset	29
8.3.12	Time start offset	29
8.3.13	Time duration	30
8.3.14	Offset to parent AB_Level.....	30
8.3.15	Offset to previous AB_Level	30
8.3.16	Offset to next AB_Level	30
8.4	AB_Level variable chunk	30
8.4.1	General	30
8.4.2	Tag.....	30
8.4.3	Level name.....	31
8.4.4	Level description	31
8.4.5	Sub-level chunk.....	31
8.5	AB_Level extra data	31
9	MPV audiobook profiles.....	31
9.1	General	31
9.2	XML schema namespace identifier	32
9.3	Use of MPV-Music profile identifier.....	32
9.4	Audiobook file	33
9.5	Audiobook profile metadata	33
9.6	MPV audiobook profile metadata	35
9.7	Dublin core metadata usage	39
9.8	Audiobook profile using MPV-Music playlists.....	41
9.8.1	General	41
9.8.2	Number of playlists per MPV file.....	42
9.8.3	Metadata usage.....	42
9.8.4	Background usage.....	43
9.8.5	Foreground usage.....	44
9.8.6	Related and rendition assets	44
9.9	Groups of assets in playlists.....	44
9.10	MPV-Audiobook schema details	47
9.10.1	XML schema definition	47
9.10.2	Audiobook metadata usage definitions.....	52
9.11	Audiobook level class identifiers.....	52
9.12	Audiobook BOOKMARK.AUB	54
9.13	MPV-Audiobook-Level metadata using <mpvp:Album>	61
9.14	<mpvm:AudioWithStills> Music and Stills Asset.....	61
9.15	MPV-Audiobook extensions to MPV-Core	64
9.15.1	Audiobook manifest file types and extensions.....	64
9.15.2	Audiobook manifest MIME media type	64
9.16	Media types and file formats.....	65
9.17	MPV-Audiobook and CEA-2003-C	65
9.18	CEA-2003-C binary to XML formats – Tag comparison	65
10	Conforming player and digital audiobook file format structure requirements.....	66
10.1	General	66
10.2	Player Requirements	66
10.3	Digital audiobook file format structure requirements	67
11	Player and digital audiobook file format structure functionality recommendations	67
11.1	General	67

11.2 Player functionality recommendations	67
11.3 Digital audiobook file format structure functionality recommendations	68
12 Digital audiobook file format structures and data constraints	68
13 Additional production options.....	68
13.1 ID3 tagging within MP3 files	68
13.2 M3U playlists.....	69
13.3 DAISY production options.....	69
14 Certification	69
Annex A (informative) Audiobook structure examples.....	70
Annex B (normative) Encoding types and identifiers	82
Annex C (informative) Use case and implementation examples: Spoken word audio.....	83
Bibliography.....	104
Figure A.1 – AB_Level structure organization in AB_Book variable length chunk area	74
Figure A.2 – View of an audiobook AB_Level hierarchy	76
Figure C.1 – Example 1 INDEX.AUB	88
Figure C.2 – Example 1 TOC.MUM	89
Figure C.3 – Example 2 INDEX.AUB	94
Figure C.4 – Example 3 INDEX.AUB	99
Figure C.5 – Example 4 INDEX.AUB	103
Table 1 – Table of contents data structure	12
Table 2 – Audiobook MultiAudio chunk format.....	14
Table 3 – AB_ExtStruct format.....	15
Table 4 – Example—AB_FixedChunk structure	16
Table 5 – Example—AB_FixedChunk structure revised	16
Table 6 – Example—AB_FixedChunk fields	17
Table 7 – Example—AB_VariableChunk data.....	17
Table 8 – AB_Header format.....	18
Table 9 – AB_Header fixed chunk format	18
Table 10 – AB_Header variable chunk format	19
Table 11 – AB_Book format	21
Table 12 – AB_Book fixed chunk format.....	21
Table 13 – AB_Book variable chunk format.....	24
Table 14 – AB_Level structure required field settings	26
Table 15 – Playlist content location identifier field settings.....	27
Table 16 – AB_Section format	28
Table 17 – AB_Level fixed chunk format	28
Table 18 – AB_Level variable chunk format	30
Table 19 – Schema namespace identifier.....	32
Table 20 – MPV-Music profile identifier.....	32
Table 21 – MPV audiobook profile properties.....	34
Table 22 – MPV audiobook profile metadata	35
Table 23 – Dublin Core metadata (DC-NMF).....	40

Table 24 – Metadata properties usage	42
Table 25 – CEA-2003-C binary and XML formats – Tag comparison	65
Table 26 – Examples of ID3 tags	69
Table A.1 – Sample table of contents	70
Table A.2 – Table of contents file structure	71
Table A.3 – File 1 TOC_TrackEntry structure	71
Table A.4 – File 2 TOC_TrackEntry structure	71
Table A.5 – File 3 TOC_TrackEntry structure	71
Table A.6 – File 4 TOC_TrackEntry structure	72
Table A.7 – Table of contents with file time offsets	72
Table A.8 – Example TOC_Playlist structure	72
Table A.9 – Example AB_Book structure	73
Table A.10 – AB_Level structure at Level 0 (audiobook)	77
Table A.11 – AB_Level at Level 1 (chapter)	77
Table A.12 – AB_Level at Level 1 (chapter)	78
Table A.13 – AB_Level at Level 1 (chapter)	78
Table A.14 – AB_Level at Level 2 (section)	79
Table A.15 – AB_Level at Level 2 (section)	79
Table A.16 – AB_Level at Level 2 (section)	80
Table A.17 – AB_Level at Level 2 (section)	80
Table A.18 – AB_Level at Level 3 (subsection)	81
Table A.19 – AB_Level at Level 3 (subsection)	81
Table B.1 – Hex representation and interpretation	82

INTERNATIONAL ELECTROTECHNICAL COMMISSION

DIGITAL AUDIOBOOK FILE FORMAT AND PLAYER REQUIREMENTS

FOREWORD

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
- 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
- 5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
- 6) All users should ensure that they have the latest edition of this publication.
- 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
- 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
- 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62571 has been prepared by technical area 10: Multimedia e-publishing and e-book, of IEC technical committee 100: Audio, video and multimedia systems and equipment.

This bilingual version (2011-07) replaces the English version.

The text of this standard is based on the following documents:

CDV	Report on voting
100/1543/CDV	100/1629/RVC

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC web site under "<http://webstore.iec.ch>" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

IECNORM.COM : Click to view the full PDF of IEC 62571:2011

DIGITAL AUDIOBOOK FILE FORMAT AND PLAYER REQUIREMENTS

1 Scope

This International Standard defines requirements and provides recommendations to publishers, software developers, content providers, and hardware manufacturers for the data structure, usability requirements, playback systems and delivery systems for audiobooks in digital file format. It should be noted that throughout this International Standard, the term audiobook is defined as any audio file or collection of audio files of primarily spoken word content that are played in a linear or specified order. Therefore, spoken word audio with occasional music, a narration of newspaper articles, or other similar spoken word audio is assimilated to audiobooks in this standard.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 11578:1996, *Information technology – Open Systems Interconnection – Remote Procedure Call (RPC)*

ISO 639-1, *Codes for the representation of names of languages – Part 1: Alpha-2 code*

ISO 3166-1, *Codes for the representation of names of countries and their subdivisions – Part 1: Country codes*

ISO 9660:1988, *Information processing – Volume and file structure of CD-ROM for information interchange*

3 General

This International Standard defines the audiobook file format structure for digital audiobook media. Goals in creating this International Standard include compatibility with music industry and multimedia standards as well as effectively presenting and navigating an audiobook. This International Standard is a compilation standard that straddles early binary architectures represented by earlier versions of CEA-2003¹, and newer XML architectures represented by the Optical Storage Technology Association's (OSTA) MusicPhotoVideo™² MPV). A conforming playback system and conforming audiobook content provides a wonderful and highly functional reading and listening experience. The goal of this International Standard is to create a broad, extensible standard for audiobook publishers, audiobook device manufacturers, and audiobook software developers in order to create the best listening experience for the audiobook consumer. An overview follows.

OSTA MultiAudio defines a binary file be placed in the root of the file format structure on a CD/MP3. The TOC.MAU serves as the table of contents for the associated MP3 files on a

¹ See Bibliography.

² MusicPhotoVideo is the trade name of a program supplied by the Optical Storage Technology Association. This information is given for the convenience of users of this document and does not constitute an endorsement by IEC of the program named.

compliant CD/MP3. Conforming MultiAudio players read this file to obtain the information the player needs to present the content to the listener.

OSTA MPV defines an eXtensible Markup Language (XML) schema to be placed in the root of the file format structure. The INDEX.AUB serves as the table of contents for the associated audiobook files, regardless of media, codec, or delivery type. Conforming MPV players read this file to obtain the information the player needs to present the content to the listener. Audiobooks conforming to this standard MUST utilize the audiobook-specific file extension, .AUB. In addition, audiobooks conforming to this standard may utilize the same MPV file extension specified by the MPV Music Profile, .MUM, in order to create additional compatibility for MPV-compliant players that do not recognize the audiobook extension.

Similarly, while OSTA MPV was primarily designed for the music industry and CD-ROM media, it provides "extensions" for enhancing the format. This International Standard "extends" OSTA MPV to support audiobooks and does not require a specific media or content delivery system. The audiobook extensions are defined as the MultiAudio binary "TOC.MAU" for CD/MP3 audiobooks, and the MPV XML "INDEX.AUB" for all digital audiobooks.

Therefore, throughout this International Standard, "digital audiobook file format structure" is used to describe the entire audiobook content, regardless of media or method of delivery. "Delivery system" is used to represent any system used for playback of the content, whether it is actual physical media or another type of delivery system, such as downloaded or streaming content. This International Standard defines required MPV data and audiobook-specific Extra Data requirements. The conforming player, by reading the INDEX.AUB or TOC.MAU, is presented with the data needed to effectively navigate and play the content in the manner the publisher intended. Conforming players and conforming audio content shall meet the requirements contained in Clause 11. In addition, conforming players and conforming audio content are encouraged, but not required, to conform to the recommendations of Clause 11.

The player identifies and reads the INDEX.AUB or TOC.MAU file in the root of the digital audiobook file format structure. A conforming player recognizes the audiobook identifier in the OSTA MPV Extra Data field, and processes the audiobook data as defined in Clauses 4 through 7 accordingly. OSTA MPV provides specific identifiers to tell the player the type of encoded audio file that is in use. OSTA MPV also provides an "unknown" identifier, which signals the device to determine the type of encoding used. Once an audiobook and its encoding method have been detected, the playback system can reference navigational points and informational fields and present them to the listener. Excessive delays in playback or moving to the next portion of the audiobook should be minimized; excessive delays are not acceptable according to this standard.

One or more audiobooks may reside on the system, but if there is more than one audiobook, they shall each reside in separate directories within the digital audiobook file format structure. The MPV file contains one or more "playlists" for each audiobook. The playlist identifies the correct linear reading order of the files to be played for each book, and the location of the digital audiobook files. A simple digital audiobook file format structure residing on physical media may need only one playlist; a more complex digital audiobook file format structure that includes choices of delivery, such as local and streaming, may have multiple playlists to coincide with each method of delivery.

Navigational levels are defined by the content publisher within the INDEX.AUB or TOC.MAU file, and may be as simple or as complex as desired. Extra Data fields identify the levels and "time offsets" to navigational points within files or file locations. Pointers to the "parent" and "child" locations are also defined within Extra Data fields. In this manner, an unlimited amount of levels and navigation points may be identified. Within the binary TOC.MAU file, Level 0 navigation is the top level; for example, the audiobook itself, or a collection of several audiobooks contained in separate folders. Level 1 navigation is the first level of nested pointers defined, and is required for compliance. In a very simple example, a digital audiobook file format structure could contain one audiobook comprised of one digital file. Each of the chapter locations are defined as Level 1 navigation points within the TOC.MAU.

The conforming player is able to present the user with Level 1 navigation by moving to each of the defined offsets.

More complex digital audiobook file format structures can be achieved by nesting levels within the INDEX.AUB. A digital audiobook file format structure can contain several layers or levels, with offsets to the next navigation point, parent level, and child level. As a more complex example, a digital audiobook file format structure could contain two audiobooks, each in a separate folder and comprised of ten chapters each, with each chapter encoded as a separate digital file. Navigation would include pointers to each separate audiobook, and enable the player to choose between one or the other. Navigation markers would point to the beginning of each chapter, allowing navigation from chapter to chapter within the audiobook. Nested levels would point to pages within the chapters.

NOTE 1 Books, chapters and pages are used as level name examples; however, there is no naming requirement for levels in order to be compliant with this International Standard.

The only constraint on the audio publisher is to encode MP3 files with a "fixed bit rate" because variable bit rate encoding makes it impossible for low-powered players to calculate the data offset from the time offset.

NOTE 2 Audiobook content providers are strongly encouraged to name the audio files using an ASCII collated sequence. This means that the unaware player would still play the audio files in order and the navigation from file to file would be whatever design the audio publisher chose for structuring the files, that is one chapter per file would yield chapter navigation; one section per file would yield section navigation. The correct linear reading order would then most likely be preserved.

There is audiobook-specific content, such as title, author, and narrator, as well as time information that allows players to calculate elapsed time and time remaining. Extra Data fields associated with each file provide the player with this information so that it can be presented to the listener. Additional enhancements are encouraged, but not required with Level Class Identifiers and the separate Bookmark.AUB file for rewriteable media. The Level Class Identifiers will allow a playback device to handle distinct audiobook levels in customized ways, depending on the type of content, or sub-grouping, of those levels. The Bookmark.AUB file contains provisions for both an autoresume point, as well as user-defined bookmarking capabilities, and is encouraged for transfer from one device to another of such. Playback device designers may also consider incorporating multi-user enhancements with bookmarking capabilities.

For a player that reads binary files, Clauses 4 to 8 detail the MultiAudio audiobook extensions, headers, playlists and levels in the MultiAudio format. For a player that is capable of reading XML, Clause 10 discusses in detail the MPV schema for audiobooks in the MPV format. For players that read both binary files and XML schema, the XML should be used.

4 MultiAudio audiobook extensions

4.1 Table of contents

This clause defines audiobook extended data structures based on OSTA MultiAudio.

The data structure for the table of contents shall be as defined in Table 1.

Table 1 – Table of contents data structure

Clause / Subclause	Description
4.2	Description
Table 2	Audiobook MultiAudio Chunk
4.2.2	Tag.Identifier
4.2.3	Tag.OrdinalNumber
4.2.4	Tag.Reserved
4.2.5	Tag.Length
4.2.6	Tag.Data
5	Audiobook Extendable Structure (AB_ExtStruct)
5.2	Tag
5.3	Offset to Extra Data
5.4	Fixed Length Chunk (AB_FixedChunk)
5.4.2	Example
5.5	Variable Length Chunk (AB_VariableChunk)
5.5.2	Example
5.6	Extra Data
6	Audiobook Header (AB_Header)
6.2	Tag
6.3	AB_Header Fixed Chunk
6.3.2	Tag
6.3.3	Audiobook Spec Major Version
6.3.4	Audiobook Spec Minor Version
6.3.5	Number of Audiobooks
6.3.6	Offset to Playlist Indexes
6.4	AB_Header Variable Chunk
6.4.2	Tag
6.4.3	Playlist Indexes
6.5	Extra Data
7	Audiobook (AB_Book)
7.2	Tag
7.3	AB_Book Fixed Chunk
7.3.2	Tag
7.3.3	Text Format
7.3.4	Flags
7.3.5	TOC_Playlist Index
7.3.6	Offset to Book Title
7.3.7	Offset to Author
7.3.8	Offset to Narrator
7.3.9	Offset to Copyright Statement
7.3.10	Offset to Audiobook UUIDs
7.3.11	Offset to Book ISBN Identifier

Table 1 (Continued)

Clause Subclause	Description
7.3.12	Offset to Book ISSN Identifier
7.3.13	Offset to Book DOI Identifier
7.3.14	Offset to Book Index Number
7.3.15	Offset to Book Shelf Number
7.3.16	Offset to Book Part Number
7.3.17	Offset to Other Identifier
7.3.18	Offset to First Audiobook Level
7.3.19	Offset to Playlist Content Location Identifier
7.4	AB_Book Variable Chunk
7.4.2	Tag
7.4.3	Book Title
7.4.4	Author
7.4.5	Narrator
7.4.6	Copyright Statement
7.4.7	Book UUID
7.4.8	Book ISBN Identifier
7.4.9	Book ISSN Identifier
7.4.10	Book DOI Identifier
7.4.11	Book Index Number
7.4.12	Book Shelf Number
7.4.13	Book Part Number
7.4.14	Other Identifier
7.4.15	First Audiobook Level
7.4.16	Playlist Content Location Identifier
7.5	Extra Data
8	Audiobook Level (AB_Level)
8.2	Tag
8.3	AB_Level Fixed Chunk
8.3.2	Tag
8.3.3	Text Format
8.3.4	Flags
8.3.5	Number of Sub-Levels
8.3.6	Level Number
8.3.7	First TOC_TrackEntry Index
8.3.8	Offset to Sub-Level Chunk
8.3.9	Offset to Level Name
8.3.10	Offset to Level Description
8.3.11	Book Time Offset
8.3.12	Time Start Offset
8.3.13	Time Duration
8.3.14	Offset to Parent AB_Level
8.3.15	Offset to Previous AB_Level
8.3.16	Offset to Next AB_Level
8.4	AB_Level Variable Chunk
8.4.2	Tag
8.4.3	Level Name
8.4.4	Level Description
8.4.5	Sub-Level Chunk
8.5	Extra Data

4.2 Description

4.2.1 General

All structures defined in CEA-2003-B were extensions of OSTA MultiAudio. These structures are retained in this International Standard. Each structure resides in the Extra Data area of the MultiAudio structures so that it does not interfere with the original intent of OSTA MultiAudio, but enhances OSTA MultiAudio with additional metadata that applies to a collection of audiobooks. All audiobook structures are contained within a MultiAudio Chunk and use the Extendable Structure format described in Clause 5.

All audiobook structures are contained with a MultiAudio Chunk structure. The Chunk shall have the values indicated in Table 2.

Table 2 – Audiobook MultiAudio chunk format

#	Structure	RBP	Length	Name	Contents
1	Tag	0	4	Identifier	9FFF0000h
2		4	2	Ordinal Number	Not Defined
3		6	2	Reserved	0
4		8	4	Length	UInt32
5	Data	12	Variable	audiobook Structure	AB_ExtStruct

4.2.2 Tag.Identifier

The identifier value should uniquely identify the MultiAudio Chunk as being an audiobook structure. The value 9FFF0000h shall designate this is an audiobook structure. A parser would then look for a Chunk with this identifier value in a MultiAudio structures Extra Data area to determine if audiobook structure is defined.

Since OSTA MultiAudio defines the range [80000000h – FFFFFFFFh] for public use, this means that other specifications may also use the value 9FFF0000h for their Chunk identifier. To ensure that the Chunk contains an audiobook structure, a parser shall then check the first 4 bytes of the Chunk's data area for a valid audiobook structure identifier. The first 4 bytes of the data area shall contain an audiobook structure Tag.Identifier field, which shall be defined by each audiobook structure.

4.2.3 Tag.OrdinalNumber

The ordinal number of the chunk shall be the index of the chunk within the array of chunk structures that make up a MultiAudio structure Extra Data area.

4.2.4 Tag.Reserved

The tag.reserved field shall be reserved for future OSTA use. This field is not defined at present, and shall not contain data.

4.2.5 Tag.Length

The Tag.Length field shall contain the length of the Chunk structure within the length of the Chunk.Tag and Chunk.Data area.

4.2.6 Data

The data area of the Chunk structure shall contain a valid audiobook structure, which shall follow the AB_ExtStruct format.

5 Audiobook extendable structure (AB_ExtStruct)

5.1 General

The audiobook extendable structure defines a binary structure that allows easy extensibility. This allows new fields to be added to the structure, while remaining backwards compatible with previous versions of the structure. All audiobook structures shall conform to the Extendable Structure format contained in Table 3.

Table 3 – AB_ExtStruct format

#	RBP	Length	Name	Contents
1	0	12	Tag	tag
2	12	4	Offset to Extra Data	UInt32
3	16	L_Fc	Fixed Length Chunk	AB_FixedChunk
4	16 + L_Fc	L_Vc	Variable Length Chunk	AB_VariableChunk
5	16 + L_Fc + L_Vc	L_Ex	Extra Data	ExtraData

5.2 Tag

Tag.Identifier field shall be set to a unique identifier that identifies the audiobook structure.

Tag.OrdinalNumber shall be set to the ordinal number of the structure. By default, the ordinal number should be set to the value 0, unless the structure is part of a list of structures.

Tag.Length shall be set to the length of the Extendable Structure in bytes. Since the structure is a MultiAudio Chunk structure, this length shall be evenly divisible by 4.

5.3 Offset to extra data

To ensure that the AB_ExtStruct can be further extended, an extra data area is included. If there is extra data included in the structure, the offset to the extra data area should contain the offset in bytes from the start of the AB_ExtStruct to the start of the Extra Data Chunk. If there is no Extra Data Chunk, this field should be set to the value 0.

5.4 Fixed length chunk (AB_FixedChunk)

5.4.1 General

The fixed length chunk is defined as a chunk structure, and provides for the extensibility of the **AB_ExtStruct**. The following values should be set:

AB_FixedChunk.Tag.Identifier shall be set to the hex value 9FFF0001h to designate that it is an AB_FixedChunk.

AB_FixedChunk.Tag.OrdinalNumber shall be set to the value 0.

AB_FixedChunk.Tag.Length shall be set to the length of the AB_FixedChunk in bytes. The length of this structure is the key to the extensibility of the structure. As new fields are added to the structure, this length grows. Software or devices that interpret this structure should first check this length to identify the version of the structure. To remain backwards compatible, fields defined in this structure should never be removed or redefined. All new fields should be added to the end of the structure.

Since the structure is defined as a MultiAudio Chunk, the length shall be evenly divisible by 4.

5.4.2 Example

The following is an example of how the structure is extended and interpreted by parsers.

An example of the AB_FixedChunk structure is defined in Table 4.

Table 4 – Example—AB_FixedChunk structure

#	RBP	Length	Name	Contents
1	0	12	AB_FixedChunk Tag	Tag
2	12	4	Field1	Uint32
3	16	2	Field2	Uint16
4	18	2	Field3	Uint16

The AB_FixedChunk.Tag.Length field of this structure would be set to the value 20, which designates a length of 20 bytes. All devices and software that can parse this structure would understand the values in Fields1-3, and that the total length of the structure is 20 bytes.

At a later date, a new field is added to the structure. The structure then is described in Table 5.

Table 5 – Example—AB_FixedChunk structure revised

#	RBP	Length	Name	Contents
1	0	12	AB_FixedChunk Tag	Tag
2	12	4	Field1	Uint32
3	16	2	Field2	Uint16
4	18	2	Field3	Uint16
5	20	4	Field4	Uint32

The previous software or device that was designed to handle Fields1-3 would ignore the new Field4. New software or devices that are designed to read the new field would still be able to interpret the previous three fields. If the newer software or device were to parse the previous, older structure, it would first check the length of the FixedChunk.Tag.Length field and realize that this structure does not define any fields greater than Field3, since the length would be only 20 bytes. Therefore, the newer software or device should gracefully handle this situation and initialize Field4 with a default value, or ignore it completely.

5.5 Variable length chunk (AB_VariableChunk)

5.5.1 General

The variable length chunk (AB_VariableChunk) structure is designed to work with the fixed length chunk (AB_FixedChunk) to handle field values that are of a variable length, such as a string. The AB_FixedChunk would define what values are stored in the AB_VariableChunk by providing offsets to the values in the AB_VariableChunk.

The AB_VariableChunk is defined as a MultiAudio Chunk structure. Therefore, the length of the structure shall be evenly divisible by 4 to ensure that it lies on a 4-byte aligned memory boundary. To ensure this alignment property, all fields in the AB_VariableChunk shall have a length that is evenly divisible by 4. Therefore, a field shall be padded with bytes to ensure that its length is evenly divisible by 4. All padding bytes shall be set to the value 0. To calculate the length of the padding, use the following equation:

$$\text{Number of padding bytes} = \text{int}((\text{TempLen} + 3)/4) * 4 - \text{TempLen}$$

where

TempLen is the size in bytes of the field before padding.

For example, if the AB_VariableChunk contains two strings (String1 = "ABC\0") and (String2 = "ABCDEF\0"), where the character '\0' designates a termination character, the first string (String1) would not have to be padded, since its length is 4 bytes. The second string (String2) would have to be padded with a single additional byte to change its length from 7 bytes to 8 bytes.

The AB_VariableChunk shall have the following values set.

- AB_VariableChunk.Tag.Identifier should be set to the hex value 9FFF0002h to designate that this is a variable length chunk.
- AB_VariableChunk.Tag.OrdinalNumber should be set to the value 1.
- AB_VariableChunk.Tag.Length should be set to the length of this structure including the tag.

The following is example of how the fixed length chunk and the variable length chunk work together to provide an extendable structure.

5.5.2 Example

In this example, the AB_FixedChunk defines the fields in Table 6.

Table 6 – Example—AB_FixedChunk fields

#	RBP	Length	Name	Contents
1	0	12	AB_FixedChunk Tag	tag
2	12	2	String1 offset	UInt32
3	16	2	Variable Field1 offset	UInt16
4	18	2	Variable Field1 Length (L_F1)	UInt16

In this example, the AB_VariableChunk would then contain the data indicated in Table 7.

Table 7 – Example—AB_VariableChunk data

#	RBP	Length	Name	Contents
1	0	12	AB_VariableChunk Tag	tag
2	12	L_Str1	String1 (NULL terminated)	byte
3	12 + L_Str1	L_F1	Variable Field1	byte

The AB_VariableChunk contains two fields, String1 and Variable Field1. Since String1 is a NULL terminated string, there is no need to provide a length in bytes of the string, since a parser would recognize the NULL character as being the end of the data. The Variable Field1 contains an arbitrary value of length L_F1 in bytes. Since this variable does not have the luxury of the NULL termination, the length of the field is included in the AB_FixedChunk.

The AB_FixedChunk would contain offsets from the start of the AB_FixedChunk to the data inside the AB_VariableChunk. If String1 were not included in the AB_VariableChunk, the offset in the AB_FixedChunk would be set to 0. Therefore, the AB_VariableChunk does not define how the data is ordered, since it can be placed in an arbitrary order. This is defined in

the AB_FixedChunk. Therefore, the AB_FixedChunk is similar to the index of a book, and the AB_VariableChunk contains the actual data, which would be the pages and chapters of the book.

5.6 Extra data

Extra data shall be defined as a Chunk containing an array of Chunk based structures. This area is used to further extend the structure.

6 Audiobook header (AB_Header)

6.1 General

The audiobook header is located in the Extra Data field of the OSTA MultiAudio TOC_Header structure. The AB_Header is contained within a Chunk structure, as described in 4.2. The ordinal number of the Chunk shall not be fixed. The AB_Header structure is defined as an AB_ExtStructure, as defined in 4.2. The AB_Header can be used to determine whether a TOC file contains any audiobooks, as well as where they are located. The AB_Header format shall be as indicated in Table 8.

Table 8 – AB_Header format

#	RBP	Length	Name	Contents
1	0	12	Tag	Tag
2	12	4	Offset to Extra Data	UInt32
3	16	L_Fc	AB_Header Fixed Chunk	AB_FixedChunk
4	16 + L_Fc	L_Vc	AB_Header Variable Chunk	AB_VariableChunk
5	16 + L_Fc + L_Vc	L_Ex	Extra Data	ExtraData

6.2 Tag

The Tag.Identifier shall be set to the hex value 44484241h (ABHD) to designate an AB_Header structure.

The Tag.OrdinalNumber shall be set to the value 0.

The Tag.Length shall be set to the length of the AB_Header structure, which includes the length of the Tag.

6.3 AB_Header fixed chunk

6.3.1 General

The AB_Header Fixed Chunk shall contain the format defined in Table 9.

Table 9 – AB_Header fixed chunk format

#	RBP	Length	Name	Contents
1	0	12	Tag	tag
2	12	2	Audiobook Spec Major Version	UInt16
3	14	2	Audiobook Spec Minor Version	UInt16
4	16	2	Number of Audiobooks (N_Bk)	UInt16
5	18	2	Offset to Playlist Indexes	UInt16

6.3.2 Tag

The Tag.Identifier sub-field shall be set to the hex value 9FFF0001h to designate that it is an AB_FixedChunk.

The Tag.OrdinalNumber sub-field shall be set to the value 0.

The Tag.Length shall be set to the length of the AB_Header Fixed Chunk structure. This International Standard specifies this length as 20 bytes.

6.3.3 Audiobook specification major version

This field shall contain the major version number of the audiobook specification. If the major version number changes, the new version of the specification is not required to be backward compatible with the previous versions of the specification.

For this International Standard, this field shall be set to the hex value 0001h, which designates that this is version 1.X and has a structure compliant with this International Standard.

NOTE Since this International Standard is backwards compatible with CEA-2003, CEA-2003-A, CEA-2003-B and ANSI/CEA-2003-C, version 1.0 is used for all versions of the standard.

6.3.4 Audiobook specification minor version

This field shall contain the minor version number of the audiobook specification. If the minor version number changes, the newer version of the specification is required to be backwards compatible with the previous versions of the specification.

For this International Standard, this field shall be set to the hex value 0001h, which designates that this is version X.1.

6.3.5 Number of audiobooks (N_Bk)

This field specifies the number of audiobooks that can be found in the MultiAudio TOC file.

6.3.6 Offset to playlist indexes

The offset to playlist indexes shall be defined as offset from the start of the AB_Header structure into the AB_Header Variable Chunk that contains an ordered list of MultiAudio TOC_Playlist indexes, see 6.3.

6.4 AB_Header variable chunk

6.4.1 General

The AB_Header Variable Chunk shall contain the fields defined in Table 10.

Table 10 – AB_Header variable chunk format

#	RBP	Length	Name	Contents
1	0	12	Tag	Tag
2	12	N_Bk * 2	Playlist Indexes	Uint16

6.4.2 Tag

Tag.Identifier sub-field should be set to the hex value 9FFF0002h to designate that this is an AB_VariableChunk.

The Tag.OrdinalNumber sub-field shall be set to the value 1.

The Tag.Length shall be set to the length of the AB_Header Variable Chunk structure in bytes.

6.4.3 Playlist indexes

The playlist indexes shall contain a list of Uint16 values that represent the indexes of the MultiAudio TOC_Playlist structures, which contain an AB_Book structure, defined in their Extra Data field. Each audiobook on the media shall have an associated TOC_Playlist structure, with a single AB_Book structure found in the Extra Data field. See Clause 7 for more information.

TOC_Playlist indexes are used instead of byte offsets to the AB_Book structures to ensure that TOC can change without breaking the audiobook structures. Since parsers that are not audiobook aware may edit the contents of the TOC, such as adding new TOC_TrackEntry structures, it may cause the byte offsets to become invalid. Therefore, it is up to the audiobook parser to map the associated indexes to the appropriate byte offsets in the file.

Since the Playlist Indexes field may have a length that it is not evenly divisible by 4, it may be necessary to include padding. See 5.5 for more information about padding.

6.5 AB_Header extra data

AB_Header extra data shall be defined as a Chunk containing an array of Chunk based structures. This area is used to further extend the AB_Header structure.

7 Audiobook book (AB_Book)

7.1 General

An audiobook is a collection of audio files that represent logical units of primarily spoken word audio. The audiobook may be represented by a single audio file, or by multiple files that constitute smaller units of the entire book. To play an audiobook, these units are played in order from first to last. Therefore, an audiobook can be represented as an ordered playlist of audio files.

Since a book is a playlist of files, the OSTA MultiAudio TOC_Playlist structure is used to represent an audiobook. Any MultiAudio compliant player could then play an audiobook by simply recognizing it as a TOC_Playlist and playing it. This allows players that do not recognize this International Standard's data structures to still play an audiobook with minimal navigation ability. To extend the navigation, as well as provide additional metadata, the AB_Book structure was designed as shown in Table 11.

Every audiobook on the media shall be represented in the MultiAudio TOC as a separate TOC_Playlist. In the TOC_Playlist structure for each audiobook, there shall be one AB_Book structure located in the Extra Data area.

The AB_Book structure is defined as an AB_ExtStructure that can be found in a Chunk structure (as defined in 4.2 within the Extra Data field of a MultiAudio TOC_Playlist). The ordinal number of this Chunk is not defined.

Table 11 – AB_Book format

#	RBP	Length	Name	Contents
1	0	12	Tag	tag
2	12	4	Offset to Extra Data	UInt32
3	16	L_Fc	AB_Book Fixed Chunk	AB_FixedChunk
4	16 + L_Fc	L_Vc	AB_Book Variable Chunk	AB_VariableChunk
5	16 + L_Fc + L_Vc	L_Ex	Extra Data	ExtraData

7.2 Tag

The Tag.Identifier shall be set to the hex value 4B424241h (ABBK) to designate an AB_Book structure.

The Tag.OrdinalNumber shall be set to the value 0.

The Tag.Length shall be set to the length of the AB_Book structure, which includes the length of the Tag.

7.3 AB_Book fixed chunk

7.3.1 General

The AB_Book Fixed Chunk shall contain the format indicated in Table 12.

Table 12 – AB_Book fixed chunk format

#	RBP	Length	Name	Contents
1	0	12	Tag	tag
2	12	2	Text Format	UInt16
3	14	2	Flags	UInt16
4	16	2	TOC_Playlist Index	UInt16
5	18	2	Offset to Book Title	UInt16
6	20	2	Offset to Author	UInt16
7	22	2	Offset to Narrator	UInt16
8	24	2	Offset to Copyright Statement	UInt16
9	26	2	Offset to Audiobook UUID	UInt16
10	28	2	Offset to Book ISBN Identifier	UInt16
11	30	2	Offset to Book ISSN Identifier	UInt16
12	32	2	Offset to Book DOI Identifier	UInt16
13	34	2	Offset to Book Index Number	UInt16
14	36	2	Offset to Book Shelf Number	UInt16
15	38	2	Offset to Book Part Number	UInt16
16	40	2	Offset to Other Identifier	UInt16
17	42	2	Offset to First Audiobook Level	UInt16
18	44	4	Offset to Playlist Content Location Identifier	UInt32

7.3.2 Tag

The Tag.Identifier sub-field shall be set to the hex value 9FFF0001h to designate that it is an AB_FixedChunk.

The Tag.OrdinalNumber sub-field shall be set to the value 0.

Tag.Length shall be set to the length of the AB_Book Fixed Chunk structure in bytes. For this International Standard, the length is 48 bytes.

7.3.3 Text format

If this field is set to 0, then all text fields shall be interpreted as ASCII strings.

If this field is set to 1, then all text fields shall be interpreted as UNICODE strings.

7.3.4 Flags

This field shall be set to the value 0 for this International Standard.

7.3.5 TOC_Playlist index

This field shall contain the index of the TOC_Playlist in the OSTA MultiAudio TOC which represents the audiobook and which contains the AB_Book structure.

7.3.6 Offset to book title

The offset to book title shall be defined as Offset from the start of the AB_Book structure to the title of the book within the AB_Book Variable Chunk field. Since the Book Title is a required field, this offset shall always be set, see 7.4.3.

7.3.7 Offset to author

The offset to author shall be defined as offset from the start of the AB_Book structure to the author of the book within the AB_Book Variable Chunk field. Since the Author is a required field, this offset shall always be set, see 7.4.4.

7.3.8 Offset to narrator

The offset to narrator shall be defined as the offset from the start of the AB_Book structure to the narrator of the book within the AB_Book Variable Chunk field. Since the Narrator is a required field, this offset shall always be set, see 7.4.5.

7.3.9 Offset to copyright statement

The offset to copyright statement shall be defined as offset from the start of the AB_Book structure to the copyright statement of the book within the AB_Book Variable Chunk field. If the book has no copyright statement, this field should be set to the value 0, see 7.4.6.

7.3.10 Offset to audiobook UUID

The offset to audiobook UUID (universally unique identifier) shall be defined as the offset from the start of the AB_Book structure to the audiobooks unique identifier. If the audiobook does not have a unique identifier, this field shall be set to the value 0, see 7.4.7.

7.3.11 Offset to book ISBN identifier

The offset to book ISBN (International Standard Book Number) identifier shall be defined as the offset from the start of the AB_Book structure to the ISBN of the book within the AB_Book Variable Chunk field. If the book has no ISBN, this field should be set to the value 0, see 7.4.8.

7.3.12 Offset to book ISSN identifier

The offset to book ISSN (International Standard Serial Number) identifier shall be defined as the offset from the start of the AB_Book structure to the ISSN identifier string of the book within the AB_Book Variable Chunk field. If the book has no ISSN, this field should be set to the value 0, see 7.4.10.

7.3.13 Offset to book DOI identifier

The offset to book DOI (Digital Object Identifier) identifier shall be identified as the offset from the start of the AB_Book structure to the DOI identifier string of the book within the AB_Book Variable Chunk field. If the book has no DOI, this field should be set to the value 0, see 7.4.10.

7.3.14 Offset to book index number

The offset to book index number shall be defined as the offset from the start of the AB_Book structure to the book index number within the AB_Book Variable Chunk field. If the book has no index number, this field should be set to the value 0, see 7.4.11.

7.3.15 Offset to book shelf number

The offset to book shelf number shall be defined as offset from the start of the AB_Book structure to a book shelf number string within the AB_Book Variable Chunk field. If the book has no shelf number, this field should be set to the value 0, see 7.4.12.

7.3.16 Offset to book part number

The offset to book part number shall be defined as the offset from the start of the AB_Book structure to the book part number string within the AB_Book Variable Chunk field. If the book has no ISSN, this field should be set to the value 0, see 7.4.13.

7.3.17 Offset to other identifier

The offset to other identifier shall be defined as the offset from the start of the AB_Book structure to an additional, undefined, string identifier for the book within the AB_Book Variable Chunk field. If the book has no additional identifier, this field should be set to the value 0, see 7.4.14.

7.3.18 Offset to first audiobook level

The offset to First Audiobook Level shall be defined as the offset from the start of the AB_Book structure to the first AB_Level structure within the AB_Book Variable Chunk field. All audiobooks shall contain one AB_Level structure to define the entire audiobook, so this offset shall always be set, see 7.4.15.

7.3.19 Offset to playlist content location identifier

The offset to playlist content location identifier shall be defined as offset from the start of the AB_Book structure to the playlist content location identifier within the AB_Book Variable Chunk field. If the field is not present or null, this field should be set to the value 0, see 7.4.16.

7.4 AB_Book variable chunk

7.4.1 General

The AB_Book Variable Chunk shall contain the format indicated in Table 13.

Table 13 – AB_Book variable chunk format

#	RBP	Length	Name	Contents
1	0	12	Tag	tag
2	12	L_Bt	Book Title	byte
3	12 + L_Bt	L_Au	Author	byte
4	12 + L_Bt + L_Au	L_Na	Narrator	byte
5	12 + L_Bt + L_Au + L_Na	L-Cs	Copyright Statement	byte
6	12 + L_Bt + L_Au + L_Na + L-Cs	L_Ui	Book UUID	byte
7	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui	L_Is	Book ISBN Identifier	byte
8	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is	L_In	Book ISSN Identifier	byte
9	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In	L_Do	Book DOI Identifier	byte
10	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do	L_In	Book Index Number	byte
11	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do + L_In	L_Sn	Book Shelf Number	byte
12	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do + L_In + L_Sn	L_Pn	Book Part Number	byte
13	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do + L_In + L_Sn + L_Pn	L_Oi	Other Identifier	byte
14	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do + L_In + L_Sn + L_Pn + L_Oi	L_Sc	First Audiobook Level	byte
15	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do + L_In + L_Sn + L_Pn + L_Oi + L_Li + L_Sc	L_Li	Content Location Identifier	Uint32

7.4.2 Tag

The Tag.Identifier sub-field should be set to the hex value 9FFF0002h to designate that this is an AB_VariableChunk.

The Tag.OrdinalNumber sub-field shall be set to the value 1.

The Tag.Length shall be set to the length of the AB_Book Variable Chunk structure in bytes.

7.4.3 Book title

Every audiobook shall contain a book title field. This field shall contain the Book Title according to the character set defined in the Text Format field located in the AB_Book Fixed Chunk. If there is no book title, the book title field shall be filled with a string termination character. It may be necessary to pad this field. See 5.5 for more information on padding.

If the AB_Book structure is located in a MultiAudio TOC_Playlist structure, the Book Title shall be the same string as Playlist Name field in the TOC_Playlist.

If the AB_Book structure is located in a MultiAudio TOC_TrackEntry structure, the Book Title shall be the same string as the Album Name field in the TOC_TrackEntry.

7.4.4 Author

Every audiobook shall contain an author field. This field shall contain the Author according to the character set defined in the Text Format field located in the AB_Book Fixed Chunk. If there is no author, the Author field shall be filled with a string termination character. It may be necessary to pad this field. See 5.5 for more information on padding.

If the AB_Book structure is located in a MultiAudio TOC_TrackEntry structure, the Author shall be the same string as the Songwriter Name field in the TOC_TrackEntry.

7.4.5 Narrator

Every audiobook shall contain a narrator field. This field shall contain the Narrator according to the character set defined in the Text Format field located in the AB_Book Fixed Chunk. If there is no narrator, the Narrator field shall be filled with a string termination character. It may be necessary to pad this field. See 5.5 for more information on padding.

If the AB_Book structure is located in a MultiAudio TOC_TrackEntry structure, the Narrator shall be the same string as the Performer Name field in the TOC_TrackEntry.

7.4.6 Copyright Statement

This field shall contain the copyright statement according to the character set defined by the Text Format field in the AB_Book Fixed Chunk structure. If there is no copyright statement, this field shall not be present. It may be necessary to pad this field. See 5.5 for more information on padding.

7.4.7 Book UUID

A UUID format was defined in the Open Software Foundation's Distributed Computer Environment RPC standard also available as ISO/IEC 11578, which defines UUIDs in an appendix.

This field contains a UUID in a 36-byte string format according to that specification. If this field is present, the UUID is not NULL (00h) terminated. Since the field shall be exactly 36 bytes, there is no need for padding.

An audiobook need not include this field. This field is recommended, although not required, so devices, as well as publishers, can uniquely identify a digital audiobook.

