ISO/IEC JTC 1/SC 34/JWG 7

Joint JTC 1/SC 34-TC 46/SC 4-IEC/TC 100/TA 10 WG: EPUB

Convenorship: KATS (Korea, Republic of)

Document type:	Final Text submitted for TR publication
Title:	Call for final review for publication of ISO/IEC TS 22424-2 Digital publishing — EPUB 3 Preservation — Part 2: Metadata requirements
Status:	This document is final text for publication of ISO/IEC TS 22424-2 Digital publishing — EPUB 3 Preservation — Part 2: Metadata requirements reflecting comments from PDTS ballots. Through four-week review in SC 34 and SC 34/JWG 7 this document will be submitted to SC 34 for publication. If you have any further comments or suggestion on this document, please let project editor (Juha Hakala, juha.hakala@helsinki.fi) by August 23, 2019.
Date of document:	2019-07-17
Source:	Project editor (Juha Hakala)
Expected action:	INFO
No. of pages:	54
Email of convenor:	samoh@g.skku.edu, zzosang@gmail.com
Committee URL:	https://isotc.iso.org/livelink/livelink/open/jtc1sc34jwg7

ISO/IEC TS 22424-2:2019

ISO/IEC JTC 1/SC 34/JWG 7

Secretariat: JISC

Digital publishing — EPUB 3 Preservation — Part 2: Metadata requirements

Édition numérique — Archivage pérenne de l'EPUB3 — Partie II : Exigences sur les métadonnées

TS stage

Warning for WDs and CDs

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 34, *Document description and processing languages*.

A list of all parts in the ISO/IEC TS 22424 series can be found on the ISO website.

Introduction

This document facilitates the long-term preservation of EPUB publications by specifying metadata elements which are required or recommended for long-term preservation (such as identifiers) and the ways in which the EPUB publication and related metadata can be packaged. EPUB versions 3.0 and later are covered; if necessary, the EPUB version applicable is specified.

Long-term preservation in general requires two things:

- making the object such as EPUB publication fit for preservation – including features to be used and feature to avoid;
- the packaging of the object (and any metadata related to it) together with any additional data such as other versions of the object and other documentation into an OAIS Submission Information Package (SIP).

Part 1 of this technical specification concentrates on the archivability of EPUB documents.

The background to this document comes from the Open Archival Information System, which is described in Part 1 of this standard.

When a Submission Information Package (SIP) is formed, mandatory preservation metadata SHALL be present in the package. Depending on the agreements made between the producer and the archive, metadata elements are stored either in the container document or the EPUB publication itself, or both. Usually an archive would expect to find all relevant metadata in the container, unless the submission agreement allows embedding of metadata into EPUB publications.

This document does not require any changes to be made to the current of future EPUB standards. However, when an EPUB publication is created or modified for submission to an archive, there are some EPUB features that should be used and others that should be avoided. Part 1 of this technical specification describes how the EPUB format SHOULD be applied. This Part 2 document concentrates on mandatory and recommended metadata elements needed for the long-term preservation of EPUB publications and their METS encoding. Part 1 recommends the usage of METS but allows also other container standards; this part concentrates on preservation metadata and its METS encoding in SIPs. Future editions of these documents MAY specify other encodings such as BITS (Book Interchange Tag Suite)[1].

In order to guarantee access to documents, OAIS archives may migrate documents into new file formats when the original formats are no longer supported by commonly used rendering tools. If the document to be migrated is an e-book in an outdated EPUB format, migration can be made to a more modern version of EPUB or, at least in principle, to another e-book format.

Generally, migration into another file format should be straightforward if the current and new format are compatible and there are efficient and reliable migration tools available. If the target format is a more modern version of the current format, compatibility should not be a problem. But if a format is rich, migration tools may not be able to render all the properties of a resource.

This document applies to EPUB versions 3 and 3.0.1. Earlier versions (EPUB 2 and 2.0.1) are not covered. Since there are no implementations of version 3.1, it is not covered in this document either. EPUB 3.2 was published in May 2019[2]. It will be taken into account in the next edition of this document.

[1] https://www.loc.gov/preservation/digital/formats/fdd/fdd000453.shtml

[2] https://w3c.github.io/publ-epub-revision/epub32/spec/epub-spec.html

This document does not cover issues related to migration between EPUB versions or from EPUB to other e-book formats. Migration to other formats is often lossy; this applies to e-book formats as well, since there are EPUB features which are not supported in other e-book formats, and vice versa. Moreover, even if the same feature is supported, technical implementations can be incompatible. For instance, if an EPUB 3 publication using fixed layout is migrated to Amazon's KF8 format, preserving fixed layout properties requires special attention since there are significant technical differences between these formats in how this feature has been implemented.

Sometimes migration cannot be applied at all; programs cannot be migrated without access to and good understanding of the source code. In such cases long-term preservation is possible only if the OAIS archive responsible is able to emulate either the program's original hardware or software environment.

Within the preservation community, emulation is considered to be a viable option for some content. For the time being there is no full understanding on how emulation will function in the long-term, but this may change with Emulation as a Service approaches coming to the market.

Metadata requirements in this document are based on the migration of file formats. Emulation is not covered (just a single example of emulation-related preservation metadata is given), although emulation is likely to be the best preservation method for fixed layout EPUB publications and interactive EPUB publications. Preservation metadata requirements for emulation-based preservation strategy may be added into a future version of this document.