7.4.8 Book ISBN Identifier

This field shall contain the ISBN of the audiobook formatted as a string of d-characters, as defined by the MultiAudio specification. If the book has no ISBN, this field shall not be present. It may be necessary to pad this field. See 5.5 for more information on padding.

7.4.9 Book ISSN Identifier

This field shall contain the ISSN of the audiobook formatted as a string of d-characters, as defined by the MultiAudio specification. If the book has no ISSN, this field shall not be present. It may be necessary to pad this field. See 5.5 for more information on padding.

7.4.10 Book DOI Identifier

This field shall contain the DOI of the audiobook formatted as a string of d-characters, as defined by the MultiAudio specification. If the book has no DOI, this field shall not be present. It may be necessary to pad this field. See 5.5 for more information on padding.

7.4.11 Book index number

This field shall contain the audiobook index number formatted as a string of d-characters, as defined by the MultiAudio specification. If the book has no index number, this field shall not be present. It may be necessary to pad this field. See 5.5 for more information on padding.

7.4.12 Book shelf number

This field shall contain the audiobook shelf number formatted as a string of d-characters, as defined in OSTA MultiAudio. If the book has no shelf number, this field shall not be present. It may be necessary to pad this field. See 5.5 for more information on padding.

7.4.13 Book part number

This field shall contain the audiobook part number formatted as a string of d-characters, as defined by the MultiAudio specification. If the book has no part number, this field shall not be present. It may be necessary to pad this field. See 5.5 for more information on padding.

7.4.14 Other identifier

This field shall contain an additional audiobook identifier formatted as a string of d-characters, as defined by the MultiAudio specification. This additional identifier is not defined by the specification, and can be set to any value that helps uniquely identify an audiobook. If the book has no additional identifier, this field shall not be present. It may be necessary to pad this field. See 5.5 for more information on padding.

7.4.15 First audiobook level

Every AB_Book structure shall contain one AB_Level structure that defines the entire book. The book is then divided into smaller sub-levels of this level. The AB_Level structure that represents the entire audiobook shall have the fields set as indicated in Table 14.

Table 14 – AB_Level structure required field settings

Subclause #	AB_Level field	Value
8.3.2	Tag.OrdinalNumber	0
8.3.6	Level Number	0
8.3.14	Offset to Parent AB_Level	0
8.3.11	Book Time Offset	0
8.3.12	Time Start Offset	0
8.3.13	Time Duration	Length of audiobook in milliseconds
8.3.15	Offset to Previous AB_Level	0
8.3.16	Offset to Next AB_Level	0

See Clause 8 for more information on the AB_Level structure.

7.4.16 Playlist content location identifier

This field shall contain the playlist content location identifier. This identifier is a number indicating the location of files for this playlist (locally, online as downloadable content, online as streaming content or any combination). This number can be derived from Table 15. If the field is null or not present, value 0 (local) should be assumed.

Table 15 – Playlist content location identifier field settings

Playlist content location identifier	Value
Local	0
Undefined	1
No content	2
Download content	3
Streaming content	4
Mixed content – Local/Download	5
Mixed content – Local/Streaming	6
Mixed content – Download/Streaming	7
Mixed content – Local/Download/Streaming	8

7.5 AB_Book extra data

AB_Book extra data shall be defined as a Chunk containing an array of Chunk based structures. This area is used to further extend the AB_Book structure.

8 Audiobook level (AB_Level)

8.1 General

An AB_Level is a new structure that is not present in OSTA MultiAudio. An AB_Level represents a logical unit of an audiobook, with a start time and duration in milliseconds of the audio file or group of audio files. Each level can represent an infinite number of smaller sub-level sections. Therefore, each level may represent the entire book, a chapter, a sub-section, or even a page. Each audiobook contains a top-level parent that is defined in the AB_Book structure. This clause represents the entire audiobook. Within this top-level section, an infinite number of sub-sections are defined.

If an AB_Level represents a group of audio files, it shall contain child sub-levels that reference each audio file's TOC_TrackEntry structure in the group. An AB_Level references an audio file using the First TOC_TrackEntry Index field in the AB_Level Fixed Chunk Field. Since this field only contains one TOC_TrackEntry index, it would be impossible for a single AB_Level structure to reference multiple audio files without defining child sub-levels. For example, if an audiobook were divided into 3 separate audio files, it would be impossible to have the top-level AB_Level structure (Level Number = 0) define no child sub-levels. At the very least, this top-level structure shall have 3 sub-levels that reference each audio file's TOC_TrackEntry structure.

The AB_Level structure is defined as an AB_ExtStructure. There is no need to contain the AB_Level structure within a Chunk structure as defined in Table 2, because it is not directly contained within a MultiAudio structures Extra Data field. AB_Section format shall be as defined in Table 16.

Table 16 – AB_Section format

#	RBP	Length	Name	Contents
1	0	12	Tag	tag
2	12	4	Offset to Extra Data	UInt32
2	16	L_Fc	AB_Level Fixed Chunk	AB_FixedChunk
3	16 + L_Fc	L_Vc	AB_Level Variable Chunk	AB_VariableChunk
4	16 + L_Fc + L_Vc	L_Ex	Extra Data	ExtraData

8.2 Tag

The Tag.Identifier shall be set to the hex value 564C4241h (ABLV) to designate an AB_Level structure.

The Tag.OrdinalNumber shall be set to the value 0.

The Tag.Length shall be set to the length of the AB_Level structure in bytes, which includes the length of the Tag.

8.3 AB_Level fixed chunk

8.3.1 General

The AB_Level Fixed Chunk shall contain the format indicated in Table 17.

Table 17 – AB_Level fixed chunk format

#	RBP	Length	Name	Contents
1	0	12	Tag	tag
2	12	2	Text Format	UInt16
3	14	2	Flags	UInt16
4	16	2	Number of Sub-Levels (N_Sc)	UInt16
5	18	2	Level Number	UInt16
6	20	2	First TOC_TrackEntry Index	UInt16
7	22	2	Offset to Sub-Level Chunk	UInt16
8	24	2	Offset to Level Name	UInt16
9	26	2	Offset to Level Description	UInt16
10	28	4	Book Time Offset	UInt32
11	32	4	Time Start Offset	UInt32
12	36	4	Time Duration	UInt32
13	40	4	Offset to Parent AB_Level	UInt32
14	44	4	Offset to Previous AB_Level	UInt32
15	48	4	Offset to Next AB_Level	UInt32

8.3.2 Tag

The Tag.Identifier sub-field shall be set to the hex value 9FFF0001h to designate that it is an AB_FixedChunk.

The Tag.OrdinalNumber sub-field shall be set to the value 0.

The Tag.Length shall be set to the length of the AB_Level Fixed Chunk structure in bytes. For this International Standard, the length is 52 bytes.

8.3.3 Text format

If this field is set to 0, then all text fields shall be interpreted as ASCII strings.

If this field is set to 1, then all text fields shall be interpreted as UNICODE strings.

8.3.4 Flags

Flags shall be set to the value 0 for this International Standard.

8.3.5 Number of sub-levels

This field shall contain the number of sub-levels that this level represents. If there are no sub-levels, this field shall be set to 0.

8.3.6 Level number

This field shall contain the level number of the level. The top level shall have this value set to 0, while each sub-level having value of one greater than the parent. The AB_Level with the level 0 represents the entire audiobook, which can be found in the AB_Book structure.

8.3.7 TOC_TrackEntry index

Each AB_Level represents a piece of an audio file, or a group of audio files. Each audio file is defined by a MultiAudio TOC_TrackEntry structure. This field shall contain the index of the TOC_TrackEntry structure within the MultiAudio TOC file that is the first audio file that this AB_Level encapsulates. If the AB_Level is a piece of one file, the index shall be the index of the TOC_TrackEntry of which the level is a piece. If the AB_Level represents a group of audio files, this field shall contain the TOC_TrackEntry index of the first audio file in the group.

8.3.8 Offset to sub-level chunk

The offset to sub-level chunk shall be defined as offset in bytes from the start of the AB_Level structure to the sub-level chunk within the AB_Level Variable Chunk field. If there are no sub-levels, this field shall be set to 0.

8.3.9 Offset to level name

The offset to level name shall be defined as offset from the start of the AB_Level structure to the Level Name. This field shall not be 0, since the Level Name is a required, see 8.4.3.

8.3.10 Offset to level description

The offset to level description shall be defined as offset from the start of the AB_Level structure to the Level Description. If the level has no description, this field shall be set to the value 0, see 8.4.4.

8.3.11 Book time offset

Book time offset shall designate the time offset in milliseconds from the start of the audiobook to the start of this level.

8.3.12 Time start offset

Time start offset shall designate the time offset in milliseconds from the start of an audio file to the start of this level. If the level represents an audio file, all sub-levels shall start at a

value that is the same or greater than this value. If the level represents a group of audio files, this field shall be set to 0.

8.3.13 Time duration

Time duration shall designate the time length of this level in milliseconds. If the level represents an audio file, the duration shall not exceed the Time Start Offset of the next level or the length of the entire audio file in milliseconds. If the level represents a group of audio files, the duration shall not exceed the total time length of the audiobook in milliseconds or the Book Time Offset of the next level.

8.3.14 Offset to parent AB_Level

Offset from the start of this AB_Level structure to the parent of this level in bytes. Since the parent level is always previous to this level within a MultiAudio file, this constitutes a negative offset. The field is defined as an unsigned value (UInt32), so the parser shall jump backward the designated number of bytes, instead of forward.

If the level has no parent level, this field shall be set to the value 0. This shall only occur at the highest level (Level Number = 0), which is the first AB_Level defined in the AB_Book structure.

8.3.15 Offset to previous AB_Level

Offset in bytes from the start of the AB_Level structure to the previous level at the same Level Number. If this is the first level at the current Level Number, this field shall be set to 0. Since the previous level is always defined before the AB_Level in the MultiAudio file, this constitutes a negative offset. The parser shall therefore jump backwards the designated number of bytes in the file, instead of forward.

8.3.16 Offset to next AB_Level

Offset in bytes from the start of the AB_Level structure to the next level at the same Level Number. If this is the last level at the current Level Number, this field shall be set to the value 0.

8.4 AB_Level variable chunk

8.4.1 General

The AB_Level Variable Chunk shall contain the format indicated in Table 18.

Table 18 – AB_Level variable chunk format

#	RBP	Length	Name	Contents
1	0	12	Tag	tag
3	12	L_Sn	Level Name	byte
4	12 + L_Sn	L_Sd	Level Description	byte
6	12 + L_Sn + L_Sd	L_Sc	Sub-Level Chunk	LevelChunk

8.4.2 Tag

The Tag.Identifier sub-field should be set to the hex value 9FFF0002h to designate that this is an AB_VariableChunk.

The Tag.OrdinalNumber sub-field shall be set to the value 1.

The Tag.Length shall be set to the length of the AB_Level Variable Chunk structure in bytes.

8.4.3 Level name

This field shall contain the name of the level according to the character set defined by the Text Format field in the AB_Level Fixed Chunk structure. The level name should be a short string that accurately and briefly describes what the level represents. If the level has no name, this field shall be filled with a string termination character. It may be necessary to pad this field. See 5.5 for more information on padding.

8.4.4 Level description

This field shall contain the description of the level according to the character set defined by the Text Format field in the AB_Level Fixed Chunk structure. The level description is a string that further defines what the level represents. If the level has no description, this field shall not be present. It may be necessary to pad this field. See 5.5 for more information on padding.

8.4.5 Sub-level chunk

Sub-level chunk shall be defined as a Chunk containing an array of AB_Level structures that represent the sub-levels of the AB_Level structure. If the AB_Level has no sub-levels, this field shall not be present. The Chunk structure shall have the following fields defined.

- a) The LevelChunk.Tag.Identifier sub-field shall be set to the hex value 9FFF0003h to designate a LevelChunk.
- b) The LevelChunk.Tag.OrdinalNumber sub-field shall be set to the value 0.
- c) LevelChunk.Tag.Length shall be set to the length in bytes of all sub-levels plus the length in bytes of the tag.

8.5 AB_Level extra data

AB_Level extra data shall be defined as a Chunk containing an array of Chunk based structures. This area is used to further extend the AB_Level structure.

9 MPV audiobook profiles

9.1 General

This clause describes the XML-based version of audiobook requirements, and is based on an adaptation of the OSTA MPV Music specification.

The MPV core specification (MPV-Core) defines the essential framework of the MPV technology. The MPV music profile 1.0 (MPV-Music) extends MPV-Core to specify how to create, manage and play music playlists.

An MPV framework mechanism, the profile, defines significant new functionality. MPV-Core specification establishes requirements for MPV profile use. MPV-Core requires that an MPV file declare which profiles it implements, and declare the namespaces of the profiles. This allows a processing application to quickly determine whether a given MPV file meets its expectations for processing.

The audiobook profile makes use of the MPV music profile 1.0 because most of the MPV music profile is applicable to audiobooks, and leveraging the MPV music profile enhances playback compatibility for audiobooks. MPV-Music includes schema and practices.

The audiobook profile (MPV-Audiobook) is an extension of MPV-Music and is fully compatible with it. Files implementing MPV-Audiobook profile should be usable in MPV-Music-aware applications and devices. This means, for example, that an MPV-Music playback application

or device can read and playback CEA Audiobooks; however, a playback device that is only MPV-Music aware ignores the MPV-Audiobook profile, and the playback experience will be less full-featured than in an MPV-Audiobook-aware player.

9.2 XML schema namespace identifier

This International Standard defines a schema namespace identifier to be used to identify the XML schema for audiobooks, see Table 19.

Table 19 – Schema namespace identifier

Namespace Identifier	http://www.ce.org/ CEA-2003-DTD, 1.0
Conventional Namespace Prefix	ab:

This identifier shall be used in the namespace declaration value when defining namespace prefixes or declaring the namespace of an element.

9.3 Use of MPV-Music profile identifier

To be compatible with MPV-Music players, an audiobook MPV file shall also include the MPV music profile declaration in the Profiles section of the MPV manifest in Table 20. An example is given in Table 20.

Table 20 – MPV-Music profile identifier

MPV Music Profile Identifier	http://www.osta.org/ mpv/music/1.0/
------------------------------	---

An example of an MPV music profile identifier usage is given below.

```
<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest
  xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:mpv="http://ns.osta.org/mpv/1.0/"
  xmlns:mpvp="http://ns.osta.org/mpv/presentation/1.0/"
  xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"
  xmlns:ab="http://ns.ce.org/audiobook/2003b/"
  xmlns:nmf="http://ns.osta.org/nmf/1.0/" >
  <nmf:Metadata>
    <ManifestProperties xmlns="http://ns.osta.org/manifest/1.0/">
      <ProfileBag>
        <Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
```

```
<Profile>http://ns.osta.org/mpv/music/1.0/</Profile>

<Profile>http://ns.ce.org/audiobook/2003b/</Profile>

</ProfileBag>

</ManifestProperties>

</nmf:Metadata>

</file:Manifest>
```

9.4 Audiobook file

Audiobook file means an MPV file that complies with the MPVMusic that also declares the MPV-Audiobook profile. An audiobook file shall also contain a single `<mpvm:Album>` element that defines the audiobook and contains a set of assets described within the `<mpv:AssetList>` element which are referenced directly or indirectly by the contents of the `<mpvm:Album>`.

9.5 Audiobook profile metadata

MPV-Audiobook extends and builds upon metadata defined in MPVMusic. MPV-Audiobook adds metadata specifically about audiobooks.

MPV-Audiobook metadata and MPV-Music metadata may be used as metadata assigned to individual MPV assets. In MPV-Audiobook, each MPV audio asset corresponds to a book chapter.

The metadata may also be used with the MPV presentation playlist (“element name `<mpvp:album>`”) to describe the metadata for the entire audiobook one time, instead of duplicating it for each audiobook chapter asset. The MPV presentation playlist is used to wrap a presentation layer on top of the individual audio assets. The MPV presentation album defines the audiobook as a whole.

MPV-Audiobook shall include the new properties defined in Table 21.

These properties are specific to audiobooks. These properties may be used within a `<nmf:Metadata>` element that is present on any MPV asset (such as `mpv:Audio`) or on the MPV presentation element `<mpvp:Album>`

Table 21 – MPV audiobook profile properties

Metadata	Audiobook Profile	Discussion
Book Title	nmf:Metadata ab:Properties ab:BookTitle	Primary metadata for the title of the book being read. When used on an asset, overrides any metadata value specified on mpvp:Album.
Book Author	nmf:Metadata ab:Properties ab:BookAuthor	Primary metadata for the author of the book being read. When used on an asset, overrides any metadata value specified on mpvp:Album.
Book Co-Author	nmf:Metadata ab:Properties BookCoauthor	Primary metadata for the co-author of the book being read. When used on an asset, overrides any metadata value specified on mpvp:Album.
Book Subtitle	nmf:Metadata ab:Properties BookSubtitle	Primary metadata for the Subtitle of the book being read. When used on an asset, overrides any metadata value specified on mpvp:Album.
Book Narrator	nmf:Metadata ab:Properties ab:BookNarrator	Primary metadata for the narrator of the book being read. When used on an asset, overrides any metadata value specified on mpvp:Album.
Book Genre	nmf:Metadata ab:Properties BookGenre	Primary metadata for the genre of the book being read. When used on an asset, overrides any metadata value specified on mpvp:Album.
Book UUID	nmf:Metadata ab:Properties ab:BookUUID	Primary metadata for book UUID.
Book ISBN Identifier	nmf:Metadata ab:Properties ab:BookISBN	Primary metadata for book ISBN.
Book ISSN Identifier	nmf:Metadata ab:Properties ab:BookISSN	Primary metadata for book ISSN.
Book DOI Identifier	nmf:Metadata ab:Properties ab:BookDOI	Primary metadata for book DOI.
Book Index Number	nmf:Metadata ab:Properties ab:BookIndexNumber	Primary metadata for book Book Index Number.
Book Shelf Number	nmf:Metadata ab:Properties ab:BookShelfNumber	Primary metadata for book Book Shelf Number.
Book Part Number	nmf:Metadata ab:Properties ab:BookPartNumber	Primary metadata for book Book Part Number.
Other Identifier	nmf:Metadata ab:Properties ab:OtherIdentifier	Primary metadata for any other book identifier.
Content Location Identifier	nmf:Metadata ab:Properties ab:ContentLocationIdentifier	Primary metadata for playlist content location identifier.

9.6 MPV audiobook profile metadata

As described in MPV-Music, a variety of metadata is available to describe audiobooks. Table 22 reviews those properties and gives an interpretation for audiobooks.

Table 22 – MPV audiobook profile metadata

Metadata	Metadata elements in MPV	Discussion regarding audiobooks
Audiobook asset Chapter	Sub-elements of the mpv:Audio asset	
Pathname	mpv:LastURL	One or more pathnames that should resolve to the chapter file. Each pathname is specific to a file system, so there can be different names for each file system on a CD.
Format	nmf:Metadata dc:Properties dc:format	Use MPV-provided list of well-known MIME types or vendor-specific extended types.
Title	nmf:Metadata dc:Properties dc:title	Primary metadata for the title of this audio asset. Such as: if a chapter, the chapter title; if a page, the page number.
Album Title	nmf:Metadata mpvm:MusicProperties mpvm:AlbumTitle	Not used as primary metadata for Audiobooks. SHOULD be populated with the value of ab:BookTitle for enhanced compatibility with MPV Music players.
Book Title	nmf:Metadata ab:Properties ab:BookTitle	Primary metadata for the title of the book being read. When used on an asset, overrides any metadata value specified on mpvp:Album.
Creator	nmf:Metadata dc:Properties dc:creator	Not used as primary metadata for Audiobooks. SHOULD be populated with the value of ab:BookAuthor for enhanced compatibility with MPV Music players.
Principal artist	nmf:Metadata mpvm:MusicProperties mpvm:PrincipalArtist	Not used as primary metadata for Audiobooks. SHOULD be populated with the value of ab:BookAuthor for enhanced compatibility with MPV Music players.
Book Author	nmf:Metadata ab:Properties ab:BookAuthor	Primary metadata for the author of the book being read. When used on an asset, overrides any metadata value specified on mpvp:Album.

Table 22 (continued)

Metadata	Metadata elements in MPV	Discussion regarding audiobooks
Contributor	nmf:Metadata dc:Properties dc:contributor	Not used as primary metadata for Audiobooks. SHOULD be populated with the value of ab:BookNarrator for enhanced compatibility with MPV Music players.
Book Narrator	nmf:Metadata ab:Properties ab:Narrator	Primary metadata for the narrator of the book being read. When used on an asset, overrides any metadata value specified on mpvp:Album.
Music by	nmf:Metadata mpvm:MusicProperties mpvm:MusicBy	Composer name(s) in free form plain text.
Lyrics by	nmf:Metadata mpvm:MusicProperties mpvm:LyricsBy	Not used.
Arranged by	nmf:Metadata mpvm:MusicProperties mpvm:ArrangedBy	Not used.
Produced By	nmf:Metadata mpvm:MusicProperties ProducedBy	Producer name(s) in free form plain text.
Genre	nmf:Metadata mpvm:MusicProperties mpvm:Genre	Predefined set of genres can be extended simply by using a new name.
Date Recorded	nmf:Metadata dcterms:Properties dcterms:created	recorded date-time, for example 2002-10-03T21:07:00Z
Identifier	nmf:Metadata dc:Properties dc:identifier	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookISBN for enhanced compatibility with MPV players of all kinds.
Book UUID	nmf:Metadata ab:Properties ab:BookUUID	Primary metadata for book UUID.
Book ISBN Identifier	nmf:Metadata ab:Properties ab:BookISBN	Primary metadata for book ISBN.
Book ISSN Identifier	nmf:Metadata ab:Properties ab:BookISSN	Primary metadata for book ISSN.
Book DOI Identifier	nmf:Metadata ab:Properties ab:BookDOI	Primary metadata for book DOI.
Book Index Number	nmf:Metadata ab:Properties ab:BookIndexNumber	Primary metadata for book Book Index Number.
Book Shelf Number	nmf:Metadata ab:Properties ab:BookShelfNumber	Primary metadata for book Book Shelf Number.
Book Part Number	nmf:Metadata ab:Properties ab:BookPartNumber	Primary metadata for book Book Part Number.
Other Identifier	nmf:Metadata ab:Properties ab:OtherIdentifier	Primary metadata for any other book identifier.
Content Location Identifier	nmf:Metadata ab:Properties ab:LocationIdentifier	Primary metadata for playlist content location identifier.

Table 22 (continued)

Metadata	Metadata elements in MPV	Discussion regarding audiobooks
Description	nmf:Metadata dc:Properties dc:description	Commentary, message, description.
Keywords	nmf:Metadata dc:Properties dc:subject	Semicolon separated list of plain text keywords or phrases for search. Semi-colon can be used by escaping with backslash, for example \;
Copyright Statement	nmf:Metadata dc:Properties dc:rights	Free-form plain text copyright statement of rights.
Language	nmf:Metadata dc:Properties dc:language	"<language> <territory>" where <language> is from [ISO 639-1] and <territory> is [ISO 3166-1] Example: "en_US"
Rendition using another encoder	mpv:Rendition mpv:renditionType="alt" mpv:AudioRef	Refers to an alternate encoding of the asset.
Rendition with another bit rate	mpv:Rendition mpv:renditionType="subsampling" mpv:AudioRef	Refers to another encoding of the asset at a different bit rate. Use mpvm:EncodedBitrate or interrogate the file for more information.
Encoded Bit rate	nmf:Metadata mpvm:MusicProperties mpvm:EncodedBitrate	Maximum bit rate in bps of the encoded asset. For variable bit rate-encoded assets, this is the maximum value; for constant bit rate encoded assets, it is the average value.
Asset artwork	mpv:Related mpv:relationship="urn:osta-org:mpv:music:artwork" mpv:StillRef	Not used.
Music video	mpv:Related mpv:relationship="urn:osta-org:mpv:music:video" mpv:VideoRef	Not used.
Online Info	nmf:Metadata mpvm:MusicProperties mpvm:MoreInfoURL	URL to follow to get more info about the audiobook.
Playing Time	nmf:Metadata mpvm:MusicProperties mpvm:PlayingTime	In seconds, duration. Decimal values are allowed, such as 134,58 s. Means 134 and 58/100 s.
Track Number	nmf:Metadata mpvm:MusicProperties mpvm:TrackNumber	Sequence order of the chapter in the original book. Starts with 1, not 0.
Number Tracks	nmf:Metadata mpvm:MusicProperties mpvm:NumTracks	Number of chapters in the original compilation. Starts with 1, not 0.
Set Number in Collection	nmf:Metadata mpvm:MusicProperties mpvm:SetNumber	Not used.
Number Sets in Collection	nmf:Metadata mpvm:MusicProperties mpvm:NumSets	Not used.

Table 22 (continued)

Metadata	Metadata elements in MPV	Discussion regarding audiobooks
PlayCount	nmf:Metadata mpvm:MusicProperties mpvm:PlayCount	Not used.
Lyrics	nmf:Metadata mpvm:MusicProperties mpvm:Lyrics	Not used.
Mood	nmf:Metadata mpvm:MusicProperties mpvm:Mood	Not used.
Tempo	nmf:Metadata mpvm:MusicProperties mpvm:Tempo	Not used.
Situation	nmf:Metadata mpvm:MusicProperties mpvm:Situation	Not used.
Medium	nmf:Metadata dcterms:Properties dcterms:medium	Not used.
Original file and filename	mpv:Related mpv:relationship="copyOf" mpv:AudioRef	Not used.
Extra Data	nmf:Metadata and mpv:Metadata	Any metadata that an application wishes to create can go into these containers.
Key-Value Pairs	nmf:Metadata mpvm:Properties mpvm:KeyValue mpvmkv:Key nmf:Metadata mpvm:Properties mpvm:KeyValue mpvmkv:Value	Not used.
Album ("Audiobook")	mpvp:Album	
Number of Entries	implicit – number of entries in the foreground	
Title	nmf:Metadata dc:Properties dc:title	Not used as primary metadata for Audiobooks. SHOULD be populated with the value of ab:BookTitle for enhanced compatibility with MPV players.
Book Title	nmf:Metadata ab:Properties ab:BookTitle	Title of the overall book.
Contributor	nmf:Metadata dc:Properties dc:contributor	Not used as primary metadata for Audiobooks. SHOULD be populated with the value of ab:BookNarrator for enhanced compatibility with MPV players.
Narrator	nmf:Metadata ab:Properties ab:BookNarrator	Primary audiobook metadata.
Creator	nmf:Metadata dc:Properties dc:creator	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookAuthor for enhanced compatibility with MPV players.
Author	nmf:Metadata ab:Properties ab:BookAuthor	Primary audiobook metadata.

Table 22 (continued)

Album Audiobook	mpvp:Album	
Principal Artist	nmf:Metadata mpvm:MusicProperties mpvm:PrincipalArtist	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookAuthor for enhanced compatibility with MPV Music players.
Description	nmf:Metadata dc:Properties dc:description	Free-form commentary, message, description in plain text.
Copyright statement	nmf:Metadata dc:Properties dc:rights	Free-form plain text of the Copyright Statement of rights.
Audiobook Content Items	mpvp:Album mpvp:Foreground	This content is played as the principal content of the audiobook playlist.
Audiobook accompaniment	mpvp:Album mpvp:Background	This content is played as background to the audiobook content in the foreground. For example, it could specify images or background music.
Audiobook Artwork	mpvp:Album mpv:Related mpv:relationship="urn:osta-org:mpv:music:artwork" mpv:StillRef	Artwork for the audiobook.
Identifier	nmf:Metadata dc:Properties dc:identifier	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookISBN for enhanced compatibility with MPV Music players.
Book UUID	nmf:Metadata ab:Properties ab:BookUUID	Primary metadata for book UUID.
Book ISBN Identifier	nmf:Metadata ab:Properties ab:BookISBN	Primary metadata for book ISBN.
Book ISSN Identifier	nmf:Metadata ab:Properties ab:BookISSN	Primary metadata for book ISSN.
Book DOI Identifier	nmf:Metadata ab:Properties ab:BookDOI	Primary metadata for book DOI.
Book Index Number	nmf:Metadata ab:Properties ab:BookIndexNumber	Primary metadata for book Book Index Number.
Book Shelf Number	nmf:Metadata ab:Properties ab:BookShelfNumber	Primary metadata for book Book Shelf Number.
Book Part Number	nmf:Metadata ab:Properties ab:BookPartNumber	Primary metadata for book Book Part Number.
Other Identifier	nmf:Metadata ab:Properties ab:OtherIdentifier	Primary metadata for any other book identifier.
Content Location Identifier	nmf:Metadata ab:Properties ab:LocationIdentifier	Primary metadata for play-list content location identifier.
Key-Value Pairs	nmf:Metadata mpvm:Properties mpvm:KeyValue mpvmkv:Key nmf:Metadata mpvm:Properties mpvm:KeyValue mpvmkv:Value	Not used.
Extra Data	mpv:Metadata and nmf:Metadata anywhere	

9.7 Dublin core metadata usage

To promote interoperability, MPV-Core makes use of the Dublin Core metadata (DC-NMF) to represent essential metadata across all types of assets. The dc:title element is used to specify the title of an audiobook asset, as the title of an image or video asset.

In 9.6, the Dublin Core metadata elements were mixed into the overall set of audiobook meta-data properties and assets. To clarify usage of DC-NMF, Table 23 includes only the Dublin Core elements.

Table 23 – Dublin Core metadata (DC-NMF)

Audio asset Chapter, Section	mpv:Audio subelements	
Creator ("Author")	nmf:Metadata dc:Properties dc:creator	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookAuthor for enhanced compatibility with MPV players.
Contributor ("Narrator")	nmf:Metadata dc:Properties dc:contributor	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookNarrator for enhanced compatibility with MPV players.
Description	nmf:Metadata dc:Properties dc:description	Commentary, message, description in plain text.
Format	nmf:Metadata dc:Properties dc:format	Use MPV-provided list of well-known MIME types.
Identifier ("ISBN")	nmf:Metadata dc:Properties dc:identifier	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookISBN for enhanced compatibility with MPV Music players.
Keywords	nmf:Metadata dc:Properties dc:subject	Semicolon separated list of plain text keywords or phrases for search. Semi-colon can be used by escaping with backslash, for example \;
Language	nmf:Metadata dc:Properties dc:language	"<language>_<territory>" where <language> is from [ISO 639-1] and <territory> is [ISO 3166-1] Example: "en_US"
Publisher	nmf:Metadata dc:Properties dc:publisher	Free-form plain text.
Rights ("Copyright Statement")	nmf:Metadata dc:Properties dc:rights	Free-form plain text copyright statement.
Title	nmf:Metadata dc:Properties dc:title	Free-form plain text of the title of the chapter.
Created Date	nmf:Metadata dcterms:Properties dcterms:created	Date-time audiobook performed or recorded.
Medium	nmf:Metadata dcterms:Properties dcterms:medium	Original audiobook release medium conformant to [ID3v240]

Table 23 (continued)

Album (audiobook) of spoken word	mpvp:Album subelements	Comments
Creator ("Author")	nmf:Metadata dc:Properties dc:creator	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookAuthor for enhanced compatibility with MPV players.
Contributor ("Narrator")	nmf:Metadata dc:Properties dc:contributor	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookNarrator for enhanced compatibility with MPV players.
Description	nmf:Metadata dc:Properties dc:description	Free-form commentary, message, description in plain text.
Identifier ("Book ISBN")	nmf:Metadata dc:Properties dc:identifier	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookISBN for enhanced compatibility with MPV Music players.
Publisher	nmf:Metadata dc:Properties dc:publisher	Free-form plain text.
Rights ("Copyright Statement")	nmf:Metadata dc:Properties dc:rights	Free-form plain text of the copyright statement.
Title	nmf:Metadata dc:Properties dc:title	Free-form plain text of the title of the audiobook.

9.8 Audiobook profile using MPV-Music playlists

9.8.1 General

MPV-Audiobook uses MPV-Music playlists to organize audio assets and other content on a storage media. An MPV-Music playlist identifies the primary assets to be presented to the user, such as a set of audiobook chapters or sections. The MPV-Music playlist also specifies the sequence in which assets should be presented and other presentation characteristics.

MPV-Music playlists are implemented based on the MPV presentation profile specification [MPV-Pres]. The MPV-Music profile specifies the details of this usage for music content. This usage is maintained and refined for audiobook use.

An MPV-Music playlist uses the <mpvp:Album> element, which may contain metadata, foreground and background content, and related and rendition assets. An example is below.

```
<mpvp:Album>
```

```
  <mpv:Metadata> ... </mpv:Metadata>
```

```
  <nmf:Metadata> ... </nmf:Metadata>
```

```
  <mpvp:Background> ... </mpvp:Background>
```

```
  <mpvp:Foreground> ... </mpvp:Foreground>
```

<mpv:Rendition> ... </mpv:Rendition>

<mpv:Related> ... </mpv:Related>

</mpvp:Album>

MPV-Music anticipates that users will organize media files into folders/directories of their own choosing and assign long filenames. Content location and playback order may not be specified solely by requiring specific directory and filenames. Instead, MPV-Music provides an approach permitting the content to be located anywhere and with any name.

In addition, many related media assets may be present on storage media, such as CD, to enhance access and playback performance and to provide enhanced playback and printing experiences. The MPV playlist allows this content to be managed to present the user a simple high-level interaction with their content.

9.8.2 Number of playlists per MPV file

An MPV file conforming to the MPV-Audiobook shall contain one playlist (mpvm:Album element).

9.8.3 Metadata usage

Metadata placed on the mpvp:Album element defines the scope of the whole album or playlist, and metadata properties for this usage shall be as defined in Table 24. For example, the mpvm:PrincipalArtist element may be used with either a single music asset or on the mpvp:Album.

Table 24 – Metadata properties usage

Album Audiobook	mpvp:Album	
Number of Entries	implicit – number of entries in the foreground	
Title	nmf:Metadata dc:Properties dc:title	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookTitle for enhanced compatibility with MPV players.
Book Title	nmf:Metadata ab:Properties ab:BookTitle	Title of the overall book from which the chapter came.
Book Subtitle	nmf:Metadata ab:Properties BookSubtitle	Subtitle of the overall book from which the chapter came.
Contributor	nmf:Metadata dc:Properties dc:contributor	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookNarrator for enhanced compatibility with MPV players.
Narrator	nmf:Metadata ab:Properties ab:BookNarrator	Primary audiobook metadata.
Creator	nmf:Metadata dc:Properties dc:creator	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookAuthor for enhanced compatibility with MPV players.
Author	nmf:Metadata ab:Properties ab:BookAuthor	Primary audiobook metadata.
Book Coauthor	nmf:Metadata ab:Properties BookCoauthor	Coauthor of the overall book from which the chapter came.

Table 24 (continued)

Album Audiobook	mpvp:Album	
Principal Artist	nmf:Metadata mpvm:MusicProperties mpvm:PrincipalArtist	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookAuthor for enhanced compatibility with MPV Music players.
Book Genre	nmf:Metadata ab:Properties BookGenre	Genre of the overall book from which the chapter came.
Description	nmf:Metadata dc:Properties dc:description	Description summary of book.
Copyright statement	nmf:Metadata dc:Properties dc:rights	Free-form plain text of the Copyright Statement of rights.
Audiobook Content Items	mpvp:Album mpvp:Foreground	This content is played as the principal content of the audiobook playlist.
Audiobook accompaniment	mpvp:Album mpvp:Background	This content is played as background to the audiobook content in the foreground. For example, it could specify images or background music.
Audiobook Artwork	mpvp:Album mpv:Related mpv:relationship="urn:osta-org:mpv:music:artwork" mpv:StillRef	Artwork for the audiobook.
Identifier	nmf:Metadata dc:Properties dc:identifier	Not used as primary metadata for audiobooks. SHOULD be populated with the value of ab:BookISBN for enhanced compatibility with MPV Music players.
Book UUID	nmf:Metadata ab:Properties ab:BookUUID	
Book ISBN Identifier	nmf:Metadata ab:Properties ab:BookISBN	
Book ISSN Identifier	nmf:Metadata ab:Properties ab:BookISSN	
Book DOI Identifier	nmf:Metadata ab:Properties ab:BookDOI	
Book Index Number	nmf:Metadata ab:Properties ab:BookIndexNumber	
Book Shelf Number	nmf:Metadata ab:Properties ab:BookShelfNumber	
Book Part Number	nmf:Metadata ab:Properties ab:BookPartNumber	
Other Identifier	nmf:Metadata ab:Properties ab:OtherIdentifier	
Content Location Identifier	nmf:Metadata ab:Properties ab:LocationIdentifier	
Key-Value Pairs	nmf:Metadata mpvm:Properties mpvm:KeyValue mpvm kv:Key nmf:Metadata mpvm:Properties mpvm:KeyValue mpvm kv:Value	Not used.
Extra Data	mpv:Metadata and nmf:Metadata anywhere	

9.8.4 Background usage

The MPV-Audiobook may contain the mpvp:Background element to provide background presentation content to the spoken word in the foreground. It is applied at the discretion of the playback application or device. If it is a simple device or application with no visual presentation capabilities, the mpvp:Background content should be ignored.

If the application or device has visual presentation capabilities, the mpvp:Background content may be processed. The recommended content is a series of mpv:StillRef elements that refer

to still images that may be displayed behind the foreground content. This can be used in both browse/menu mode as well as playback mode. All the MPV-Pres properties may be applied, such as transitions. These are honored at the discretion of the playback application.

9.8.5 Foreground usage

An MPV-Audiobook shall include a foreground list mpvp:Foreground in mpvp:Album, which shall be identified using the mpvp:AudioRef element. mpvp:AudioRef defines the set of primary spoken word items in the playlist and their sequence. In addition to the referenced audio assets, the mpv:AssetList in the MPV file may also contain still image, video, and other kinds of assets. mpvp:Album explicitly identifies which assets in the mpv:AssetList are primary.

9.8.6 Related and rendition assets

mpvp:Album may identify renditions and related assets. Cover artwork may be specified using a mpv:Related element with an urn:osta-org:mpv:music:artwork relationship identifier.

9.9 Groups of assets in playlists

There may be situations in which a playlist has so many assets that it is inconvenient to navigate one item at a time. The playback application or device may choose to implement a FastForward or FastRewind function that jumps ahead by multiples of items.

MPV-generating applications may also choose to group assets within playlists. The playback device may choose to implement FastForward or FastRewind functions by jumping ahead by groups; group navigation may also be explicit, if desired by the playback device.

Groups are specified using standard MPV functionality. A group of assets to be played sequentially shall be identified using the mpv:Seq asset. When a playback application or device encounters a mpv:Seq asset, the application or device should play the contents sequentially.

When mpv:Seqs are used with a mpvp:Album, only the mpv:Seq assets shall be listed in the album. When the playback application or device reads mpv:Seq assets, then these assets shall play the mpv:Seq contents in sequence. See below for an example of MPV asset group's implementation.

```
<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest
xmlns:file="http://ns.osta.org/manifest/1.0/"
xmlns:ab="http://ns.ce.org/audiobook/2003b/"
xmlns:mpv="http://ns.osta.org/mpv/1.0/"
xmlns:mpvp="http://ns.osta.org/mpv/presentation/1.0/"
xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://http://purl.org/dc/terms/"
xmlns:nmf="http://ns.osta.org/nmf/1.0/"
  <nmf:Metadata>
    <ManifestProperties xmlns="http://ns.osta.org/manifest/1.0/">
```

```

    <ProfileBag>
        <Profile>http://ns.ce.org/audiobook/2003b/</Profile>
        <Profile>http://ns.osta.org/mpv/presentation/1.0/</Profile>
        <Profile>http://ns.osta.org/mpv/music/1.0/</Profile>
    </ProfileBag>
</ManifestProperties>
<Properties xmlns="http://purl.org/dc/terms/">
    <created>2003-01-29T15:51:41Z"</created>
    <modified>2003-01-29T15:51:41Z"/></modified>
</Properties>
</nmf:Metadata>

<mpvp:Album>
<nmf:Metadata>
    <dc:Properties>
        <dc:creator>Ima Smoothtalker</dc:creator>
        <dc:identifier>put an ISBN number here</dc:identifier>
        <dc:rights>Copyright (c)2003 HomeMade Press<dc:rights/>
        <dc:title>Life Along the Willamette</dc:title>
    </dc:Properties>
    <mpvm:MusicProperties>
        <mpvm:AlbumTitle>Life Along the
Willamette</mpvm:AlbumTitle>
        <mpvm:Genre>Spoken Word:Non-Fiction</mpvm:Genre>
    </mpvm:MusicProperties>
    <ab:AudiobookProperties>
        <ab:BookAuthor>Ura Writer</ab:BookAuthor>
        <ab:BookISBN>put an ISBN number here</ab:BookISBN>
        <ab:BookNarrator>Ima Smoothtalker</ab:BookNarrator>
        <ab:BookTitle>Life Along the Willamette</ab:BookTitle>
    </ab:AudiobookProperties>
</nmf:Metadata>
<mpvp:Foreground>
    <mpv:SeqRef mpv:idRef="IDSeq0001"/> <!-- chapter 1 -->

```

```

        <mpv:SeqRef mpv:idRef="IDSeq0002"/> <!-- chapter 2 -->
    </mpvp:Foreground>
</mpvp:Album>

<mpv:AssetList>
    <mpv:Seq mpv:id="IDSeq0001"> <!-- chapter 1 -->
        <mpv:AudioRef mpv:idRef="ID000001"/> <!-- page -->
        <mpv:AudioRef mpv:idRef="ID000002"/> <!-- page -->
        <mpv:AudioRef mpv:idRef="ID000003"/> <!-- page -->
    </mpv:Seq>
    <mpv:Seq mpv:id="IDSeq0002"> <!-- chapter 2 -->
        <mpv:AudioRef mpv:idRef="ID000004"/> <!-- page -->
        <mpv:AudioRef mpv:idRef="ID000005"/> <!-- page -->
        <mpv:AudioRef mpv:idRef="ID000006"/> <!-- page -->
    </mpv:Seq>
        <mpv:Audio mpv:id="ID000001">
            <nmf:Metadata>
                <Properties xmlns="http://purl.org/dc/elements/1.1">
                    <contributor>Ima Smoothtalker</contributor>
                    <creator>Ura Writer</creator>
                    <format>audio/mpeg</format>
                    <title>page 1</title>
                </Properties>
                <MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0/">
                    <AlbumTitle>Life Along the Willamette</AlbumTitle>
                    <Genre>Spoken Word:Non-Fiction</Genre>
                    <PlayingTime>3.48</PlayingTime>
                </MusicProperties>
            </nmf:Metadata>
            <mpv:LastURL>audio/page1.mp3</mpv:LastURL>
        </mpv:Audio>
        <mpv:Audio mpv:id="ID000002">

```

```

<nmf:Metadata>
  <Properties xmlns="http://purl.org/dc/elements/1.1">
    <contributor>Ima Smoothtalker</contributor>
    <creator>Ura Writer</creator>
    <format>audio/mpeg</format>
    <title>page 2</title>
  </Properties>
  <MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0/">
    <AlbumTitle>Life Along the Willamette</AlbumTitle>
    <Genre>Spoken Word:Non-Fiction</Genre>
    <PlayingTime>3.24</PlayingTime>
  </MusicProperties>
</nmf:Metadata>
<mpv:LastURL>audio/page2.mp3</mpv:LastURL>
</mpv:Audio>
</mpv:AssetList>
</file:Manifest>

```

9.10 MPV-Audiobook schema details

9.10.1 XML schema definition

The XML Schema for an MPV-Audiobook shall be as specified below.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema target Namespace="http://ns.ce.org/audiobook/2003b/"
  xmlns=http://ns.ce.org/audiobook/2003b/
  xmlns:nmf="http://ns.osta.org/nmf/1.0/tools/"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:nmf=http://ns.osta.org/nmf/1.0/
  elementFormDefault="qualified" attributeFormDefault="qualified">
  <xs:import namespace="http://ns.osta.org/nmf/1.0/" schemaLocation="nmf.xsd"/>
  <xs:annotation>
    <xs:documentation>The CEA R6 WG11 Audiobook Properties
schema</xs:documentation>
  </xs:annotation>

```

<!--name for BySchemaProperties element: AudiobookProperties-->

<xs:element name="AudiobookProperties" type="BySchemaPropsType" substitutionGroup="nmf:BySchemaPropsBase"/>

<!--top-level schema element type. Note that to be MPV NMF compliant, the order of the elements MUST be alphabetical. -->

<xs:complexType name="BySchemaPropsType">

<xs:complexContent>

<xs:extension base="nmf:BySchemaPropsType">

<xs:sequence>

<xs:element ref="BookAuthor" minOccurs="0"/>

<xs:element ref="BookDOI" minOccurs="0"/>

<xs:element ref="BookIndexNumber" minOccurs="0"/>

<xs:element ref="BookISBN" minOccurs="0"/>

<xs:element ref="BookISSN" minOccurs="0"/>

<xs:element ref="BookNarrator" minOccurs="0"/>

<xs:element ref="BookPartNumber" minOccurs="0"/>

<xs:element ref="BookShelfNumber" minOccurs="0"/>

<xs:element ref="BookTitle" minOccurs="0"/>

<xs:element ref="BookUUID" minOccurs="0"/>

<xs:element ref="OtherIdentifier" minOccurs="0"/>

<xs:element ref="LocationIdentifier" minOccurs="0"/>

<xs:element ref="unitType" minOccurs="0"/>

<xs:element ref="unitContext" minOccurs="0"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:element name="BookTitle" type="BookTitleType"/>

<xs:complexType name="BookTitleType">

<xs:simpleContent>

<xs:extension base="xs:string"/>

</xs:simpleContent>

</xs:complexType>

```
<xs:element name="BookAuthor" type=" BookAuthorType"/>
<xs:complexType name=" BookAuthorType">
  <xs:simpleContent>
    <xs:extension base="xs:string"/>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="BookNarrator" type=" BookNarratorType"/>
<xs:complexType name=" BookNarratorType">
  <xs:simpleContent>
    <xs:extension base="xs:string"/>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="BookUUID" type=" BookUUIDType"/>
<xs:complexType name=" BookUUIDType">
  <xs:simpleContent>
    <xs:extension base="xs:string"/>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="BookGenre" type="BookGenreType"/>
<xs:simpleType name="BookGenreType">
  <xs:union memberTypes="BookGenreBaseType xs:anyURI"/>
</xs:simpleType>
<xs:simpleType name="BookGenreBaseType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Spoken Word:Fiction"/>
    <xs:enumeration value="Spoken Word:Non-Fiction"/>
    <xs:enumeration value="Audiobook:Fiction"/>
    <xs:enumeration value="Audiobook:Non-Fiction"/>
  </xs:restriction>
```

```
</xs:simpleType>
```

```
<xs:element name="BookISSN" type="BookISSNType"/>
```

```
<xs:complexType name="BookISSNType">
```

```
  <xs:simpleContent>
```

```
    <xs:extension base="xs:string"/>
```

```
  </xs:simpleContent>
```

```
</xs:complexType>
```

```
<xs:element name="BookDOI" type="BookDOIType"/>
```

```
<xs:complexType name="BookDOIType">
```

```
  <xs:simpleContent>
```

```
    <xs:extension base="xs:string"/>
```

```
  </xs:simpleContent>
```

```
</xs:complexType>
```

```
<xs:element name="BookIndexNumber" type="BookIndexNumberType"/>
```

```
<xs:complexType name="BookIndexNumberType">
```

```
  <xs:simpleContent>
```

```
    <xs:extension base="xs:string"/>
```

```
  </xs:simpleContent>
```

```
</xs:complexType>
```

```
<xs:element name="BookShelfNumber" type="BookShelfNumberType"/>
```

```
<xs:complexType name="BookShelfNumberType">
```

```
  <xs:simpleContent>
```

```
    <xs:extension base="xs:string"/>
```

```
  </xs:simpleContent>
```

```
</xs:complexType>
```

```
<xs:element name="BookPartNumber" type="BookPartNumberType"/>
```

```
<xs:complexType name="BookPartNumberType">
```

```
  <xs:simpleContent>
```

```
<xs:extension base="xs:string"/>
</xs:simpleContent>
</xs:complexType>

<xs:element name="OtherIdentifier" type="OtherIdentifierType"/>
<xs:complexType name="OtherIdentifierType">
  <xs:simpleContent>
    <xs:extension base="xs:string"/>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="LocationIdentifier" type="LocationIdentifierType"/>
<xs:complexType name="LocationIdentifierType">
  <xs:simpleContent>
    <xs:extension base="xs:string"/>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="unitType" type="unitTypeType"/>
<xs:complexType name="unitTypeType">
  <xs:simpleContent>
    <xs:extension base="xs:string"/>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="unitContext" type="unitContextType"/>
<xs:complexType name="unitContextType">
  <xs:simpleContent>
    <xs:extension base="xs:string"/>
  </xs:simpleContent>
</xs:complexType>
</xs:schema>
```

9.10.2 Audiobook metadata usage definitions

MPV-Audiobook properties shall be defined as:

- a) **BookAuthor** – This element shall contain the audiobook author name. This is a required element and shall be present in every audiobook. If no author exists, the value shall be blank.
- b) **BookCoAuthor** – This element shall contain the audiobook co-author name. If more than one co-author exists, multiple values may be present. If no co-author exists, this element shall not be present.
- c) **BookDOI** – This element shall contain the DOI (Digital Object Identifier). If no DOI exists, this element shall not be present.
- d) **BookGenre** – This element shall contain the audiobook genre. If more than one genre exists, multiple values may be present. If no genre value exists, this element shall not be present.
- e) **BookIndexNumber** – This element shall contain the audiobook index number. If no index number exists, this element shall not be present.
- f) **BookISBN** – This element shall contain the audiobook ISBN (International Standard Book Number). If no ISBN exists, this element shall not be present.
- g) **BookISSN** – This element shall contain the audiobook ISSN (International Standard Serial Number). If no ISSN exists, this element shall not be present.
- h) **BookNarrator** – This element shall contain the audiobook narrator name. This is a required element and shall be present in every audiobook. If more than one narrator exists, multiple values may be present. If no narrator exists, the value shall be blank.
- i) **BookPartNumber** – This element shall contain the audiobook part number. If no part number exists, this element shall not be present.
- j) **BookShelfNumber** – This element shall contain the audiobook shelf number. If no shelf number exists, this element shall not be present.
- k) **BookSubTitle** – This element shall contain the audiobook subtitle. If no subtitle exists, this element shall not be present.
- l) **BookTitle** – This element shall contain the audiobook title. This is a required element and shall be present in every audiobook. If there is no title, the value shall be blank.
- m) **BookUUID** – This element shall contain the audiobook UUID (Universally Unique Identifier). If no UUID exists, this element shall not be present.
- n) **OtherIdentifier** – This element shall contain an additional audiobook identifier, which is not defined by this standard and can be any value that helps uniquely identify the audiobook. If no other identifier exists, this element shall not be present.
- o) **LocationIdentifier** – This element shall contain the playlist content location identifier. This identifier is a number indicating the location of files for this playlist (locally, online as downloadable content, online as streaming content or any combination). This number can be derived from Table 15. If the field is null or not present, value 0 (local) should be assumed.

9.11 Audiobook level class identifiers

The following tags will allow a playback device to handle distinct audio book levels in customized ways, depending on the type of content, or sub-grouping, of those levels.

The goal is to provide additional metadata that follows more consistent convention than level names intended for user display, allowing an aware audio book playback platform to offer more intelligent handling of particular audio book levels.

Examples:

```
<mpv:Seq>
  <mpv:Audio>
    <ab:unitType>chapter</ab:unitType>
    <ab:unitContext>book/novel</ab:unitContext>

<mpv:Seq>
  <mpv:Audio>
    <ab:unitType>quatrain, stanza, verse, paragraph</ab:unitType>
    <ab:unitContext>anthology/poems</ab:unitContext>
```

Support for this metadata is optional and will vary among playback platforms. The degree of support is left to platform designers, and might be determined by intended capabilities of the platform, anticipated content for the platform, or the targeted customers and market. For instance, a playback platform might offer special handling strictly for a line of audiobook titles sold by the platform maker or its publishing partners, or specifically for religious titles, or more specifically for the scripture of a specific faith.

A platform's special handling might be as basic as displaying a genre-appropriate icon in the user interface, altering the navigation scheme to better suit a particular audio book genre, or saving a user's own preferences for particular genres. A platform's special handling could be keyed by unit keywords alone ("chapter"), by genre alone ("scripture"), or by various combinations of unit, genre and sub-genres.

Platforms can be designed to respond to numerous keywords, or just one or two. In this way, a platform can potentially treat chapters differently than pages or sentences, treat novels differently than scripture, treat chapters in novels differently from chapters in scripture, or treat scripture of one faith differently from scripture of another faith.

Unit – This element consists of a single term such as “chapter” or “paragraph”, or a comma-separated list that begins with precise, genre-specific terms and progresses to generic, widely-supported terms (“quatrain, stanza, verse, paragraph”). The element is optional, and the text is completely customizable by content providers.

A comma-separated list can serve as a fallback mechanism, allowing playback platforms that don't recognize a more precise keyword to provide special handling based on the generic alternatives. In addition, a comma-separated list allows content providers to include keywords or transliterated spellings from other languages (“Ayah, verset, paragraphe, verse, paragraph”) if these terms are more appropriate to the language of the audio content, the target audience or the anticipated playback platform.

Customary **Unit** Keywords:

act, album, anthology, article, author, basket, book, branch, chapter, class, clip, collection, contributor, date, day, division, entry, event, group, half, hymn, instructions, issue date, landmark, lecture, lesson, letter, level, line, location, mark, measure, meeting, monologue, note, number, octave, page, paragraph, part, passage, performance, period, phrase, piece, poem, poet, point of interest, portion, program, quarter, recension, scene, section, segment, sentence, session, set, song, speech, subject, stage, stanza, story, teaching, testament, text, third, time, topic, tour, unit, verse, version, volume, weekday, word.

Context – This element identifies a genre and, optionally, more granular sub-genres separated by slashes, such as “reference”, “book/novel”, “performance/comedy”, or “performance/stage/play/one-act”. The element is optional, and the text is completely customizable by content providers.

Customary **Context** Keyword Pairings:

anthology: anthology/literature, anthology/miscellaneous, anthology/poems, anthology/stories

collection: collection/literature, collection/miscellaneous, collection/poems, collection/stories

book: book/novel, book/poems, book/textbook, book/cookbook

broadcast: broadcast/online, broadcast/radio, broadcast/tv

event: event/class, event/conference, event/meeting, event/tour

education: education/class, education/conference, education/textbook, education/tour

instruction: instruction/class, instruction/conference, education/textbook, instruction/tour

performance: performance/comedy, performance/music, performance/reading,
performance/spoken, performance/stage, performance/variety

periodical: periodical/journal, periodical/magazine, periodical/newsletter,
periodical/newspaper

reference: reference/cookbook, reference/dictionary, reference/guide, reference/encyclopedia,
reference/textbook, reference/thesaurus

scripture: scripture/abrahamic, scripture/buddhist, scripture/christian, scripture/hindu,
scripture/islam, scripture/judaism, scripture/judeochristian, scripture/ids, scripture/sikh

9.12 Audiobook BOOKMARK.AUB

The optional audiobook bookmarks are XML files and stored as an external file to the INDEX.AUB files. This bookmark file could be written to a rewritable media (internal FLASH memory) and reference a “playlist” from a read-only media (for example, a CD). The bookmark file could be written to an external rewritable flash memory and then accessed by a second device. The bookmark could be written to a network location and accessed remotely by other devices via a network (wired or wireless).

The following MPV example file includes two types of bookmarks.

AutoResume – This is a bookmark automatically generated by the system based on certain events; for example, the system is turned off or an audiobook disc is removed from the system. Having a separate bookmark file that can be transferred from one compliant playback device to another is an optional feature that greatly enhances the user experience.

UserGenerated – This is a bookmark generated by an action from the user (pressing a “Bookmark” button) to mark an important segment of the audiobook.

Bookmark file name MUST be called BOOKMARK.AUB and must be unique on a device to enable reader devices to easily find saved bookmarks. Extensions within the bookmark file are allowed as long as they adhere to XML conventions.

There can only be ONE AutoResume bookmark per user in a BOOKMARK.AUB file.

The bookmarks can be of length greater than zero indicating an entire passage. The BOOKMARK.AUB only has the metadata for the first audio file. Therefore, if the bookmark passage starts on one audio file and ends on another, then it is the playback device's

responsibility to find and determine the end location on the second audio file based on the original .aub volume description file.

The bookmark contains minimum high-level metadata about the bookmark (e.g., Book name, author, chapter number/name, page number/name) so that the consumer can visually differentiate between the various bookmarks and the device doesn't have to open any other file to present the bookmark file to the consumer.

There is a CreatedDate tag which can be automatically created by the device to describe each bookmark. In addition, an optional MarkName tag is available which can be populated by consumer using an input device (keyboard, remote control, etc.) to give the bookmark a more descriptive name. AutoResume bookmarks should not have a MarkName.

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

```
<file:Manifest xmlns:file="http://ns.osta.org/manifest/1.0/"
xmlns:mpv="http://ns.osta.org/mpv/1.0/" xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"
```

```
xmlns:mpvb="http://ns.osta.org/mpv/basic/1.0/"
```

```
xmlns:dc="http://ns.osta.org/nmf/1.0/dc/"
```

```
xmlns:nmf="http://ns.osta.org/nmf/1.0/"
```

```
xmlns:aub="http://www.cea.org/dtd/2003b/1.1">
```

```
<nmf:Metadata>
```

```
<ManifestProperties xmlns="http://ns.osta.org/manifest/1.0/">
```

```
<ProfileBag>
```

```
<Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
```

```
</ProfileBag>
```

```
</ManifestProperties>
```

```
</nmf:Metadata>
```

```
<!--beginning of bookmarks-->
```

```
<mpvb:MarkedAssets>
```

```
<!--Bookmark 1: Autoresume-->
```

```
<mpv:MarkList mpv:markType="cea:2003b:Autoresume:2005">
```

```
<mpv:AudioRef mpv:manifestLinkIdRef="ML001" mpv:idRef="AU0003">
```

```
<ab:AudioBookProperties>
```

```
<ab:bookAuthor>Joe Bestseller</ab:bookAuthor>
```

```

<ab:bookNarrator>Norma Dialects</ab:bookNarrator>

<ab:bookTitle>Use Case Example 1</ab:bookTitle>
</ab:AudioBookProperties>

  <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
      <contributor>Joe Bestseller</contributor>
      <creator>Joe Bestseller</creator>
      <title>Use Case Example 1</title>
    </Properties>
    <MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0">
      <AlbumTitle>Use Case Example 1</AlbumTitle>
      <Genre>Thriller</Genre>
    </MusicProperties>

<!--Note the next two sections contain metadata about the particular UnitType (chapter or
page, for example) which is the smallest unit type where the Bookmark is located-->

<!--Chapter 1-->

      <mpv:Seq>
        <mpv:Audio>
          <ab:unitType>chapter</ab:unitType>
            <nmf:Metadata>
              <Properties
xmlns="http://purl.org/dc/elements/1.1">
                <title>Chapter 1</title>
                <description>Chapter
1</description>
              </Properties>
            <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
              <AlbumTitle>Use Case Example 1</AlbumTitle>
              <Genre>spoken word</Genre>
            </MusicProperties>
          </nmf:Metadata>

```

<!--Begin Pages-->

```

    <mpv:Seq>
      <mpv:Audio>
        <ab:unitType>page</ab:unitType>

        <nmf:Metadata>
          <Properties xmlns="http://purl.org/dc/elements/1.1">
            <title>Page
1</title>
            <description>Page 1</description>
          </Properties>
        </nmf:Metadata>
      </mpv:Audio>
    </mpv:Seq>
  </mpv:Audio>
</mpv:Seq>

```

<!--This is the actual bookmark in time offset from beginning of file. Bookmark can be of length greater than zero indicating a passage.-->

```

    <Properties xmlns="http://www.cea.org/dtd/2003b/1.1">
      <CreateDate>2005-05-25T17:44:47+01:00</CreateDate>
      <MarkUser>User 1</MarkUser>
      <StartAt>(insert annodex time code here)</StartAt>
      <StopAt>(insert annodex time code here)</StopAt>
    </Properties>
  </nmf:Metadata>
</mpv:AudioRef>
</mpv:MarkList>

```

<!--Bookmark 2: user input -->

```

<mpv:MarkList mpv:markType="cea:2003b:UserInputBookmark:2005">
  <mpv:AudioRef mpv:manifestLinkIdRef="ML001" mpv:idRef="AU0002">

```

<!--Bookmark 2: does it need a name?-->

<ab:bookmark>My Favorite Passage</ab:bookmark>

<mpv:Seq>

<mpv:Audio>

<ab:AudioBookProperties>

<ab:unitType>Collection</ab:unitType>

<ab:unitContext>collection/stories</ab:unitContext>

<ab:bookISBN>UC-0002-000</ab:bookISBN>

<ab:bookAuthor>Joe Bestseller</ab:bookAuthor>

<ab:bookNarrator>Norma Dialects</ab:bookNarrator>

<ab:bookTitle>Use Case Example 2</ab:bookTitle>

</ab:AudioBookProperties>

<ab:unitType>Collection</ab:unitType>

<ab:unitContext>collection/stories</ab:unitContext>

<nmf:Metadata>

<Properties xmlns="http://purl.org/dc/elements/1.1">

<contributor>Norma Dialects</contributor>

<creator>Joe Bestseller</creator>

<format>audio/mpeg</format>

<title>Use Case Example 2</title>

</Properties>

<MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0">

<AlbumTitle>Use Case Example 2</AlbumTitle>

<Genre>Suspense</Genre>

<PlayingTime>420</PlayingTime>

</MusicProperties>

<!--Story 1-->

<mpv:Seq>

<mpv:Audio>

<ab:unitType>story</ab:unitType>

```

<nmf:Metadata>
  <Properties xmlns="http://purl.org/dc/elements/1.1">
    <title>Story Title</title>
    <description>Story Title</description>
  </Properties>
  <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
    <AlbumTitle>Collection Title</AlbumTitle>
    <Genre>Collection/Stories</Genre>
  </MusicProperties>
</nmf:Metadata>

<!--Chapter 1-->

<mpv:Seq>
<mpv:Audio>
  <ab:unitType>chapter</ab:unitType>
  <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
      <title>Chapter 1</title>
      <description>Chapter 1</description>
    </Properties>
  </nmf:Metadata>

</mpv:Audio>

<!--Pages-->

<mpv:Audio>
  <ab:unitType>page</ab:unitType>
  <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
      <title>Page 1</title>
      <description>Chapter 1</description>
    </Properties>

```

```

        </nmf:Metadata>
    </mpv:Audio>

    </mpv:Seq>
</mpv:Audio>

</mpv:Seq>
</mpv:Audio>
</mpv:Seq>

<nmf:Metadata>
  <Properties xmlns="http://www.cea.org/dtd/2003b/1.1">
    <CreateDate>2005-05—25T17:22:23+01:00</CreateDate>
    <MarkName>Great Passage on Love</MarkName>
    <MarkUser>User 2</MarkUser>
    <StartAt>(insert annodex time code here)</StartAt>
    <StopAt>(insert annodex time code here)</StopAt>
  </Properties>
</nmf:Metadata>
</mpv:AudioRef>
</mpv:MarkList>
</mpvb:MarkedAssets>

<mpv:AssetList>
  <mpv:ManifestLink mpv:id="AU001">
    <mpv:LastURL>
      <mpv:LastURL
1">CHAPTER1.MP3</mpv:LastURL>
      </mpv:LastURL>
    </mpv:ManifestLink>
  </mpv:AssetList>
</file:Manifest>
  mpv:filesystem="ISO9660-

```

9.13 MPV-Audiobook-Level metadata using <mpvp:Album>

This International Standard specifies that MPV-Audiobook content metadata properties on an mpvp:Album apply also to the content referenced unless explicitly overridden by asset-level metadata. For example, the <ab:BookTitle> property applies to all of the <mpv:Audio> items referenced directly or indirectly by the <mpvp:Album> element.

Recognized metadata at the Audiobook-level includes Dublin Core, mpvm:MusicProperties, ab:AudiobookProperties, and properties from the MPV Presentation profile.

9.14 <mpvm:AudioWithStills> Music and Stills Asset

This International Standard does not specify use of the <mpvm:AudioWithStills> element at this time. See below for an example of MPV-Audiobook profile implementation.

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

```
<file:Manifest                                xmlns:file="http://ns.osta.org/manifest/1.0/"
xmlns:ab="http://ns.ce.org/audiobook/2003b/"    xmlns:mpv="http://ns.osta.org/mpv/1.0/"
xmlns:mpvp="http://ns.osta.org/mpv/presentation/1.0/"
xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"  xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://http://purl.org/dc/terms/" xmlns:nmf="http://ns.osta.org/nmf/1.0/">
```

```
    <nmf:Metadata>
```

```
        <ManifestProperties xmlns="http://ns.osta.org/manifest/1.0/">
```

```
            <ProfileBag>
```

```
                <Profile>http://ns.ce.org/audiobook/2003b/</Profile>
```

```
                <Profile>http://ns.osta.org/mpv/presentation/1.0/</Profile>
```

```
                <Profile>http://ns.osta.org/mpv/music/1.0/</Profile>
```

```
            </ProfileBag>
```

```
        </ManifestProperties>
```

```
        <Properties xmlns="http://purl.org/dc/terms/">
```

```
            <created>2003-01-29T15:51:41Z</created>
```

```
            <modified>2003-01-29T15:51:41Z/></modified>
```

```
        </Properties>
```

```
    </nmf:Metadata>
```

```
    <mpvp:Album>
```

```
        <nmf:Metadata>
```

```
            <dc:Properties>
```

```
                <dc:creator>Ima Smoohtalker</dc:creator>
```

```
                <dc:identifier>put an ISBN number here</dc:identifier>
```

```
                <dc:rights>Copyright (c)2003 HomeMade Press<dc:rights/>
```

```

        <dc:title>Life Along the Willamette</dc:title>
    </dc:Properties>
    <mpvm:MusicProperties>
        <mpvm:AlbumTitle>Life Along the
Willamette</mpvm:AlbumTitle>
        <mpvm:Genre>Spoken Word:Non-Fiction</mpvm:Genre>
    </mpvm:MusicProperties>
    <ab:AudiobookProperties>
        <ab:BookAuthor>Ura Writer</ab:BookAuthor>
        <ab:BookISBN>put an ISBN number here</ab:BookISBN>
        <ab:BookNarrator>Ima Smoothtalker</ab:BookNarrator>
        <ab:BookTitle>Life Along the Willamette</ab:BookTitle>
    </ab:AudiobookProperties>
</nmf:Metadata>
<mpvp:Foreground>
    <mpv:SeqRef mpv:idRef="IDSeq0001"/> <!-- chapter 1 -->
    <mpv:SeqRef mpv:idRef="IDSeq0002"/> <!-- chapter 2 -->
</mpvp:Foreground>
</mpvp:Album>

<mpv:AssetList>
    <mpv:Seq mpv:id="IDSeq0001"> <!-- chapter 1 -->
        <mpv:AudioRef mpv:idRef="ID000001"/> <!-- page -->
        <mpv:AudioRef mpv:idRef="ID000002"/> <!-- page -->
        <mpv:AudioRef mpv:idRef="ID000003"/> <!-- page -->
    </mpv:Seq>
    <mpv:Seq mpv:id="IDSeq0002"> <!-- chapter 2 -->
        <mpv:AudioRef mpv:idRef="ID000004"/> <!-- page -->
        <mpv:AudioRef mpv:idRef="ID000005"/> <!-- page -->
        <mpv:AudioRef mpv:idRef="ID000006"/> <!-- page -->
    </mpv:Seq>
    <mpv:Audio mpv:id="ID000001"> <!-- other metadata inherited from Audiobook
-->

```

-->