Supporting emulation might require just information about appropriate tools in the submission agreement or in the related documentation. A more sustainable approach is to include a description of the emulation environment (hardware and/or software) in the premis:object section of the PREMIS metadata record in the SIP. During ingest this information is copied into AIP. If migration is used, hardware and software environments needed for rendering the versions of the document in the AIP can be specified separately as access environments.

Ambition level of migration may vary. Usually it is to preserve the intellectual content, since retaining also the original look and feel of preserved documents is considered to be too demanding. If semantics and layout are interlinked, it is important to keep also the original EPUB publication in order to facilitate preservation of the semantics via emulation-based access to the original content.

Migration both requires and produces preservation metadata. For instance, staff in the archives has to figure out which tools can be used to carry out the migration, and what weak points they may have. The intention of the preservation community is to maintain this information in format libraries such as PRONOM[3] . When a new Archival Information Package (AIP) is created after a migration, the package should contain both the old and the new representation of the migrated document and preservation metadata describing the migration event and the possible differences between the document versions[4]. Depending on their needs and archived resources archive users can then make a choice between the original, which is authentic but possibly difficult to render, and the migrated document, which should be easy to use but less authentic. In practice, finding access software to outdated versions of preserved documents may be difficult. The OAIS archive, on the other hand, can migrate the original document again when better tools can be used, or if there are significant issues in migrated documents.

Metadata elements that SHALL be included in SIPs are a priori essential for digital preservation. For instance, if there is no digital signature present and a secure transfer channel has not been used, it is impossible to guarantee the information entering the archive has not changed during transfer or that it

[3] http://www.nationalarchives.gov.uk/PRONOM/Default.aspx

[4] This document is only concerned with those metadata elements which are to be included in SIPs. Preservation metadata needed in AIPs (which describes the preservation related events such as migration) is beyond the scope.

is coming from a correct source. Moreover, if the data has already been tampered with before it enters the archive, all subsequent preservation actions may be useless.

This document does not specify generic conformance requirements for EPUB publications, but may make some restrictions to the use of EPUB specifications. The generic conformance requirements made in the EPUB Contents Documents Specification apply to EPUB publications in SIPs as well.

Part 1 of this document defined a set of requirements for archivable EPUB publications. Below is a short summary of these requirements:

- SIPs SHALL contain the entire EPUB publication including the fonts used. All publication resources SHOULD be embedded in the EPUB container, including audio and video resources. Linked resources MAY be used if the archive is able to retrieve the resources during ingest and incorporate them into the AIP. SIPs SHALL NOT contain viruses or anything else not part of the submitted EPUB publications.
- Preview EPUB publication MAY be submitted, if it will be updated with the final version of the publication once it is available, or if it is not possible to submit the final version. If both preview and final version are submitted, it is a good practice to keep both in AIP, and indicate in metadata which one is which.
- Submitted resources SHOULD NOT be DRM protected, encrypted, or obfuscated. If any of these mechanisms have been used, the archive SHALL be permitted to remove them during ingest.
- If an EPUB content document in a SIP contains scripting, the EPUB publication SHALL contain a fallback for the content in question. In the EPUB context scripting enables the use of JavaScript applications for e.g. image manipulation or enabling dynamic changes of the content. Preserving such functionality in the long-term can be difficult and might require emulation.
- If there are core media type resources or foreign resources in EPUB publications that need to be preserved, their file formats SHOULD be approved for ingest and/or preservation[5]. If there is un-archivable content, these resources SHALL be an archivable fallback and the files that cannot be preserved (except in bit level) SHALL be encoded in a way that they SHALL NOT be checked during ingest but stored as such in AIP.
- EPUB reader or readers that is/are known to be able to render the submitted publication correctly SHOULD be specified in the SIP.
- Canonical fragment identifiers[6] SHOULD NOT be used in EPUB publications submitted to an archive, because if/when the publication is migrated into another file format, these identifiers may stop functioning properly.
- Fixed layout EPUB publications can be submitted. However, if their meaning is not dependent on the layout, reflowable versions SHOULD be submitted as well, since preserving the original look and feel can be impossible in the long term. Even if a SIP contains also a reflowable version of the publication, the SIP SHOULD contain metadata required for the emulation-based preservation such as description of the original usage environment.

For a more complete description of the above requirements, please consult Part 1 of this document.

[5] Acceptable formats shall be defined in the submission agreement.

[6] http://www.idpf.org/epub/linking/cfi/epub-cfi.html

Digital publishing — EPUB 3 Preservation — Part 2: Metadata requirements

1 Scope

This document supports long-term preservation of EPUB publications via a dual strategy. First, it considers EPUB features from long-term preservation point. Some EPUB features are forbidden and some others required, depending on how they relate to long-term preservation. An EPUB document constructed according to these guidelines are suitable for preservation. In this respect, this document is related to EPUB in the same way than PDF/A is related to PDF.

Second, this document makes EPUB compliant with current practices of digital archives and technical requirements of long-term preservation applications. The former tend to rely on Open Archival Information Systems (OAIS) in their operations; the latter prefer to ingest electronic documents only in containers which conform to standards such as METS (Metadata Encoding and Transmission Standard).

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 TS 30135 (all parts), *Information technology — Digital publishing — EPUB3*

ISO 14721. *Space data and information transfer systems – Open archival information system (OAIS) – Reference model*

ISO 15836-1:2017. *Information and documentation –The Dublin Core metadata element set – Part 1: Core elements.*

METS. *Metadata Encoding & Transmission Standard. Version 1.11.* [online]. Library of Congress, 2015. Available from: https://www.loc.gov/standards/mets/

PREMIS. *PREMIS Data Dictionary for Preservation Metadata. Version 3.0.* [online]. Library of Congress, 2015. Available from http://www.loc.gov/standards/premis/

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply. Unless stated otherwise, the terms have been adopted from ISO 14721:2012.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— IEC Electropedia: available at http://www.electropedia.org/

— ISO Online browsing platform: available at https://www.iso.org/obp

3.1
access functional entity
OAIS functional entity that contains the services and functions, which make the archival information holdings and related services visible to Consumers

3.2

administrative metadata

metadata that provides information to help manage a resource, such as when and how it was created, file type and other technical information, and access rights

[SOURCE: Understanding metadata]

3.3

archival information package

AIP

Information Package consisting of Content Information and associated Preservation Description Information (PDI), which is preserved within an OAIS

3.4

archive

OAIS archive

organization that intends to preserve information for access and use by a Designated Community

3.5

authenticity

property than an entity is what it claims to be

[SOURCE: ISO/IEC 27000]

Note 1 to entry: Authenticity is judged on the basis of evidence.

3.6

bit preservation

term used to denote a very basic level of preservation of digital resource as it has been submitted (literally the preservation of the **bits** forming a digital resource)

Note 1 to entry: This may include maintaining onsite and offsite backup copies, virus checking, fixity-checking, and periodic refreshing to a new storage medium.

Note 2 to entry: Bit preservation is not digital preservation but it does provide a building block for the more complete set of digital preservation practices and processes that ensure the survival of digital content and also its usability, display, context and interpretation over time.

[SOURCE: Digital preservation handbook, Glossary]

3.7

consumer

role played by those persons or client systems, who interact with OAIS services to find preserved information of interest and to access that information in detail

Note 1 to entry: This can include other OAISs, as well as internal OAIS persons or systems.

3.8

content information

set of information that is the original target of preservation or that includes part or all of that information

Note 1 to entry: It is an Information Object composed of its Content Data Object and its Representation Information.

3.9
context information
information that documents the relationships of the Content Information to its environment

Note 1 to entry: This includes reasons why the Content Information was created and how it relates to other Content Information objects.

3.10
core media type resource
a publication resource that is a core media type and may therefore be included in the EPUB publication withouth the provision of fallbacks.

[SOURCE: EPUB Publications 3.0.1]

Note 1 to entry: Core media types have been specified in chapter 5.1. of the EPUB publications specification, version 3.0.1. For instance, core media types for still images are image/gif, image/jpg, image/png and image/svg+xml. Any other still image file format is foreign and requires a fallback, meaning the same resource expressed in another foreign format or core media type.

3.11
data, pl
reinterpretable representation of information in a formalized manner suitable for communication, interpretation, or processing

[SOURCE: ISO 5127:2017]

Note 1 to entry: Data are often understood as taking the form of a set of values of qualitative or quantitative variables.

3.12
data dictionary
organized and constructed (electronic data base) compilation of descriptions of data concepts that provides a consistent means for documenting, storing and retrieving the syntactical form (i.e. representational form) and the meaning and connotation of each data concept

[SOURCE: ISO 24531:2013]

Note 1 to entry: PREMIS[7] is a data dictionary.

3.13
descriptive metadata
descriptive information
metadata about a resource for example for discovery and identification

Note 1 to entry: These can include elements such as title, abstract, author, and keywords.

[SOURCE: Understanding metadata]

3.14
designated community
identified group of potential Consumers who should be able to understand a particular set of information

[7] PREMIS Data Dictionary for Preservation Metadata (https://www.loc.gov/standards/premis/) is a leading metadata specification for metadata needed for long-term preservation.

Note 1 to entry: A Designated Community may be composed of multiple user communities. The community is defined by an Archive, though this definition may change later on.

3.15
digital preservation
series of managed activities necessary to ensure continued access to digital materials for as long as necessary

Note 1 to entry: Digital preservation refers to all of the actions required to maintain access to digital materials beyond the limits of media failure or technological and organizational change

Note 2 to entry: Those materials may be records created during the day-to-day business of an organization; "born-digital" materials created for a specific purpose (e.g. teaching resources); or the products of digitisation projects.

EXAMPLE 1 **Short-term preservation -** Access to digital materials either for a defined period of time while use is predicted but which does not extend beyond the foreseeable future and/or until it becomes inaccessible because of changes in technology.

EXAMPLE 2 **Medium-term preservation -** Continued access to digital materials beyond changes in technology for a defined period of time but not indefinitely.

EXAMPLE 3 **Long-term preservation -** Continued access to digital materials, or at least to the information contained in them, indefinitely.