```

<nmf:Metadata>
  <Properties xmlns="http://purl.org/dc/elements/1.1">
    <format>audio/mpeg</format>
    <title>page 1</title>
  </Properties>
  <MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0/">
    <PlayingTime>3.48</PlayingTime>
  </MusicProperties>
</nmf:Metadata>
<mpv:LastURL>audio/page1.mp3</mpv:LastURL>
</mpv:Audio>
<mpv:Audio mpv:id="ID000002">
<!--this explicit metadata overrides Audiobook; happens to be the same here -->
  <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
      <contributor>Ima Smoothtalker</contributor>
      <creator>Ura Writer</creator>
      <format>audio/mpeg</format>
      <title>page 2</title>
    </Properties>
    <MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0/">
      <AlbumTitle>Life Along the Willamette</AlbumTitle>
      <Genre>Spoken Word:Non-Fiction</Genre>
      <PlayingTime>3.24</PlayingTime>
    </MusicProperties>
  </nmf:Metadata>
  <mpv:LastURL>audio/page2.mp3</mpv:LastURL>
</mpv:Audio>
...
</mpv:AssetList>
</file:Manifest>

```

9.15 MPV-Audiobook extensions to MPV-Core

9.15.1 Audiobook manifest file types and extensions

9.15.1.1 File extension

For systems in which file type is carried by the file name extension, such as Microsoft Windows and Unix, the MPV-Audiobook manifest file shall utilize an extension. This International Standard defines two extensions a manifest may carry. Each MPV-Audiobook shall utilize one of the following audiobook-specific MPV file extensions as follows.

- a) **.aub** – This extension identifies a file to be an MPVAudiobook profile (AUB) file. Usage is case insensitive. This extension may be registered by an application to provide default and alternate processors of audiobook manifests.
- b) **.mum** – This extension identifies a file to be a Music Manifest (MUM). Audiobooks compliant with this International Standard may utilize the same MPV file extension specified by the MPV Music Profile-- .mum. Usage is case insensitive. This extension may be registered by an application to provide default and alternate processors of music manifests.

9.15.1.2 Apple Macintosh file type

The Apple Macintosh operating system uses an internal file type stored as a resource value of the data fork of a file. Audiobooks compliant with this International Standard shall utilize an audiobook-specific MPV file type, which is as follows.

- a) **.aub** – This Apple Macintosh file type identifies the file to contain an audiobook manifest. Usage is case sensitive. This extension may be registered by an application to provide a default processor of audiobook manifests.
- b) **.mum** – This Apple Macintosh file type identifies the file to contain a music manifest. Audiobooks compliant with this International Standard may, in addition, utilize the same file type as MPV music manifests on Macintosh systems. Usage is case sensitive. This extension may be registered by an application to provide a default processor of music manifests.

Some applications examine leading characters of a file in an attempt to determine its file type. No byte sequences can be counted on to always be present, but generally all XML documents in the UTF-8 charsets begin with hexadecimal 3C 3F 78 6D 6C, ("<?xml"). While this identifies the document as an XML document, it does not identify the document as a music manifest. Parsing the document to locate the outer element defined by the manifest schema further identifies the document.

9.15.2 Audiobook manifest MIME media type

MIME media types are widely used in internet applications to indicate the type of a file or content in a manner external of the file and independent of the name of the file or any information embedded in the file [MIME-2]. IANA maintains a registry of MIME media types, and a set of MIME media types is found at [MIMETYPES-REG].

Audiobooks compliant with this International Standard may utilize the same MIME media type specified by the MPV-Music.

- a) **application/vnd.osta-org.mum+xml** – This MIME media type identifies the content to be a Music Manifest. Usage is case sensitive. This media type may be registered with internet browsers by an application to provide the default processor of a music profile.
- b) **application/vnd.ce-org.aub+xml** – This is the audiobook-specific MIME media type for an audiobook manifest. This MIME media type identifies the content to be a audiobook manifest. Usage is case sensitive. This media type may be registered with internet browsers by an application to provide the default processor of an audiobook.

9.16 Media types and file formats

CEA-2003-C defines additional audio formats to those defined by the MPV music profile.

9.17 MPV-Audiobook and CEA-2003-C

Subclause 9.17 defines the MPV profile for audiobooks (MPV-Audiobook), and enables MPV developers to implement support for audiobook content. To comply with CEA-2003-C, developers shall comply with all normative elements of CEA-2003-C.

9.18 CEA-2003-C binary to XML formats – Tag comparison

Table 25 compares CEA-2003-C binary and XML format tags.

Table 25 – CEA-2003-C binary and XML formats – Tag comparison

CEA-2003-C metadata tags based on MPV	CEA-2003-A metadata tags based on MultiAudio
Music Asset ("Song", "Track")	Audiobook Asset ("Chapter")
Pathname	Pathname
Format	Format
Title	Chapter Title
Album Title	Book Title
Performed by	Narrator
Principal Artist	Author
Band, Orchestra, Accompanist	N/A
Music by	Music by
Lyrics by	N/A
Arranged by	N/A
Produced by	Produced by
Genre	Genre
Date Recorded	Date Recorded
Identifier	Book ISBN Number
Description	Description
Key words	N/A
Rights	Copyright Statement
Language	Language
Rendition using another encoder	N/A
Rendition using another bit rate	N/A
Encoded Bit rate	Encoded Bit rate
Asset Artwork	N/A
Music Video	N/A
Online Info	Online Info
Playing Time	Playing Time
Track Number	Chapter Number
Number Tracks	Number of Chapters
Set Number in Collection	N/A
Number Sets in Collection	N/A
Play Count	N/A
Lyrics	N/A

Table 25 (continued)

CEA-2003-C metadata tags based on MPV	CEA-2003-A metadata tags based on MultiAudio
Mood	N/A
Tempo	N/A
Situation	N/A
Medium	N/A
Original file and filename	N/A
Extra Data	N/A
Key-Value Pairs	N/A
Album ("Playlist") of Music	Audiobook ("Playlist")
Number of Entries	Number of Entries
Title	Book Title
Performed by	Narrator
Principal Artist	Author
Description	Description
Identifier	Book ISBN Number
Rights	Copyright Statement
Album Artwork	Audiobook Artwork
Music Playlist Items	Audiobook Playlist Items
Playlist Accompaniment	N/A
Key-Value Pairs	N/A
Extra Data	Book ISSN Identifier
	Book DOI Identifier
	Audiobook UUIDs
	Book Index Number
	Book Shelf Number
	Book Part Number
	Other Identifier

10 Conforming player and digital audiobook file format structure requirements

10.1 General

Requirements for audiobook compatible players and digital audiobook file formats are contained in 10.2 or 10.3, respectively.

10.2 Player Requirements

To conform to CEA-2003-C, players shall:

- read and process the INDEX.AUB (INDEX.MUM) or TOC.MAU file in the root of the digital audiobook file format structure, process the audiobook required data and Extended Data identified in Clauses 4 through 8;
- present audio files in their correct order as specified in the INDEX.AUB or TOC.MAU file and shall read and process Audiobook Extended Data, as specified in Clauses 4 through 8;
- recover from a missing or 'file not found' situation, and offer alternatives or a meaningful message to the user; such as using alternate playlists as described in Clause 7;
- support Level 1 navigation; for example, book chapters;

- e) support moving forward and backward in the content in time (fast forward and fast reverse);
- f) support accurate presentation of the time elapsed in the current audio object;
- g) play content continuously, without excessive, unintended pauses;
- h) support forward and reverse navigation to next or previous Level 1 elements, such as chapters, when such level and level elements are available in digital audiobook file format structures, as defined in Clause 7;
- i) hold the current state of play (automatic bookmark/auto resume) for the current play position in the current audiobook in such a manner as to automatically resume at that position – both when playback is stopped and then resumed, and when power is restored following a power-off condition, provided in the latter case that the same audiobook is present when power is restored.

10.3 Digital audiobook file format structure requirements

To conform to CEA-2003-C, digital audiobook file format structures shall:

- a) contain the INDEX.AUB or TOC.MAU file, including the required data and Extended Data as identified in Clauses 4 through 8;
- b) identify levels and level elements required for Level 1 navigation, such as chapters;
- c) provide description of delivery method, if not entirely local files, utilizing the Playlist Content Location Identifier as defined in Clause 7, Table 15;
- d) identify encoding type of files as specified in the OSTA MPV specification set, Appendix A. (See Annex B for OSTA list of “well known” encoding type identifiers.)

11 Player and digital audiobook file format structure functionality recommendations

11.1 General

Although not required for compliance, CEA-2003-C makes the following recommendations to further enhance the audiobook playback experience. Requirements for audiobook compatible players and digital audiobook file format structures are contained in 10.2 or CEA-2004, respectively.

11.2 Player functionality recommendations

It is recommended that players optionally should:

- a) process Extra Data, such as author and narrator, as defined in Clause 7 and present it to the listener;
- b) support forward and reverse navigation to next or previous Level 2 or higher elements, such as chapters, when such levels and level elements are available in audiobook data structures on audiobook media as defined in Clause 8;
- c) support the presentation of detailed information about the listener's current location in the audiobook, for example, Level Name and Level Description as described in Clause 8;
- d) move the bookmarks associated with content when the content is moved to another player;
- e) provide audible prompts describing title, location, movement, etc;
- f) support the presentation of additional audiobook data, for example, current section name, as defined in 7.3.17;
- g) support the presentation of the total time of an audiobook as defined in 8.3.13;
- h) support the presentation of time remaining in the current audiobook;
- i) present the total time elapsed in the title;

- j) store the closing bookmark for multiple content titles and support resuming of playback at the stored position, as long as digital audiobook file format structures are not removed (enhanced automatic bookmarking);
- k) store the closing bookmark for several audiobooks and support resuming of playback at the stored position. These stored multiple bookmarks should be retained despite stopping playback of any of the audiobooks or powering off;
- l) hold state for multiple digital audiobook file format structures;
- m) provide user-settable bookmarks (distinct from "automatic bookmark");
- n) provide support for accelerated fast forward and fast reverse movement through audiobook content;
- o) provide a mechanism to re-read immediately preceding content, for example, single key press to skip back 15 s and continue to read;
- p) present time elapsed as HH:MM:SS as defined in 8.3.13;
- q) provide a range of playback speeds while preserving the pitch of the narrator's voice. 75 % to 200 % is considered good, and 50 % to 300 % is considered ideal;
- r) play files in ASCII-collated order in the absence of playlist, in order to support legacy audiobooks as defined in Clause 3, NOTE 2;
- s) process any extra data fields as specified in Clause 5.

11.3 Digital audiobook file format structure functionality recommendations

It is recommended that digital audiobook file format structures optionally should:

- a) contain optional Extra Data fields as defined in Clauses 4 through 8;
- b) contain audiobook information in greater detail, for example, current level name and level description, as defined in 7.3.17;
- c) contain total time of an audiobook in HH:MM:SS format, as defined in 8.3.13.

12 Digital audiobook file format structures and data constraints

Data shall be formatted in either single channel (mono) or two-channel (stereo) audio. Supporting more than two channels is not required nor recommended, as player behavior is not predictable in this case.

The digital audiobook file format structures shall make use of a file system that allows long file names (more than 8.3 characters).

The playlist directory file shall be located in the root directory of the digital audiobook file format structure. Multiple audiobooks within one file structure, for instance, a collection of short stories, containing two or more books shall be formatted with each book's data in its own subfolder. Audiobook file format structures containing only one book may be formatted with data in a corresponding subfolder or in the root of the structure.

13 Additional production options

13.1 ID3 tagging within MP3 files

The following information is illustrative of tagging. CEA-2003-C requires a TOC.mau file to exist in the root directory of the audiobook file format structure. The TOC.mau contains information relating to each track, as well as the order in which the tracks are to be played. The audiobook publisher should be aware that although ID3 tags are not required, if they exist, these files may be utilized by authoring software to help create the TOC.mau file.

ID3 tags store information about an MP3 within the file itself. If the audiobook is created from MP3 files that already contain ID3 tag information, the relevant tags can be scanned and used in the appropriate TOC.mau fields. ID3 tags are standardized in ID3 specifications, and examples are found in Table 26. Since ID3 tagging software is readily available, developers of TOC.mau authoring software may elect to extract the information directly from these tags, rather than creating TOC.mau information manually.

13.2 M3U playlists

Playlists in *.m3u format may also be exploited in the creation of a TOC.mau file. An *.m3u playlist contains a list of files in the order that these files are to be played. Again, M3U authoring software is readily available, and developers of TOC.mau authoring software may elect to extract the track information from existing *.m3u playlists, rather than entering track information manually.

Table 26 – Examples of ID3 tags

	ID3	MAU	Extended MAU
Title	TALB	Album Name	Book Title
ISBN			ISBN
Author	TCOM	Songwriter	Author
Narrator	TPE1	Performer	Narrator
Genre	TCON	Genre	
Copyright	TCOP		Copyright Statement
Chapter/SubSection	TIT2	Track Name	

13.3 DAISY production options

The DAISY Consortium's specifications for Digital Talking Books (DTB) are compatible with this extension to MultiAudio for audiobooks. Both DAISY 2.02 and DAISY 3, the ANSI/NISO Z39.86-2002 can coexist within the same digital audiobook file format structure without interfering with this MultiAudio extension. A CEA-2003-C conforming player shall ignore the files it does not support. The additional XML navigation and control files present for the DAISY DTB would then be ignored.

A DAISY DTB that also contains the TOC.MAU file that conforms to this specification enables the content to be correctly rendered in either a DAISY conforming player, or a CEA certified player. The expected behavior for a DAISY conforming player would be to render the DAISY DTB normally with the full features of the DTB. If the DAISY control files are not present, the players would "fall back" to the MultiAudio file and use the audiobook extensions to provide the expected behaviors. In this way, both audiobooks and players can be both DAISY and CEA audiobook certified.

It is expected that many producers of DAISY DTB want the added compatibility that providing a conforming MultiAudio audiobook TOC.MAU file provides. All the data needed to create the MultiAudio file exists in the DAISY DTB formatted content. It is anticipated that software will be written to generate the TOC.MAU file from existing DAISY DTBs. Please look for news about this utility on the DAISY Web site in the tools section.

14 Certification

A logo has been designed to communicate compliance with the provisions of CEA-2004, and its companion standards, including CEA-2003-C. For further information concerning the logo, and provisions for its use, contact productmarks@ce.org.

Annex A (informative)

Audiobook structure examples

A.1 General

CEA-2003-A referenced the OSTA MultiAudio Audiobook Extended Data Structures Specification. Annex A contains examples of how an audiobook could be produced and structured within the MultiAudio TOC file.

The sample book in the Annex A example is titled “Audiobooks For Beginners”, and uses MP3 digital files placed on a CD-ROM as an example format, although with CEA-2003-A, any type or combination of digital files and media may be used. If this were an actual book, the table of contents of the book would look like Table A.1.

Table A.1 – Sample table of contents

“Audiobooks for Beginners” Table of Contents	
Section	Pages
Chapter 1: What is an Audiobook?	1 – 20
1.1) Description of an Audiobook	1 – 5
1.2) The OSTA MultiAudio Specification	5 – 20
Chapter 2: The Audiobook Structure	21 – 30
Chapter 3: How to use the Specification	31 – 50
3.1) How to Structure the Files on the Media	31 – 45
3.1.1) File Locations	31 – 38
3.1.2) Media Types	39 – 45
3.2) Example Uses	46 – 50

A.2 Track entry structure

If the entire book were a single audio file, it would be 5 000 000 ms (83 min). Since this is too long to manage, the example audiobook is split into 4 separate files, representing logical pieces of the audiobook’s table of contents, as shown in Table A.2.

Each of the files requires MultiAudio TOC_TrackEntry structures for navigation from file to file, as specified in Table A.2 through Table A.6.

Table A.2 – Table of contents file structure

Section	Page	File	Book Time
Chapter 1: What is an audiobook?	1 – 5	FILE 1	0 – 800 000 ms
1.1) Description of an audiobook			
1.2) The OSTA MultiAudio Specification	5 – 20	FILE 2	800 000 – 2 000 000 ms
Chapter 2: The Audiobook Structure	21 – 30	FILE 3	2 000 000 – 3 000 000 ms
Chapter 3: How to use the Specification	31 – 50	FILE 4	3 000 000 – 5 000 000 ms
3.1) How to structure the files on the media			
3.1.1) File Locations			
3.1.2) Media Types			
3.2) Example Uses			

Table A.3 – File 1 TOC_TrackEntry structure

TOC_TrackEntry (Tag.OrdinalNumber = 3)		
AB_Book field	TOC_TrackEntry field	Value
NA	Track Name	"1.1 – Description of an Audiobook"
Narrator	Performer Name	"John Smith"
Author	Songwriter Name	"Jane Doe"
Book Title	Album Name	"Audiobooks For Beginners"
AB_Level Field	Playing Time	800 000 ms
NA	Pathname	"AB For Beginners\Chapter1-1.mp3"

Table A.4 – File 2 TOC_TrackEntry structure

TOC_TrackEntry (Tag.OrdinalNumber = 6)		
AB_Book Field	TOC_TrackEntry Field	Value
NA	Track Name	"1.2 – MultiAudio Spec"
Narrator	Performer Name	"John Smith"
Author	Songwriter Name	"Jane Doe"
Book Title	Album Name	"audiobooks For Beginners"
AB_Level Field	Playing Time	1 200 000 ms
NA	Pathname	"AB For Dummies\Chapter1-2.mp3"

Table A.5 – File 3 TOC_TrackEntry structure

TOC_TrackEntry (Tag.OrdinalNumber = 5)		
AB_Book Field	TOC_TrackEntry Field	Value
NA	Track Name	"2 – The AB Structures"
Narrator	Performer Name	"John Smith"
Author	Songwriter Name	"Jane Doe"
Book Title	Album Name	"Audiobooks For Beginners"
AB_Level Field	Playing Time	1 000 000 ms
NA	Pathname	"AB For Beginners\Chapter2.mp3"

Table A.6 – File 4 TOC_TrackEntry structure

TOC_TrackEntry (Tag.OrdinalNumber = 8)		
AB_Book Field	TOC_TrackEntry Field	Value
NA	Track Name	"3 – How to Use the Spec"
Narrator	Performer Name	"John Smith"
Author	Songwriter Name	"Jane Doe"
Book Title	Album Name	"Audiobooks For Beginners"
AB_Level Field	Playing Time	2 000 000 ms
NA	Pathname	"AB For Beginners\Chapter3.mp3"

A.3 TOC_Playlist and AB_Book structures

Although their ordinal numbers are not sequential, these files need to be played sequentially to listen to the audiobook from logical start to finish. Therefore, a TOC_Playlist is needed to represent this ordered collection of files. The TOC_Playlist for this audiobook would contain the information in Table A.8.

The TOC_Playlist at ordinal number 2 in the MultiAudio TOC file would contain a single AB_Book structure located in the Extra Data field. This AB_Book structure would contain additional book information as well as the AB_Level structure, which defines how the audiobook is organized, much like a book's Table of Contents as shown in Table A.7. For this example, the AB_Book structure would contain the information in Table A.9. The AB_Book structure would be located in the TOC_Playlist extra data area.

Table A.7 – Table of contents with file time offsets

Table of Contents			
Section	Page	File	Book Time
Chapter 1: What is an Audiobook?	1 – 20	FILE 1	0 – 2 000 000 ms
1.1) Description of an Audiobook	1 – 5	FILE 1	0 – 800 000 ms
1.2) The OSTA MultiAudio Specification	5 – 20	FILE 2	800 000 – 2 000 000 ms
Chapter 2: The Audiobook Structure	21 – 30	FILE 3	2 000 000 – 3 000 000 ms
Chapter 3: How to use the Specification	31 – 50	FILE 4	3 000 000 – 5 000 000 ms
3.1) How to structure the files on the media	31 – 45	FILE 4	3 000 000 – 4 200 000 ms
3.1.1) File Locations	31 – 38	FILE 4	3 000 000 – 3 600 000 ms
3.1.2) Media Types	39 – 45	FILE 4	3 600 000 – 4 200 000 ms
3.2) Example Uses	46 – 50	FILE 4	4 200 000 – 5 000 000 ms

Table A.8 – Example TOC_Playlist structure

TOC_Playlist (Tag.OrdinalNumber = 2)		
AB_Book Field	TOC_Playlist Field	Value
Book Title	Playlist Name	"Audiobooks For Beginners"
NA	Playlist Description	"Compressed audiobook"
NA	Track Indexes	3,6,5,8
NA	Extra Data	Contains AB_Book Structure

Table A.9 – Example AB_Book structure

AB_Book		
AB_Book Field	TOC_Playlist Field	Value
Book Title	Playlist Name	"Audiobooks For Beginners"
Narrator	NA	"John Smith"
Author	NA	"Jane Doe"
audiobook Level	NA	AB_Level Structure

A.4 AB_Level structure

Each AB_Level represents a logical unit of time in the entire audiobook. An AB_Level can contain an infinite number of sub-levels that represents smaller, logical units, the parent level. These sub-levels, in turn, can represent further, smaller, sub-levels. This is much the same way a computer is organized into files/directories. The top-level directory represents the entire media as one logical unit. Sub-directories under this top-level directory represent smaller pieces of the whole. A device navigates the AB_Level structure in much the same way a computer is navigated; digging into a directory to see its contents, and going back to see the parent directories. Instead of files and directories, the AB_Level represents logical units of a book.

The top-level structure, Level Number 0, represents the entire audiobook. The sub-levels under this top-level can be chapters, sections, pages, or words. The specification does not define what these levels are called, just how they are logically structured. It is up to the publisher to determine what these levels represent, and up to the device that implements the specification to determine how the structures should be best navigated.

For this example, the top-level AB_Level structure represents the example audiobook "Audiobooks For Beginners". Each sub-level under this top-level structure represents a chapter in the book. Sub-levels may be defined as desired; chapters are used for illustration in this example book.

As Figure A.1 shows, each AB_Level structure contains smaller sub-levels that, when combined, would represent the parent level. Although the visual reference in Figure A.1 shows spacing between the start and finish of each level, there is no spacing between levels. Each sub-level at the same Level Number begins at the end of the previous sibling level. The first sub-level begins at the same point as the parent level. The last sub-level also ends at the same point in time as the parent level.

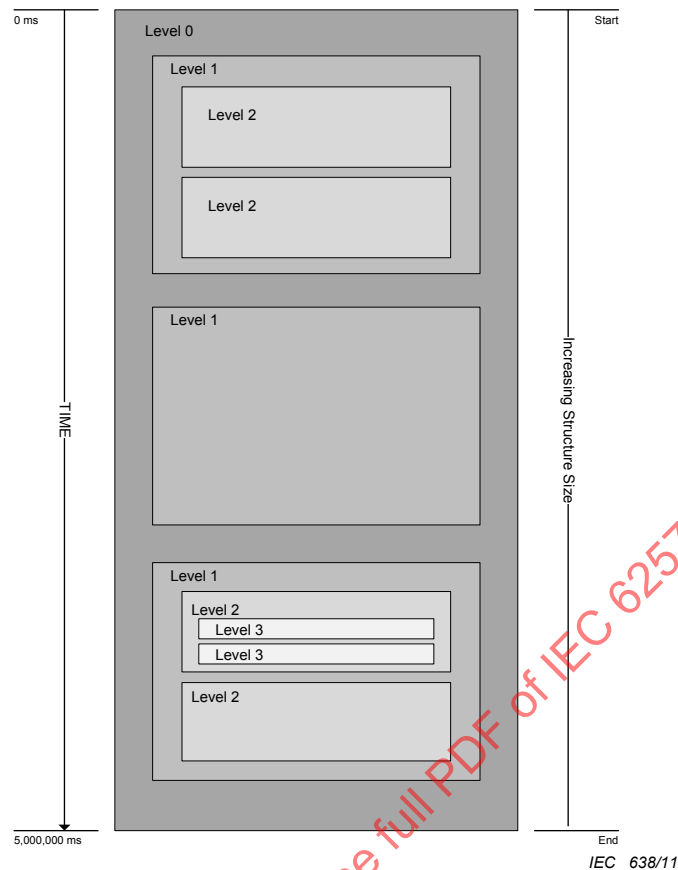


Figure A.1 – AB_Level structure organization in AB_Book variable length chunk area

A.5 AB_Level hierarchy

In Figure A.2, each AB_Level is placed according to its level number, which is designated along the top of the figure. Each horizontal band represents a separate audio file for the book, which means that each band represents a different TOC_TrackEntry structure located in the MultiAudio TOC.

In addition, each vertical band represents an AB_Level structure as shown in Tables A.10 to A.15. As shown, these TOC_TrackEntries are specified per AB_Level structure by the FTI (First TOC_TrackEntry Index) field. This field represents the first TOC_TrackEntry that the AB_Level encapsulates. Each level contains three time offsets:

- BT (Book Time Offset): Time offset from the start of the audiobook to the start of the AB_Level in milliseconds if the audiobook were a single audio file.
- FT (File Time Offset): Time offset from the start of the audio file designated by the FTI field to the start of the AB_Level in milliseconds. If an AB_Level represents a group of audio files, this value is always 0.
- TD (Time Duration): Length of time in milliseconds that the AB_Level represents.

Therefore, at any AB_Level, a device can start playing the contents of the level by using the First TOC_TrackEntry Index field to access the audio file that represents the AB_Level. It would then use the File Time Offset to jump to the start of the AB_Level within an audio file and play for the number of milliseconds designated by the Time Duration field. If the device were to bookmark a designated part of an audiobook, it would simply need to record the current Book Time Offset value for the portion of the book that is being played to access it at a later time.

Continuing to structure the audiobook presented in Table A.3, each band in Figure A.2 represents a separate AB_Level structure at a designated level. At level 0, the AB_Level represents the entire audiobook and encompasses all four of the separate audio files that make up the audiobook.

The AB_Level structures of Table A.16 through Table A.19. These AB_Level structures actually represent smaller pieces of the audio file in TOC_TrackEntry 8. They therefore represent time offsets into File 4.

IECNORM.COM : Click to view the full PDF of IEC 62571:2011

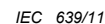


Figure A.2 – View of an audiobook AB_Level hierarchy

Table A.10 – AB_Level structure at Level 0 (audiobook)

Figure A.2 Element	AB_Level field	Field value
8.3.6 LN	8.3.6 Level Number	0
8.3.9 Name	8.3.9 Level Name	“Audiobooks For Beginners”
8.3.10 Description	8.3.10 Level Description	“OSTA MultiAudio Audiobook Extended Data Structures”
8.3.5 N_Sc	8.3.5 Number of Sub-Levels	3
8.3.8 Child Off	8.3.8 Offset to Sub-Level	See 8.3.8
8.3.7 FTI	8.3.7 First TOC_TrackEntry Index	3
8.3.11 BT	8.3.11 Book Time Offset	0 ms (Start of Book)
8.3.12 FT	8.3.12 File Time Offset	0 ms (Group of Files)
8.3.13 TD	8.3.13 Time Duration	5 000 000 ms (Entire Book)
8.3.14 Parent Off	8.3.14 Offset to Parent AB_Level	0 (Has no Parent)
8.3.15 Prev Off	8.3.15 Offset to Previous AB_Level	0 (Has no previous sibling)
8.3.16 Next Off	8.3.16 Offset to Next AB_Level	0 (Has no next sibling)

Table A.11 – AB_Level at Level 1 (chapter)

Figure A.2 Element	AB_Level field	Field value
8.3.6 LN	8.3.6 Level Number	1
8.3.9 Name	8.3.9 Level Name	“Chapter 1”
8.3.10 Description	8.3.10 Level Description	“What is an Audiobook?”
8.3.5 N_Sc	8.3.5 Number of Sub-Levels	2
8.3.8 Child Off	8.3.8 Offset to Sub-Level	See 8.3.8
8.3.7 FTI	8.3.7 First TOC_TrackEntry Index	3
8.3.11 BT	8.3.11 Book Time Offset	0 ms
8.3.12 FT	8.3.12 File Time Offset	0 ms (Group of Files)
8.3.13 TD	8.3.13 Time Duration	2 000 000 ms
8.3.14 Parent Off	8.3.14 Offset to Parent AB_Level	See 8.3.14
8.3.15 Prev Off	8.3.15 Offset to Previous AB_Level	0 (Has no previous sibling)
8.3.16 Next Off	8.3.16 Offset to Next AB_Level	See 8.3.16

Table A.12 – AB_Level at Level 1 (chapter)

Figure A.2 Element	AB_Level field	Field value
8.3.6 LN	8.3.6 Level Number	1
8.3.9 Name	8.3.9 Level Name	“Chapter 2”
8.3.10 Description	8.3.10 Level Description	“The Audiobook Structures”
8.3.5 N_Sc	8.3.5 Number of Sub-Levels	0
8.3.8 Child Off	8.3.8 Offset to Sub-Level	0 (has no sub-levels)
8.3.7 FTI	8.3.7 First TOC_TrackEntry Index	5
8.3.11 BT	8.3.11 Book Time Offset	2 000 000 ms
8.3.12 FT	8.3.12 File Time Offset	0 ms
8.3.13 TD	8.3.13 Time Duration	1 000 000 ms
8.3.14 Parent Off	8.3.14 Offset to Parent AB_Level	See 8.3.14
8.3.15 Prev Off	8.3.15 Offset to Previous AB_Level	See 8.3.15
8.3.16 Next Off	8.3.16 Offset to Next AB_Level	See 8.3.16

Table A.13 – AB_Level at Level 1 (chapter)

Figure A.2 Element	AB_Level field	Field value
8.3.6 LN	8.3.6 Level Number	1
8.3.9 Name	8.3.9 Level Name	“Chapter 3”
8.3.10 Description	8.3.10 Level Description	“How to use the Specification”
8.3.5 N_Sc	8.3.5 Number of Sub-Levels	2
8.3.8 Child Off	8.3.8 Offset to Sub-Level	See 8.3.8
8.3.7 FTI	8.3.7 First TOC_TrackEntry Index	8
8.3.11 BT	8.3.11 Book Time Offset	3 000 000 ms
8.3.12 FT	8.3.12 File Time Offset	0 ms (Group of Files)
8.3.13 TD	8.3.13 Time Duration	2 000 000 ms
8.3.14 Parent Off	8.3.14 Offset to Parent AB_Level	See 8.3.14
8.3.15 Prev Off	8.3.15 Offset to Previous AB_Level	See 8.3.15
8.3.16 Next Off	8.3.16 Offset to Next AB_Level	0 (Has no next sibling)

Table A.14 – AB_Level at Level 2 (section)

First child of Table A.11		
Figure A.2 Element	AB_Level Field	Field Value
8.3.6 LN	8.3.6 Level Number	2
8.3.9 Name	8.3.9 Level Name	"Section 1.1"
8.3.10 Description	8.3.10 Level Description	"Description of an Audiobook"
8.3.5 N_Sc	8.3.5 Number of Sub-Levels	0
8.3.8 Child Off	8.3.8 Offset to Sub-Level	0 (Has no sub-levels)
8.3.7 FTI	8.3.7 First TOC_TrackEntry Index	3
8.3.11 BT	8.3.11 Book Time Offset	0 ms
8.3.12 FT	8.3.12 File Time Offset	0 ms
8.3.13 TD	8.3.13 Time Duration	800 000 ms
8.3.14 Parent Off	8.3.14 Offset to Parent AB_Level	See 8.3.14
8.3.15 Prev Off	8.3.15 Offset to Previous AB_Level	0 (Has no previous sibling)
8.3.16 Next Off	8.3.16 Offset to Next AB_Level	See 8.3.16

Table A.15 – AB_Level at Level 2 (section)

Second Child of Table A.11		
Figure A.2 Element	AB_Level Field	Field Value
8.3.6 LN	8.3.6 Level Number	2
8.3.9 Name	8.3.9 Level Name	"Section 1.2"
8.3.10 Description	8.3.10 Level Description	"The OSTA MultiAudio Specification"
8.3.5 N_Sc	8.3.5 Number of Sub-Levels	0
8.3.8 Child Off	8.3.8 Offset to Sub-Level	0 (Has no sub-levels)
8.3.7 FTI	8.3.7 First TOC_TrackEntry Index	6
8.3.11 BT	8.3.11 Book Time Offset	800 000 ms
8.3.12 FT	8.3.12 File Time Offset	0 ms
8.3.13 TD	8.3.13 Time Duration	1 200 000 ms
8.3.14 Parent Off	8.3.14 Offset to Parent AB_Level	See 8.3.14
8.3.15 Prev Off	8.3.15 Offset to Previous AB_Level	See 8.3.15
8.3.16 Next Off	8.3.16 Offset to Next AB_Level	0 (Has no next sibling)

Table A.16 – AB_Level at Level 2 (section)

First child of Table A.13		
Figure A.2 Element	AB_Level field	Field value
8.3.6 LN	8.3.6 Level Number	2
8.3.9 Name	8.3.9 Level Name	"Section 3.1"
8.3.10 Description	8.3.10 Level Description	"How to Structure the Files on the Media"
8.3.5 N_Sc	8.3.5 Number of Sub-Levels	2
8.3.8 Child Off	8.3.8 Offset to Sub-Level	See 8.3.8
8.3.7 FTI	8.3.7 First TOC_TrackEntry Index	8
8.3.11 BT	8.3.11 Book Time Offset	3 000 000 ms
8.3.12 FT	8.3.12 File Time Offset	0 ms
8.3.13 TD	8.3.13 Time Duration	1 200 000 ms
8.3.14 Parent Off	8.3.14 Offset to Parent AB_Level	See 8.3.14
8.3.15 Prev Off	8.3.15 Offset to Previous AB_Level	0 (Has no previous sibling)
8.3.16 Next Off	8.3.16 Offset to Next AB_Level	See 8.3.16

Table A.17 – AB_Level at Level 2 (section)

Second child of Table A.13		
Figure A.2 Element	AB_Level field	Field value
8.3.6 LN	8.3.6 Level Number	2
8.3.9 Name	8.3.9 Level Name	"Section 3.2"
8.3.10 Description	8.3.10 Level Description	"Example Uses"
8.3.5 N_Sc	8.3.5 Number of Sub-Levels	0
8.3.8 Child Off	8.3.8 Offset to Sub-Level	0 (Has no sub-levels)
8.3.7 FTI	8.3.7 First TOC_TrackEntry Index	8
8.3.11 BT	8.3.11 Book Time Offset	4 200 000 ms
8.3.12 FT	8.3.12 File Time Offset	1 200 000 ms
8.3.13 TD	8.3.13 Time Duration	800 000 ms
8.3.14 Parent Off	8.3.14 Offset to Parent AB_Level	See 8.3.14
8.3.15 Prev Off	8.3.15 Offset to Previous AB_Level	See 8.3.15
8.3.16 Next Off	8.3.16 Offset to Next AB_Level	0 (Has no next sibling)

Table A.18 – AB_Level at Level 3 (subsection)

First child of Table A.16		
Figure A.2 Element	AB_Level field	Field value
8.3.6 LN	8.3.6 Level Number	3
8.3.9 Name	8.3.9 Level Name	"Subsection 3.1.1"
8.3.10 Description	8.3.10 Level Description	"File Locations"
8.3.5 N_Sc	8.3.5 Number of Sub-Levels	0
8.3.8 Child Off	8.3.8 Offset to Sub-Level	0 (Has no sub-levels)
8.3.7 FTI	8.3.7 First TOC_TrackEntry Index	8
8.3.11 BT	8.3.11 Book Time Offset	3 000 000 ms
8.3.12 FT	8.3.12 File Time Offset	0 ms
8.3.13 TD	8.3.13 Time Duration	600 000 ms
8.3.14 Parent Off	8.3.14 Offset to Parent AB_Level	See 8.3.14
8.3.15 Prev Off	8.3.15 Offset to Previous AB_Level	0 (Has no previous sibling)
8.3.16 Next Off	8.3.16 Offset to Next AB_Level	See 8.3.16

Table A.19 – AB_Level at Level 3 (subsection)

Second child of Table A.16		
Figure A.2 Element	AB_Level field	Field value
8.3.6 LN	8.3.6 Level Number	3
8.3.9 Name	8.3.9 Level Name	"Subsection 3.1.2"
8.3.10 Description	8.3.10 Level Description	"Media Types"
8.3.5 N_Sc	8.3.5 Number of Sub-Levels	0
8.3.8 Child Off	8.3.8 Offset to Sub-Level	0 (Has no sub-levels)
8.3.7 FTI	8.3.7 First TOC_TrackEntry Index	8
8.3.11 BT	8.3.11 Book Time Offset	3 600 000 ms
8.3.12 FT	8.3.12 File Time Offset	600 000 ms
8.3.13 TD	8.3.13 Time Duration	600 000 ms
8.3.14 Parent Off	8.3.14 Offset to Parent AB_Level	See 8.3.14
8.3.15 Prev Off	8.3.15 Offset to Previous AB_Level	0 (Has no previous sibling)
8.3.16 Next Off	8.3.16 Offset to Next AB_Level	See 8.3.16

Annex B (normative)

Encoding types and identifiers

Hex presentation of encoding types and identifiers are specified in Table B.1.

Table B.1 – Hex representation and interpretation

String	Hex representation	Interpretation
UNKNOWN	55 4E 4B 4E 4F 57 4E 00	Unknown encoding
MP3	4d 50 33 00	Digital Audio file compressed from PCM (RAW) form using MPEG-1, Layer 3 method
WMA	57 4d 41 00	Digital Audio file compressed from PCM (RAW) form, typically using the Windows Media Audio codec
WAV	57 41 56 00	Simple Microsoft audio file format, commonly containing uncompressed (PCM) audio
ATRAC3	41 54 52 41 43 4b 33 00	Digital Audio file compressed from PCM (RAW) form using Sony's Adaptive Transform Acoustic Coding 3
MPEG2_AAC	4d 50 45 47 32 5F 41 41 43 00	MPEG-2 Advanced Audio Coding – Successor to MPEG-1 Layer-3 compression
MPEG4_AAC	4d 50 45 47 34 5F 41 41 43 00	MPEG-4 Advanced Audio Coding – Has all the features of MPEG-2 AAC plus Perceptual Noise Substitution (PNS), a Long Term Predictor (LTP) and extensions to support scalability
RM	[[need hex representation]]	RealAudio, Digital Audio file compressed from PCM (RAW) form using a variety of different codecs
TWINVQ	54 57 49 4e 53 51 00	Transform-Domain Weighted Interleave Vector Quantization
OGG_VORBIS	4F 47 47 5F 56 4F 52 42 49 53 00	Open source, patent-free, royalty-free audio codec

Annex C (informative)

Use case and implementation examples: Spoken word audio

C.1 General

These examples are not meant to be all-inclusive or prohibit other types of spoken word presentation, but are included to give content providers and device manufacturers examples of achieving the goals of standards compliance and an outstanding spoken word audio experience for the consumer. All of the following summaries and examples of files can be downloaded for review at <http://sourceforge.net/projects/audiobook>. This site also includes a web-based program that can be used to generate a simple INDEX.AUB file.

C.2 Example 1: Book/Novel, physical media

A consumer purchases an audiobook in MP3-CD format. The content provider formats the CEA/APA audiobook format audiobook so that each file contains one chapter of the audiobook, and includes the binary toc.mau file in the root of the CD. The consumer listens to the audiobook at different times and different days, but on the same playback device.

The audiobook compatible playback device will recognize the audiobook format audiobook and enable defaults and navigational options for audiobook navigation. The player will enable auto-resume playback, allowing the consumer to stop listening to the audiobook for a period of time and return to the same point in the audiobook when playback is resumed. The device displays the chapter name and page number currently playing. The consumer is able to view information such as the book title, book author, and book narrator by changing display in some way, for example button push.

For a working sample of a binary toc.mau for Example 1, see <http://sourceforge.net/projects/audiobook>. For a sample index.aub file for Example 1, see Figure C.1. For a sample toc.mum file, see Figure C.2.

C.3 Example 2: Collection/Stories, download

A consumer has downloaded an audiobook in WMA format. The content provider provides the WMA file, which contains a collection of short stories in one file, and the audiobook format toc.aub and bookmark.aub, both XML files. The length of the file is 7 h and the content provider has provided navigational data for three levels: Story, chapter, and page. The consumer will begin listening to the spoken word audio in a vehicle with an audiobook compatible portable media device connected through USB to the in-dash receiver. By means of navigational buttons on the portable media device, the consumer will navigate from one story to another, bookmarking a favorite in order to return easily to that story another time. The consumer will stop playback of the audio in the vehicle at some point and synchronize the portable media device with an audiobook compatible home entertainment system.

The audiobook compatible portable media device will record bookmark information, such as the auto-resume point and the short story the consumer bookmarked for future reference. This information was saved in the bookmark.aub file by the portable media device. Upon synchronization with the home entertainment center, the consumer will resume listening where playback was stopped, and can return to favorite stories that were bookmarked while in the vehicle.

For a sample index.aub file for Example 2, see Figure C.3.

C.4 Example 3: Scripture/Judeo-Christian, file transfer

A consumer has purchased the spoken word audiobook format recording of the new international version bible. The content provider has formatted the 76 h of spoken word in one-hour MP3 files and provides both the binary toc.mau and XML toc.aub. The table of contents files include metadata for display and navigation by testament, book, chapter and verse.

The audiobook compatible playback software installed on the consumer's computer displays a nested series of all levels and names of those levels in a familiar, table of contents presentation. The consumer may begin play anywhere within the spoken word audio, search for a particular testament, book, chapter, or verse, and bookmark favorites for easy playback, or choose to listen to the collection from beginning to end with auto-resume capability whenever playback is stopped.

For a working sample of a binary toc.mau for Example 3, see <http://sourceforge.net/projects/audiobook>. For a sample index.aub file for Example 3, see Figure C.4.

C.5 Example 4: Periodical/Newspaper, wireless subscription

A consumer has a newspaper podcast subscription delivered in MP3 format automatically each day to his Bluetooth-enabled PDA. The consumer listens to each morning's broadcast during commute in a vehicle equipped with an audiobook compatible stereo head unit which wirelessly connects through Bluetooth to the audiobook compatible PDA. The content provider has formatted the XML toc.aub with two levels of navigation, consisting of article, title and paragraph number. Additional information regarding the podcast spoken word audio, such as newspaper name, issue date, and editor name can be displayed on the PDA by choosing an "Info" button. The PDA passes information regarding the toc.aub to the head unit, which by default displays the article title and author's name on a two-line display. Interaction from the consumer (button push) can change the display to show paragraph number or issue date.

On arrival at destination, the consumer will use a Bluetooth headset to continue listening to the newspaper podcast, displaying information and navigating articles by interacting with the PDA. If the PDA is powered down or playback is stopped before the podcast is complete, the audiobook compatible PDA will enable auto-resume features to restart where the consumer stopped playback.

For a sample index.aub file for Example 4, see Figure C.5.

```

<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest xmlns:file="http://ns.osta.org/manifest/1.0/"
xmlns:ab="http://ns.ce.org/audiobook/2003b/"
xmlns:mpv="http://ns.osta.org/mpv/1.0/"
xmlns:mpvp="http://ns.osta.org/mpv/presentation/1.0/"
xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://http://purl.org/dc/terms/"
xmlns:nmf="http://ns.osta.org/nmf/1.0/">
  <mpv:AssetList>
    <mpv:Seq>
      <mpv:Audio>
        <ab:AudioBookProperties>
          <ab:unitType>Book</ab:unitType>
          <ab:unitContext>book/novel</ab:unitContext>
          <ab:bookISBN>UC-0001-000</ab:bookISBN>
          <ab:bookAuthor>Joe Bestseller</ab:bookAuthor>
          <ab:bookNarrator>Norma Dialects</ab:bookNarrator>
          <ab:bookTitle>Use Case Example 1</ab:bookTitle>
        </ab:AudioBookProperties>
        <nmf:Metadata>
          <Properties xmlns="http://purl.org/dc/elements/1.1">
            <contributor>Joe Bestseller</contributor>
            <creator>Joe Bestseller</creator>
            <format>audio/mpeg</format>
            <title>Use Case Example 1</title>
          </Properties>
          <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
            <AlbumTitle>Use Case Example 1</AlbumTitle>
            <Genre>Thriller</Genre>
            <PlayingTime>16200</PlayingTime>
          </MusicProperties>
        </nmf:Metadata>
      </mpv:Audio>
    </mpv:Seq>
  </mpv:AssetList>
<!--Chapter 1-->
  <mpv:Seq>
    <mpv:Audio>
      <ab:unitContext>book/novel</ab:unitContext>
      <ab:unitType>chapter</ab:unitType>
      <nmf:Metadata>
        <Properties
xmlns="http://purl.org/dc/elements/1.1">
          <format>audio/mpeg</format>
          <title>Chapter 1</title>
          <description>Chapter
1</description>
        </Properties>
        <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
          <AlbumTitle>Use Case Example 1</AlbumTitle>
          <Genre>spoken word</Genre>
          <PlayingTime>2700</PlayingTime>
        </MusicProperties>
      </nmf:Metadata>
      <mpv:lastURL>C:\chapter1.mp3</mpv:lastURL>
      <ab:timeOffset>0000</ab:timeOffset>
    </mpv:Audio>
  </mpv:Seq>
<!--Begin Pages-->

```

```

<mpv:Seq>
  <mpv:Audio>
    <ab:unitType>page</ab:unitType>
    <nmf:Metadata>
      <Properties xmlns="http://purl.org/dc/elements/1.1">
        <format>audio/mpeg</format>
        <title>Page
1</title>
        <description>Page 1</description>
      </Properties>
      <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
        <PlayingTime>60</PlayingTime>
      </MusicProperties>
    </nmf:Metadata>
    <mpv:lastURL>C:\chapter1.mp3</mpv:lastURL>
    <ab:timeOffset>0000</ab:timeOffset>
  </mpv:Audio>
  <mpv:Audio>
    <ab:unitType>page</ab:unitType>
    <nmf:Metadata>
      <Properties xmlns="http://purl.org/dc/elements/1.1">
        <format>audio/mpeg</format>
        <title>Page
2</title>
        <description>Page 2</description>
      </Properties>
      <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
        <PlayingTime>120</PlayingTime>
      </MusicProperties>
    </nmf:Metadata>
    <mpv:lastURL>C:\chapter1.mp3</mpv:lastURL>
    <ab:timeOffset>60</ab:timeOffset>
  </mpv:Audio>
  <mpv:Audio>
    <ab:unitType>page</ab:unitType>
    <nmf:Metadata>
      <Properties xmlns="http://purl.org/dc/elements/1.1">
        <format>audio/mpeg</format>
        <title>Page
3</title>
        <description>Page 2</description>
      </Properties>
      <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
        <PlayingTime>60</PlayingTime>
      </MusicProperties>
    </nmf:Metadata>
    <mpv:lastURL>C:\chapter1.mp3</mpv:lastURL>
    <ab:timeOffset>180</ab:timeOffset>
  </mpv:Audio>

<!--Chapter 2-->

<mpv:Seq>
  <mpv:Audio>
    <ab:unitContext>book/novel</ab:unitContext>

```

```

        <ab:unitType>chapter</ab:unitType>
        <nmf:Metadata>
            <Properties
xmlns="http://purl.org/dc/elements/1.1">
                <format>audio/mpeg</format>
                <title>Tom Plays, Fights, and
Hides</title>
                <description>Chapter
2</description>
            </Properties>
            <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
                <AlbumTitle>Book Title</AlbumTitle>
                <Genre>spoken word</Genre>
                <PlayingTime>3000</PlayingTime>
            </MusicProperties>
        </nmf:Metadata>
        <mpv:lastURL>C:\chapter2.mp3</mpv:lastURL>
        <ab:timeOffset>240</ab:timeOffset>

<!--Begin Pages-->

        <mpv:Seq>
            <mpv:Audio>
                <ab:unitType>page</ab:unitType>
                <nmf:Metadata>
                    <Properties xmlns="http://purl.org/dc/elements/1.1">
                        <format>audio/mpeg</format>
                        <title>Page
4</title>
                        <description>Page 4</description>
                    </Properties>
                    <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
                        <PlayingTime>60</PlayingTime>
                    </MusicProperties>
                    <nmf:Metadata>
                        <mpv:lastURL>C:\chapter2.mp3</mpv:lastURL>
                        <ab:timeOffset>0000</ab:timeOffset>
                    </mpv:Audio>
                    <mpv:Audio>
                        <ab:unitType>page</ab:unitType>
                        <nmf:Metadata>
                            <Properties xmlns="http://purl.org/dc/elements/1.1">
                                <format>audio/mpeg</format>
                                <title>Page
5</title>
                                <description>Page 5</description>
                            </Properties>
                            <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
                                <PlayingTime>60</PlayingTime>
                            </MusicProperties>
                            <nmf:Metadata>
                                <mpv:lastURL>C:\chapter2.mp3</mpv:lastURL>
                                <ab:timeOffset>60</ab:timeOffset>
                            </mpv:Audio>
                            <mpv:Audio>
                                <ab:unitType>page</ab:unitType>

```

```

<nmf:Metadata>
  <Properties xmlns="http://purl.org/dc/elements/1.1">
    <format>audio/mpeg</format>
    <title>Page
6</title>

    <description>Page 6</description>
  </Properties>
  <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
    <PlayingTime>60</PlayingTime>
  </MusicProperties>
</nmf:Metadata>

  <mpv:lastURL>C:\chapter2.mp3</mpv:lastURL>
    <ab:timeOffset>120</ab:timeOffset>
    </mpv:Audio>
  </mpv:Seq>
</mpv:Audio>
  </mpv:Seq>
</mpv:Seq>
  </mpv:Audio>
  </mpv:Seq>
</mpv:Audio>
  </mpv:Seq>
</mpv:AssetList>
</file:Manifest>

```

IEC 640/11

Figure C.1 – Example 1 INDEX.AUB

```

<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest
xmlns:file="http://ns.osta.org/manifest/1.0/"
xmlns:ab="http://ns.ce.org/audiobook/2003b"
xmlns:mpv="http://ns.osta.org/mpv/1.0/"
xmlns:mpvp="http://ns.osta.org/mpv/presentation/1.0/"
xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://purl.org/dc/terms/"
xmlns:nmf="http://ns.osta.org/nmf/1.0/">
<nmf:Metadata>
<ManifestProperties xmlns="http://ns.osta.org/manifest/1.0/">
  <ProfileBag>
    <Profile>http://ns.ce.org/audiobook/2003b/</Profile>
    <Profile>http://ns.ce.org/mpv/presentation/1.0/</Profile>
    <Profile>http://ns.ce.org/mpv/music/1.0/</Profile>
  </ProfileBag>
</ManifestProperties>
  <Properties xmlns="http://purl.org/dc/terms/">
    <Created>4/18/2006 12:45:01 PM</Created>
    <Modified>4/18/2006 12:45:01 PM</Modified>
  </Properties>
</nmf:Metadata>
<mpvp:Album>
  <nmf:Metadata>
    <dc:Properties>
      <dc:creator>Joe Bestseller</dc:creator>
      <dc:identifier>UC-0001-000</dc:identifier>
      <dc:rights>2006</dc:rights>
      <dc:title>Use Case Example 1</dc:title>
      <dc:contributor>Norma Dialects</dc:contributor>
    </dc:Properties>
  </nmf:Metadata>
</mpvp:Album>
</file:Manifest>

```

IEC 641/11

Figure C.2 – Example 1 TOC.MUM

```

<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest xmlns:file="http://ns.osta.org/manifest/1.0/"
xmlns:ab="http://ns.ce.org/audiobook/2003b/"
xmlns:mpv="http://ns.osta.org/mpv/1.0/"
xmlns:mpvp="http://ns.osta.org/mpv/presentation/1.0/"
xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://http://purl.org/dc/terms/"
xmlns:nmf="http://ns.osta.org/nmf/1.0/">
  <mpv:AssetList>
    <mpv:Seq>
      <mpv:Audio>
        <ab:AudioBookProperties>
          <ab:unitType>Collection</ab:unitType>
          <ab:unitContext>collection/stories</ab:unitContext>
          <ab:bookISBN>UC-0002-000</ab:bookISBN>
          <ab:bookAuthor>Joe Bestseller</ab:bookAuthor>
          <ab:bookNarrator>Norma Dialects</ab:bookNarrator>
          <ab:bookTitle>Use Case Example 2</ab:bookTitle>
        </ab:AudioBookProperties>
        <ab:unitType>Collection</ab:unitType>
        <ab:unitContext>collection/stories</ab:unitContext>
        <nmf:Metadata>
          <Properties xmlns="http://purl.org/dc/elements/1.1">
            <contributor>Norma Dialects</contributor>
            <creator>Joe Bestseller</creator>
            <format>audio/wma</format>
            <title>Use Case Example 2</title>
          </Properties>
          <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
            <AlbumTitle>Use Case Example 2</AlbumTitle>
            <Genre>Suspense</Genre>
            <PlayingTime>420</PlayingTime>
          </MusicProperties>
        </nmf:Metadata>
      </mpv:Audio>
    </mpv:Seq>
  </mpv:AssetList>
<!--Story 1-->
  <mpv:Seq>
    <mpv:Audio>
      <ab:unitType>story</ab:unitType>
      <nmf:Metadata>
        <Properties
xmlns="http://purl.org/dc/elements/1.1">
          <format>audio/wma</format>
          <title>Story Title</title>
          <description>Story
Title</description>
        </Properties>
      </nmf:Metadata>
      <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
        <AlbumTitle>Collection Title</AlbumTitle>
        <Genre>Collection/Stories</Genre>
        <PlayingTime>900</PlayingTime>
      </MusicProperties>
    </mpv:Audio>
  </mpv:Seq>
  <mpv:lastURL>C:\story.wma</mpv:lastURL>
<ab:timeOffset>0000</ab:timeOffset>

```

<!--Chapter 1-->

```

    <mpv:Seq>
      <mpv:Audio>

    <ab:unitType>chapter</ab:unitType>
      <nmf:Metadata>
        <Properties xmlns="http://purl.org/dc/elements/1.1">
          <format>audio/wma</format>
          <title>Chapter
1</title>
          <description>Chapter 1</description>
        </Properties>
        <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
          <PlayingTime>300</PlayingTime>
        </MusicProperties>
      </nmf:Metadata>
      <mpv:lastURL>C:\story.wma</mpv:lastURL>
      <ab:timeOffset>0000</ab:timeOffset>
    </mpv:Audio>

<!--Pages-->

    <mpv:Seq>
      <mpv:Audio>
        <ab:unitType>page</ab:unitType>
        <nmf:Metadata>
          <Properties xmlns="http://purl.org/dc/elements/1.1">
            <format>audio/wma</format>
            <title>Page
1</title>
            <description>Chapter 1</description>
          </Properties>
          <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
            <PlayingTime>60</PlayingTime>
          </MusicProperties>
        </nmf:Metadata>
        <mpv:lastURL>C:\story.wma</mpv:lastURL>
        <ab:timeOffset>0000</ab:timeOffset>
      </mpv:Audio>
      <mpv:Audio>
        <ab:unitType>page</ab:unitType>
        <nmf:Metadata>
          <Properties xmlns="http://purl.org/dc/elements/1.1">
            <format>audio/wma</format>
            <title>Page
2</title>
            <description>page 2</description>
          </Properties>
          <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
            <PlayingTime>60</PlayingTime>
          </MusicProperties>
        </nmf:Metadata>
        <mpv:lastURL>C:\story.wma</mpv:lastURL>
        <ab:timeOffset>0060</ab:timeOffset>
      </mpv:Audio>
    </mpv:Audio>
  
```

```

<ab:unitType>page</ab:unitType>
<nmf:Metadata>
  <Properties xmlns="http://purl.org/dc/elements/1.1">
    <format>audio/wma</format>
    <title>Page
3</title>
    <description>Page 3</description>
  </Properties>
  <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
    <PlayingTime>60</PlayingTime>
  </MusicProperties>
  </nmf:Metadata>
  <mpv:lastURL>C:\story.wma</mpv:lastURL>
    <ab:timeOffset>0120</ab:timeOffset>
  </mpv:Audio>
  <mpv:Audio>
    <ab:unitType>page</ab:unitType>
  <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
      <format>audio/wma</format>
      <title>Page
4</title>
      <description>Page 4</description>
    </Properties>
    <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
      <PlayingTime>60</PlayingTime>
    </MusicProperties>
    </nmf:Metadata>
    <mpv:lastURL>C:\story.wma</mpv:lastURL>
      <ab:timeOffset>180</ab:timeOffset>
    </mpv:Audio>
    <mpv:Audio>
      <ab:unitType>page</ab:unitType>
    <nmf:Metadata>
      <Properties xmlns="http://purl.org/dc/elements/1.1">
        <format>audio/wma</format>
        <title>Page
5</title>
        <description>Page 5</description>
      </Properties>
      <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
        <PlayingTime>60</PlayingTime>
      </MusicProperties>
      </nmf:Metadata>
      <mpv:lastURL>C:\story.wma</mpv:lastURL>
        <ab:timeOffset>240</ab:timeOffset>
      </mpv:Audio>
<!--Story 2-->
    <mpv:Seq>
    <mpv:Audio>
      <ab:unitType>story</ab:unitType>
    <nmf:Metadata>
      <Properties
xmlns="http://purl.org/dc/elements/1.1">
        <format>audio/wma</format>

```

```

<title>Story Title</title>
<description>Story
Title</description>
</Properties>
<MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
  <AlbumTitle>Collection Title</AlbumTitle>
  <Genre>Collection/Stories</Genre>
  <PlayingTime>900</PlayingTime>
</MusicProperties>
</nmf:Metadata>
<mpv:lastURL>C:\story.wma</mpv:lastURL>
<ab:timeOffset>300</ab:timeOffset>
</mpv:Audio>

<!--Chapter 2-->
  <mpv:Seq>
  <mpv:Audio>
    <ab:unitType>chapter</ab:unitType>
  <nmf:Metadata>
  <Properties xmlns="http://purl.org/dc/elements/1.1">
    <format>audio/wma</format>
    <title>Chapter 1</title>
    <description>Chapter 1</description>
  </Properties>
  <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
    <PlayingTime>300</PlayingTime>
  </MusicProperties>
  </nmf:Metadata>
  <mpv:lastURL>C:\story.wma</mpv:lastURL>
  <ab:timeOffset>360</ab:timeOffset>
</mpv:Audio>

<!--Pages-->
  <mpv:Seq>
  <mpv:Audio>
    <ab:unitType>page</ab:unitType>
  <nmf:Metadata>
  <Properties xmlns="http://purl.org/dc/elements/1.1">
    <format>audio/wma</format>
    <title>Page 1</title>
    <description>page 1</description>
  </Properties>
  <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
    <PlayingTime>60</PlayingTime>
  </MusicProperties>
  </nmf:Metadata>
  <mpv:lastURL>C:\story.wma</mpv:lastURL>
  <ab:timeOffset>420</ab:timeOffset>
  </mpv:Audio>
  <mpv:Audio>
    <ab:unitType>page</ab:unitType>
  <nmf:Metadata>
  <Properties xmlns="http://purl.org/dc/elements/1.1">
    <format>audio/wma</format>
    <title>Page 2</title>
    <description>Page 2</description>
  </Properties>

```

```

<MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
  <PlayingTime>60</PlayingTime>
</MusicProperties>
</nmf:Metadata>

<mpv:lastURL>C:\story.wma</mpv:lastURL>
  <ab:timeOffset>480</ab:timeOffset>
</mpv:Audio>
  <mpv:Audio>
    <ab:unitType>page</ab:unitType>
  <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
      <format>audio/wma</format>
      <title>Page
3</title>
      <description>Page 3</description>
    </Properties>
    <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
      <PlayingTime>60</PlayingTime>
    </MusicProperties>
    </nmf:Metadata>