[SOURCE: Digital preservation handbook, Glossary]

3.16
digital rights management
DRM
packaging, distributing, controlling, and tracking content based on rights and licensing information

[SOURCE: ISO 19153:2014]

3.17
digital signature
signature
data appended to, or a cryptographic transformation of, a data unit that allows the recipient of the data unit to prove the source and integrity of the data unit and protect against forgery, e.g. by the recipient

[SOURCE: ISO/IEC 19784-1:2006]

3.18
dissemination information package
DIP
information package, derived from one or more AIPs, sent by an Archive to a Consumer in response to a request in the OAIS

3.19
distributable object
component of an EPUB publication that can be reused in other contexts

Note 1 to entry: A Distributable Object can be a complete EPUB Content Document (e.g., a chapter of a book), a section of such a document (e.g., an exercise or a promotional excerpt), a media resource (e.g., a video or interactive feature), or a combination of such resources that are not necessarily contiguous within the parent EPUB publication but are intended to be able to be distributed as a unit.

[SOURCE: EPUB Distributable Objects 1.0]

3.20

electronic book

e-book

non-serial digital document, licensed or not, where searchable text is prevalent, and which can be seen in analogy to a print book

Note 1 to entry: The use of e-books is, in many cases, dependent on a dedicated device and/or a special reader or viewing software.

[SOURCE: ISO 2789:2013]

3.21

EPUB container

ZIP based packaging and distribution format for EPUB publications

[SOURCE: EPUB Publications 3.0.1]

3.22

EPUB content document

publication resource that conforms to one of the EPUB content document definitions

[SOURCE: EPUB Publications 3.0.1]

3.23

EPUB navigation document

specialization of the XHTML content document, containing human- and machine-readable global navigation information

[SOURCE: EPUB Publications 3.0.1]

3.24

EPUB publication

collection of one or more renditions conforming to the EPUB specifications, packaged in an EPUB container

[SOURCE: EPUB Publications 3.0.1]

3.25

EPUB reading system

system that processes EPUB publications for presentation to a user in a manner compliant with EPUB specifications

[SOURCE: EPUB Publications 3.0.1]

3.26

fallback

mechanism with which versions of the same resource in different file formats can be linked to one another

[SOURCE: EPUB Publications 3.0.1]

Note 1 to entry: A reading system that does not support the file format of a foreign resource shall traverse the fallback chain until it finds a version it can render.

3.27

fixity information

information that documents the authentication mechanisms and provides authentication keys to ensure that the Content Information object has not been altered in an undocumented manner

[SOURCE: ISO 13527:2010]

3.28

foreign resource

publication resource that is not a core media type

[SOURCE: EPUB Publications 3.0.1]

3.29

identifier

data string or pointer that establishes the identity of an item, institution, or person alone or in combination with other elements.

[SOURCE: ISO 8459:2009]

Note 1 to entry: EPUB 3 specifies Unique Identifiers and Release Identifiers; the latter is a combination of a Unique Identifier and the last modification data of the rendition of the resource.

3.30

independently understandable

characteristic of information that is sufficiently complete to allow it to be interpreted, understood, and used by the Designated Community without having to resort to special resources not widely available, including named individuals

3.31

information

any type of knowledge that can be exchanged

Note 1 to entry: In an exchange, this is represented by data

EXAMPLE a string of bits (the data) accompanied by a description on how to interpret the string of bits as numbers representing temperature observations measured in degrees Celsius (the representation information)

3.32

information package

logical container composed of optional content information and optional associated preservation description information

3.33

ingest functional entity

OAIS functional entity that contains the services and functions that accept SIPs from producers, prepares AIPs for storage, and ensures AIPs and their supporting descriptive information become established within the OAIS

3.34

long-term

period of time long enough to raise concerns about the impact of changing technologies, including support for new media and data formats, and of a changing designated community, on the information being held in an OAIS

Note 1 to entry: This period extends into the indefinite future.

3.35

long-term preservation

act of maintaining information, independently understandable by a designated community, with evidence supporting its authenticity over the long-term

3.36

manifest

EPUB manifest element provides an exhaustive list of the Publication Resources that constitute the given Rendition, each represented by an item element.

[SOURCE: EPUB Publications 3.0.1]

3.37

metadata

data about other Data, documents, or records that describe their content, context, structure, format, provenance, and/or rights.

[SOURCE: ISO 5127:2017]

3.38

METS

Metadata Encoding and Transmission Standard, a standard for presenting metadata using XML.

[SOURCE: Digital preservation handbook, Glossary]

3.39

migration

means of overcoming technological obsolescence by transferring digital resources from one hardware/software generation to the next

Note 1 to entry: The purpose of migration is to preserve the intellectual content of digital objects and to retain the ability for clients to retrieve, display, and otherwise use them in the face of constantly changing technology.

Note 2 to entry: Migration differs from the refreshing of storage media in that it is not always possible to make an exact digital copy or replicate original features and appearance and still maintain the compatibility of the resource with the new generation of technology.

[SOURCE: Digital preservation handbook, Glossary]

3.40

Open Archival Information System

OAIS

archive, consisting of an organization, which may be a part of a larger organization, of people and systems, that has accepted the responsibility to preserve Information and make it available to a Designated Community. It has a set of responsibilities, as defined in section 4, which allow an OAIS Archive to be distinguished from other uses of the term 'Archive'.

Note 1 to entry: The term 'Open' in OAIS is used to imply that this Recommendation and future related Recommendations and standards are developed in open forums, but it does not imply access to the Archive is unrestricted.

Note 2 to entry: The OAIS abbreviation is also commonly used to refer to the Open Archival Information System Reference Model standard which defined the term. The standard is a conceptual framework describing the environment, functional components, and information objects associated with a system responsible for long-term preservation.