    <mpv:lastURL>C:\story.wma</mpv:lastURL>
      <ab:timeOffset>540</ab:timeOffset>
    </mpv:Audio>
    </mpv:Seq>
    </mpv:Seq>
    </mpv:Seq>
    </mpv:Seq>
    </mpv:Seq>
    </mpv:Seq>
    </mpv:Audio>
    </mpv:Seq>
    </mpv:Audio>
    </mpv:Seq>
  </mpv:AssetList>
</file:Manifest>

```

IEC 642/11

Figure C.3 – Example 2 INDEX.AUB

```

<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest xmlns:file="http://ns.osta.org/manifest/1.0/"
xmlns:ab="http://ns.ce.org/audiobook/2003b/"
xmlns:mpv="http://ns.osta.org/mpv/1.0/"
xmlns:mpvp="http://ns.osta.org/mpv/presentation/1.0/"
xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://http://purl.org/dc/terms/"
xmlns:nmf="http://ns.osta.org/nmf/1.0/">
  <mpv:AssetList>
    <mpv:Seq>
      <mpv:Audio>
        <ab:AudioBookProperties>
          <ab:unitType>Scripture</ab:unitType>
          <ab:unitContext>scripture/judeo-
christian</ab:unitContext>
          <ab:bookISBN>UC-0003-000</ab:bookISBN>
          <ab:bookAuthor>various</ab:bookAuthor>
          <ab:bookNarrator>Norma Dialects</ab:bookNarrator>
          <ab:bookTitle>Use Case Example 3</ab:bookTitle>
          <ab:unitType>Scripture</ab:unitType>
        </ab:AudioBookProperties>
        <ab:unitContext>Scripture/Judeo-Christian</ab:unitContext>
        <nmf:Metadata>
          <Properties xmlns="http://purl.org/dc/elements/1.1">
            <contributor>Norma Dialects</contributor>
            <creator>Zondervan</creator>
            <format>audio/mpeg</format>
            <title>New International Version Bible</title>
          </Properties>
          <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
            <AlbumTitle>New      International      Version
Bible</AlbumTitle>
            <Genre>spoken word</Genre>
            <PlayingTime>4560</PlayingTime>
          </MusicProperties>
        </nmf:Metadata>
      </mpv:Audio>
    </mpv:Seq>
  </mpv:AssetList>
<!--Testament-->
  <mpv:Seq>
    <mpv:Audio>
      <ab:unitType>scripture/testament</ab:unitType>
      <nmf:Metadata>
        <Properties
xmlns="http://purl.org/dc/elements/1.1">
          <format>audio/mpeg</format>
          <title>Old Testamanet</title>
          <description>Old
Testamanet</description>
        </Properties>
        <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
          <AlbumTitle>Old Testamanet</AlbumTitle>
          <Genre>scripture/judeo-
christian</Genre>
          <PlayingTime>2280</PlayingTime>
        </MusicProperties>
      </nmf:Metadata>
    </mpv:Audio>
  </mpv:Seq>

```

```

<mpv:lastURL>C:\oldtestament1.mp3</mpv:lastURL>
    <ab:timeOffset>0000</ab:timeOffset>

<!--Book-->

    <mpv:Seq>
        <mpv:Audio>
            <ab:unitType>book</ab:unitType>
            <nmf:Metadata>
                <Properties xmlns="http://purl.org/dc/elements/1.1">
                    <format>audio/mpeg</format>

                    <title>Genesis</title>

                    <description>Book 1 - Genesis</description>
                </Properties>
                <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
                    <PlayingTime>120</PlayingTime>
                </MusicProperties>
                </nmf:Metadata>

                <mpv:lastURL>C:\oldtestament1.mp3</mpv:lastURL>
                    <ab:timeOffset>0000</ab:timeOffset>
                </mpv:Audio>

<!--Chapter-->

    <mpv:Seq>
        <mpv:Audio>
            <ab:unitType>chapter</ab:unitType>
            <nmf:Metadata>
                <Properties xmlns="http://purl.org/dc/elements/1.1">
                    <format>audio/mpeg</format>
                    <title>Chapter
1</title>

                    <description>Chapter 1</description>
                </Properties>
                <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
                    <PlayingTime>120</PlayingTime>
                </MusicProperties>
                </nmf:Metadata>

                <mpv:lastURL>C:\oldtestament1.mp3</mpv:lastURL>
                    <ab:timeOffset>0000</ab:timeOffset>
                </mpv:Audio>

<!--Verse-->

    <mpv:Seq>
        <mpv:Audio>
            <ab:unitType>verse</ab:unitType>
            <nmf:Metadata>
                <Properties xmlns="http://purl.org/dc/elements/1.1">
                    <format>audio/mpeg</format>
                    <title>verse
1</title>

                    <description>verse 1</description>
                </Properties>

```

```

<MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
  <PlayingTime>60</PlayingTime>
</MusicProperties>
</nmf:Metadata>

<mpv:lastURL>C:\oldtestament1.mp3</mpv:lastURL>
  <ab:timeOffset>0</ab:timeOffset>
</mpv:Audio>
<mpv:Audio>

<ab:unitType>verse</ab:unitType>
  <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
      <format>audio/mpeg</format>
      <title>verse
2</title>
      <description>verse 2</description>
    </Properties>
  </MusicProperties>
  <PlayingTime>60</PlayingTime>
</MusicProperties>
</nmf:Metadata>

<mpv:lastURL>C:\oldtestament1.mp3</mpv:lastURL>
  <ab:timeOffset>60</ab:timeOffset>
</mpv:Audio>
<!--Book-->

  <mpv:Seq>
    <mpv:Audio>
      <ab:unitType>book</ab:unitType>
      <nmf:Metadata>
        <Properties xmlns="http://purl.org/dc/elements/1.1">
          <format>audio/mpeg</format>
          <title>Exodus</title>
          <description>Book 2 - Exodus</description>
        </Properties>
      </MusicProperties>
      <PlayingTime>120</PlayingTime>
    </MusicProperties>
    </nmf:Metadata>

    <mpv:lastURL>C:\oldtestament2.mp3</mpv:lastURL>
      <ab:timeOffset>0</ab:timeOffset>
    </mpv:Audio>

  <!--Chapter-->

    <mpv:Seq>
      <mpv:Audio>
        <ab:unitType>chapter</ab:unitType>
        <nmf:Metadata>
          <Properties xmlns="http://purl.org/dc/elements/1.1">
            <format>audio/mpeg</format>

```

Chapter 1</title>

<title>Exodus

<description>Chapter 1</description>

</Properties>

<MusicProperties

xmlns="http://ns.osta.org/mpv/music/1.0">

<PlayingTime>120</PlayingTime>

</MusicProperties>

</nmf:Metadata>

<mpv:lastURL>C:\oldtestament2.mp3</mpv:lastURL>

<ab:timeOffset>0000</ab:timeOffset>

</mpv:Audio>

<!--Verse-->

<mpv:Seq>

<mpv:Audio>

<ab:unitType>verse</ab:unitType>

<nmf:Metadata>

<Properties xmlns="http://purl.org/dc/elements/1.1">

<format>audio/mpeg</format>

<title>verse

1</title>

<description>verse 1</description>

</Properties>

<MusicProperties

xmlns="http://ns.osta.org/mpv/music/1.0">

<PlayingTime>60</PlayingTime>

</MusicProperties>

</nmf:Metadata>

<mpv:lastURL>C:\oldtestament2.mp3</mpv:lastURL>

<ab:timeOffset>0</ab:timeOffset>

</mpv:Audio>

<mpv:Audio>

<ab:unitType>verse</ab:unitType>

<nmf:Metadata>

<Properties xmlns="http://purl.org/dc/elements/1.1">

<format>audio/mpeg</format>

<title>verse

2</title>

<description>verse 2</description>

</Properties>

<MusicProperties

xmlns="http://ns.osta.org/mpv/music/1.0">

<PlayingTime>60</PlayingTime>

</MusicProperties>

</nmf:Metadata>

```
<mpv:lastURL>C:\oldtestament2.mp3</mpv:lastURL>
      <ab:timeOffset>60</ab:timeOffset>
      </mpv:Audio>
      </mpv:Seq>
      </mpv:Seq>
      </mpv:Seq>
      </mpv:Seq>
      </mpv:Seq>
      </mpv:Seq>
      </mpv:Seq>
      </mpv:Audio>
      </mpv:Seq>
      </mpv:Audio>
      </mpv:Seq>
      </mpv:AssetList>
</file:Manifest>
```

IEC 643/11

Figure C.4 – Example 3 INDEX.AUB

IECNORM.COM : Click to view the full PDF of IEC 62571:2011

```

<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest xmlns:file="http://ns.osta.org/manifest/1.0/"
xmlns:ab="http://ns.ce.org/audiobook/2003b/"
xmlns:mpv="http://ns.osta.org/mpv/1.0/"
xmlns:mpvp="http://ns.osta.org/mpv/presentation/1.0/"
xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://http://purl.org/dc/terms/"
xmlns:nmf="http://ns.osta.org/nmf/1.0">
  <mpv:AssetList>
    <mpv:Seq>
      <mpv:Audio>
        <ab:AudioBookProperties>
          <ab:unitType>newspaper</ab:unitType>
          <ab:unitContext>periodical/newspaper</ab:unitContext>
          <ab:bookISBN>UC-0004-000</ab:bookISBN>
          <ab:bookAuthor>My Town Newspaper</ab:bookAuthor>
          <ab:bookNarrator>Norma Dialects</ab:bookNarrator>
          <ab:bookTitle>Use Case Example 4</ab:bookTitle>
          <ab:unitType>newspaper</ab:unitType>
        </ab:AudioBookProperties>
        <ab:unitContext>periodical/newspaper</ab:unitContext>
        <nmf:Metadata>
          <Properties xmlns="http://purl.org/dc/elements/1.1">
            <contributor>Norma Dialects</contributor>
            <creator>Newspaper Publisher</creator>
            <format>audio/mpeg</format>
            <title>The New York Times</title>
          </Properties>
          <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
            <AlbumTitle>The New York Times</AlbumTitle>
            <Genre>spoken word</Genre>
            <PlayingTime>4560</PlayingTime>
          </MusicProperties>
        </nmf:Metadata>
      </mpv:Audio>
    </mpv:Seq>
  </mpv:AssetList>
<!--News Paper-->
  <mpv:Seq>
    <mpv:Audio>
      <ab:unitType>periodical/newspaper</ab:unitType>
      <nmf:Metadata>
        <Properties
xmlns="http://purl.org/dc/elements/1.1">
          <format>audio/mpeg</format>
          <title>The New York Times</title>
          <description>The New York
Times</description>
        </Properties>
        <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
          <AlbumTitle>The New York Times</AlbumTitle>
          <Genre>periodical/newspaper</Genre>
          <PlayingTime>2280</PlayingTime>
        </MusicProperties>
      </nmf:Metadata>
      <mpv:lastURL>C:\nytimes.mp3</mpv:lastURL>
    </mpv:Audio>
  </mpv:Seq>
</mpv:AssetList>

```

```

<ab:timeOffset>0000</ab:timeOffset>

<!--Article-->

    <mpv:Seq>
        <mpv:Audio>

            <ab:unitType>article</ab:unitType>
                <nmf:Metadata>
                    <Properties xmlns="http://purl.org/dc/elements/1.1">
                        <format>audio/mpeg</format>
                        <title>Audiobook
Industry Finalizes Standards</title>
                        <description>Audiobook Industry Finalizes
Standards - Wednesday, April 26th 2006</description>
                        <ab:bookAuthor>Clark Kent</ab:bookAuthor>
                    </Properties>
                    <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
                        <PlayingTime>600</PlayingTime>
                    </MusicProperties>
                    </nmf:Metadata>

                    <mpv:lastURL>C:\nytimes1.mp3</mpv:lastURL>
                        <ab:timeOffset>0000</ab:timeOffset>
                    </mpv:Audio>

<!--Paragraph-->

    <mpv:Seq>
        <mpv:Audio>

            <ab:unitType>paragraph</ab:unitType>
                <nmf:Metadata>
                    <Properties xmlns="http://purl.org/dc/elements/1.1">
                        <format>audio/mpeg</format>
                        <title>Paragraph
1</title>
                        <description>Paragraph 1</description>
                    </Properties>
                    <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
                        <PlayingTime>120</PlayingTime>
                    </MusicProperties>
                    </nmf:Metadata>

                    <mpv:lastURL>C:\nytimes.mp3</mpv:lastURL>
                        <ab:timeOffset>0000</ab:timeOffset>
                    </mpv:Audio>

                <mpv:Audio>

            <ab:unitType>paragraph</ab:unitType>
                <nmf:Metadata>
                    <Properties xmlns="http://purl.org/dc/elements/1.1">
                        <format>audio/mpeg</format>
                        <title>Paragraph
2</title>
                        <description>Paragraph 2</description>
                    </Properties>
                    <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
                        <PlayingTime>120</PlayingTime>
                    </MusicProperties>

```

```

</nmf:Metadata>

<mpv:lastURL>C:\nytimes.mp3</mpv:lastURL>
    <ab:timeOffset>120</ab:timeOffset>
    </mpv:Audio>
    <mpv:Audio>

    <ab:unitType>paragraph</ab:unitType>
    <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
        <format>audio/mpeg</format>
        <title>Paragraph
3</title>
        <description>Paragraph 3</description>
    </Properties>
    <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
        <PlayingTime>120</PlayingTime>
    </MusicProperties>
    </nmf:Metadata>

    <mpv:lastURL>C:\nytimes.mp3</mpv:lastURL>
    <ab:timeOffset>240</ab:timeOffset>

<!--Article-->

    <mpv:Seq>
    <mpv:Audio>

    <ab:unitType>article</ab:unitType>
    <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
        <format>audio/mpeg</format>
        <title>This is the
Second Article</title>
        <description>This is the Second Article -
Wednesday, April 26th 2006</description>
        <ab:bookAuthor>Lois Lane</ab:bookAuthor>
    </Properties>
    <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
        <PlayingTime>600</PlayingTime>
    </MusicProperties>
    </nmf:Metadata>

    <mpv:lastURL>C:\nytimes1.mp3</mpv:lastURL>
    <ab:timeOffset>240</ab:timeOffset>
    </mpv:Audio>

<!--Paragraph-->

    <mpv:Seq>
    <mpv:Audio>

    <ab:unitType>paragraph</ab:unitType>
    <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
        <format>audio/mpeg</format>
        <title>Paragraph
1</title>
        <description>Paragraph 1</description>

```

```

</Properties>
<MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
  <PlayingTime>120</PlayingTime>
</MusicProperties>
</nmf:Metadata>

<mpv:lastURL>C:\nytimes1.mp3</mpv:lastURL>
  <ab:timeOffset>360</ab:timeOffset>
</mpv:Audio>
<mpv:Audio>

<ab:unitType>paragraph</ab:unitType>
  <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
      <format>audio/mpeg</format>
      <title>Paragraph
2</title>
      <description>Paragraph 2</description>
    </Properties>
    <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
      <PlayingTime>120</PlayingTime>
    </MusicProperties>
    </nmf:Metadata>

    <mpv:lastURL>C:\nytimes1.mp3</mpv:lastURL>
      <ab:timeOffset>480</ab:timeOffset>
    </mpv:Audio>
    <mpv:Audio>

    <ab:unitType>paragraph</ab:unitType>
      <nmf:Metadata>
        <Properties xmlns="http://purl.org/dc/elements/1.1">
          <format>audio/mpeg</format>
          <title>Paragraph
3</title>
          <description>Paragraph 3</description>
        </Properties>
        <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
          <PlayingTime>120</PlayingTime>
        </MusicProperties>
        </nmf:Metadata>