3.41

package document

publication resource that describes one rendition of an EPUB publication, as defined in package document. The package document carries meta information about the Rendition, provides a manifest of resources and defines the default reading order.

[SOURCE: EPUB Publications 3.0.1]

Note 1 to entry: It specifies all tools required to render the document, provides an exhaustive list of resources belonging to the document, and defines their default reading order.

3.42

PDF

Portable Document Format, a set of formats and open standards maintained by the International Organization for Standardization for producing and sharing electronic documents

Note 1 to entry: Originally developed by Adobe Systems.

[SOURCE: Digital preservation handbook, Glossary]

3.43

PDF/A

versions of the PDF standard intended for archival use

[SOURCE: Digital preservation handbook, Glossary]

3.44

pre-ingest

actions required before data can be submitted into an OAIS archive, including negotiation of data acquisitions, checking rights and access criteria, licensing, and data submission

Note 1 to entry: This area also includes activities involving data producer support and training.

Note 2 to entry: Pre-ingest is not a function in the standard OAIS model, but activities in this area can form a significant part of a producer's responsibilities.

[SOURCE: UK Data Archive. Archive training manual8]

3.45

preservation description information

PDI

information necessary for the adequate preservation of Content Information that can be categorized as provenance, reference, fixity, context, and rights information

3.46

preservation metadata

metadata containing information needed to archive and preserve a resource

[SOURCE: Understanding metadata]

3.47

preservation planning functional entity

OAIS functional entity that provides the services and functions for monitoring the environment of the OAIS and that provides recommendations and preservation plans to ensure information stored in the

[8] http://www.data-archive.ac.uk/curate/archive-training-manual/pre-ingest

OAIS remains accessible to, understandable by, and sufficiently usable by the designated community over the long-term, even if the original computing environment becomes obsolete

3.48

producer

role played by those persons or client systems that provide the information to be preserved

Note 1 to entry: This can include other OAISs or internal OAIS persons or systems.

3.49

provenance information

information that documents the history of the Content Information

Note 1 to entry: This information states the origin or source of the Content Information, any changes that may have taken place since it was generated, and who has had custody of it.

Note 2 to entry: The Archive is responsible for creating and preserving Provenance Information from the point of ingest; however, earlier Provenance Information should be provided by the Producer. Provenance Information adds to the evidence to support authenticity.

3.50

publication resource

resource that has the content or instructions contributing to the logic and rendering of at least one rendition of an EPUB publication

EXAMPLE Examples of publication resources include a rendition’s Package Document, EPUB Content Document, EPUB style sheets, audio, video, images, and embedded fonts and scripts.

3.51

reading system

system that processes EPUB publications for presentation to a user in a manner conformant with EPUB specification

[SOURCE: Modified from EPUB Publications 3.0.1]

3.52

reference information

information that is used as an Identifier for the Content Information

Note 1 to entry: This also includes Identifiers that allow outside systems to refer unambiguously to a particular Content Information.

EXAMPLE an ISBN is a type of Reference Information.

3.53

reference model

framework for understanding significant relationships among entities in an environment and for the development of consistent standards or specifications supporting that environment

Note 1 to entry: A Reference Model is based on a small number of unifying concepts and may be used as a basis for education and explaining standards to a non-specialist.

3.54

reformatting

copying information content from one storage medium to a different storage medium (media reformatting) or converting from one file format to a different file format (file reformatting)

[SOURCE: Digital preservation handbook, Glossary]

3.55

refreshing

copying information content from one storage media to the same storage media

[SOURCE: Digital preservation handbook, Glossary]

3.56

release identifier

identifier that allows any instance of an EPUB publication to be compared against another to determine if they are identical, different versions, or unrelated

[SOURCE: EPUB Publications 3.0.1]

Note 1 to entry: Release Identifiers consist of a unique identifier and the last-modified date of the document.

3.57

remotely-hosted resource

objects hosted outside the EPUB Container.

3.58

rendition

one rendering of the content of an EPUB publication, as expressed by an EPUB package

[SOURCE: EPUB Publications 3.0.1]

3.59

repository system

long-term preservation system used by an archive

3.60

rights management metadata

information that identifies the access restrictions concerning the Content Information, including the legal framework, licensing terms, and access control

Note 1 to entry: This contains the access and distribution conditions stated in the Submission Agreement, related to both preservation (by the OAIS) and final usage (by the Consumer).

Note 2 to entry: It also includes specifications for the application of rights enforcement measures.

3.61

spine

EPUB spine element defines the default reading order of the EPUB Publication content by defining an ordered list of manifest item references.

[SOURCE : EPUB Publications 3.0.1]

3.62

structural metadata

metadata that indicates how compound objects are put together, for example how the pages of a document are arranged to form chapters

[SOURCE: Understanding metadata]

3.63

submission agreement

agreement reached between an OAIS archive and a Producer that specifies a data model and any other arrangements needed for the data submission session

Note 1 to entry: This data model identifies the format/content and the logical constructs used by the Producer and how they are represented on each media delivery or in a telecommunication session.

3.64

submission information package

SIP

information package that is delivered by a Producer to an OAIS to be used to construct or update one or more AIPs and/or the associated descriptive information.