        <mpv:lastURL>C:\nytimes1.mp3</mpv:lastURL>
          <ab:timeOffset>600</ab:timeOffset>
        </mpv:Audio>
        </mpv:Seq>
        </mpv:Seq>
        </mpv:Audio>
        </mpv:Seq>
        </mpv:Seq>
        </mpv:Audio>
        </mpv:Seq>
        </mpv:Audio>
      </mpv:Seq>
    </mpv:AssetList>
  </file:Manifest>

```

Figure C.5 – Example 4 INDEX.AUB

Bibliography

ISO/IEC 11172-3:1993, *Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s – Part 3: Audio*

ISO/IEC 14496-3:2009, *Information technology – Coding of audio-visual objects – Part 3: Audio*

ISO/IEC 13818-7:2006, *Information technology – Generic coding of moving pictures and associated audio information – Part 7: Advanced Audio Coding (AAC)*

ANSI/CEA-2003-C, *Digital Audiobook File Format and Player Requirements*

ID3 Specifications, see <http://www.id3.org/>

ANSI/NISO Z39.86-2002, *Specifications for the Digital Talking Book*; Daisy Consortium

ID3 Version 2.3, see <http://www.id3.org/>

IECNORM.COM : Click to view the full PDF of IEC 62571:2011

IECNORM.COM : Click to view the full PDF of IEC 62571:2017

SOMMAIRE

AVANT-PROPOS	111
1 Domaine d'application	113
2 Références normatives	113
3 Généralités	113
4 Extensions MultiAudio du livre audio	116
4.1 Table des matières	116
4.2 Description	118
4.2.1 Généralités	118
4.2.2 Tag.Identifier	118
4.2.3 Tag.OrdinalNumber	118
4.2.4 Tag.Reserved	118
4.2.5 Tag.Length	118
4.2.6 Data	119
5 Structure extensible de livre audio (AB_ExtStruct)	119
5.1 Généralités	119
5.2 Balise (<i>Tag</i>)	119
5.3 Décalage de la zone de données supplémentaires (<i>Extra Data</i>)	119
5.4 Bloc de longueur fixe (AB_FixedChunk)	119
5.4.1 Généralités	119
5.4.2 Exemple	120
5.5 Bloc de longueur variable (AB_VariableChunk)	121
5.5.1 Généralités	121
5.5.2 Exemple	121
5.6 Données supplémentaires	122
6 En-tête de livre audio (AB_Header)	122
6.1 Généralités	122
6.2 Balise	123
6.3 Bloc fixe de AB_Header	123
6.3.1 Généralités	123
6.3.2 Balise	123
6.3.3 Version majeure de la spécification de livre audio	123
6.3.4 Version mineure de la spécification de livre audio	124
6.3.5 Nombre de livres audio (N_Bk)	124
6.3.6 Décalage des index de liste d'écoute	124
6.4 Bloc variable de AB_Header	124
6.4.1 Généralités	124
6.4.2 Balise	124
6.4.3 Index des listes d'écoute (<i>Playlist indexes</i>)	124
6.5 Données supplémentaires de AB_Header	125
7 Livre audio (AB_Book)	125
7.1 Généralités	125
7.2 Balise	126
7.3 Bloc fixe de AB_Book	126
7.3.1 Généralités	126
7.3.2 Balise	127

7.3.3	Format texte (<i>Text format</i>).....	127
7.3.4	Indicateurs (<i>Flags</i>).....	127
7.3.5	Index de TOC_Playlist.....	127
7.3.6	Décalage du titre du livre.....	127
7.3.7	Décalage de l'auteur.....	127
7.3.8	Décalage du narrateur.....	127
7.3.9	Décalage de la déclaration de copyright.....	127
7.3.10	Décalage de l'identificateur unique universel de livre audio (<i>audiobook UUID</i>).....	127
7.3.11	Décalage du numéro ISBN du livre.....	128
7.3.12	Décalage du numéro ISSN du livre.....	128
7.3.13	Décalage de l'identificateur DOI du livre.....	128
7.3.14	Décalage du numéro d'index du livre (<i>book index number</i>).....	128
7.3.15	Décalage de la cote du livre (<i>book shelf number</i>).....	128
7.3.16	Décalage du numéro de référence du livre (<i>book part number</i>).....	128
7.3.17	Décalage d'un autre identificateur.....	128
7.3.18	Décalage du premier niveau de livre audio (<i>AudioBook Level</i> , <i>AB_Level</i>).....	128
7.3.19	Décalage de l'identificateur de l'emplacement du contenu de la liste d'écoute.....	128
7.4	Bloc (<i>Chunk</i>) variable de AB_Book.....	129
7.4.1	Généralités.....	129
7.4.2	Balise.....	129
7.4.3	Titre du livre (<i>Book title</i>).....	129
7.4.4	Auteur.....	130
7.4.5	Narrateur.....	130
7.4.6	Déclaration de copyright.....	130
7.4.7	Identificateur unique universel du livre.....	130
7.4.8	Numéro ISBN du livre.....	130
7.4.9	Numéro ISSN du livre.....	131
7.4.10	Identificateur DOI du livre.....	131
7.4.11	Numéro d'index du livre.....	131
7.4.12	Cote du livre (<i>Book shelf number</i>).....	131
7.4.13	Numéro de référence du livre (<i>Book part number</i>).....	131
7.4.14	Autre identificateur.....	131
7.4.15	Premier niveau de livre audio (<i>First audiobook level</i>).....	131
7.4.16	Identificateur de l'emplacement du contenu liste d'écoute (<i>Playlist</i> <i>content location identifier</i>).....	132
7.5	Données supplémentaires de AB_Book (<i>AB_Book extra data</i>).....	132
8	Niveau de livre audio (AB_Level).....	132
8.1	Généralités.....	132
8.2	Balise.....	133
8.3	Bloc fixe de AB_Level.....	133
8.3.1	Généralités.....	133
8.3.2	Balise.....	134
8.3.3	Format texte (<i>Text format</i>).....	134
8.3.4	Indicateurs.....	134
8.3.5	Nombre de sous-niveaux.....	134
8.3.6	Numéro de niveau (<i>Level Number</i>).....	134
8.3.7	Index de TOC_TrackEntry (<i>TOC_TrackEntry index</i>).....	135

8.3.8	Décalage du bloc de sous-niveau (<i>Offset to sub-level chunk</i>)	135
8.3.9	Décalage du nom de niveau (<i>Offset to level name</i>)	135
8.3.10	Décalage de la description de niveau (<i>Offset to level description</i>)	135
8.3.11	Décalage temporel du livre	135
8.3.12	Décalage d'heure de début	135
8.3.13	Durée	135
8.3.14	Décalage du AB_Level parent (<i>Offset to parent AB_Level</i>)	135
8.3.15	Décalage du AB_Level précédent	136
8.3.16	Décalage du AB_Level suivant (<i>Offset to next AB_Level</i>)	136
8.4	Bloc variable de AB_Level	136
8.4.1	Généralités	136
8.4.2	Balise (<i>Tag</i>)	136
8.4.3	Nom de niveau (<i>Level name</i>)	136
8.4.4	Description du niveau (<i>Level description</i>)	136
8.4.5	Bloc de sous-niveau (<i>Sub-level chunk</i>)	137
8.5	Données supplémentaires de AB_Level (<i>AB_Level extra data</i>)	137
9	Profils MPV audiobook	137
9.1	Généralités	137
9.2	Identificateur d'espace de noms en schéma XML	137
9.3	Utilisation d'un identificateur de profil MPV-Music	138
9.4	Fichier de livre audio	139
9.5	Métadonnées du profil de livre audio	139
9.6	Métadonnées du profil de livre audio MPV	141
9.7	Utilisation des métadonnées Dublin Core	146
9.8	Profil de livre audio utilisant les listes d'écoute MPV-Music	147
9.8.1	Généralités	147
9.8.2	Nombre de listes d'écoute par fichier MPV	148
9.8.3	Utilisation des métadonnées	148
9.8.4	Utilisation en arrière-plan	150
9.8.5	Utilisation en avant-plan	150
9.8.6	Actifs associés et d'interprétation (<i>rendition</i>)	150
9.9	Groupes d'actifs dans les listes d'écoute	150
9.10	Détails du schéma d'un livre audio MPV	153
9.10.1	Définition de schéma XML	153
9.10.2	Définitions de l'utilisation des métadonnées de livre audio	158
9.11	Identificateurs de classe de niveau de livre audio	159
9.12	Fichier BOOKMARK.AUB du livre audio	160
9.13	Métadonnées de niveau de livre audio MPV utilisant <mpvp:Album>	167
9.14	Actif musical et fixe <mpvm:AudioWithStills>	167
9.15	Extensions MPV-Audiobook à MPV-Core	170
9.15.1	Types de fichier et extensions de manifeste de livre audio	170
9.15.2	Type de support MIME de manifeste de livre audio	171
9.16	Types de support et formats de fichier	171
9.17	MPV-Audiobook et CEA-2003-C	171
9.18	Formats des fichiers binaires à XML CEA-2003-C – Comparaison de balises	171
10	Respect des exigences quant aux lecteurs et à la structure de format de fichier de livre audio numérique	173
10.1	Généralités	173
10.2	Exigences quant aux lecteurs	173

10.3 Exigences des structures de format de fichier de livre audio numérique	174
11 Recommandations pour la fonctionnalité des lecteurs et des structures de format de fichier de livre audio numérique	174
11.1 Généralités.....	174
11.2 Recommandations sur les fonctionnalités des lecteurs	174
11.3 Recommandations sur la fonctionnalité des structures de format de fichier du livre audio numérique	175
12 Structures de format de fichier de livre audio numérique et contraintes sur les données	175
13 Options supplémentaires de production	176
13.1 Balisage ID3 dans les fichiers MP3	176
13.2 Listes d'écoute M3U.....	176
13.3 Options de production DAISY	176
14 Certification	177
Annexe A (informative) Exemples de structure de livre audio	178
Annexe B (normative) Types de codage et identificateurs	190
Annexe C (informative) Exemples de cas d'utilisation et d'implémentation: audio de mots parlés.....	191
Bibliographie.....	212
Figure A.1 – Organisation de la structure de AB_Level dans la zone des blocs de longueur variable de AB_Book (<i>AB_Book Variable Length Chunk Area</i>).....	182
Figure A.2 – Vue d'une hiérarchie de AB_Level d'un livre audio.....	184
Figure C.1 – Fichier INDEX.AUB de l'Exemple 1	196
Figure C.2 – Fichier TOC.MUM de l'Exemple 1	197
Figure C.3 – Fichier INDEX.AUB de l'Exemple 2	202
Figure C.4 – Fichier INDEX.AUB de l'Exemple 3	207
Figure C.5 – Fichier INDEX.AUB de l'Exemple 4	211
Tableau 1 – Structure de données de la table des matières	116
Tableau 2 – Format de bloc MultiAudio de livre audio	118
Tableau 3 – Format AB_ExtStruct	119
Tableau 4 – Exemple de structure de AB_FixedChunk	120
Tableau 5 – Exemple de la structure de AB_FixedChunk révisée	120
Tableau 6 – Exemple—Champs de AB_FixedChunk	122
Tableau 7 – Exemple—Données de AB_VariableChunk	122
Tableau 8 – Format de AB_Header	123
Tableau 9 – Format du bloc fixe de AB_Header	123
Tableau 10 – Format du bloc variable de AB_Header.....	124
Tableau 11 – Format de AB_Book	126
Tableau 12 – Format du bloc fixe de AB_Book.....	126
Tableau 13 – Format du bloc variable de AB_Book	129
Tableau 14 – Paramètres de champ requis pour la structure de AB_Level	132
Tableau 15 – Paramètres de champ d'identificateur de l'emplacement de contenu	132
Tableau 16 – Format de AB_Section.....	133

Tableau 17 – Format du bloc fixe de AB_Level	134
Tableau 18 – Format du bloc variable de AB_Level	136
Tableau 19 – Identificateur d'espace de noms de schéma	138
Tableau 20 – Identificateur du profil MPV-Music	138
Tableau 21 – Propriétés du profil de livre audio MPV	140
Tableau 22 – Métadonnées du profil de livre audio MPV	141
Tableau 23 – Métadonnées Dublin Core (DC-NMF).....	146
Tableau 24 – Utilisation des propriétés de métadonnées	148
Tableau 25 – Formats binaires et XML CEA-2003-C – Comparaison de balises	172
Tableau 26 – Exemples de balises ID3	176
Tableau A.1 – Modèle de table des matières.....	178
Tableau A.2 – Structure du fichier table des matières	179
Tableau A.3 – Structure TOC_TrackEntry du fichier 1	179
Tableau A.4 – Structure TOC_TrackEntry du fichier 2	179
Tableau A.5 – Structure TOC_TrackEntry du fichier 3	180
Tableau A.6 – Structure TOC_TrackEntry du fichier 4	180
Tableau A.7 – Table des matières avec décalages temporels des fichiers	181
Tableau A.8 – Structure de TOC_Playlist de l'exemple	181
Tableau A.9 – Structure AB_Book de l'exemple	181
Tableau A.10 – Structure de AB_Level au Niveau 0 (livre audio)	185
Tableau A.11 – AB_Level au Niveau 1 (chapitre).....	185
Tableau A.12 – AB_Level au Niveau 1 (chapitre)	186
Tableau A.13 – AB_Level au Niveau 1 (chapitre)	186
Tableau A.14 – AB_Level au Niveau 2 (section)	187
Tableau A.15 – AB_Level au Niveau 2 (section)	187
Tableau A.16 – AB_Level au Niveau 2 (section)	188
Tableau A.17 – AB_Level au Niveau 2 (section)	188
Tableau A.18 – AB_Level au Niveau 3 (sous-section)	189
Tableau A.19 – AB_Level au Niveau 3 (sous-section)	189
Tableau B.1 – Représentation hexadécimale et interprétation	190

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

EXIGENCES RELATIVES AU FORMAT DE FICHIER DE LIVRE AUDIO ET AUX LECTEURS

AVANT-PROPOS

- 1) La Commission Electrotechnique Internationale (CEI) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, la CEI – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de la CEI"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec la CEI, participent également aux travaux. La CEI collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
- 2) Les décisions ou accords officiels de la CEI concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de la CEI intéressés sont représentés dans chaque comité d'études.
- 3) Les Publications de la CEI se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de la CEI. Tous les efforts raisonnables sont entrepris afin que la CEI s'assure de l'exactitude du contenu technique de ses publications; la CEI ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
- 4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de la CEI s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de la CEI dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de la CEI et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
- 5) La CEI elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de la CEI. La CEI n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
- 6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
- 7) Aucune responsabilité ne doit être imputée à la CEI, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de la CEI, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de la CEI ou de toute autre Publication de la CEI, ou au crédit qui lui est accordé.
- 8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
- 9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de la CEI peuvent faire l'objet de droits de brevet. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets et de ne pas avoir signalé leur existence.

La Norme Internationale CEI 62571 a été établie par le domaine technique 10: Multimedia e-publishing and e-book¹, du comité d'études 100 de la CEI: Systèmes et appareils audio, vidéo et multimédia.

La présente version bilingue, publiée en 2011-07, correspond à la version anglaise.

Le texte anglais de cette norme est issu des documents 100/1543/CDV et 100/1629/RVC. Le rapport de vote 100/1629/RVC donne toute information sur le vote ayant abouti à l'approbation de cette norme.

La version française de cette norme n'a pas été soumise au vote.

Cette publication a été rédigée selon les Directives ISO/CEI, Partie 2.

¹ Publication et livre électronique multimédia.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de stabilité indiquée sur le site web de la CEI sous "<http://webstore.iec.ch>" dans les données relatives à la publication recherchée. A cette date, la publication sera

- reconduite,
- supprimée,
- remplacée par une édition révisée, ou
- amendée.

IECNORM.COM : Click to view the full PDF of IEC 62571:2011

EXIGENCES RELATIVES AU FORMAT DE FICHIER DE LIVRE AUDIO ET AUX LECTEURS

1 Domaine d'application

La présente Norme internationale définit les exigences et formule des recommandations à l'intention des éditeurs, développeurs de logiciels, fournisseurs de contenu et fabricants de matériel sur la structure de données, les exigences en facilité d'utilisation, les systèmes de lecture et les systèmes de remise des livres audio au format de fichier numérique. Il convient de noter que dans l'ensemble de cette Norme internationale, le terme livre audio désigne un fichier audio ou un assemblage de fichiers audio contenant surtout du langage parlé, fichiers lus ou linéairement ou dans un ordre spécifié. Donc, de l'audio fait de mots parlés avec de la musique de temps en temps, de narration d'articles de journaux, et d'autres discours de ce genre, est dans cette norme considéré comme livre audio.

2 Références normatives

Les documents de référence suivants sont indispensables pour l'application du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document référencé s'applique (y compris les éventuels amendements).

ISO/CEI 11578:1996, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Appel de procédures à distance (RPC)*

ISO 639-1, *Codes pour la représentation des noms de langue – Partie 1: Code Alpha-2*

ISO 3166-1, *Codes pour la représentation des noms de pays et de leurs subdivisions – Partie 1: Codes de pays*

ISO 9660:1988, *Traitement de l'information – Structure de volume et de fichier des disques optiques compacts à mémoire fixe (CD-ROM) destinés à l'échange d'information*

3 Généralités

La présente Norme internationale définit la structuration des formats de fichiers des livres audio pour le média numérique livre audio. Cette Norme internationale a pour but la compatibilité avec les normes de l'industrie musicale, du multimédia, des outils efficaces de présentation et de navigation pour livre audio. Cette norme internationale est une compilation qui recouvre les architectures binaires représentées par des versions précédentes de CEA-2003, et des architectures XML plus récentes représentées par la norme MPV MusicPhotoVideo™² de l'Optical Storage Technology Association (OSTA). Un système de lecture et un contenu de livre audio conformes rend la lecture et l'écoute agréables et très fonctionnelles. L'objectif de cette Norme internationale est de créer une norme étendue et extensible à l'usage des éditeurs de livres audio, des fabricants de lecteurs de livres audio et des développeurs de logiciels de livre audio pour assurer au consommateur du livre audio l'écoute la plus plaisante. En voici une présentation générale.

² MusicPhotoVideo est le nom de marque d'un programme fourni par l'Optical Storage Technology Association. Ces informations sont données pour la commodité des utilisateurs de ce document et n'impliquent pas que le programme désigné ait été approuvé par l'IEC/CEI.

OSTA MultiAudio définit qu'un fichier binaire est placé à la racine d'une structure de format de fichiers sur un CD/MP3. Le fichier TOC.MAU sert de table des matières aux fichiers MP3 associés sur un CD/MP3 conforme. Les lecteurs MultiAudio conformes lisent ce fichier pour obtenir les informations dont ils ont besoin pour présenter le contenu à l'auditeur.

La norme OSTA MPV définit un schéma XML (*eXtensible Markup Language*) à placer à la racine de la structure de format de fichier. Le fichier INDEX.AUB sert de table des matières aux fichiers livre audio associés, quels que soient le support, le codeur-décodeur ou le support. Les lecteurs MPV conformes lisent ce fichier pour trouver les indications de présentation du contenu à l'auditeur. Les livres audio conformes à cette norme DOIVENT utiliser l'extension de fichier spécifique au livre audio, .AUB. En outre, les livres audio conformes à cette norme peuvent utiliser la même extension de fichier spécifiée par le Profil MPV Music, .MUM, afin de créer une compatibilité supplémentaire pour les lecteurs conformes à MPV qui ne reconnaissent pas l'extension du livre audio.

De même, alors que la norme MPV de l'OSTA était d'abord destinée à l'industrie musicale et aux CD-ROM, elle fournit des "extensions" pour améliorer ce format. La présente Norme internationale "étend" MPV OSTA pour prendre en charge les livres audio et ne requiert pas un support ou un système de distribution de contenu spécifique. Les extensions de livre audio sont définies par le fichier binaire MultiAudio TOC.MAU pour les livres audio sur CD/MP3 et par le fichier MPV XML INDEX.AUB pour tous les livres audio numériques.

Par conséquent, dans l'intégralité de cette Norme internationale, l'expression "structure de format de fichier de livre audio numérique" sert à décrire l'ensemble du contenu d'un livre audio, quel que soit le support ou la distribution. "Système de distribution" désigne tout système servant à restituer le contenu, que ce soit un véritable support physique ou un autre système de distribution, par exemple le téléchargement ou le streaming. Cette Norme internationale définit les données MPV requises ainsi que les exigences liées aux autres données spécifiques aux livres audio. Le lecteur conforme à la norme, en lisant les fichiers INDEX.AUB ou TOC.MAU, reçoit les données nécessaires pour parcourir efficacement et lire le contenu de la façon voulue par l'éditeur. Les lecteurs conformes et les contenus audio conformes doivent satisfaire aux exigences présentées à l'Article 11. En outre, il est encouragé, mais non obligatoire, que les lecteurs conformes et les contenus audio conformes suivent les recommandations de l'Article 11.

Le lecteur identifie et lit le fichier INDEX.AUB ou TOC.MAU à la racine de la structure du format de fichier du livre audio numérique. Un lecteur conforme reconnaît l'identificateur du livre audio dans le champ "Extra Data OSTA MPV" et en traite les données selon les Articles 4 à 7. OSTA MPV fournit des identificateurs spécifiques qui indiquent au lecteur le type de fichier audio codé utilisé. OSTA MPV fournit également un identificateur "inconnu" qui indique à l'appareil d'avoir à déterminer le type de codage utilisé. Une fois qu'un livre audio et sa méthode de codage sont détectés, le système de lecture peut référencer des points de navigation et des champs d'information et les présenter à l'auditeur. Il est recommandé de minimiser les retards à la restitution ou au passage à la partie suivante du livre audio; des retards excessifs ne sont, selon la présente norme, pas admissibles.

Un ou plusieurs livres audio peuvent être sur le système; s'il y en a plusieurs, ils doivent être dans des répertoires séparés de la structure de format de fichier du livre audio numérique. Le fichier MPV contient une ou plusieurs listes d'écoute (*playlist*) pour chaque livre audio. Cette liste d'écoute détermine l'ordre correct de lecture linéaire des fichiers à restituer pour chaque livre ainsi que l'emplacement des fichiers du livre audio numérique. Une structure de format de fichier de livre audio simple résidant sur un support physique peut n'avoir besoin que d'une seule liste d'écoute; une structure de format de fichier de livre audio numérique plus complexe avec des options de distribution, par exemple "local" ou "streaming", peut avoir plusieurs listes d'écoute, une par procédé de distribution.

Des niveaux de navigation sont définis par l'éditeur de contenu dans le fichier INDEX.AUB ou TOC.MAU, et peuvent être aussi simples ou complexes que souhaité. Les champs de données supplémentaires (*Extra Data*) identifient les niveaux et les "décalages temporels"

appliqués aux points de navigation dans les fichiers ou les emplacements de fichiers. Des pointeurs vers les emplacements "parent" et "enfant" sont également définis au sein des champs de données supplémentaires (*Extra Data*). Ainsi, un nombre illimité de niveaux et de points de navigation peut être identifié. Dans le fichier binaire TOC.MAU, la navigation de Niveau 0 est le niveau le plus élevé; par exemple, on peut parcourir le livre audio lui-même ou une collection de plusieurs livres audio contenus dans des dossiers séparés. La navigation de Niveau 1 (*Level 1*) est le premier niveau défini de pointeurs imbriqués, et est requis pour la conformité. Dans un exemple très simple, une structure de format de fichier de livre audio numérique pourrait contenir un livre audio constitué d'un fichier numérique. Chacun des emplacements de chapitre est dans le fichier TOC.MAU défini comme point de navigation de Niveau 1. Le lecteur conforme est capable de donner à l'utilisateur une navigation au Niveau 1 en se déplaçant vers chacun des décalages définis.

Des structures plus complexes de format de fichier de livre audio numérique peuvent être réalisées en imbriquant des niveaux dans le fichier INDEX.AUB. Une structure de format de fichier de livre audio peut contenir plusieurs couches ou niveaux avec des décalages menant aux points de navigation suivants, au niveau "parent" et au niveau "enfant". Dans un exemple plus complexe, la structure de format de fichier de livre audio numérique peut contenir deux livres audio, chacun dans un dossier séparé et constitué de dix chapitres, chaque chapitre étant codé comme fichier numérique séparé. Dans ce cas, la navigation comprendrait des pointeurs vers chacun des livres audio et permettrait au lecteur de choisir l'un ou l'autre. Des marqueurs de navigation pointerait vers le début de chaque chapitre, permettant de naviguer de chapitre en chapitre dans le livre audio. Des niveaux imbriqués pointerait vers les pages des chapitres.

NOTE 1 Les livres, chapitres et pages sont utilisés comme exemples de noms de niveau; cependant, il n'y a pas d'exigence quant aux dénominations des niveaux, qui serait imposée par la présente Norme internationale.

La seule contrainte pour l'éditeur de livres audio est de coder les fichiers MP3 à un "débit binaire fixe" car un codage à débit binaire variable rendrait aux lecteurs de faible puissance impossible le calcul du décalage des données (*data offset*) à partir du décalage temporel (*time offset*).

NOTE 2 Les fournisseurs de contenu de livres audio sont vivement encouragés à nommer les fichiers audio selon l'ordre d'un classement alphabétique suivant les caractères ASCII. Cela permet à un lecteur, ignorant de lire quand même les fichiers audio dans le bon ordre et à la navigation entre fichiers d'être celle choisie par l'éditeur de livres audio pour structurer les fichiers. En d'autres termes, un chapitre par fichier conduirait à une navigation par chapitres, une section par fichier à une navigation par sections. L'ordre correct de lecture linéaire a alors plus de chances d'être conservé.

Il y a un contenu spécifique au livre audio, tel que le titre, l'auteur et le narrateur, ainsi qu'une information temporelle qui permettent aux lecteurs de calculer le temps écoulé et le temps restant. Les champs de données supplémentaires (*Extra Data*) associées à chaque fichier fournissent cette information au lecteur pour qu'elle puisse être donnée à l'auditeur. Des améliorations supplémentaires sont encouragées mais non obligatoires pour les Identificateurs de classe de niveau (*Level Class Identifier*) et le fichier Bookmark.AUB séparé destiné aux supports réinscriptibles. Les Identificateurs de classe de niveau permettront à l'appareil de lecture de gérer des niveaux distincts de livre audio de façon personnalisée, en fonction du type de contenu ou sous-groupage, de ces niveaux. Le fichier Bookmark.AUB est pourvu à la fois d'un point de reprise automatique et de fonctionnalités d'insertion de signets par l'utilisateur et est conseillé pour le transfert entre deux appareils de même type. Les concepteurs d'appareils de lecture peuvent également penser à intégrer des améliorations multiutilisateurs dotées de fonctionnalités d'insertion de signets.

Pour un lecteur lisant des fichiers binaires, les Articles 4 à 8 répertorient les extensions de livres audio MultiAudio, les en-têtes, les listes d'écoute (*playlists*) et les niveaux du format MultiAudio. Pour un lecteur capable de lire un fichier XML, l'Article 10 présente en détail le schéma MPV des livres audio de format MPV. Pour les lecteurs qui lisent à la fois les fichiers binaires et le schéma XML, il convient d'utiliser le format XML.

4 Extensions MultiAudio du livre audio

4.1 Table des matières

Le présent article définit les structures de données étendues de livre audio fondées sur la norme OSTA MultiAudio.

La structure de données de la table des matières doit être celle définie au Tableau 1.

Tableau 1 – Structure de données de la table des matières

Article / Paragraphe	Description
4.2	Description
Tableau 2	Bloc MultiAudio de livre audio (<i>Audiobook MultiAudio Chunk</i>)
4.2.2	Tag.Identifier
4.2.3	Tag.OrdinalNumber
4.2.4	Tag.Reserved
4.2.5	Tag.Length
4.2.6	Tag.Data
5	Structure extensible de livre audio (AB_ExtStruct) (<i>Audiobook Extendable Structure</i>)
5.2	Balise (<i>Tag</i>)
5.3	Décalage des données supplémentaires (<i>Offset to Extra Data</i>)
5.4	Bloc de longueur fixe (AB_FixedChunk) (<i>Fixed Length Chunk</i>)
5.4.2	Exemple
5.5	Bloc de longueur variable (AB_VariableChunk) (<i>Variable Length Chunk</i>)
5.5.2	Exemple
5.6	Données supplémentaires (<i>Extra Data</i>)
6	En-tête de livre audio (AB_Header) (<i>Audiobook Header</i>)
6.2	Balise
6.3	Bloc fixe AB_Header (<i>AB_Header Fixed Chunk</i>)
6.3.2	Balise
6.3.3	Version majeure de la spécification de livre audio (<i>Audiobook Spec Major Version</i>)
6.3.4	Version mineure de la spécification de livre audio (<i>Audiobook Spec Minor Version</i>)
6.3.5	Numéro des livres audio (<i>Number of Audiobooks</i>)
6.3.6	Décalage d'index de liste d'écoute (<i>Offset to Playlist Indexes</i>)
6.4	Bloc variable AB_Header
6.4.2	Balise
6.4.3	Index de liste d'écoute (<i>Playlist Indexes</i>)
6.5	Données supplémentaires (<i>Extra Data</i>)
7	Livre audio (AB_Book)
7.2	Balise
7.3	Bloc fixe AB_Book
7.3.2	Balise
7.3.3	Format texte (<i>Text Format</i>)
7.3.4	Indicateurs ou Drapeaux (<i>Flags</i>)
7.3.5	Index TOC_Playlist
7.3.6	Décalage du titre de livre (<i>Offset to Book Title</i>)
7.3.7	Décalage de l'auteur (<i>Offset to Author</i>)

Tableau 1 (suite)

Article / Paragraphe	Description
7.3.8	Décalage du narrateur (<i>Offset to Narrator</i>)
7.3.9	Décalage de la déclaration de Copyright (<i>Offset to Copyright Statement</i>)
7.3.10	Décalage des identificateurs uniques universels de livre audio (<i>Offset to Audiobook UUIDs</i>)
7.3.11	Décalage du numéro ISBN du livre (<i>Offset to Book ISBN Identifier</i>)
7.3.12	Décalage du numéro ISSN du livre
7.3.13	Décalage de l'identificateur DOI du livre
7.3.14	Décalage du numéro d'index du livre
7.3.15	Décalage de la cote du livre
7.3.16	Décalage du numéro de référence du livre
7.3.17	Décalage d'un autre identificateur
7.3.18	Décalage du premier niveau de livre audio
7.3.19	Décalage de l'identificateur d'emplacement de contenu de liste d'écoute
7.4	Bloc variable AB_Book
7.4.2	Balise
7.4.3	Titre du livre
7.4.4	Auteur
7.4.5	Narrateur
7.4.6	Déclaration de Copyright
7.4.7	Identificateur unique universel du livre (<i>Book UUID</i>)
7.4.8	Numéro ISBN du livre
7.4.9	Numéro ISSN du livre
7.4.10	Identificateur DOI du livre
7.4.11	Numéro d'index du livre
7.4.12	Cote du livre
7.4.13	Numéro de référence du livre
7.4.14	Autre identificateur
7.4.15	Premier niveau de livre audio
7.4.16	Identificateur d'emplacement de contenu de liste d'écoute
7.5	Données supplémentaires (<i>Extra Data</i>)
8	Niveau de livre audio (AB_Level)
8.2	Balise
8.3	Bloc fixe AB_Level
8.3.2	Balise
8.3.3	Format texte (<i>Text Format</i>)
8.3.4	Indicateurs
8.3.5	Nombre de sous-niveaux
8.3.6	Numéro de niveau
8.3.7	Premier index TOC_TrackEntry
8.3.8	Décalage du bloc de sous-niveau
8.3.9	Décalage du nom de niveau
8.3.10	Décalage de la description de niveau
8.3.11	Décalage temporel du livre
8.3.12	Décalage d'heure de début
8.3.13	Durée
8.3.14	Décalage du AB_Level parent
8.3.15	Décalage du AB_Level précédent
8.3.16	Décalage du AB_Level suivant
8.4	Bloc variable de AB_Level
8.4.2	Balise
8.4.3	Nom de niveau
8.4.4	Description du niveau
8.4.5	Bloc de sous-niveau
8.5	Données supplémentaires (<i>Extra Data</i>)

4.2 Description

4.2.1 Généralités

Toutes les structures définies dans la norme CEA-2003-B étaient des extensions de OSTA MultiAudio. Ces structures sont conservées dans la présente Norme internationale. Chaque structure est dans la zone de données supplémentaires (*Extra Data*) des structures MultiAudio de sorte qu'elle n'interfère pas avec la fonction d'origine de la norme OSTA MultiAudio, mais l'améliore avec des métadonnées supplémentaires qui s'appliquent à une collection de livres audio. Toutes les structures de livre audio sont contenues dans un Bloc MultiAudio (*MultiAudio Chunk*) et utilisent le format de structure extensible (*Extendable Structure*) décrit dans l'Article 5.

Toutes les structures de livre audio sont contenues dans une structure de Bloc MultiAudio. Le Bloc doit avoir les valeurs indiquées dans le Tableau 2.

Tableau 2 – Format de bloc MultiAudio de livre audio

#	Structure	RBP	Longueur	Nom	Contenu
1	Balise	0	4	Identificateur	9FFF0000h
2		4	2	Nombre ordinal	Non défini
3		6	2	Réservé	0
4		8	4	Longueur	UInt32
5	Données	12	Variable	Structure de livre audio	AB_ExtStruct

4.2.2 Tag.Identifier

Il convient que la valeur d'identificateur identifie de façon unique le Bloc MultiAudio comme une structure de livre audio. La valeur 9FFF0000h doit lui donner la qualité de structure du livre audio. Un analyseur recherchera ensuite un Bloc avec cette valeur d'identificateur dans la zone des données supplémentaires (*Extra Data*) des structures MultiAudio afin de déterminer si la structure de livre audio est définie.

La norme OSTA MultiAudio affecte l'intervalle [80000000h – FFFFFFFFh] à une utilisation publique, ce qui signifie que d'autres spécifications peuvent également utiliser la valeur 9FFF0000h pour leur identificateur de Bloc. Pour s'assurer que le Bloc contient une structure de livre audio, l'analyseur doit ensuite rechercher dans les 4 premiers octets de la zone de données du Bloc un identificateur valide de structure de livre audio. Les 4 premiers octets de la zone de données doivent contenir un champ Tag.Identifier de structure de livre audio qui doit être défini par chaque structure de livre audio.

4.2.3 Tag.OrdinalNumber

Le nombre ordinal du bloc doit être l'index du bloc dans la matrice de structures de bloc qui constituent la zone de données supplémentaires (*Extra Data*) d'une structure MultiAudio.

4.2.4 Tag.Reserved

Le champ tag.reserved doit être réservé pour une utilisation ultérieure par l'OSTA. Ce champ n'est pas défini à présent et ne doit pas contenir de données.

4.2.5 Tag.Length

Le champ Tag.Length doit contenir la longueur de la structure de bloc (*Chunk structure*) dans la longueur des zones Chunk.Tag et Chunk.Data.

4.2.6 Data

La zone data de la structure de Bloc (*Chunk structure*) doit contenir une structure de livre audio valide, qui doit suivre le format AB_ExtStruct.

5 Structure extensible de livre audio (AB_ExtStruct)

5.1 Généralités

La structure extensible de livre audio définit une structure binaire qui permet une extensibilité aisée. Cela permet d'ajouter de nouveaux champs à la structure, tout en conservant une compatibilité rétroactive avec les versions précédentes de la structure. Toutes les structures de livre audio doivent être conformes au format de structure extensible (*Extendable Structure format*) contenu dans le Tableau 3.

Tableau 3 – Format AB_ExtStruct

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise (<i>Tag</i>)	balise
2	12	4	Décalage de la zone de données supplémentaires (<i>Offset to Extra Data</i>)	Uint32
3	16	L_Fc	Bloc de longueur fixe (<i>Fixed Length Chunk</i>)	AB_FixedChunk
4	16 + L_Fc	L_Vc	Bloc de longueur variable (<i>Variable Length Chunk</i>)	AB_VariableChunk
5	16 + L_Fc + L_Vc	L_Ex	Données supplémentaires (<i>Extra Data</i>)	Données supplémentaires (<i>Extra Data</i>)

5.2 Balise (*Tag*)

Un identificateur unique qui identifie la structure de livre audio doit figurer dans le champ Tag.Identifier.

Le nombre ordinal de la structure doit figurer dans le champ Tag.OrdinalNumber. Par défaut, il convient que ce nombre ordinal ait la valeur 0, à moins que la structure n'appartienne à une liste de structures.

La longueur de la structure extensible en octets doit figurer dans Tag.Length. Comme cette structure est une structure de Bloc MultiAudio, cette longueur doit être exactement divisible par 4.

5.3 Décalage de la zone de données supplémentaires (*Extra Data*)

Pour garantir que AB_ExtStruct puisse être étendue ultérieurement, une zone de données supplémentaires est incluse. Si la structure contient des données supplémentaires, il convient que le décalage de la zone de données supplémentaires (*Extra Data*) contienne le décalage en octets à partir du début de AB_ExtStruct jusqu'au début du Bloc de données supplémentaires (*Extra Data*). S'il n'y a pas de bloc de données supplémentaires (*Extra Data Chunk*) il est recommandé de donner à ce champ la valeur 0.

5.4 Bloc de longueur fixe (AB_FixedChunk)

5.4.1 Généralités

Le bloc de longueur fixe est défini comme une structure de bloc (*chunk structure*) et assure l'extensibilité de **AB_ExtStruct**. Il convient de définir les valeurs suivantes:

AB_FixedChunk.Tag.Identifier doit être mis à la valeur hexadécimale 9FFF0001h afin de montrer qu'il s'agit d'un bloc AB_FixedChunk.

AB_FixedChunk.Tag.OrdinalNumber doit être mis à la valeur 0.

AB_FixedChunk.Tag.Length doit être mis au nombre d'octets de la longueur de AB_FixedChunk. L'extensibilité de la structure dépend essentiellement de la longueur de cette structure. Au fur et à mesure que de nouveaux champs sont ajoutés à la structure, cette longueur s'accroît. Il convient que le logiciel ou les appareils qui interprètent cette structure vérifient d'abord cette longueur pour identifier la version de la structure. Il est, afin d'assurer une compatibilité rétroactive, recommandé que les champs de cette structure ne soient jamais supprimés ou redéfinis. Il convient d'ajouter tous les nouveaux champs à la fin de la structure.

Puisque cette structure est définie comme un Bloc MultiAudio, sa longueur doit être exactement divisible par 4.

5.4.2 Exemple

L'exemple suivant montre comment la structure est étendue et interprétée par les analyseurs.

Un exemple de structure de AB_FixedChunk est défini dans le Tableau 4.

Tableau 4 – Exemple de structure de AB_FixedChunk

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise AB_FixedChunk	Balise (<i>Tag</i>)
2	12	4	Champ 1 (<i>Field1</i>)	UInt32
3	16	2	Champ 2 (<i>Field2</i>)	UInt16
4	18	2	Champ 3 (<i>Field3</i>)	UInt16

Le champ AB_FixedChunk.Tag.Length de cette structure devrait être mis à la valeur 20, qui désigne une longueur de 20 octets. Tous les appareils et le logiciel qui peuvent analyser cette structure peuvent comprendre les valeurs des champs 1 à 3 et que la longueur totale de la structure est de 20 octets.

A une date ultérieure, un nouveau champ est ajouté à la structure. La nouvelle structure est décrite au Tableau 5.

Tableau 5 – Exemple de la structure de AB_FixedChunk révisée

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise AB_FixedChunk	Balise
2	12	4	Champ 1	UInt32
3	16	2	Champ 2	UInt16
4	18	2	Champ 3	UInt16
5	20	4	Champ 4	UInt32

Le logiciel ou l'appareil précédent qui était conçu pour gérer les champs 1 à 3 ignore le nouveau champ 4. Les nouveaux logiciels ou appareils conçus pour lire le nouveau champ restent capables d'interpréter les trois champs précédents. Si le logiciel ou l'appareil plus récent devait analyser la structure plus ancienne précédente, il vérifierait d'abord la longueur du champ FixedChunk.Tag.Length et réaliserait que cette structure ne définit pas de champs supérieurs au champ 3, la longueur n'étant que de 20 octets. Par conséquent, il convient que

le logiciel ou l'appareil le plus récent gère au mieux cette situation et initialise le champ 4 avec une valeur par défaut, ou l'ignore complètement.

5.5 Bloc de longueur variable (AB_VariableChunk)

5.5.1 Généralités

La structure du bloc de longueur variable (AB_VariableChunk) est conçue pour fonctionner avec le bloc de longueur fixe (AB_FixedChunk) afin de gérer les valeurs des champs de longueur variable, telles qu'une chaîne. AB_FixedChunk définit quelles valeurs sont stockées dans le bloc AB_VariableChunk en fournissant des décalages pour les valeurs contenues dans AB_VariableChunk.

Le bloc AB_VariableChunk est défini en tant que structure de Bloc MultiAudio (*MultiAudio Chunk structure*). En conséquence, la longueur de la structure doit être exactement divisible par 4 pour garantir qu'elle s'insère bien dans une mémoire gérée par groupes de 4 octets. Pour garantir cela, tous les champs du bloc AB_VariableChunk doivent avoir une longueur exactement divisible par 4. Par conséquent, un champ doit être rempli d'octets de bourrage pour garantir que sa longueur soit exactement divisible par 4. Tous les octets de bourrage doivent être mis à la valeur 0. Pour calculer la longueur du bourrage, utiliser l'équation suivante:

$$\text{Nombre d'octets de bourrage} = \text{int}((\text{TempLen} + 3)/4) * 4 - \text{TempLen}$$

où

TempLen est la taille en octets du champ avant bourrage.

Par exemple, si le bloc AB_VariableChunk contient deux chaînes (String1 = "ABC\0") et (String2 = "ABCDEF\0"), où le caractère '\0' désigne un caractère de terminaison, la première chaîne (String1) n'a pas à être complétée car sa longueur est de 4 octets. La seconde chaîne (String2) devrait être complétée par un seul octet supplémentaire afin que sa longueur passe de 7 à 8 octets.

AB_VariableChunk doit avoir les valeurs de consigne suivantes.

- AB_VariableChunk.Tag.Identifier sera mis à la valeur hexadécimale 9FFF0002h pour indiquer qu'il s'agit d'un bloc de longueur variable.
- AB_VariableChunk.Tag.OrdinalNumber sera mis à la valeur 1.
- AB_VariableChunk.Tag.Length sera mis à la longueur de cette structure, balise comprise.

L'exemple suivant montre comment le bloc de longueur fixe et le bloc de longueur variable fonctionnent conjointement pour produire une structure extensible.

5.5.2 Exemple

Dans cet exemple, AB_FixedChunk définit les champs du Tableau 6.

Tableau 6 – Exemple—Champs de AB_FixedChunk

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise de AB_FixedChunk	balise
2	12	2	Décalage de chaîne 1 (<i>String1</i>)	Uint32
3	16	2	Décalage de champ 1 (<i>Field1</i>) variable	Uint16
4	18	2	Longueur de champ 1 (<i>Field1</i>) variable (<i>L_F1</i>)	Uint16

Dans cet exemple, AB_VariableChunk contiendrait alors les données indiquées dans le Tableau 7.

Tableau 7 – Exemple—Données de AB_VariableChunk

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise de AB_VariableChunk	balise
2	12	L_Str1	String1 (terminée par la valeur NULL)	octet
3	12 + L_Str1	L_F1	Variable Field1	octet

Le bloc AB_VariableChunk contient deux champs, String1 et Variable Field1. Etant donné que String1 est une chaîne terminée par la valeur NULL, il n'est pas nécessaire de fournir la longueur de la chaîne en octets car un analyseur reconnaîtra que le caractère NULL se trouve à la fin des données. Variable Field1 contient une valeur arbitraire de longueur L_F1 en octets. Cette variable ne bénéficiant pas de la terminaison NULL, la longueur du champ est incluse dans AB_FixedChunk.

Dans cet exemple, AB_FixedChunk contiendrait des décalages du début de AB_FixedChunk jusqu'aux données contenues dans AB_VariableChunk. Si String1 n'était pas incluse dans AB_VariableChunk, le décalage dans AB_FixedChunk serait mis à 0. Par conséquent, AB_VariableChunk ne définit pas l'ordre des données car celles-ci peuvent être placées dans un ordre arbitraire. Cet ordre est défini dans AB_FixedChunk. Ainsi, le bloc AB_FixedChunk est l'équivalent de l'index d'un livre et AB_VariableChunk contient les véritables données, c'est-à-dire les pages et les chapitres du livre.

5.6 Données supplémentaires

Les données supplémentaires doivent être définies sous forme d'un Bloc (*Chunk*) contenant une matrice de structures à base de Blocs. Cette zone sert aux extensions supplémentaires de la structure.

6 En-tête de livre audio (AB_Header)

6.1 Généralités

L'en-tête de livre audio est situé dans le champ Extra Data de la structure de TOC_Header de OSTA MultiAudio. AB_Header est contenue dans une structure de Bloc, comme décrit en 4.2. Le nombre ordinal du Bloc ne doit pas être fixé. La structure de AB_Header est définie comme une structure AB_ExtStructure, comme défini en 4.2. AB_Header peut être utilisé pour déterminer si un fichier table des matières (*TOC file*, *Table Of Content file*) contient des livres audio, ainsi que l'emplacement de ces livres. Le format de AB_Header doit être comme indiqué dans le Tableau 8.

Tableau 8 – Format de AB_Header

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise	Balise
2	12	4	Décalage des données supplémentaires (<i>Extra Data</i>)	Uint32
3	16	L_Fc	Bloc fixe AB_Header	AB_FixedChunk
4	16 + L_Fc	L_Vc	Bloc variable AB_Header	AB_VariableChunk
5	16 + L_Fc + L_Vc	L_Ex	Données supplémentaires (<i>Extra Data</i>)	ExtraData

6.2 Balise

Tag.Identifier doit être mis à la valeur hexadécimale 44484241h (ABHD) afin de désigner une structure AB_Header.

Tag.OrdinalNumber doit être mis à la valeur 0.

Tag.Length doit avoir pour valeur la longueur de la structure AB_Header, longueur de la Balise comprise.

6.3 Bloc fixe de AB_Header

6.3.1 Généralités

Le bloc fixe de AB_Header doit contenir le format défini dans le Tableau 9.

Tableau 9 – Format du bloc fixe de AB_Header

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise	balise
2	12	2	Version majeure de spécification de livre audio	Uint16
3	14	2	Version mineure de spécification de livre audio	Uint16
4	16	2	Nombre de livres audio (N_Bk)	Uint16
5	18	2	Décalage des index de liste d'écoute	Uint16

6.3.2 Balise

Le sous-champ Tag.Identifier doit être la valeur décimale 9FFF0001h pour indiquer qu'il s'agit d'un bloc AB_FixedChunk.

Le sous-champ Tag.OrdinalNumber doit être mis à la valeur 0.

Tag.Length doit être mis à la valeur de la longueur de la structure de bloc fixe de AB_Header. La présente Norme internationale spécifie que cette longueur est de 20 octets.

6.3.3 Version majeure de la spécification de livre audio

Ce champ doit contenir le numéro de la version majeure de la spécification de livre audio. Si le numéro de version majeure change, il n'est pas obligatoire que la nouvelle version de la spécification conserve une compatibilité rétroactive avec les versions précédentes de la spécification.

Pour la présente Norme internationale, ce champ doit prendre la valeur hexadécimale 0001h, qui indique qu'il s'agit de la version 1.X et que sa structure est conforme à cette Norme.

NOTE Etant donné que cette Norme internationale conserve une compatibilité rétroactive avec les normes CEA-2003, CEA-2003-A, CEA-2003-B et ANSI/CEA-2003-C, la version 1.0 est utilisée pour toutes les versions de la norme.

6.3.4 Version mineure de la spécification de livre audio

Ce champ doit contenir le numéro de la version mineure de la spécification du livre audio. Si le numéro de version mineure change, il est obligatoire que la nouvelle version de la spécification conserve une compatibilité rétroactive avec les versions précédentes de la spécification.

Pour la présente Norme internationale, ce champ doit prendre la valeur hexadécimale 0001h, qui indique qu'il s'agit de la version X.1.

6.3.5 Nombre de livres audio (N_Bk)

Ce champ spécifie le nombre de livres audio qui peuvent être trouvés dans le fichier table des matières MultiAudio (*MultiAudio TOC file*).

6.3.6 Décalage des index de liste d'écoute

Le décalage des index de liste d'écoute doit être défini comme le décalage entre le début de la structure de AB_Header jusqu'au bloc variable de AB_Header (*AB_Header Variable Chunk*) qui contient une liste ordonnée des index de TOC_Playlist MultiAudio, voir 6.3.

6.4 Bloc variable de AB_Header

6.4.1 Généralités

Le bloc variable de AB_Header doit contenir les champs définis dans le Tableau 10.

Tableau 10 – Format du bloc variable de AB_Header

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise	Balise
2	12	N_Bk * 2	Index de liste d'écoute	Uint16

6.4.2 Balise

Il convient que le sous-champ Tag.Identifier prenne la valeur hexadécimale 9FFF0002h pour indiquer qu'il s'agit d'un bloc AB_VariableChunk.

Le sous-champ Tag.OrdinalNumber doit prendre la valeur 1.

Tag.Length doit prendre pour valeur en octets de la longueur de la structure de bloc variable de AB_Header (*AB_Header Variable Chunk*).

6.4.3 Index des listes d'écoute (*Playlist indexes*)

Les index des listes d'écoute doivent contenir une liste de valeurs Uint16 qui représentent les index de ces structures de TOC_Playlist MultiAudio qui contiennent une structure de AB_Book, définie dans leur champ de données supplémentaires (*Extra Data*). Chaque livre audio du support doit avoir une structure associée de TOC_Playlist, avec une seule structure de AB_Book présente dans le champ de données supplémentaires (*Extra Data*). Voir l'Article 7 pour plus d'informations.

Les index de TOC_Playlist sont utilisés à la place des décalages d'octets des structures de AB_Book pour que la table des matières puisse être modifiée sans que cela ne casse les

structures du livre audio. Comme des analyseurs qui ne reconnaissent pas les livres audio peuvent modifier les contenus de la table des matières, par exemple ajouter de nouvelles structures de TOC_TrackEntry, les décalages d'octets peuvent se trouver invalidés. Par conséquent, il revient à l'analyseur de livres audio de mettre en correspondance les index associés et les décalages d'octets appropriés dans le fichier.

Etant donné que le champ des index des listes d'écoute (*Playlist Indexes*) peut avoir une longueur qui n'est pas exactement divisible par 4, il peut être nécessaire d'ajouter du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

6.5 Données supplémentaires de AB_Header

Les données supplémentaires de AB_Header doivent être définies comme un Bloc contenant une matrice de structures de Blocs (*Chunk*). Cette zone sert aux extensions supplémentaires de la structure de AB_Header.

7 Livre audio (AB_Book)

7.1 Généralités

Un livre audio est une collection de fichiers audio qui représentent des unités logiques principalement d'audio de mots parlés. Un livre audio peut être représenté par un fichier audio unique ou par plusieurs fichiers qui constituent des unités plus petites du livre complet. Lors de la lecture d'un livre audio, ces unités sont lues en suivant l'ordre de la première à la dernière. Par conséquent, un livre audio peut être représenté par une liste d'écoute (*playlist*) ordonnée de fichiers audio.

Un livre étant une liste d'écoute de fichiers, la structure de TOC_Playlist MultiAudio de l'OSTA permet de représenter un livre audio. Tout lecteur compatible MultiAudio pourrait alors lire un livre audio en le reconnaissant simplement comme une TOC_Playlist, et en jouant cette liste. Cela permet aux lecteurs qui ne reconnaissent pas les structures de données de cette Norme internationale de lire tout de même un livre audio avec des capacités de navigation minimales. La structure de AB_Book a été conçue afin d'étendre la navigation et de fournir des métadonnées supplémentaires.

Chaque livre audio du support doit dans la table des matières MultiAudio (*MultiAudio TOC*) être représenté comme une liste d'écoute TOC_Playlist. Dans la structure de TOC_Playlist de chaque livre audio, il doit y avoir une structure de AB_Book située dans la zone Extra Data.

La structure de AB_Book est définie comme une structure AB_ExtStructure présente dans une structure de Bloc (telle qu'elle est définie en 4.2 au sein du champ Extra Data d'une liste d'écoute TOC_Playlist MultiAudio). Le nombre ordinal de ce Bloc n'est pas défini.

Tableau 11 – Format de AB_Book

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise	balise
2	12	4	Décalage des données supplémentaires (<i>Extra Data</i>)	Uint32
3	16	L_Fc	Bloc fixe de AB_Book	AB_FixedChunk
4	16 + L_Fc	L_Vc	Bloc variable de AB_Book	AB_VariableChunk
5	16 + L_Fc + L_Vc	L_Ex	Données supplémentaires (<i>Extra Data</i>)	ExtraData

7.2 Balise

Tag.Identifier doit avoir la valeur hexadécimale 4B424241h (ABBK) pour désigner une structure de AB_Book.

Tag.OrdinalNumber doit prendre la valeur 0.

Tag.Length doit avoir pour valeur la longueur de la structure de AB_Book, longueur de la Balise comprise.

7.3 Bloc fixe de AB_Book

7.3.1 Généralités

Le bloc fixe de AB_Book doit contenir le format défini dans le Tableau 12.

Tableau 12 – Format du bloc fixe de AB_Book

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise	balise
2	12	2	Format texte (Text Format)	Uint16
3	14	2	Indicateurs	Uint16
4	16	2	Index de TOC_Playlist	Uint16
5	18	2	Décalage du titre de livre	Uint16
6	20	2	Décalage de l'auteur	Uint16
7	22	2	Décalage du narrateur	Uint16
8	24	2	Décalage de la déclaration de Copyright	Uint16
9	26	2	Décalage de l'identificateur unique universel de livre audio	Uint16
10	28	2	Décalage du numéro ISBN du livre	Uint16
11	30	2	Décalage du numéro ISSN du livre	Uint16
12	32	2	Décalage de l'identificateur DOI du livre	Uint16
13	34	2	Décalage du numéro d'index du livre	Uint16
14	36	2	Décalage de la cote du livre	Uint16
15	38	2	Décalage du numéro de référence du livre	Uint16
16	40	2	Décalage d'un autre identificateur	Uint16
17	42	2	Décalage du premier niveau de livre audio	Uint16
18	44	4	Décalage de l' identificateur d'emplacement de contenu de liste d'écoute	Uint32

7.3.2 Balise

Le sous-champ Tag.Identifier doit prendre la valeur décimale 9FFF0001h pour indiquer qu'il s'agit d'un bloc AB_FixedChunk.

Le sous-champ Tag.OrdinalNumber doit avoir la valeur 0.

Tag.Length doit avoir la longueur de la structure de bloc fixe de AB_Book en octets. Pour cette Norme internationale, la longueur est de 48 octets.

7.3.3 Format texte (*Text format*)

Si ce champ est défini par 0, tous les champs de texte doivent être interprétés comme des chaînes ASCII.

Si ce champ est défini par 1, tous les champs de texte doivent être interprétés comme des chaînes UNICODE.

7.3.4 Indicateurs (*Flags*)

Ce champ doit prendre la valeur 0 pour cette Norme internationale.

7.3.5 Index de TOC_Playlist

Ce champ doit contenir l'index de la liste d'écoute TOC_Playlist dans la table des matières MultiAudio de l'OSTA qui représente le livre audio et contient la structure de AB_Book.

7.3.6 Décalage du titre du livre

Le décalage du titre du livre doit être défini comme le décalage entre le début de la structure de AB_Book et le titre du livre dans le champ de Bloc variable de AB_Book. Le Titre du livre étant un champ obligatoire, ce décalage doit toujours être défini, voir 7.4.3.

7.3.7 Décalage de l'auteur

Le décalage de l'auteur doit être défini comme le décalage entre le début de la structure de AB_Book et l'auteur du livre dans le champ de Bloc variable de AB_Book. L'Auteur étant un champ obligatoire, ce décalage doit toujours être défini, voir 7.4.4.

7.3.8 Décalage du narrateur

Le décalage du narrateur doit être défini comme le décalage entre le début de la structure de AB_Book et le narrateur du livre dans le champ de Bloc variable de AB_Book. Le Narrateur étant un champ obligatoire, ce décalage doit toujours être défini, voir 7.4.5.

7.3.9 Décalage de la déclaration de copyright

Le décalage de la déclaration de copyright doit être défini comme le décalage entre le début de la structure de AB_Book jusqu'à la déclaration de copyright dans le champ de Bloc variable de AB_Book. Si le livre ne possède pas de déclaration de copyright, il convient de donner à ce champ la valeur 0, voir 7.4.6.

7.3.10 Décalage de l'identificateur unique universel de livre audio (*audiobook UUID*)

Le décalage de l'identificateur unique universel (*UUID (universally unique identifier)*) doit être défini comme le décalage entre le début de la structure de AB_Book et l'identificateur unique des livres audio. Si le livre audio ne possède pas d'identificateur unique, ce champ doit prendre la valeur 0, voir 7.4.7.

7.3.11 Décalage du numéro ISBN du livre

Le décalage du numéro ISBN (International Standard Book Number) du livre doit être défini comme le décalage entre le début de la structure de AB_Book et le numéro ISBN du livre dans le champ de Bloc variable de AB_Book. Si le livre ne possède pas de numéro ISBN, il convient que ce champ prenne la valeur 0, voir 7.4.8.

7.3.12 Décalage du numéro ISSN du livre

Le décalage du numéro ISSN (International Standard Serial Number) du livre doit être défini comme le décalage entre le début de la structure de AB_Book et la chaîne du numéro ISSN du livre dans le champ de Bloc variable de AB_Book. Si le livre ne possède pas de numéro ISSN, il convient que ce champ prenne la valeur 0, voir 7.4.10.

7.3.13 Décalage de l'identificateur DOI du livre

Le décalage de l'identificateur DOI (Digital Object Identifier) doit être défini comme le décalage entre le début de la structure de AB_Book et la chaîne de l'identificateur DOI du livre dans le champ de Bloc variable de AB_Book. Si le livre ne possède pas d'identificateur DOI, il convient que ce champ prenne la valeur 0, voir 7.4.10.

7.3.14 Décalage du numéro d'index du livre (*book index number*)

Le décalage du numéro d'index du livre doit être défini comme le décalage entre le début de la structure de AB_Book jusqu'au numéro d'index du livre dans le champ de Bloc variable de AB_Book. Si le livre ne possède pas de numéro d'index, il convient que ce champ prenne la valeur 0, voir 7.4.11.

7.3.15 Décalage de la cote du livre (*book shelf number*)

Le décalage de la cote du livre doit être défini comme le décalage entre le début de la structure de AB_Book et la cote du livre dans le champ de Bloc variable de AB_Book. Si le livre ne possède pas de cote, il convient que ce champ prenne la valeur 0, voir 7.4.12.

7.3.16 Décalage du numéro de référence du livre (*book part number*)

Le décalage du numéro de référence du livre doit être défini comme le décalage entre le début de la structure de AB_Book et le numéro de référence du livre dans le champ de Bloc variable de AB_Book. Si le livre ne possède pas de numéro de référence, il convient que ce champ prenne la valeur 0, voir 7.4.13.

7.3.17 Décalage d'un autre identificateur

Le décalage d'un autre identificateur doit être défini comme le décalage entre le début de la structure de AB_Book jusqu'à l'identificateur de chaîne non défini supplémentaire du livre dans le champ de Bloc variable de AB_Book. Si le livre ne possède pas d'identificateur supplémentaire, il convient que ce champ prenne la valeur 0, voir 7.4.14.

7.3.18 Décalage du premier niveau de livre audio (*AudioBook Level, AB_Level*)

Le décalage du Premier niveau de livre audio doit être défini comme le décalage entre le début de la structure de AB_Book et la structure du premier AB_Level dans le champ de Bloc variable de AB_Book. Tous les livres audio doivent contenir une structure de AB_Level pour définir l'intégralité du livre audio et ce décalage doit donc toujours être défini, voir 7.4.15.

7.3.19 Décalage de l'identificateur de l'emplacement du contenu de la liste d'écoute

Le décalage de l'identificateur d'emplacement du contenu d'une liste d'écoute doit être défini comme le décalage entre le début de la structure de AB_Book jusqu'à l'identificateur de l'emplacement du contenu de la liste d'écoute dans le champ de Bloc variable de AB_Book. Si

le champ n'existe pas ou est indéfini (null), il convient que ce champ prenne la valeur 0, voir 7.4.15.

7.4 Bloc (*Chunk*) variable de AB_Book

7.4.1 Généralités

Le bloc variable de AB_Book doit contenir le format défini dans le Tableau 13.

Tableau 13 – Format du bloc variable de AB_Book

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise	balise
2	12	L_Bt	Titre du livre (<i>book title</i>)	octet
3	12 + L_Bt	L_Au	Auteur	octet
4	12 + L_Bt + L_Au	L_Na	Narrateur	octet
5	12 + L_Bt + L_Au + L_Na	L-Cs	Déclaration de Copyright	octet
6	12 + L_Bt + L_Au + L_Na + L-Cs	L_Ui	Identificateur unique universel du livre	octet
7	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui	L_Is	Numéro ISBN du livre	octet
8	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is	L_In	Numéro ISSN du livre	octet
9	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In	L_Do	Identificateur DOI du livre	octet
10	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do	L_In	Numéro d'index du livre (<i>index number</i>)	octet
11	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do + L_In	L_Sn	Cote du livre (<i>shelf number</i>)	octet
12	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do + L_In + L_Sn	L_Pn	Numéro de référence du livre (<i>part number</i>)	octet
13	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do + L_In + L_Sn + L_Pn	L_Oi	Autre identificateur	octet
14	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do + L_In + L_Sn + L_Pn + L_Oi	L_Sc	Premier niveau de livre audio	octet
15	12 + L_Bt + L_Au + L_Na + L-Cs + L_Ui + L_Is + L_In + L_Do + L_In + L_Sn + L_Pn + L_Oi + L_Li + L_Sc	L_Li	Identificateur d'emplacement de contenu	Uint32

7.4.2 Balise

Il convient que le sous-champ Tag.Identifier prenne la valeur hexadécimale 9FFF0002h pour indiquer qu'il s'agit d'un bloc AB_VariableChunk.

Le sous-champ Tag.OrdinalNumber doit être mis à la valeur 1.

Tag.Length doit avoir pour valeur la longueur de la structure de bloc variable de AB_Header en octets.

7.4.3 Titre du livre (*Book title*)

Chaque livre audio doit contenir un champ titre du livre. Ce champ doit contenir le Titre du livre en fonction du jeu de caractères défini dans le champ de Format texte (*Text Format*) situé dans le Bloc fixe de AB_Book. S'il n'y a pas de titre du livre, le champ de titre doit être rempli par un caractère de terminaison de chaîne. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

Si la structure de AB_Book est située dans une structure de TOC_Playlist MultiAudio, le Titre du livre doit être la même chaîne que le champ nom de liste d'écoute (*Playlist Name*) dans TOC_Playlist.

Si la structure de AB_Book est située dans une structure de TOC_TrackEntry MultiAudio, le Titre du livre doit être la même chaîne que le champ nom d'album (*Album Name*) dans TOC_TrackEntry.

7.4.4 Auteur

Chaque livre audio doit contenir un champ d'auteur. Ce champ doit contenir l'Auteur en fonction du jeu de caractères défini dans le champ de Format texte (*Text Format*) situé dans le Bloc fixe de AB_Book. S'il n'y a pas d'auteur, le champ Auteur doit être rempli par un caractère de terminaison de chaîne. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

Si la structure de AB_Book est située dans une structure de TOC_TrackEntry MultiAudio, l'Auteur doit être la même chaîne que le champ nom d'auteur (*Songwriter Name*) dans TOC_TrackEntry.

7.4.5 Narrateur

Chaque livre audio doit contenir un champ de narrateur. Ce champ doit contenir le Narrateur en fonction du jeu de caractères défini dans le champ de Format texte (*Text Format*) situé dans le Bloc fixe de AB_Book. S'il n'y a pas d'auteur, le champ Narrateur doit être rempli par un caractère de terminaison de chaîne. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

Si la structure de AB_Book est située dans une structure de TOC_TrackEntry MultiAudio, le Narrateur doit être la même chaîne que le champ de nom d'interprète (*Performer Name*) dans TOC_TrackEntry.

7.4.6 Déclaration de copyright

Ce champ doit contenir la déclaration de copyright en fonction du jeu de caractères défini dans le champ de Format texte (*Text Format*) situé dans la structure de Bloc fixe de AB_Book. S'il n'existe pas de déclaration de copyright, ce champ ne doit pas exister. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

7.4.7 Identificateur unique universel du livre

Un format d'Identificateur unique universel a été défini dans la norme RPC d'environnement informatique distribué de l'Open Software Foundation, également disponible sous ISO/CEI 11578, qui définit les identificateurs uniques universels en annexes.

Ce champ contient un Identificateur unique universel dans un format de chaîne de 36 octets prévu par cette spécification. Si ce champ est présent, l'Identificateur unique universel n'est pas terminé par la valeur NULL (00h). Etant donné que le champ doit être d'une longueur exacte de 36 octets, le bourrage n'est pas nécessaire.

Un livre audio peut ne pas inclure ce champ. Ce champ est recommandé mais non obligatoire et permet aux appareils ainsi qu'aux éditeurs d'identifier de façon unique un livre audio numérique.

7.4.8 Numéro ISBN du livre

Ce champ doit contenir le numéro ISBN du livre audio sous forme d'une chaîne de d-caractères, comme il est défini dans la spécification MultiAudio. Si le livre ne possède pas

de numéro ISBN, ce champ ne doit pas exister. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

7.4.9 Numéro ISSN du livre

Ce champ doit contenir le numéro ISSN du livre audio sous forme d'une chaîne de d-caractères, comme défini dans la spécification MultiAudio. Si le livre ne possède pas de numéro ISSN, ce champ ne doit pas exister. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

7.4.10 Identificateur DOI du livre

Ce champ doit contenir l'identificateur DOI du livre audio sous forme d'une chaîne de d-caractères, comme il est défini dans la spécification MultiAudio. Si le livre ne possède pas d'identificateur DOI, ce champ ne doit pas exister. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

7.4.11 Numéro d'index du livre

Ce champ doit contenir le numéro d'index du livre audio sous forme d'une chaîne de d-caractères, comme il est défini dans la spécification MultiAudio. Si le livre ne possède pas de numéro d'index, ce champ ne doit pas exister. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

7.4.12 Cote du livre (*Book shelf number*)

Ce champ doit contenir la cote du livre audio sous forme d'une chaîne de d-caractères, comme il est défini dans la norme MultiAudio OSTA. Si le livre ne possède pas de cote, ce champ ne doit pas exister. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

7.4.13 Numéro de référence du livre (*Book part number*)

Ce champ doit contenir le numéro de référence du livre audio sous forme d'une chaîne de d-caractères, comme défini dans la spécification MultiAudio. Si le livre ne possède pas de numéro de référence, ce champ ne doit pas exister. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

7.4.14 Autre identificateur

Ce champ doit contenir un identificateur de livre audio supplémentaire sous forme d'une chaîne de d-caractères, comme défini dans la spécification MultiAudio. Cet identificateur supplémentaire n'est pas défini par la spécification, et peut être défini par toute valeur aidant à identifier un livre audio de façon unique. Si le livre ne possède pas d'identificateur supplémentaire, ce champ ne doit pas exister. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

7.4.15 Premier niveau de livre audio (*First audiobook level*)

Chaque structure de AB_Book doit contenir une structure de AB_Level qui définit l'ensemble du livre. Le livre est ensuite divisé en sous-niveaux plus petits de ce niveau. La structure de AB_Level qui représente le livre audio complet doit comporter les champs indiqués dans le Tableau 14.

Tableau 14 – Paramètres de champ requis pour la structure de AB_Level

Numéro de paragraphe	Champ AB_Level	Valeur
8.3.2	Tag.OrdinalNumber	0
8.3.6	Numéro de niveau (<i>Level Number</i>)	0
8.3.14	Décalage du AB_Level parent	.
		0
8.3.11	Décalage temporel du livre	0
8.3.12	Décalage de heure de début	0
8.3.13	Durée	Longueur du livre audio en millisecondes
8.3.15	Décalage du AB_Level précédent	0
8.3.16	Décalage du AB_Level suivant	0

Voir l'Article 8 pour plus d'informations sur la structure de AB_Level.

7.4.16 Identificateur de l'emplacement du contenu liste d'écoute (*Playlist content location identifier*)

Ce champ doit contenir l'identificateur de l'emplacement du contenu liste d'écoute. Il s'agit d'un nombre indiquant l'emplacement des fichiers de cette liste d'écoute (l'emplacement peut être local, en ligne sous forme d'un contenu téléchargeable, en ligne sous forme d'un contenu en streaming ou une combinaison de ces emplacements). Ce nombre peut être obtenu dans le Tableau 15. Si ce champ est indéfini (null) ou inexistant, il convient de supposer la valeur 0 (local).

Tableau 15 – Paramètres de champ d'identificateur de l'emplacement de contenu

Identificateur d'emplacement de contenu de liste d'écoute	Valeur
Local	0
Indéfini	1
Aucun contenu	2
Contenu téléchargeable	3
Contenu en streaming	4
Contenu mixte – Local/Téléchargé	5
Contenu mixte – Local/Streaming	6
Contenu mixte – Téléchargé/Streaming	7
Contenu mixte – Local/Téléchargé/Streaming	8

7.5 Données supplémentaires de AB_Book (*AB_Book extra data*)

Les données supplémentaires de AB_Book sont définies comme un bloc (*Chunk*) contenant une matrice de structures à base de blocs (*Chunk based structures*). Cette zone permet d'apporter une extension supplémentaire de la structure de AB_Book.

8 Niveau de livre audio (AB_Level)

8.1 Généralités

Un niveau AB_Level est une nouvelle structure qui n'existe pas dans MultiAudio de l'OSTA. Un AB_Level représente l'unité logique d'un livre audio, avec une heure de début et une

durée en millisecondes du fichier audio ou du groupe de fichiers audio. Chaque niveau peut représenter un nombre infini de sections de sous-niveaux plus petits. Par conséquent, chaque niveau peut représenter le livre complet, un chapitre, une sous-section ou même une page. Chaque livre audio contient un parent de niveau le plus élevé qui est défini dans la structure de AB_Book. Cet article représente l'ensemble du livre audio. Au sein de cette section de niveau le plus élevé, un nombre illimité de sous-sections sont définies.

Si un AB_Level représente un groupe de fichiers audio, il doit contenir des sous-niveaux enfant qui référencent la structure de TOC_TrackEntry de chaque fichier audio du groupe. Un niveau AB_Level renvoie à un fichier audio en employant le premier champ de l'index TOC_TrackEntry dans le champ AB_Level Fixed Chunk. Etant donné que ce champ ne contient qu'un index TOC_TrackEntry, il serait impossible à une structure de AB_Level unique de référencer plusieurs fichiers audio sans définir des sous-niveaux enfant. Par exemple, si un livre audio est divisé en 3 fichiers audio séparés, il est impossible que la structure de AB_Level de niveau le plus élevé (Numéro de niveau = 0) ne définisse aucun sous-niveau enfant. Au minimum, cette structure de niveau le plus élevé doit posséder 3 sous-niveaux qui référencent la structure TOC_TrackEntry de chaque fichier audio.

La structure AB_Level est définie comme une AB_ExtStructure. Il n'est pas nécessaire de renfermer la structure de AB_Level dans une structure de bloc (*Chunk*) comme définie au tableau 2, parce qu'elle n'est pas directement contenue à l'intérieur d'un champ de données supplémentaires des structures MultiAudio (*MultiAudio structures Extra Data*). Le format de AB_Section doit être comme défini dans le Tableau 16.

Tableau 16 – Format de AB_Section

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise	balise
2	12	4	Décalage des données supplémentaires (<i>Extra Data</i>)	Uint32
2	16	L_Fc	Bloc fixe de AB_Level	AB_FixedChunk
3	16 + L_Fc	L_Vc	Bloc variable de AB_Level	AB_VariableChunk
4	16 + L_Fc + L_Vc	L_Ex	Données supplémentaires (<i>Extra Data</i>)	ExtraData

8.2 Balise

Le Tag.Identifier doit être la valeur hexadécimale 564C4241h (ABLV) pour indiquer une structure de AB_Level.

Tag.OrdinalNumber doit être la valeur 0.

Tag.Length doit être la longueur de la structure AB_Level en octets, longueur de la Balise comprise.

8.3 Bloc fixe de AB_Level

8.3.1 Généralités

Le bloc fixe de AB_Level doit contenir le format défini dans le Tableau 17.

Tableau 17 – Format du bloc fixe de AB_Level

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise	balise
2	12	2	Format texte (<i>Text Format</i>)	Uint16
3	14	2	Indicateurs	Uint16
4	16	2	Nombre de sous-niveaux (N_Sc)	Uint16
5	18	2	Numéro de niveau (<i>Level Number</i>)	Uint16
6	20	2	Premier index TOC_TrackEntry	Uint16
7	22	2	Décalage du bloc de sous-niveau	Uint16
8	24	2	Décalage du nom de niveau	Uint16
9	26	2	Décalage de la description de niveau	Uint16
10	28	4	Décalage temporel du livre	Uint32
11	32	4	Décalage d'heure de début	Uint32
12	36	4	Durée	Uint32
13	40	4	Décalage du AB_Level parent	Uint32
14	44	4	Décalage du AB_Level précédent	Uint32
15	48	4	Décalage du AB_Level suivant	Uint32

8.3.2 Balise

Le sous-champ Tag.Identifier doit être la valeur décimale 9FFF0001h pour indiquer qu'il s'agit d'un bloc AB_FixedChunk.

Le sous-champ Tag.OrdinalNumber doit être la valeur 0.

Tag.Length doit être la longueur de la structure de bloc fixe de AB_Level en octets. Pour cette Norme internationale, la longueur est de 52 octets.

8.3.3 Format texte (*Text format*)

Si ce champ est défini par 0, tous les champs de texte doivent être interprétés comme des chaînes ASCII.

Si ce champ est défini par 1, tous les champs de texte doivent être interprétés comme des chaînes UNICODE.

8.3.4 Indicateurs

Les indicateurs doivent être définis à la valeur 0 pour la présente Norme internationale.

8.3.5 Nombre de sous-niveaux

Ce champ doit contenir le nombre de sous-niveaux que ce niveau représente. S'il n'existe pas de sous-niveau, ce champ doit être 0.

8.3.6 Numéro de niveau (*Level Number*)

Ce champ doit contenir le numéro du niveau. Pour le niveau le plus élevé, cette valeur doit être définie par 0, tandis que chaque sous-niveau a une valeur supérieure d'un par rapport au parent. Le AB_Level de niveau 0 représente le livre audio complet présent dans la structure de AB_Book.

8.3.7 Index de TOC_TrackEntry (*TOC_TrackEntry index*)

Chaque AB_Level représente une partie d'un fichier audio, ou un groupe de fichiers audio. Chaque fichier audio est défini par une structure de TOC_TrackEntry MultiAudio. Ce champ doit contenir l'index de la structure de TOC_TrackEntry dans le fichier table des matières MultiAudio (*MultiAudio TOC*) qui est le premier fichier audio encapsulé par cet AB_Level. Si le AB_Level est une partie d'un fichier, l'index doit être l'index de la TOC_TrackEntry à laquelle le niveau appartient. Si le AB_Level représente un groupe de fichiers audio, ce champ doit contenir l'index de TOC_TrackEntry du premier fichier audio du groupe.

8.3.8 Décalage du bloc de sous-niveau (*Offset to sub-level chunk*)

Le décalage du bloc de sous-niveau doit être défini comme le décalage en octets du bloc de sous-niveau dans le champ de bloc variable de AB_Level (*AB_Level Variable Chunk*) par rapport au début de la structure de AB_Level. S'il n'existe pas de sous-niveaux, ce champ doit être 0.

8.3.9 Décalage du nom de niveau (*Offset to level name*)

Le décalage du nom de niveau doit être défini comme le décalage du nom de niveau (*Level Name*) par rapport au début de la structure de AB_Level. Ce champ ne doit pas être 0 car le nom de niveau est obligatoire, voir 8.4.3.

8.3.10 Décalage de la description de niveau (*Offset to level description*)

Le décalage de la description de niveau doit être défini comme le décalage de la description de niveau par rapport au début de la structure de AB_Level. Si le niveau ne possède pas de description, ce champ doit être la valeur 0, voir 8.4.4.

8.3.11 Décalage temporel du livre

Le décalage temporel du livre doit indiquer le décalage temporel en millisecondes à partir du début du livre audio jusqu'au début de ce niveau.

8.3.12 Décalage d'heure de début

Le décalage d'heure de début doit indiquer le décalage temporel en millisecondes à partir du début d'un fichier audio jusqu'au début de ce niveau. Si le niveau représente un fichier audio, tous les sous-niveaux doivent commencer à une valeur égale ou supérieure à cette valeur. Si le niveau représente un groupe de fichiers audio, ce champ doit être 0.

8.3.13 Durée

La durée doit indiquer la durée de ce niveau en millisecondes. Si le niveau représente un fichier audio, la durée ne doit pas dépasser le Time Start Offset (décalage d'heure de début) du niveau suivant ou la longueur du fichier audio complet en millisecondes. Si le niveau représente un groupe de fichiers audio, la durée ne doit pas dépasser la durée totale du livre audio en millisecondes ou le Décalage temporel du livre de niveau suivant.

8.3.14 Décalage du AB_Level parent (*Offset to parent AB_Level*)

Il s'agit du décalage entre le début de la structure de AB_Level et le parent de ce niveau en octets. Etant donné que le niveau parent précède toujours ce niveau dans un fichier MultiAudio, cela constitue un décalage négatif. Ce champ est défini par une valeur non signée (UInt32), et l'analyseur doit donc revenir en arrière du nombre d'octets indiqué et non avancer.

Si le niveau ne comporte pas de niveau parent, ce champ doit être la valeur 0. Cela ne doit se produire qu'au niveau le plus haut (Numéro de niveau (*Level Number*) = 0), qui est le premier AB_Level défini dans la structure de AB_Book.

8.3.15 Décalage du AB_Level précédent

Il s'agit du décalage en octets à partir du début de la structure de AB_Level jusqu'au niveau précédent sur le même Numéro de niveau. Si c'est le premier niveau pour le Numéro de niveau en cours, ce champ doit être 0. Etant donné que le niveau précédent est toujours défini avant le AB_Level dans le fichier MultiAudio, cela constitue un décalage négatif. L'analyseur doit par conséquent sauter en arrière le nombre d'octets indiqué, au lieu d'avancer, dans le fichier.

8.3.16 Décalage du AB_Level suivant (*Offset to next AB_Level*)

Il s'agit du décalage en octets à partir du début de la structure de AB_Level jusqu'au niveau suivant sur le même Numéro de niveau. S'il s'agit du dernier niveau sur le Numéro de niveau en cours, ce champ doit être la valeur 0.

8.4 Bloc variable de AB_Level

8.4.1 Généralités

Le bloc variable de AB_Level doit contenir le format défini dans le Tableau 18.

Tableau 18 – Format du bloc variable de AB_Level

#	RBP	Longueur	Nom	Contenu
1	0	12	Balise	balise
3	12	L_Sn	Nom de niveau	octet
4	12 + L_Sn	L_Sd	Description du niveau	octet
6	12 + L_Sn + L_Sd	L_Sc	Bloc de sous-niveau	LevelChunk

8.4.2 Balise (*Tag*)

Il est recommandé que le sous-champ Tag.Identifier prenne la valeur hexadécimale 9FFF0002h pour indiquer qu'il est un bloc AB_VariableChunk.

Le sous-champ Tag.OrdinalNumber doit avoir la valeur 1.

Tag.Length doit être la longueur de la structure de bloc variable de AB_Level en octets.

8.4.3 Nom de niveau (*Level name*)

Ce champ doit contenir le nom du niveau exprimé dans le jeu de caractères défini par le champ de Format texte (*Text Format*) dans la structure de Bloc fixe de AB_Level (*AB_Level Fixed Chunk*). Il convient que le nom de niveau soit une chaîne courte décrivant brièvement et avec précision ce que le niveau représente. Si le niveau ne porte pas de nom, le champ doit être rempli par un caractère de terminaison de chaîne. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

8.4.4 Description du niveau (*Level description*)

Ce champ doit contenir la description du niveau exprimée dans le jeu de caractères défini par le champ de Format texte (*Text Format*) dans la structure de Bloc fixe de AB_Level (*AB_Level Fixed Chunk*). La description du niveau est une chaîne qui définit plus en détail ce que le niveau représente. Si le niveau ne possède pas de description, ce champ ne doit pas exister. Il peut être nécessaire de remplir ce champ par du bourrage. Voir 5.5 pour plus d'informations sur le bourrage.

8.4.5 Bloc de sous-niveau (*Sub-level chunk*)

Le bloc de sous-niveau doit être défini comme un Bloc contenant une matrice de structures AB_Level qui représente les sous-niveaux de la structure de AB_Level. Si AB_Level ne possède pas de sous-niveaux, ce champ ne doit pas exister. Les champs suivants doivent être définis pour la structure de Bloc:

- a) Le sous-champ LevelChunk.Tag.Identifier doit avoir la valeur hexadécimale 9FFF0003h pour indiquer un LevelChunk.
- b) Le sous-champ LevelChunk.Tag.OrdinalNumber doit avoir la valeur 0.
- c) LevelChunk.Tag.Length doit être la longueur en octets de tous les sous-niveaux plus la longueur en octets de la balise.

8.5 Données supplémentaires de AB_Level (*AB_Level extra data*)

Les données supplémentaires de AB_Level doivent être définies comme un Bloc contenant une matrice de structures à base de blocs (*Chunk*). Cette zone permet d'apporter une extension supplémentaire à la structure de AB_Level.

9 Profils MPV audiobook

9.1 Généralités

Cet article décrit la version sur base XML des exigences relatives au livre audio et repose sur une adaptation de la spécification MPV Music de l'OSTA.

La spécification de noyau MPV (MPV-Core) définit le cadre indispensable de la technologie MPV. Le profil musical MPV 1.0 (MPV-Music) étend MPV-Core afin de spécifier comment créer, gérer et lire des listes d'écoute (*playlist*) de morceaux de musique.

Un mécanisme de base de MPV, le profil, définit une nouvelle fonctionnalité importante. La spécification MPV-Core spécifie l'utilisation du profil MPV. MPV-Core requiert qu'un fichier MPV déclare quels profils il met en œuvre et déclare les espaces de noms (*namespaces*) des profils. Cela permet d'utiliser une application de traitement afin de déterminer rapidement si un fichier MPV donné trouve les traitements qu'il attend.

Le profil de livre audio utilise le profil musical MPV 1.0 car la plupart des profils musicaux MPV sont applicables aux livres audio et parce que prendre appui sur le profil musical MPV améliore la compatibilité en lecture des livres audio. MPV-Music comprend des schémas et des exemples pratiques.

Le profil de livre audio (MPV-Audiobook) est une extension de MPV-Music et est entièrement compatible avec elle. Il convient que les fichiers implémentant le profil MPV-Audiobook soient utilisables dans des applications et dispositifs reconnaissant MPV-Music. Cela signifie, par exemple qu'une application ou un appareil de lecture MPV-Music peut lire des livres audio CEA; cependant, un appareil de lecture ne reconnaissant que le profil MPV-Music ignore le profil MPV-Audiobook et le ressentira lors de la lecture du fichier sera moins complet qu'avec un lecteur sachant traiter les MPV-audiobook.

9.2 Identificateur d'espace de noms en schéma XML

La présente Norme internationale définit un identificateur d'espace de noms de schéma à utiliser pour identifier le schéma XML des livres audio, voir le Tableau 19.

Tableau 19 – Identificateur d'espace de noms de schéma

Identificateur d'espace de noms	http://www.cei.org/CEI-2003-DTD/1.0/
Préfixe d'espace de noms conventionnel	ab:

Cet identificateur doit être utilisé dans la valeur de déclaration d'espace de noms (*namespace declaration value*) lorsque l'on définit des préfixes d'espace de noms ou que l'on déclare l'espace de noms d'un élément.

9.3 Utilisation d'un identificateur de profil MPV-Music

Pour être compatible avec les lecteurs Musique MPV, un fichier MPV de livre audio doit également comprendre la déclaration du profil Musique MPV dans la section Profils du manifeste MPV dans le Tableau 20. Un exemple est donné au Tableau 20.

Tableau 20 – Identificateur du profil MPV-Music

Identificateur du profil MPV Music	http://ns.osta.org/mpv/music/1.0/
------------------------------------	---

Un exemple de l'utilisation d'un identificateur de profil MPV-Music est présenté ci-dessous.

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

<file:Manifest

  xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:mpv="http://ns.osta.org/mpv/1.0/"
  xmlns:mpvp="http://ns.osta.org/mpv/presentation/1.0/"
  xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"
  xmlns:ab="http://ns.cei.org/audiobook/2003b/"
  xmlns:nmf="http://ns.osta.org/nmf/1.0/" >

<nmf:Metadata>

  <ManifestProperties xmlns="http://ns.osta.org/manifest/1.0/">

    <ProfileBag>

      <Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>

      <Profile>http://ns.osta.org/mpv/music/1.0/</Profile>

      <Profile>http://ns.cei.org/audiobook/2003b/</Profile>

    </ProfileBag>

  </ManifestProperties>
```

</nmf:Metadata>

</file:Manifest>

9.4 Fichier de livre audio

Un fichier de livre audio désigne un fichier MPV conforme au profil MPVMusic qui déclare aussi le profil MPV-Audiobook. Un fichier de livre audio doit aussi contenir un élément <mpvm:Album> unique qui définit le livre audio et contient un jeu d'actifs (*assets*) décrits dans l'élément <mpv:AssetList> qui sont référencés directement ou indirectement par le contenu du <mpvm:Album>.

9.5 Métadonnées du profil de livre audio

MVP-Audiobook étend et développe les métadonnées définies dans MPVMusic. MPV-Audiobook ajoute des métadonnées particulières aux livres audio.

Les métadonnées MPV-Audiobook et les métadonnées MPV-Music peuvent être utilisées en tant que métadonnées affectées à des actifs MPV (*MPV assets*) individuels. Dans MPV-Audiobook, chaque actif audio MPV correspond à un chapitre du livre.

Les métadonnées peuvent également être utilisées avec la liste d'écoute de présentation MPV (*MPV presentation playlist*) ("nom d'élément <mpvp:album>") afin de décrire en une fois les métadonnées pour tout le livre audio au lieu de les dupliquer pour chaque actif d'un chapitre du livre audio. La liste d'écoute de présentation MPV sert à mettre une couche de présentation au-dessus des actifs audio individuels. L'album de présentation MPV définit le livre audio comme un tout.

Le livre audio MPV (*MPV-Audiobook*) doit inclure les nouvelles propriétés définies dans le Tableau 21.

Ces propriétés sont spécifiques aux livres audio. Elles peuvent être utilisées dans un élément <nmf:Metadata> présent sur n'importe quel actif MPV (tel que mpv:Audio) ou sur l'élément de présentation MPV <mpvp:Album>.

Tableau 21 – Propriétés du profil de livre audio MPV

Métadonnées	Profil de livre audio	Description
Titre du livre	nmf:Metadata ab:Properties ab:BookTitle	Métadonnées principales du titre du livre en cours de lecture. Lorsqu'elles sont utilisées sur un actif (asset), elles se substituent aux valeurs de métadonnées spécifiées sur mpvp:Album.
Auteur du livre	nmf:Metadata ab:Properties ab:BookAuthor	Métadonnées principales concernant l'auteur du livre en cours de lecture. Lorsqu'elles sont utilisées sur un actif, elles se substituent aux valeurs de métadonnées spécifiées sur mpvp:Album.
Co-auteur du livre	nmf:Metadata ab:Properties BookCoauthor	Métadonnées principales concernant le co-auteur du livre en cours de lecture. Lorsqu'elles sont utilisées sur un actif, elles se substituent aux valeurs de métadonnées spécifiées sur mpvp:Album.
Sous-titre du livre	nmf:Metadata ab:Properties BookSubtitle	Métadonnées principales concernant le sous-titre du livre en cours de lecture. Lorsqu'elles sont utilisées sur un actif, elles se substituent aux valeurs de métadonnées spécifiées sur mpvp:Album.
Narrateur du livre	nmf:Metadata ab:Properties ab:BookNarrator	Métadonnées principales concernant le narrateur du livre en cours de lecture. Lorsqu'elles sont utilisées sur un actif, elles se substituent aux valeurs de métadonnées spécifiées sur mpvp:Album.
Genre du livre	nmf:Metadata ab:Properties BookGenre	Métadonnées principales concernant le genre du livre en cours de lecture. Lorsqu'elles sont utilisées sur un actif, elles se substituent aux valeurs de métadonnées spécifiées sur mpvp:Album.
Identificateur unique universel du livre	nmf:Metadata ab:Properties ab:BookUUID	Métadonnées principales concernant l'identificateur unique universel du livre.
Numéro ISBN du livre	nmf:Metadata ab:Properties ab:BookISBN	Métadonnées principales concernant le numéro ISBN du livre.
Numéro ISSN du livre	nmf:Metadata ab:Properties ab:BookISSN	Métadonnées principales concernant le numéro ISSN du livre.
Identificateur DOI du livre	nmf:Metadata ab:Properties ab:BookDOI	Métadonnées principales concernant l'identificateur DOI du livre.
Numéro d'index du livre	nmf:Metadata ab:Properties ab:BookIndexNumber	Métadonnées principales concernant le numéro d'index du livre.
Cote du livre	nmf:Metadata ab:Properties ab:BookShelfNumber	Métadonnées principales concernant la cote du livre.
Numéro de référence du livre	nmf:Metadata ab:Properties ab:BookPartNumber	Métadonnées principales concernant le numéro de référence du livre.
Autre identificateur	nmf:Metadata ab:Properties ab:OtherIdentifier	Métadonnées principales concernant un autre identificateur du livre.
Identificateur d'emplacement de contenu	nmf:Metadata ab:Properties ab:ContentLocationIdentifier	Métadonnées principales concernant l'identificateur d'emplacement de contenu de liste d'écoute.

9.6 Métadonnées du profil de livre audio MPV

Comme il est décrit dans MPV-Music, une gamme de métadonnées est disponible pour décrire les livres audio. Le Tableau 22 répertorie ces propriétés et en donne une interprétation pour les livres audio.

Tableau 22 – Métadonnées du profil de livre audio MPV

Métadonnées	Éléments de métadonnées dans MPV	Description concernant les livres audio
Chapitre d'actif de livre audio	Sous-éléments de l'actif mpv:Audio	
Nom de chemin	mpv:LastURL	Un ou plusieurs noms de chemins dont la résolution est le fichier du chapitre. Chaque nom de chemin est spécifique à un système de fichiers; il peut donc y avoir des noms différents pour chaque système de fichiers sur un CD-ROM.
Format	nmf:Metadata dc:Properties dc:format	Utiliser la liste fournie par MPV des types MIME recensés ou des types étendus spécifiques au fournisseur.
Titre	nmf:Metadata dc:Properties dc:title	Métadonnées principales concernant le titre d'un actif audio. Par exemple: s'il s'agit d'un chapitre, le titre du chapitre; s'il s'agit d'une page, le numéro de page.
Titre de l'album	nmf:Metadata mpvm:MusicProperties mpvm:AlbumTitle	Non utilisé comme métadonnées principales pour les Livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookTitle pour une compatibilité améliorée avec les lecteurs MPV Music.
Titre du livre	nmf:Metadata ab:Properties ab:BookTitle	Métadonnées principales concernant le titre du livre en cours de lecture. Lorsqu'elles sont utilisées sur un actif, elles se substituent aux valeurs de métadonnées spécifiées sur mpvp:Album.
Créateur	nmf:Metadata dc:Properties dc:creator	Non utilisé comme métadonnées principales pour les Livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookAuthor pour une compatibilité améliorée avec les lecteurs MPV Music.
Artiste principal	nmf:Metadata mpvm:MusicProperties mpvm:PrincipalArtist	Non utilisé comme métadonnées principales pour les Livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookAuthor pour une compatibilité améliorée avec les lecteurs MPV Music.

Tableau 22 (suite)

Métadonnées	Éléments de métadonnées dans MPV	Description concernant les livres audio
Auteur du livre	nmf:Metadata ab:Properties ab:BookAuthor	Métadonnées principales concernant l'auteur du livre en cours de lecture. Lorsqu'elles sont utilisées sur un actif, elles se substituent aux valeurs de métadonnées spécifiées sur mpvp:Album.
Contributeur	nmf:Metadata dc:Properties dc:contributor	Non utilisé comme métadonnées principales pour les Livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookNarrator pour une compatibilité améliorée avec les lecteurs MPV Music.
Narrateur du livre	nmf:Metadata ab:Properties ab:Narrator	Métadonnées principales concernant le narrateur du livre en cours de lecture. Lorsqu'elles sont utilisées sur un actif, elles se substituent aux valeurs de métadonnées spécifiées sur mpvp:Album.
Auteur de la musique	nmf:Metadata mpvm:MusicProperties mpvm:MusicBy	Nom(s) du (des) compositeur(s) en format texte libre.
Auteur des paroles	nmf:Metadata mpvm:MusicProperties mpvm:LyricsBy	Non utilisé.
Arrangé par	nmf:Metadata mpvm:MusicProperties mpvm:ArrangedBy	Non utilisé.
Produit par	nmf:Metadata mpvm:MusicProperties ProducedBy	Nom(s) du (des) producteur(s) en format texte libre.
Genre	nmf:Metadata mpvm:MusicProperties mpvm:Genre	Un jeu prédéfini de genres peut être étendu simplement à l'aide d'un nouveau nom.
Date d'enregistrement	nmf:Metadata dcterms:Properties dcterms:created	Date et heure d'enregistrement, par exemple 2002-10-03T21:07:00Z
Identificateur	nmf:Metadata dc:Properties dc:identifier	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookISBN pour une compatibilité améliorée avec les lecteurs MPV de toutes catégories.
Identificateur unique universel du livre	nmf:Metadata ab:Properties ab:BookUUID	Métadonnées principales concernant l'identificateur unique universel du livre.
Numéro ISBN du livre	nmf:Metadata ab:Properties ab:BookISBN	Métadonnées principales concernant le numéro ISBN du livre.
Numéro ISSN du livre	nmf:Metadata ab:Properties ab:BookISSN	Métadonnées principales concernant le numéro ISSN du livre.
Identificateur DOI du livre	nmf:Metadata ab:Properties ab:BookDOI	Métadonnées principales concernant l'identificateur DOI du livre.
Numéro d'index du livre	nmf:Metadata ab:Properties ab:BookIndexNumber	Métadonnées principales concernant le numéro d'index du livre.
Cote du livre	nmf:Metadata ab:Properties ab:BookShelfNumber	Métadonnées principales concernant la cote du livre.
Numéro de référence du livre	nmf:Metadata ab:Properties ab:BookPartNumber	Métadonnées principales concernant le numéro de référence du livre.

Tableau 22 (suite)

Métadonnées	Éléments de métadonnées dans MPV	Description concernant les livres audio
Autre identificateur	nmf:Metadata ab:Properties ab:OtherIdentifier	Métadonnées principales concernant un autre identificateur du livre.
Identificateur d'emplacement de contenu	nmf:Metadata ab:Properties ab:LocationIdentifier	Métadonnées principales concernant l'identificateur d'emplacement de contenu de liste d'écoute.
Description	nmf:Metadata dc:Properties dc:description	Commentaire, message, description.
Mots clés	nmf:Metadata dc:Properties dc:subject	Liste séparée par des deux-points, des mots clés ou expressions de recherche sous forme de texte. Les deux-points peuvent être utilisés par l'échappement avec une barre oblique inverse, par exemple \;
Déclaration de Copyright	nmf:Metadata dc:Properties dc:rights	Texte de format libre de déclaration des droits de copyright.
Langue	nmf:Metadata dc:Properties dc:language	"<language>_<territory>" où <language> provient de [ISO 639-1] et <territory> est [ISO 3166-1] Exemple: "en_US"
Reproduction utilisant un autre codeur	mpv:Rendition mpv:renditionType="alt" mpv:AudioRef	Fait référence à un autre codage de l'actif.
Reproduction utilisant un autre débit binaire	mpv:Rendition mpv:renditionType="subsampling" mpv:AudioRef	Fait référence à un autre codage de l'actif à un débit binaire différent. Utiliser mpvm:EncodedBitrate ou consulter le fichier pour plus d'informations.
Débit binaire codé	nmf:Metadata mpvm:MusicProperties mpvm:EncodedBitrate	Débit binaire maximal en bits/s de l'actif codé. Pour les actifs à débit binaire codé variable, il s'agit de la valeur maximale; pour les actifs codés à débit binaire constant, il s'agit de la valeur moyenne.
Graphique de l'actif	mpv:Related mpv:relationship="urn:osta-org:mpv:music:artwork" mpv:StillRef	Non utilisé.
Vidéo de la musique	mpv:Related mpv:relationship="urn:osta-org:mpv:music:video" mpv:VideoRef	Non utilisé.
Informations en ligne	nmf:Metadata mpvm:MusicProperties mpvm:MoreInfoURL	URL à suivre pour obtenir des informations supplémentaires sur le livre audio.
Temps d'écoute	nmf:Metadata mpvm:MusicProperties mpvm:PlayingTime	Durée en secondes. Les valeurs décimales sont autorisées, telles que 134,58 s., ce qui signifie 134 et 58/100 s.
Numéro de piste	nmf:Metadata mpvm:MusicProperties mpvm:TrackNumber	Ordre séquentiel du chapitre dans le livre d'origine. Commence à 1 et non 0.
Nombre de pistes	nmf:Metadata mpvm:MusicProperties mpvm:NumTracks	Nombre de chapitres dans la compilation d'origine. Commence par 1 et non 0.
Numéro du jeu dans la collection	nmf:Metadata mpvm:MusicProperties mpvm:SetNumber	Non utilisé.
Nombre de jeux dans la collection	nmf:Metadata mpvm:MusicProperties mpvm:NumSets	Non utilisé.

Tableau 22 (suite)

Métadonnées	Éléments de métadonnées dans MPV	Description concernant les livres audio
Nombre de jeux dans la collection	nmf:Metadata mpvm:MusicProperties mpvm:NumSets	Non utilisé.
Comptage de lecture	nmf:Metadata mpvm:MusicProperties mpvm:PlayCount	Non utilisé.
Paroles	nmf:Metadata mpvm:MusicProperties mpvm:Lyrics	Non utilisé.
Mode	nmf:Metadata mpvm:MusicProperties mpvm:Mood	Non utilisé.
Tempo	nmf:Metadata mpvm:MusicProperties mpvm:Tempo	Non utilisé.
Situation	nmf:Metadata mpvm:MusicProperties mpvm:Situation	Non utilisé.
Support	nmf:Metadata dcterms:Properties dcterms:medium	Non utilisé.
Nom de fichier et fichier d'origine	mpv:Related mpv:relationship="copyOf" mpv:AudioRef	Non utilisé.
Données supplémentaires	nmf:Metadata et mpv:Metadata	Toutes les métadonnées qu'une application souhaite créer peuvent être placées dans ces conteneurs.
Paires clé-valeur	nmf:Metadata mpvm:Properties mpvm:KeyValue mpvmkv:Key nmf:Metadata mpvm:Properties mpvm:KeyValue mpvmkv:Value	Non utilisé.
Album (livre audio)	mpvp:Album	
Nombre d'entrées	implicite – nombre d'entrées en arrière-plan	
Titre	nmf:Metadata dc:Properties dc:title	Non utilisé comme métadonnées principales pour les Livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookTitle pour une compatibilité améliorée avec les lecteurs MPV.
Titre du livre	nmf:Metadata ab:Properties ab:BookTitle	Titre du livre complet.
Contributeur	nmf:Metadata dc:Properties dc:contributor	Non utilisé comme métadonnées principales pour les Livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookNarrator pour une compatibilité améliorée avec les lecteurs MPV.
Narrateur	nmf:Metadata ab:Properties ab:BookNarrator	Métadonnées principales du livre audio.
Créateur	nmf:Metadata dc:Properties dc:creator	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookAuthor pour une compatibilité améliorée avec les lecteurs MPV.
Auteur	nmf:Metadata ab:Properties ab:BookAuthor	Métadonnées principales du livre audio.
Artiste principal	nmf:Metadata mpvm:MusicProperties mpvm:PrincipalArtist	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookAuthor pour une compatibilité améliorée avec les lecteurs MPV Music.
Description	nmf:Metadata dc:Properties dc:description	Commentaire de format libre, message, description sous forme de texte normal.

Tableau 22 (suite)

Métadonnées	Éléments de métadonnées dans MPV	Description concernant les livres audio
Déclaration de Copyright	nfm:Metadata dc:Properties dc:rights	Texte normal de format libre de déclaration des droits de copyright.
Éléments du contenu du livre audio	mpvp:Album mpvp:Foreground	Ce contenu est lu en tant que contenu principal de la liste d'écoute de livre audio.
Accompagnement du livre audio	mpvp:Album mpvp:Background	Ce contenu est lu en arrière-plan du contenu du livre audio en avant-plan. Par exemple, il peut spécifier des images ou une musique d'arrière-plan.
Graphique du livre audio	mpvp:Album mpv:Related mpv:relationship="urn:osta-org:mpv:music:artwork" mpv:StillRef	Graphique du livre audio.
Identificateur	nfm:Metadata dc:Properties dc:identifier	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookISBN pour une compatibilité améliorée avec les lecteurs MPV Music.
Identificateur unique universel du livre	nfm:Metadata ab:Properties ab:BookUUID	Métadonnées principales concernant l'identificateur unique universel du livre.
Numéro ISBN du livre	nfm:Metadata ab:Properties ab:BookISBN	Métadonnées principales concernant le numéro ISBN du livre.
Numéro ISSN du livre	nfm:Metadata ab:Properties ab:BookISSN	Métadonnées principales concernant le numéro ISSN du livre.
Identificateur DOI du livre	nfm:Metadata ab:Properties ab:BookDOI	Métadonnées principales concernant l'identificateur DOI du livre.
Numéro d'index du livre	nfm:Metadata ab:Properties ab:BookIndexNumber	Métadonnées principales concernant le numéro d'index du livre.
Cote du livre	nfm:Metadata ab:Properties ab:BookShelfNumber	Métadonnées principales concernant la cote du livre.
Numéro de référence du livre	nfm:Metadata ab:Properties ab:BookPartNumber	Métadonnées principales concernant le numéro de référence du livre.
Autre identificateur	nfm:Metadata ab:Properties ab:OtherIdentifier	Métadonnées principales concernant un autre identificateur du livre.
Identificateur d'emplacement de contenu	nfm:Metadata ab:Properties ab:LocationIdentifier	Métadonnées principales concernant l'identificateur d'emplacement de contenu de liste d'écoute.
Paires clé-valeur	nfm:Metadata mpvm:Properties mpvm:KeyValue mpvmkv:Key nfm:Metadata mpvm:Properties mpvm:KeyValue mpvmkv:Value	Non utilisé.
Données supplémentaires	mpv:Metadata and nfm:Metadata anywhere	

9.7 Utilisation des métadonnées Dublin Core

Afin de promouvoir l'interopérabilité, MPV-Core utilise les métadonnées Dublin Core (DC-NMF) pour représenter les métadonnées essentielles sur tous types d'actifs. L'élément dc:title sert à spécifier le titre d'un actif de livre audio, tout comme le titre d'une image ou d'un actif vidéo.

En 9.6, les éléments de métadonnées Dublin Core ont été combinés en un jeu global de propriétés et d'actifs de métadonnées de livre audio. Pour clarifier l'utilisation de DC-NMF, le Tableau 23 ne comporte que les éléments Dublin Core.

Tableau 23 – Métadonnées Dublin Core (DC-NMF)

Actif audio Chapitre, Section	Sous-éléments mpv:Audio	
Créateur ("Auteur")	nmf:Metadata dc:Properties dc:creator	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookAuthor pour une compatibilité améliorée avec les lecteurs MPV.
Contributeur ("Narrateur")	nmf:Metadata dc:Properties dc:contributor	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookNarrator pour une compatibilité améliorée avec les lecteurs MPV.
Description	nmf:Metadata dc:Properties dc:description	Commentaire, message, description sous forme de texte normal.
Format	nmf:Metadata dc:Properties dc:format	Utiliser la liste fournie par MPV des types MIME recensés.
Identificateur ("ISBN")	nmf:Metadata dc:Properties dc:identifier	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookISBN pour une compatibilité améliorée avec les lecteurs MPV Music.
Mots clés	nmf:Metadata dc:Properties dc:subject	Liste séparée par des deux-points des mots clés ou expressions de recherche sous forme de texte normal. Les deux-points peuvent être utilisés par l'échappement avec une barre oblique inverse, par exemple \;
Langue	nmf:Metadata dc:Properties dc:language	"<language>_<territory>" où <language> provient de [ISO 639-1] et <territory> est [ISO 3166-1] Exemple: "en_US"
Editeur	nmf:Metadata dc:Properties dc:publisher	Texte normal de format libre.
Droits ("Déclaration de Copyright")	nmf:Metadata dc:Properties dc:rights	Déclaration de copyright en texte normal de format libre.
Titre	nmf:Metadata dc:Properties dc:title	Titre du chapitre en texte normal de format libre.
Date de création	nmf:Metadata dcterms:Properties dcterms:created	Date et heure auxquelles le livre audio a été réalisé ou enregistré.
Support	nmf:Metadata dcterms:Properties dcterms:medium	Support d'édition du livre audio d'origine conforme à [ID3v240]

Tableau 23 (suite)

Album (livre audio) de mots parlés	Sous-éléments mpvp:Album	Observations
Créateur ("Auteur")	nmf:Metadata dc:Properties dc:creator	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookAuthor pour une compatibilité améliorée avec les lecteurs MPV.
Contributeur ("Narrateur")	nmf:Metadata dc:Properties dc:contributor	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookNarrator pour une compatibilité améliorée avec les lecteurs MPV.
Description	nmf:Metadata dc:Properties dc:description	Commentaire de format libre, message, description sous forme de texte normal.
Identificateur ("Numéro ISBN du livre")	nmf:Metadata dc:Properties dc:identifier	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookISBN pour une compatibilité améliorée avec les lecteurs MPV Music.
Editeur	nmf:Metadata dc:Properties dc:publisher	Texte normal de format libre.
Droits ("Déclaration de Copyright")	nmf:Metadata dc:Properties dc:rights	Déclaration de copyright en texte normal de format libre.
Titre	nmf:Metadata dc:Properties dc:title	Titre du livre audio en texte normal de format libre.

9.8 Profil de livre audio utilisant les listes d'écoute MPV-Music

9.8.1 Généralités

MPV-Audiobook utilise les listes d'écoute MPV-Music pour organiser les actifs audio et les autres contenus sur un support de stockage. Une liste d'écoute MPV-Music identifie les actifs principaux à présenter à l'utilisateur, tels qu'un jeu de chapitres ou de sections de livre audio. La liste d'écoute MPV-Music spécifie également l'ordre de présentation des actifs ainsi que d'autres caractéristiques de présentation.

Les listes d'écoute MPV-Music sont remplies en fonction de la spécification de profil de présentation MPV [MPV-Pres]. Le profil MPV-Music spécifie les détails de cette utilisation pour les contenus musicaux. Cette utilisation est tenue à jour et affinée pour les livres audio.

Une liste d'écoute MPV-Music utilise l'élément <mpvp:Album>, qui peut contenir des métadonnées, un contenu d'avant-plan et d'arrière-plan et des actifs associés et d'interprétation (*rendition*). Voici un exemple.

```
<mpvp:Album>
```

```
  <mpv:Metadata> ... </mpv:Metadata>
```

```
  <nmf:Metadata> ... </nmf:Metadata>
```

```
  <mpvp:Background> ... </mpvp:Background>
```

```
  <mpvp:Foreground> ... </mpvp:Foreground>
```

```
  <mpv:Rendition> ... </mpv:Rendition>
```

```
  <mpv:Related> ... </mpv:Related>
```

</mpvp:Album>

MPV-Music anticipe que les utilisateurs organiseront les fichiers du support (*media files*) en dossiers/répertoires de leur choix et leur affecteront des noms de fichiers longs. L'emplacement du contenu et l'ordre de lecture ne peuvent pas être spécifiés seulement par le biais de répertoires et noms de fichiers spécifiques. A la place, MPV-Music fournit une approche permettant au contenu de se trouver à n'importe quel endroit et sous n'importe quel nom.

En outre, de nombreux actifs du support apparentés peuvent être présents sur le support de stockage, tel qu'un CD, afin d'améliorer l'accès et les performances de lecture et fournir un ressenti amélioré en lecture et en impression. La liste d'écoute MPV permet de gérer ce contenu pour donner à l'utilisateur une interaction simple et de haut niveau avec son contenu.

9.8.2 Nombre de listes d'écoute par fichier MPV

Un fichier MPV conforme à MPV-Audiobook doit contenir une liste d'écoute (élément mpvm:Album).

9.8.3 Utilisation des métadonnées

Les métadonnées placées sur l'élément mpvp:Album définissent l'étendue de l'ensemble de l'album ou de la liste d'écoute et les propriétés de métadonnées destinées à cette utilisation doivent être définies dans le Tableau 24. Par exemple, l'élément mpvm:PrincipalArtist peut être utilisé avec un actif musical unique ou sur mpvp:Album.

Tableau 24 – Utilisation des propriétés de métadonnées

Album livre audio	mpvp:Album	
Nombre d'entrées	implicite – nombre d'entrées en avant-plan	
Titre	nmf:Metadata dc:Properties dc:title	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookTitle pour une compatibilité améliorée avec les lecteurs MPV.
Titre du livre	nmf:Metadata ab:Properties ab:BookTitle	Titre du livre entier dont provient le chapitre.
Sous-titre du livre	nmf:Metadata ab:Properties BookSubtitle	Sous-titre du livre entier dont provient le chapitre.
Contributeur	nmf:Metadata dc:Properties dc:contributor	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookNarrator pour une compatibilité améliorée avec les lecteurs MPV.
Narrateur	nmf:Metadata ab:Properties ab:BookNarrator	Métadonnées principales du livre audio.
Créateur	nmf:Metadata dc:Properties dc:creator	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookAuthor pour une compatibilité améliorée avec les lecteurs MPV.
Auteur	nmf:Metadata ab:Properties ab:BookAuthor	Métadonnées principales du livre audio.