3.65

unique identifier

primary identifier of an EPUB publication, which may be shared by one or several renditions of the same EPUB publication that conform to the EPUB standard and embody the same content.

[SOURCE: EPUB Publications 3.0.1]

3.66

XHTML content document

EPUB content document that conforms to the profile for HTML defined in XHTML Content Documents

[SOURCE: EPUB Publications 3.0.1]

Note 1 to entry: see EPUB Content Documents 3.0.1, chapter 2.

4 Abbreviated terms

AIP	Archival Information Package
DIP	Dissemination Information Package
DRM	Digital Rights Management
OAIS	Open Archival Information System
PDI	Preservation Description Information
SIP	Submission Information Package

5 Syntax

This document provides examples of how metadata elements SHOULD be expressed using either

1) Metadata Encoding and Transmission Standard (METS[9]) version 1.11 and PREMIS Data Dictionary for Preservation Metadata (PREMIS[10]) version 2.2, and/or
2) EPUB version 3.0 and 3.0.1

[9] http://www.loc.gov/standards/mets/

[10] http://www.loc.gov/standards/premis/

for encoding SIPs. Other container standards MAY be added to the future editions of this document.

This dual approach was chosen because there are different options available for a producer to turn existing EPUB publications into SIPs:

1) All metadata (mandatory and otherwise) may be embedded in the EPUB publication.
2) Mandatory metadata is copied from EPUB document to the METS container if and when it is already present, or created and placed in the METS container (recommended approach).
3) Option 2, but a container standard other than METS is used.

The first option looks appealing because that way it would be relatively easy to create EPUB publications suitable for long-term preservation, especially if the mandatory metadata elements are already present (and if the EPUB publication itself does not have features unsuitable for preservation).

Unfortunately this approach has some issues:

- Commonly used repository systems expect information packages based on container standards such as METS. Current versions of these applications may not able to process SIPs which contain only an EPUB publication.
- Depending on the mandatory metadata required, it may not be possible to include all preservation metadata into EPUB publication.
- If there is no container document, it may be difficult to send multiple EPUB publications in a single SIP, or partial updates (for instance, only descriptive metadata about a publication that has already been archived.

Options 2 and 3 are based on the idea that there are two independent specifications, the core EPUB specification (currently version 3.1), and a container specification (this document). This allows the two communities (EPUB and digital archivists) to cooperate without putting unnecessary constraints on each other. Both specifications are independent from one another, which makes it easier to manage them.

From a technical point of view, the main strength of the second option is that METS containers are almost universally accepted in long-term preservation applications. One reason for the popularity of the standard is that it is flexible – it is possible to embed any descriptive or administrative metadata into a METS document. Whatever mandatory metadata will be agreed upon by the producer and the OAIS archive, METS can be used as a container.

The option of using some other container standard than METS or EPUB has not been examined while preparing this document. ISO/IEC JTC 1/SC 34 JWG7 decided early on to use METS due to its technical features and popularity among long-term preservation application vendors as well as libraries, archives, and museums. If and when other options emerge in the future, it is possible to extend this document to support other container standards as well.

The main weakness of METS approach is that currently very few publishers support it. Unless production processes change radically, a common solution will be to submit e-books in EPUB format as such, with accompanying ONIX metadata. In this approach, the producer (which can be the OAIS archive) creates the METS SIP during pre-ingest, using the data and metadata delivered by the publisher. The publisher does not need to know METS, but EPUB documents themselves and the accompanying metadata SHOULD meet the requirements made in the submission agreement.

This document requires that each SIP SHALL have a METS document with mandatory descriptive and administrative metadata elements embedded, using e.g. Dublin Core and PREMIS formats. The use of a separate, METS based preservation layer enables the current long-term preservation applications to ingest EPUB publications. Producers and OAIS archives MAY also choose other approaches, such as embedding all metadata in EPUB publications or using another container standard. Whichever strategy is chosen, it should be planned out carefully.

In the hybrid approach, some descriptive and administrative metadata needed during ingest MAY not be copied from the EPUB document to the METS document. In order to use this metadata, the OAIS archive SHALL have reading systems or other applications which are able to render EPUB publications and extract the relevant metadata from them.

This document does not require copying of EPUB structural metadata to METS documents. Therefore the structural metadata in METS is simple, only specifying the location of EPUB publication or publications in the SIP but not their internal structure. EPUB reading systems would not be able to use the structural metadata in a METS document, because they utilize structural metadata in the EPUB spine element when publications are rendered.

In order to eliminate uncertainty concerning the syntax and semantics of SIPs, submission agreements SHALL specify a METS profile or profiles which can be used to facilitate packaging of EPUB publications. This document can be used as a basis for these profiles. The profile can be part of the submission agreement, or linked to it. The latter approach was chosen in the Finnish Digital Library initiative; the benefit is that submission agreements will be relatively simple because technical details are stated in the document "Metadata requirements and preparing content for digital preservation"[11]. Finnish Digital Library initiative has published also a separate document titled "File formats"[12], which lists the file formats suitable for ingest and preservation. Unfortunately this document does not contain guidelines on how these file formats should be applied. EPUB is an example of a file format which is in principle archivable, but in practice can be used in a way which may makes long-term preservation challenging. The purpose of Part 1 of this document is to provide guidelines for creation of archivable EPUB publications.

Specifications, such as the ones created in Finnish Digital Library initiative, SHALL be sufficiently detailed; for instance, they SHALL specify all mandatory metadata elements and all archivable or ingestible file formats. Otherwise SIPs may lack crucial data, or contain files that cannot be processed. Of course even this may not be sufficient; in addition to only saying that MXF, TIFF and EPUB are archivable formats, it is also necessary to specify what type of MXF videos, TIFF images and EPUB publications are acceptable. Digital archiving projects like the National Digital Library in Finland do not necessarily have a mandate or resources for such work; that is why specifications like this one for EPUB, AS-07 for archivable MXF[13] and TI/A[14] for archival of TIFF images are needed.

If just listing all the archivable file formats is not enough, it is also insufficient to provide just a list of mandatory preservation metadata elements. Element specific guidelines are often necessary. For instance, it is not enough to just say that SIPs must contain identifiers for EPUB publications. Producer and OAIS archive SHALL also agree on what needs to be identified (for instance, EPUB publications, their component parts, metadata records), which identifiers (ISBNs, DOIs, URNs, etc.) are accepted and – just to give an EPUB specific example on identifier usage – whether EPUB release identifiers are acceptable. Metadata is crucial in digital archiving, because it affects all the steps in the preservation process – ingest, archival, and dissemination. When a producer and an OAIS archive decide on which identifiers to use, this may have an impact not only on SIPs, but also on Archival Information Packages (AIPs) and Dissemination Information Packages (DIPs) the archive will be able to create.

If the SIP does not meet the requirements, usually the ingest process fails and OAIS archive asks the provider to fix the problem. But submission agreement can specify other approaches; for instance, if the provider does not have sufficient technical skills, the OAIS archive or a third party could take care of fixing technical problems in submitted EPUB documents may be submission agreement. It might even be possible to ignore certain minor issues during ingest, although even minor problems may endanger long term preservation.

[11] http://digitalpreservation.fi/files/Metadata-1.7.0-en.pdf

[12] http://digitalpreservation.fi/files/File-Formats-1.6.1-en.pdf

[13] https://www.amwa.tv/projects/AS-07.shtml

[14] http://ti-a.org/

NOTE Sometimes it is not possible or practical to create SIPs which meet all the requirements. For instance, a SIP MAY contain the same resource both in the original (non-archivable) and archivable formats. In such case, METS encoding SHOULD indicate that the original file is not validated during ingest. Omission of mandatory metadata element(s) SHOULD be agreed upon between the producer and the OAIS archive in advance, in order to avoid ingest failures.

6 Packaging metadata

This chapter covers mainly metadata about the SIP (container) which is usually submitted using METS elements and attributes.

NOTE It is not possible to make a clear division between descriptive and administrative metadata. For instance package creator information is normally just administrative metadata. But if the package creator has modified the EPUB publication to make sure that SIP meets the requirements of the submission agreement, the creator may have performed tasks which normally belong to the editor of the publication. The name of the editor is regarded as descriptive metadata.

6.1 Package creator / submitter information

Both the name of the original creator of the package and the name of the submitting organization SHALL be included in the METS header, if the submitting organization has made any changes to the package. If the submitting organization has not modified the content, the creator name is sufficient.

If a secure transmission channel is used and it allows identification of the submitting organization, submitter information MAY be omitted.

Creator / submitter identifier SHOULD be included, if the name alone does not uniquely identify the organization. The identifier SHOULD be an ISNI or another standard identifier. The identifier system in use SHALL be indicated.

Examples

SIP creator:

```
<mets:metsHdr CREATEDATE="2017-07-15T12:00:00" RECORDSTATUS=”NEW”>
  <mets:agent ROLE=”CREATOR” TYPE=”ORGANIZATION”>
    <mets:name> National library of Finland </mets:name>
[...]
</mets:metsHdr>

SIP submitter:

<mets:metsHdr CREATEDATE="2018-02-11T08:00:00" RECORDSTATUS=”NEW”>
  <mets:agent ROLE=”PRESERVATION” TYPE=”ORGANIZATION”>
    <mets:name> Kansalliskirjasto </mets:name>
    <mets:note> ISNI 0000 0001 2033 7602 </mets:note>
 [...]
</mets:metsHdr>
```

6.2 Package status

The METS header RECORDSTATUS attribute value “REPLACEMENT” SHOULD be used to indicate the status of the package if the package is resubmitted. If the attribute is not present, its value is assumed to be “NEW”.

Example

Modified SIP to replace one sent earlier:

```
<mets:metsHdr CREATEDATE="2018-01-10T17:12:55" RECORDSTATUS=”REPLACEMENT”>
 [...]
</mets:metsHdr>
```

6.3 Package identifier

Every SIP SHALL have a unique identifier. The submission agreement SHALL specify the identifier type or types used (for instance, UUID).

NOTE In practice, some producers MAY prefer to use alternative methods, such as time stamp added to the file name. Such arrangements SHALL be specified in the submission agreements.

SIPs themselves are not preserved after the ingest process is finished, but the SIP identifier MAY be preserved both in the repository system and in producer's production systems, if there is a possibility the SIP identifier could be needed later on.

There are two encoding options, the first one of which is mandatory:

1) An identifier SHALL be located in the root element of the METS document using the OBJID attribute, which identifies the METS object as a whole.
2) A SIP identifier MAY also be expressed in a PREMIS metadata record, if it is intended as a persistent identifier.