Tableau 24 (suite)

Album livre audio	mpvp:Album	
Co-auteur du livre	nmf:Metadata ab:Properties BookCoauthor	Co-auteur du livre global dont provient le chapitre.
Artiste principal	nmf:Metadata mpvm:MusicProperties mpvm:PrincipalArtist	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookAuthor pour une compatibilité améliorée avec les lecteurs MPV Music.
Genre du livre	nmf:Metadata ab:Properties BookGenre	Genre du livre global dont provient le chapitre.
Description	nmf:Metadata dc:Properties dc:description	Description récapitulative du livre.
Déclaration de Copyright	nmf:Metadata dc:Properties dc:rights	Texte normal de format libre de déclaration des droits de copyright.
Éléments du contenu du livre audio	mpvp:Album mpvp:Foreground	Ce contenu est lu en tant que contenu principal de la liste d'écoute de livre audio.
Accompagnement du livre audio	mpvp:Album mpvp:Background	Ce contenu est lu en arrière-plan du contenu du livre audio en avant-plan. Par exemple, il peut spécifier des images ou une musique d'arrière-plan.
Graphique du livre audio	mpvp:Album mpv:Related mpv:relationship="urn:osta-org:mpv:music:artwork" mpv:StillRef	Graphique du livre audio.
Identificateur	nmf:Metadata dc:Properties dc:identifier	Non utilisé comme métadonnées principales pour les livres audio. IL CONVIENT de le renseigner par la valeur de ab:BookISBN pour une compatibilité améliorée avec les lecteurs MPV Music.
Identificateur unique universel du livre	nmf:Metadata ab:Properties ab:BookUUID	
Numéro ISBN du livre	nmf:Metadata ab:Properties ab:BookISBN	
Numéro ISSN du livre	nmf:Metadata ab:Properties ab:BookISSN	
Identificateur DOI du livre	nmf:Metadata ab:Properties ab:BookDOI	
Numéro d'index du livre	nmf:Metadata ab:Properties ab:BookIndexNumber	
Cote du livre	nmf:Metadata ab:Properties ab:BookShelfNumber	
Numéro de référence du livre	nmf:Metadata ab:Properties ab:BookPartNumber	
Autre identificateur	nmf:Metadata ab:Properties ab:OtherIdentifier	
Identificateur d'emplacement de contenu	nmf:Metadata ab:Properties ab:LocationIdentifier	
Paires clé-valeur	nmf:Metadata mpvm:Properties mpvm:KeyValue mpvm kv:Key nmf:Metadata mpvm:Properties mpvm:KeyValue mpvm kv:Value	Non utilisé.
Données supplémentaires	mpv:Metadata et nmf:Metadata à tout emplacement	

9.8.4 Utilisation en arrière-plan

MPV-Audiobook peut contenir l'élément `mpvp:Background` afin de fournir du contenu de présentation d'arrière-plan aux mots parlés en avant-plan. Il est appliqué au bon vouloir de l'application ou de l'appareil de lecture. S'il s'agit d'un appareil ou d'une application simple sans fonctionnalités de présentation visuelle, il convient d'ignorer le contenu `mpvp:Background`.

Si l'application ou l'appareil possède des fonctionnalités de présentation visuelle, le contenu `mpvp:Background` peut être traité. Le contenu recommandé est une série d'éléments `mpv:StillRef` qui renvoient à des images fixes qui peuvent être affichées à l'arrière du contenu d'avant-plan. Il peut être utilisé en mode parcourir/menu ainsi qu'en mode lecture. Toutes les propriétés MPV-Pres peuvent être appliquées, telles les transitions. Celles-ci sont appliquées selon l'application de lecture.

9.8.5 Utilisation en avant-plan

Un livre audio MPV doit inclure une liste d'avant-plan `mpvp:Foreground` dans `mpvp:Album`, qui doit être identifiée à l'aide de l'élément `mpvp:AudioRef`. Ce dernier définit le jeu des principaux éléments de mots parlés dans la liste d'écoute et leur séquence. En plus des actifs audio référencés, la liste `mpv:AssetList` dans le fichier MPV peut également contenir des images fixes, des vidéos et d'autres types d'actifs. `mpvp:Album` identifie explicitement quels sont les actifs principaux dans `mpv:AssetList`.

9.8.6 Actifs associés et d'interprétation (*rendition*)

`mpvp:Album` peut identifier des actifs associés et d'interprétation. Un graphique de couverture peut être spécifié à l'aide d'un élément `mpv:Related` avec un identificateur de relation `urn:osta-org:mpv:music:artwork`.

9.9 Groupes d'actifs dans les listes d'écoute

Des situations peuvent survenir dans lesquelles une liste d'écoute possède un si grand nombre d'actifs qu'il n'est pas pratique de parcourir un élément à la fois. L'application ou l'appareil de lecture peut choisir de mettre en œuvre la fonction d'Avance rapide ou de Retour rapide qui permet de sauter plusieurs éléments.

Les applications génératrices de MPV peuvent également choisir de grouper les actifs dans des listes d'écoute. L'appareil de lecture peut choisir de mettre en œuvre les fonctions d'Avance rapide et de Retour rapide en sautant des groupes; la navigation par groupes peut également être explicite si l'appareil de lecture le souhaite.

Les groupes sont spécifiés à l'aide de la fonctionnalité MPV standard. Un groupe d'actifs à lire de façon séquentielle doit être identifié à l'aide de l'actif `mpv:Seq`. Lorsqu'une application ou un appareil de lecture rencontre un actif `mpv:Seq`, il convient que l'application ou le dispositif lise le contenu de façon séquentielle.

Lorsque `mpv:Seqs` sont utilisés avec `mpvp:Album`, seuls les actifs `mpv:Seq` doivent être répertoriés dans l'album. Lorsque l'application ou l'appareil lit les actifs `mpv:Seq`, ces actifs doivent lire le contenu `mpv:Seq` dans l'ordre. Voir ci-dessous un exemple de l'implémentation d'un groupe d'actifs MPV.

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

```
<file:Manifest
```

```
xmlns:file="http://ns.osta.org/manifest/1.0/"
```

```
xmlns:ab="http://ns.cei.org/audiobook/2003b/"
```

xmlns:mpv="http://ns.osta.org/mpv/1.0/"

xmlns:mpvp="http://ns.osta.org/mpv/presentation/1.0/"

xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"

xmlns:dc="http://purl.org/dc/elements/1.1/"

xmlns:dcterms="http://http://purl.org/dc/terms/"

xmlns:nmf="http://ns.osta.org/nmf/1.0/">

<nmf:Metadata>

<ManifestProperties xmlns="http://ns.osta.org/manifest/1.0/">

<ProfileBag>

<Profile>http://ns.ce.org/audiobook/2003b/</Profile>

<Profile>http://ns.osta.org/mpv/presentation/1.0/</Profile>

<Profile>http://ns.osta.org/mpv/music/1.0/</Profile>

</ProfileBag>

</ManifestProperties>

<Properties xmlns="http://purl.org/dc/terms/">

<created>2003-01-29T15:51:41Z"</created>

<modified>2003-01-29T15:51:41Z"/></modified>

</Properties>

</nmf:Metadata>

<mpvp:Album>

<nmf:Metadata>

<dc:Properties>

<dc:creator>Ima Smoohtalker</dc:creator>

<dc:identifier>put an ISBN number here</dc:identifier>

<dc:rights>Copyright (c)2003 HomeMade Press<dc:rights/>

<dc:title>Life Along the Willamette</dc:title>

</dc:Properties>

<mpvm:MusicProperties>

<mpvm:AlbumTitle>Life Along the

Willamette</mpvm:AlbumTitle>

<mpvm:Genre>Spoken Word:Non-Fiction</mpvm:Genre>

</mpvm:MusicProperties>

```

<ab:AudiobookProperties>
    <ab:BookAuthor>Ura Writer</ab:BookAuthor>
    <ab:BookISBN>put an ISBN number here</ab:BookISBN>
    <ab:BookNarrator>Ima Smoothtalker</ab:BookNarrator>
    <ab:BookTitle>Life Along the Willamette</ab:BookTitle>
</ab:AudiobookProperties>
</nmf:Metadata>
<mpvp:Foreground>
    <mpv:SeqRef mpv:idRef="IDSeq0001"/> <!-- chapter 1 -->
    <mpv:SeqRef mpv:idRef="IDSeq0002"/> <!-- chapter 2 -->
</mpvp:Foreground>
</mpvp:Album>

<mpv:AssetList>
<mpv:Seq mpv:id="IDSeq0001"> <!-- chapter 1 -->
    <mpv:AudioRef mpv:idRef="ID000001"/> <!-- page -->
    <mpv:AudioRef mpv:idRef="ID000002"/> <!-- page -->
    <mpv:AudioRef mpv:idRef="ID000003"/> <!-- page -->
</mpv:Seq>
<mpv:Seq mpv:id="IDSeq0002"> <!-- chapter 2 -->
    <mpv:AudioRef mpv:idRef="ID000004"/> <!-- page -->
    <mpv:AudioRef mpv:idRef="ID000005"/> <!-- page -->
    <mpv:AudioRef mpv:idRef="ID000006"/> <!-- page -->
</mpv:Seq>

    <mpv:Audio mpv:id="ID000001">
        <nmf:Metadata>
            <Properties xmlns="http://purl.org/dc/elements/1.1">
                <contributor>Ima Smoothtalker</contributor>
                <creator>Ura Writer</creator>
                <format>audio/mpeg</format>
                <title>page 1</title>
            </Properties>

```

```

    <MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0/">
      <AlbumTitle>Life Along the Willamette</AlbumTitle>
      <Genre>Spoken Word:Non-Fiction</Genre>
      <PlayingTime>3.48</PlayingTime>
    </MusicProperties>
  </nmf:Metadata>
  <mpv:LastURL>audio/page1.mp3</mpv:LastURL>
</mpv:Audio>
<mpv:Audio mpv:id="ID000002">
  <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
      <contributor>Ima Smoothtalker</contributor>
      <creator>Ura Writer</creator>
      <format>audio/mpeg</format>
      <title>page 2</title>
    </Properties>
    <MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0/">
      <AlbumTitle>Life Along the Willamette</AlbumTitle>
      <Genre>Spoken Word:Non-Fiction</Genre>
      <PlayingTime>3.24</PlayingTime>
    </MusicProperties>
  </nmf:Metadata>
  <mpv:LastURL>audio/page2.mp3</mpv:LastURL>
</mpv:Audio>
</mpv:AssetList>
</file:Manifest>

```

9.10 Détails du schéma d'un livre audio MPV

9.10.1 Définition de schéma XML

Le Schéma XML d'un livre audio MPV doit être comme indiqué ci-dessous.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schematarget Namespace="http://ns.cei.org/audiobook/2003b/"
xmlns=http://ns.cei.org/audiobook/2003b/

```

```

xmlns:nmft="http://ns.osta.org/nmf/1.0/tools/"
xmlns:xs="http://www.w3.org/2001/XMLSchema"

xmlns:nmf=http://ns.osta.org/nmf/1.0/

elementFormDefault="qualified" attributeFormDefault="qualified">

    <xs:import namespace="http://ns.osta.org/nmf/1.0/" schemaLocation="nmf.xsd"/>

    <xs:annotation>

        <xs:documentation>The CEA R6 WG11 Audiobook Properties
schema</xs:documentation>

    </xs:annotation>

    <!--name for BySchemaProperties element: AudiobookProperties-->

    <xs:element name="AudiobookProperties" type="BySchemaPropsType"
substitutionGroup="nmf:BySchemaPropsBase"/>

    <!--top-level schema element type. Note that to be MPV NMF compliant, the order of
the elements MUST be alphabetical. -->

    <xs:complexType name="BySchemaPropsType">
        <xs:complexContent>
            <xs:extension base="nmf:BySchemaPropsType">
                <xs:sequence>
                    <xs:element ref="BookAuthor" minOccurs="0"/>
                    <xs:element ref="BookDOI" minOccurs="0"/>
                    <xs:element ref="BookIndexNumber" minOccurs="0"/>
                    <xs:element ref="BookISBN" minOccurs="0"/>
                    <xs:element ref="BookISSN" minOccurs="0"/>
                    <xs:element ref="BookNarrator" minOccurs="0"/>
                    <xs:element ref="BookPartNumber" minOccurs="0"/>
                    <xs:element ref="BookShelfNumber" minOccurs="0"/>
                    <xs:element ref="BookTitle" minOccurs="0"/>
                    <xs:element ref="BookUUID" minOccurs="0"/>
                    <xs:element ref="OtherIdentifier" minOccurs="0"/>
                    <xs:element ref="LocationIdentifier" minOccurs="0"/>
                    <xs:element ref="unitType" minOccurs="0"/>
                    <xs:element ref="unitContext" minOccurs="0"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>

```

```
</xs:complexType>
```

```
<xs:element name="BookTitle" type=" BookTitleType"/>
```

```
<xs:complexType name=" BookTitleType">
```

```
  <xs:simpleContent>
```

```
    <xs:extension base="xs:string"/>
```

```
  </xs:simpleContent>
```

```
</xs:complexType>
```

```
<xs:element name="BookAuthor" type=" BookAuthorType"/>
```

```
<xs:complexType name=" BookAuthorType">
```

```
  <xs:simpleContent>
```

```
    <xs:extension base="xs:string"/>
```

```
  </xs:simpleContent>
```

```
</xs:complexType>
```

```
<xs:element name="BookNarrator" type=" BookNarratorType"/>
```

```
<xs:complexType name=" BookNarratorType">
```

```
  <xs:simpleContent>
```

```
    <xs:extension base="xs:string"/>
```

```
  </xs:simpleContent>
```

```
</xs:complexType>
```

```
<xs:element name="BookUUID" type=" BookUUIDType"/>
```

```
<xs:complexType name=" BookUUIDType">
```

```
  <xs:simpleContent>
```

```
    <xs:extension base="xs:string"/>
```

```
  </xs:simpleContent>
```

```
</xs:complexType>
```

```
<xs:element name="BookGenre" type="BookGenreType"/>
```

```
<xs:simpleType name="BookGenreType">
```

```
  <xs:union memberTypes="BookGenreBaseType xs:anyURI"/>
```

```
</xs:simpleType>
```

```
<xs:simpleType name="BookGenreBaseType">
```

```
  <xs:restriction base="xs:string">
```

```
<xs:enumeration value="Spoken Word:Fiction"/>
```

```
<xs:enumeration value="Spoken Word:Non-Fiction"/>
```

```
<xs:enumeration value="Audiobook:Fiction"/>
```

```
<xs:enumeration value="Audiobook:Non-Fiction"/>
```

```
  </xs:restriction>
```

```
</xs:simpleType>
```

```
<xs:element name="BookISSN" type="BookISSNType"/>
```

```
<xs:complexType name="BookISSNType">
```

```
  <xs:simpleContent>
```

```
    <xs:extension base="xs:string"/>
```

```
  </xs:simpleContent>
```

```
</xs:complexType>
```

```
<xs:element name="BookDOI" type="BookDOIType"/>
```

```
<xs:complexType name="BookDOIType">
```

```
  <xs:simpleContent>
```

```
    <xs:extension base="xs:string"/>
```

```
  </xs:simpleContent>
```

```
</xs:complexType>
```

```
<xs:element name="BookIndexNumber" type="BookIndexNumberType"/>
```

```
<xs:complexType name="BookIndexNumberType">
```

```
  <xs:simpleContent>
```

```
    <xs:extension base="xs:string"/>
```

```
  </xs:simpleContent>
```

```
</xs:complexType>
```

```
<xs:element name="BookShelfNumber" type="BookShelfNumberType"/>
```

```
<xs:complexType name="BookShelfNumberType">
```

```
<xs:simpleContent>
  <xs:extension base="xs:string"/>
</xs:simpleContent>
</xs:complexType>

<xs:element name="BookPartNumber" type="BookPartNumberType"/>
<xs:complexType name="BookPartNumberType">
  <xs:simpleContent>
    <xs:extension base="xs:string"/>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="OtherIdentifier" type="OtherIdentifierType"/>
<xs:complexType name="OtherIdentifierType">
  <xs:simpleContent>
    <xs:extension base="xs:string"/>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="LocationIdentifier" type="LocationIdentifierType"/>
<xs:complexType name="LocationIdentifierType">
  <xs:simpleContent>
    <xs:extension base="xs:string"/>
  </xs:simpleContent>
</xs:complexType>

<xs:element name="unitType" type="unitTypeType"/>
<xs:complexType name="unitTypeType">
  <xs:simpleContent>
    <xs:extension base="xs:string"/>
  </xs:simpleContent>
</xs:complexType>
```

```

<xs:element name="unitContext" type="unitContextType"/>

<xs:complexType name="unitContextType">
    <xs:simpleContent>
        <xs:extension base="xs:string"/>
    </xs:simpleContent>
</xs:complexType>

</xs:schema>

```

9.10.2 Définitions de l'utilisation des métadonnées de livre audio

Les propriétés de livre audio MPV (*MPV-Audiobook*) doivent être définies comme suit:

- a) **BookAuthor** – Cet élément doit contenir le nom de l'auteur du livre audio. Il s'agit d'un élément obligatoire qui doit être présent dans tous les livres audio. S'il n'y a pas d'auteur, la valeur doit être vide.
- b) **BookCoAuthor** – Cet élément doit contenir le nom du co-auteur du livre audio. S'il existe plusieurs co-auteurs, plusieurs valeurs peuvent être présentes. S'il n'y a pas de co-auteur, cet élément ne doit pas figurer.
- c) **BookDOI** – Cet élément doit contenir l'identificateur DOI (Digital Object Identifier). S'il n'y a pas d'identificateur DOI, cet élément ne doit pas figurer.
- d) **BookGenre** – Cet élément doit contenir le genre du livre audio. S'il existe plusieurs genres, plusieurs valeurs peuvent être présentes. Si la valeur de genre n'existe pas, cet élément ne doit pas figurer.
- e) **BookIndexNumber** – Cet élément doit contenir le numéro d'index du livre audio. Si le numéro d'index n'existe pas, cet élément ne doit pas figurer.
- f) **BookISBN** – Cet élément doit contenir le numéro ISBN (International Standard Book Number) du livre. S'il n'y a pas de numéro ISBN, cet élément ne doit pas figurer.
- g) **BookISSN** – Cet élément doit contenir le numéro ISSN (International Standard Serial Number) du livre. S'il n'y a pas de numéro ISSN, cet élément ne doit pas figurer.
- h) **BookNarrator** – Cet élément doit contenir le nom du narrateur du livre audio. Il s'agit d'un élément obligatoire qui doit être présent dans tous les livres audio. S'il existe plusieurs narrateurs, plusieurs valeurs peuvent être présentes. S'il n'y a pas de narrateur, la valeur doit être vide.
- i) **BookIndexNumber** – Cet élément doit contenir le numéro de référence du livre audio. Si le numéro de référence n'existe pas, cet élément ne doit pas figurer.
- j) **BookShelfNumber** – Cet élément doit contenir la cote du livre audio. Si la cote n'existe pas, cet élément ne doit pas figurer.
- k) **BookSubTitle** – Cet élément doit contenir le sous-titre du livre audio. S'il n'y a pas de sous-titre, cet élément ne doit pas figurer.
- l) **BookTitle** – Cet élément doit contenir le titre du livre audio. Il s'agit d'un élément obligatoire qui doit être présent dans tous les livres audio. S'il n'y a pas de titre, la valeur doit être vide.
- m) **BookUUID** – Cet élément doit contenir l'identificateur unique universel du livre audio. S'il n'y a pas d'identificateur unique universel, cet élément ne doit pas figurer.
- n) **OtherIdentifier** – Cet élément doit contenir tout identificateur de livre audio supplémentaire qui n'est pas défini par cette norme et peut comprendre toute valeur permettant d'identifier le livre audio de façon unique. S'il n'existe pas d'autre identificateur, cet élément ne doit pas figurer.
- o) **LocationIdentifier** – Cet élément doit contenir l'identificateur d'emplacement de contenu de liste d'écoute. Il s'agit d'un nombre indiquant l'emplacement des fichiers de cette liste

d'écoute (l'emplacement peut être local, en ligne sous forme d'un contenu téléchargeable, en ligne sous forme d'un contenu en streaming ou une combinaison de ces emplacements). Ce nombre peut être obtenu dans le Tableau 15. Si ce champ est indéfini (null) ou inexistant, il convient de supposer la valeur 0 (local).

9.11 Identificateurs de classe de niveau de livre audio

Les balises suivantes permettront à l'appareil de lecture de gérer des niveaux distincts de livre audio de façon personnalisée, en fonction du type de contenu ou du sous-groupage de ces niveaux.

L'objectif est de fournir des métadonnées supplémentaires qui suivent une convention plus cohérente que les noms de niveau destinés à l'affichage utilisateur, ce qui permet d'obtenir une plateforme de lecture de livre audio reconnaissant ces métadonnées afin d'offrir une gestion plus intelligente des niveaux de livre audio particuliers.

Exemples:

```
<mpv:Seq>
  <mpv:Audio>
    <ab:unitType>chapter</ab:unitType>
    <ab:unitContext>book/novel</ab:unitContext>

<mpv:Seq>
  <mpv:Audio>
    <ab:unitType>quatrain, stanza, verse, paragraph</ab:unitType>
    <ab:unitContext>anthology/poems</ab:unitContext>
```

Le support de ces métadonnées est facultatif et variera selon les plateformes de lecture. Le niveau de support est laissé au choix des concepteurs de plate-forme et peut être déterminé par les fonctionnalités prévues de la plate-forme, son contenu anticipé ou les clients et le marché ciblés. Par exemple, une plateforme de lecture peut offrir un traitement spécial réservé strictement pour une ligne de titres de livres audio vendus par le concepteur de plate-forme ou ses partenaires d'édition, ou spécifiquement pour des titres religieux ou plus spécifiquement pour les textes religieux d'une religion donnée.

Le traitement spécial d'une plateforme peut avoir la simplicité de l'affichage d'une icône de genre dans l'interface utilisation, en modifiant le schéma de navigation pour mieux s'adapter à un genre de livre audio particulier ou en sauvegardant les préférences d'un utilisateur pour des genres spécifiques. Le traitement spécial d'une plateforme peut être activé par des mots clés d'unité uniquement ("chapitre"), par genre uniquement ("textes religieux"), ou par différentes combinaisons de l'unité, du genre et des sous-genres.

Les plateformes peuvent être conçues pour répondre à plusieurs mots clés ou seulement un ou deux. Ainsi, une plateforme peut potentiellement traiter les chapitres différemment des pages ou des phrases, traiter les romans différemment des textes religieux, traiter les chapitres des romans différemment des chapitres des textes religieux ou traiter les textes d'une religion différemment des textes d'une autre religion.

Unité – Cet élément est constitué d'un seul terme tel que "chapitre" ou "paragraphe", ou d'une liste séparée par des virgules qui commence par des termes précis spécifiques au genre et évolue vers des termes génériques largement pris en charge ("quatrain, strophe, vers, paragraphe"). L'élément est facultatif et le texte est entièrement personnalisable par les fournisseurs de contenu.

Une liste de termes séparés par des virgules peut servir de mécanisme de rétromigration, permettant aux plates-formes de lecture qui ne reconnaissent pas un mot clé plus précis de

fournir un traitement spécial à partir de solutions alternatives génériques. De plus, une liste de termes séparés par des virgules permet aux fournisseurs de contenu d'inclure des mots clés ou des orthographes translittérées à partir d'autres langues ("Ayah, verset, paragraphe, vers, paragraph") si ces termes sont plus appropriés pour la langue du contenu audio, le public ciblé ou la plate-forme de lecture anticipée.

Mots clés "unitaires" usuels (*Customary Unit Keywords*):

act, album, anthology, article, author, basket, book, branch, chapter, class, clip, collection, contributor, date, day, division, entry, event, group, half, hymn, instructions, issue date, landmark, lecture, lesson, letter, level, line, location, mark, measure, meeting, monologue, note, number, octave, page, paragraph, part, passage, performance, period, phrase, piece, poem, poet, point of interest, portion, program, quarter, recension, scene, section, segment, sentence, session, set, song, speech, subject, stage, stanza, story, teaching, testament, text, third, time, topic, tour, unit, verse, version, volume, weekday, word.

Contexte – Cet élément identifie un genre et, en option, des sous-genres plus précis séparés par des barres obliques, tels que "référence", "livre/roman", "représentation/comédie", ou "représentation/scène/pièce/acte-un". Cet élément est facultatif et le texte peut être entièrement personnalisé par les fournisseurs de contenu.

Paires de mots clés de **Contexte** usuelles (*Customary Context Keyword Pairings*):

anthology: anthology/literature, anthology/miscellaneous, anthology/poems, anthology/stories

collection: collection/literature, collection/miscellaneous, collection/poems, collection/stories

book: book/novel, book/poems, book/textbook, book/cookbook

broadcast: broadcast/online, broadcast/radio, broadcast/tv

event: event/class, event/conference, event/meeting, event/tour

education: education/class, education/conference, education/textbook, education/tour

instruction: instruction/class, instruction/conference, education/textbook, instruction/tour

performance: performance/comedy, performance/music, performance/reading, performance/spoken, performance/stage, performance/variety

periodical: periodical/journal, periodical/magazine, periodical/newsletter, periodical/newspaper

reference: reference/cookbook, reference/dictionary, reference/guide, reference/encyclopedia, reference/textbook, reference/thesaurus

scripture: scripture/abrahamic, scripture/buddhist, scripture/christian, scripture/hindu, scripture/islam, scripture/judaism, scripture/judeochristian, scripture/ids, scripture/sikh

9.12 Fichier BOOKMARK.AUB du livre audio

Les signets de livre audio facultatifs sont des fichiers XML et sont stockés sous forme de fichiers externes INDEX.AUB. Ce fichier de signet pourrait être écrit sur un support réinscriptible (mémoire FLASH interne) et référencer une liste d'écoute pour un support en lecture seule (par exemple, un CD-ROM). Le fichier de signet pourrait être écrit dans une mémoire flash réinscriptible externe puis accédé par un second appareil. Le signet pourrait être écrit à un emplacement du réseau et atteint à distance par d'autres appareils via un réseau (réseau de câbles ou réseau sans fil).

Le fichier d'exemple MPV suivant comprend deux types de signets.

AutoResume – Il s'agit d'un signet généré automatiquement par le système en fonction de certains événements; par exemple, lorsque le système est mis hors tension ou que le disque de livre audio est retiré du système. Disposer d'un fichier de signet séparé qui peut être transféré d'un appareil de lecture conforme vers un autre est une fonction facultative qui améliore considérablement l'expérience utilisateur.

UserGenerated – Il s'agit d'un signet généré par une action effectuée par l'utilisateur (activation de la touche "Signet") pour marquer un segment important du livre audio.

Le nom du fichier de signet DOIT être appelé BOOKMARK.AUB et doit être unique sur un appareil afin que les appareils de lecture trouvent facilement les signets sauvegardés. Les extensions du fichier de signet sont autorisées à condition de respecter les conventions XML.

Il ne peut y avoir qu'UN signet AutoResume par utilisateur dans un fichier BOOKMARK.AUB.

Les signets peuvent être d'une longueur supérieure à zéro pour indiquer un passage entier. Le fichier BOOKMARK.AUB ne possède les métadonnées que pour le premier fichier audio. Par conséquent, si le passage de signet commence sur un fichier audio et finit sur un autre, il incombe à l'appareil de lecture de trouver et déterminer l'emplacement final sur le second fichier audio à partir du fichier de description de volume .aub d'origine.

Le signet contient des métadonnées de haut niveau minimales sur le signet (par exemple, nom du livre, auteur, nom/numéro de chapitre, nom/numéro de page) de sorte que le consommateur puisse différencier visuellement les différents signets et que l'appareil n'ait pas à ouvrir un autre fichier pour présenter le fichier de signet au consommateur.

Il existe une balise CreatedDate que l'appareil peut créer automatiquement afin de décrire chaque signet. En outre, une balise facultative MarkName est disponible et peut être remplie par le consommateur à l'aide d'un dispositif d'entrée (clavier, télécommande, etc.) pour donner un nom plus descriptif au signet. Il convient que les signets AutoResume ne possèdent pas de balise MarkName.

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

```
<file:Manifest xmlns:file="http://ns.osta.org/manifest/1.0/"
xmlns:mpv="http://ns.osta.org/mpv/1.0/" xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"
```

```
xmlns:mpvb="http://ns.osta.org/mpv/basic/1.0/"
```

```
xmlns:dc="http://ns.osta.org/nmf/1.0/dc/"
```

```
xmlns:nmf="http://ns.osta.org/nmf/1.0/"
```

```
xmlns:aub="http://www.cea.org/dtd/2003b/1.1">
```

```
<nmf:Metadata>
```

```
<ManifestProperties xmlns="http://ns.osta.org/manifest/1.0/">
```

```
<ProfileBag>
```

```
<Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
```

```
</ProfileBag>
```

```
</ManifestProperties>
```

```
</nmf:Metadata>
```

<!--beginning of bookmarks-->

<mpvb:MarkedAssets>

<!--Bookmark 1: Autoresume-->

<mpv:MarkList mpv:markType="cea:2003b:Autoresume:2005">

<mpv:AudioRef mpv:manifestLinkIdRef="ML001" mpv:idRef="AU0003">

<ab:AudioBookProperties>

<ab:bookAuthor>Joe Bestseller</ab:bookAuthor>

<ab:bookNarrator>Norma Dialects</ab:bookNarrator>

<ab:bookTitle>Use Case Example 1</ab:bookTitle>

</ab:AudioBookProperties>

<nmf:Metadata>

<Properties xmlns="http://purl.org/dc/elements/1.1">

<contributor>Joe Bestseller</contributor>

<creator>Joe Bestseller</creator>

<title>Use Case Example 1</title>

</Properties>

<MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0">

<AlbumTitle>Use Case Example 1</AlbumTitle>

<Genre>Thriller</Genre>

</MusicProperties>

<!--A noter que les deux sections suivantes contiennent des métadonnées concernant l'UnitType particulier (chapitre ou page, par exemple) qui est le plus petit type d'unité dans lequel le Signet est placé-->

<!--Chapter 1-->

<mpv:Seq>

<mpv:Audio>

```

        <ab:unitType>chapter</ab:unitType>
        <nmf:Metadata>
            <Properties
xmlns="http://purl.org/dc/elements/1.1">
                <title>Chapter 1</title>
                <description>Chapter
1</description>
            </Properties>
            <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
                <AlbumTitle>Use Case Example 1</AlbumTitle>
                <Genre>spoken word</Genre>
            </MusicProperties>
        </nmf:Metadata>

<!--Begin Pages-->

        <mpv:Seq>
            <mpv:Audio>
                <ab:unitType>page</ab:unitType>
                <nmf:Metadata>
                    <Properties xmlns="http://purl.org/dc/elements/1.1">
                        <title>Page
1</title>
                        <description>Page 1</description>
                    </Properties>
                </nmf:Metadata>
            </mpv:Audio>
        </mpv:Seq>
    </mpv:Audio>
</mpv:Seq>

```

<!--Il s'agit du signet réel dans le décalage temporel à partir du début du fichier. Le signet peut avoir une longueur supérieure à zéro ce qui indique un passage.-->

```
<Properties xmlns="http://www.cea.org/dtd/2003b/1.1">
```

```
<CreateDate>2005-05—25T17:44:47+01:00</CreateDate>
```

```

    <MarkUser>User 1</MarkUser>

    <StartAt>(insert annodex time code here)</StartAt>

    <StopAt>(insert annodex time code here)</StopAt>

    </Properties>

    </nmf:Metadata>

    </mpv:AudioRef>

    </mpv:MarkList>

<!--Bookmark 2: user input -->

    <mpv:MarkList mpv:markType="cea:2003b:UserInputBookmark:2005">

    <mpv:AudioRef mpv:manifestLinkIdRef="ML001" mpv:idRef="AU0002">

<!--Bookmark 2: does it need a name?-->

    <ab:bookmark>My Favorite Passage</ab:bookmark>

<mpv:Seq>
    <mpv:Audio>
        <ab:AudioBookProperties>
            <ab:unitType>Collection</ab:unitType>
            <ab:unitContext>collection/stories</ab:unitContext>
            <ab:bookISBN>UC-0002-000</ab:bookISBN>
            <ab:bookAuthor>Joe Bestseller</ab:bookAuthor>
            <ab:bookNarrator>Norma Dialects</ab:bookNarrator>
            <ab:bookTitle>Use Case Example 2</ab:bookTitle>
        </ab:AudioBookProperties>
        <ab:unitType>Collection</ab:unitType>
        <ab:unitContext>collection/stories</ab:unitContext>
    </mpv:Audio>
    </mpv:Seq>
    </nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
        <contributor>Norma Dialects</contributor>
        <creator>Joe Bestseller</creator>
        <format>audio/mpeg</format>
    </Properties>

```

```

        <title>Use Case Example 2</title>
    </Properties>
    <MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0">
        <AlbumTitle>Use Case Example 2</AlbumTitle>
        <Genre>Suspense</Genre>
        <PlayingTime>420</PlayingTime>
    </MusicProperties>

<!--Story 1-->
    <mpv:Seq>
        <mpv:Audio>
            <ab:unitType>story</ab:unitType>
            <nmf:Metadata>
                <Properties xmlns="http://purl.org/dc/elements/1.1">
                    <title>Story Title</title>
                    <description>Story Title</description>
                </Properties>
                <MusicProperties
xmlns="http://ns.osta.org/mpv/music/1.0">
                    <AlbumTitle>Collection Title</AlbumTitle>
                    <Genre>Collection/Stories</Genre>
                </MusicProperties>
            </nmf:Metadata>

<!--Chapter 1-->

            <mpv:Seq>
                <mpv:Audio>
                    <ab:unitType>chapter</ab:unitType>
                <nmf:Metadata>
                    <Properties xmlns="http://purl.org/dc/elements/1.1">
                        <title>Chapter 1</title>
                        <description>Chapter 1</description>
                    </Properties>

```

```

</nmf:Metadata>

</mpv:Audio>

<!--Pages-->

<mpv:Audio>
  <ab:unitType>page</ab:unitType>
  <nmf:Metadata>
    <Properties xmlns="http://purl.org/dc/elements/1.1">
      <title>Page 1</title>
      <description>Chapter 1</description>
    </Properties>
  </nmf:Metadata>
</mpv:Audio>

</mpv:Seq>

</mpv:Audio>

</mpv:Seq>

</mpv:Audio>

</mpv:Seq>

<nmf:Metadata>
  <Properties xmlns="http://www.cea.org/dtd/2003b/1.1">
    <CreateDate>2005-05—25T17:22:23+01:00</CreateDate>
    <MarkName>Great Passage on Love</MarkName>
    <MarkUser>User 2</MarkUser>
    <StartAt>(insert annodex time code here)</StartAt>
    <StopAt>(insert annodex time code here)</StopAt>
  </Properties>
</nmf:Metadata>
</mpv:AudioRef>
</mpv:MarkList>
</mpvb:MarkedAssets>

```

```

<mpv:AssetList>
  <mpv:ManifestLink mpv:id="AU001">
    <mpv:LastURL>
      <mpv:LastURL mpv:filesystem="ISO9660-1">CHAPTER1.MP3</mpv:LastURL>
    </mpv:LastURL>
  </mpv:ManifestLink>
</mpv:AssetList>
</file:Manifest>

```

9.13 Métadonnées de niveau de livre audio MPV utilisant <mpvp:Album>

La présente Norme internationale spécifie que les propriétés de métadonnées de contenu de livre audio MPV sur un mpvp:Album s'appliquent également au contenu référencé à moins qu'elles ne soient explicitement remplacées par des métadonnées de niveau d'actif. Par exemple, la propriété <ab:BookTitle> s'applique à tous les éléments <mpvp:Audio> référencés directement ou indirectement par l'élément <mpvp:Album>.

Les métadonnées reconnues au niveau du livre audio comprennent Dublin Core, mpvm:MusicProperties, ab:AudiobookProperties et les propriétés provenant du profil de présentation MPV.

9.14 Actif musical et fixe <mpvm:AudioWithStills>

Cette Norme internationale ne spécifie pas l'utilisation de l'élément <mpvm:AudioWithStills> à ce stade. Voir ci-dessous un exemple de mise en œuvre du profil de livre audio MPV.

```

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

<file:Manifest xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:ab="http://ns.ce.org/audiobook/2003b/"
  xmlns:mpvp="http://ns.osta.org/mpv/presentation/1.0/"
  xmlns:mpvm="http://ns.osta.org/mpv/music/1.0/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:nmf="http://ns.osta.org/nmf/1.0/">

  <nmf:Metadata>
    <ManifestProperties xmlns="http://ns.osta.org/manifest/1.0/">
      <ProfileBag>
        <Profile>http://ns.ce.org/audiobook/2003b/</Profile>
        <Profile>http://ns.osta.org/mpv/presentation/1.0/</Profile>
        <Profile>http://ns.osta.org/mpv/music/1.0/</Profile>
      </ProfileBag>
    </ManifestProperties>

```

```

<Properties xmlns="http://purl.org/dc/terms/">
  <created>2003-01-29T15:51:41Z"</created>
  <modified>2003-01-29T15:51:41Z"/></modified>
</Properties>
</nmf:Metadata>

<mpvp:Album>
  <nmf:Metadata>
    <dc:Properties>
      <dc:creator>Ima Smoothtalker</dc:creator>
      <dc:identifier>put an ISBN number here</dc:identifier>
      <dc:rights>Copyright (c)2003 HomeMade Press<dc:rights/>
      <dc:title>Life Along the Willamette</dc:title>
    </dc:Properties>
    <mpvm:MusicProperties>
      <mpvm:AlbumTitle>Life Along the
Willamette</mpvm:AlbumTitle>
      <mpvm:Genre>Spoken Word:Non-Fiction</mpvm:Genre>
    </mpvm:MusicProperties>
    <ab:AudiobookProperties>
      <ab:BookAuthor>Ura Writer</ab:BookAuthor>
      <ab:BookISBN>put an ISBN number here</ab:BookISBN>
      <ab:BookNarrator>Ima Smoothtalker</ab:BookNarrator>
      <ab:BookTitle>Life Along the Willamette</ab:BookTitle>
    </ab:AudiobookProperties>
  </nmf:Metadata>
  <mpvp:Foreground>
    <mpv:SeqRef mpv:idRef="IDSeq0001"/> <!-- chapter 1 -->
    <mpv:SeqRef mpv:idRef="IDSeq0002"/> <!-- chapter 2 -->
  </mpvp:Foreground>
</mpvp:Album>

```

```

<mpv:AssetList>
  <mpv:Seq mpv:id="IDSeq0001"> <!-- chapter 1 -->
    <mpv:AudioRef mpv:idRef="ID000001"/> <!-- page -->
    <mpv:AudioRef mpv:idRef="ID000002"/> <!-- page -->
    <mpv:AudioRef mpv:idRef="ID000003"/> <!-- page -->
  </mpv:Seq>
  <mpv:Seq mpv:id="IDSeq0002"> <!-- chapter 2 -->
    <mpv:AudioRef mpv:idRef="ID000004"/> <!-- page -->
    <mpv:AudioRef mpv:idRef="ID000005"/> <!-- page -->
    <mpv:AudioRef mpv:idRef="ID000006"/> <!-- page -->
  </mpv:Seq>
  <mpv:Audio mpv:id="ID000001"> <!-- other metadata inherited from Audiobook
-->
    <nmf:Metadata>
      <Properties xmlns="http://purl.org/dc/elements/1.1">
        <format>audio/mpeg</format>
        <title>page 1</title>
      </Properties>
      <MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0/">
        <PlayingTime>3.48</PlayingTime>
      </MusicProperties>
    </nmf:Metadata>
    <mpv:LastURL>audio/page1.mp3</mpv:LastURL>
  </mpv:Audio>
  <mpv:Audio mpv:id="ID000002">
<!--this explicit metadata overrides Audiobook; happens to be the same here -->
    <nmf:Metadata>
      <Properties xmlns="http://purl.org/dc/elements/1.1">
        <contributor>Ima Smoothtalker</contributor>
        <creator>Ura Writer</creator>
        <format>audio/mpeg</format>
        <title>page 2</title>
      </Properties>
      <MusicProperties xmlns="http://ns.osta.org/mpv/music/1.0/">

```

```

        <AlbumTitle>Life Along the Willamette</AlbumTitle>

        <Genre>Spoken Word:Non-Fiction</Genre>

        <PlayingTime>3.24</PlayingTime>

    </MusicProperties>

</nmf:Metadata>

    <mpv:LastURL>audio/page2.mp3</mpv:LastURL>

</mpv:Audio>

...

    </mpv:AssetList>

</file:Manifest>

```

9.15 Extensions MPV-Audiobook à MPV-Core

9.15.1 Types de fichier et extensions de manifeste de livre audio

9.15.1.1 Extension de fichier

Pour les systèmes dans lesquels le type de fichier est indiqué par l'extension de nom de fichier, tels que Microsoft Windows et Unix, le manifeste MPV-Audiobook doit utiliser une extension. Cette Norme internationale définit les deux extensions qu'un manifeste peut porter. Chaque livre audio MPV doit utiliser une des extensions de fichier MPV spécifiques au livre audio suivantes.

- a) **.aub** – Cette extension identifie un fichier en tant que fichier (AUB) de profil MPV-Audiobook. Sa syntaxe n'est pas sensible à la casse. Cette extension peut être enregistrée par une application pour fournir des processeurs par défaut et de remplacement de manifestes de livre audio.
- b) **.mum** – Cette extension identifie un fichier en tant que Manifeste musical (MUM). Les livres audio conformes à cette Norme internationale peuvent utiliser l'extension de fichier MPV spécifiée par le Profil MPV Music-- .mum. La syntaxe n'est pas sensible à la casse. Cette extension peut être enregistrée par une application pour fournir des processeurs par défaut et de remplacement de manifestes musicaux.

9.15.1.2 Type de fichier Apple Macintosh

Le système d'exploitation Apple Macintosh utilise un type de fichier interne stocké en tant que valeur de ressource de la fourchette de données d'un fichier. Les livres audio conformes à cette Norme internationale doivent utiliser un type de fichier MPV spécifique au livre audio, comme suit.

- a) **.aub** – Ce type de fichier Apple Macintosh identifie le fichier qui doit contenir un manifeste de livre audio. La syntaxe est sensible à la casse. Cette extension peut être enregistrée par une application pour fournir un processeur par défaut de manifestes de livre audio.
- b) **.mum** – Ce type de fichier Apple Macintosh identifie le fichier qui doit contenir un manifeste musical. En outre, les livres audio conformes à cette Norme internationale peuvent utiliser le même type de fichier que les manifestes MPV Music sur les systèmes Macintosh. La syntaxe est sensible à la casse. Cette extension peut être enregistrée par une application pour fournir un processeur par défaut de manifestes musicaux.

Certaines applications examinent les caractères d'interlignage d'un fichier afin d'essayer d'en déterminer le type. Aucune séquence d'octets ne peut être présente de façon certaine mais généralement tous les documents XML du jeu de caractères UTF-8 commencent par la valeur