If a private identifier system is used, the name of the creator of the package (if the creator is not the producer) MAY be part of the identifier. This makes it possible to identify the creator, and the OAIS archive is able to contact that organization directly – instead of the producer – if there are technical problems during the ingestion process.

NOTE Elements within the METS document may be identified using ID attribute, which uses the XML ID data type for identifiers. Therefore the first character of the ID attribute value must be a letter. OBJID attribute uses data type string and has no restrictions on the first character.

Examples

Package identifier in the root of a METS document:

```
<mets:mets OBJID="urn:uuid:A1B0D67E-2E81-4DF5-9E67-A64CBE366809"
xsi:schemaLocation="http://www.loc.gov/METS/ http://www.loc.gov/standards/mets/mets.xsd">
[...]
</mets:mets>
```

Publication identifier used as a package identifier in a Dublin Core record embedded in an EPUB publication:

```
<dc:identifier id="pub-id">urn:uuid:A1B0D67E-2E81-4DF5-9E67-A64CBE366809</dc:identifier>
    <meta refines="#pub-id" property="identifier-type" scheme="xsd:string">uuid</meta>
```

Publication identifier SHALL NOT be used as package identifiers. A SIP can contain multiple EPUB publications; one EPUB publication can be submitted in multiple SIPs and even if a SIP contains just one publication it may be necessary to re-send the SIP with other package identifier.

Example

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

```
    <dc:identifier id="pub-id">urn:uuid:A1B0D67E-2E81-4DF5-9E67-
A64CBE366809</dc:identifier>
    <meta property="dcterms:modified">2011-01-01T12:00:00Z</meta>
    …
</metadata>

results in the Package ID:

urn:uuid:A1B0D67E-2E81-4DF5-9E67-A64CBE366809@2011-01-01T12:00:00Z
```

6.4 Work and publication identifiers

According to the EPUB specification, each EPUB publication SHALL have a (globally) unique identifier. However, revised publications do not need to have a new standard identifier if only minor changes have been made, such as metadata updates or errata fixes. In such cases, usage of release identifiers (which consist of e.g. ISBN and the publication date) is recommended in the EPUB specification, but not mandatory. This approach is similar to the one in the ISBN standard.

In order to facilitate long-term preservation, each rendition and version of an EPUB publication submitted to an OAIS archive SHALL have an identifier, and the submission agreement or other guidelines SHALL specify the identifier systems allowed. If the archive's repository system cannot process EPUB release identifiers (for instance because the system assumes each e-book has its own ISBN or other standard identifier), release identifiers assigned by the publisher SHOULD be replaced with identifiers the OAIS archive is able to use during pre-ingest by the producer.

Identifiers belonging to the different manifestations of a work SHOULD be included in the metadata records describing these manifestations. In addition, a work identifier MAY be used in order to facilitate interlinking of manifestations of a work.

NOTE ISBN is universally used for identification of books, but there is no widely used identifier system for textual works. ISTC has not been successful, and following the closure of the ISTC International Centre the future of the identifier is uncertain.

Component parts of EPUB publications SHALL have separate identifiers if they are submitted as independent publications. For instance, if each chapter of an e-book is submitted as a separate EPUB publication, they SHALL have their own identifiers even if all chapters (EPUB publications) are sent in the same SIP.

Producers MAY provide identifiers to fragments within publications, such as paragraphs or sentences within a text. If such identifiers or other methods are used to provide links between for instance a text and an audio version of the text within an EPUB 3 document, the OAIS archive SHALL maintain the links even when the text or audio file is migrated into a new format, if submission agreement requires that such functionality is retained and if the new audio format allows such linking.

The submission agreement SHALL specify the encoding of publication identifiers. There are at least two encoding options, of which one SHALL be selected in the agreement:

1) An identifier is included in the Dublin Core metadata embedded in the EPUB publication, as specified in the EPUB publications specification. The EPUB specification requires that the identifier of a publication is provided in the Dublin Core element Identifier. The EPUB META element MAY be used to indicate the identifier type, using an authorized list such as ONIX Code List 5[15].
2) Identifier is expressed using the <premis:objectIdentifier> element. If so, PREMIS encoding SHALL specify the identifier type.

[15] http://www.stison.com/onix/codelists/onix-codelist-5.htm

NOTE 1 Digital preservation systems may be unable to handle EPUB release identifiers. For instance, their duplicate-check algorithms may expect standard identifiers such as ISBNs for books. It is possible to build a digital preservation system capable of using EPUB release identifiers, but as of this writing no such systems exist in the library sector.

NOTE 2 If the repository system cannot process EPUB release identifiers and ISBN or other standard identifiers cannot be applied, it is possible to use custom made identifier systems to identify renditions. The submission agreement SHALL specify such systems. They SHOULD be used with caution, since if and when archived data is transferred to a new OAIS archive, non-standard identifier systems can become a problem.

NOTE 3 Producers should have guidelines on how to use identifiers. For instance, if ISBN cannot be used to identify all submitted books, alternative solutions should be clearly specified.

Examples

Identifiers in a Dublin Core record (including an identifier for the resource itself and its source):

```
<dc:identifier id="isbn-id">urn:isbn:9780101010101</dc:identifier>
    <meta refines="#isbn-id" property="identifier-type" scheme="onix:codelist5">15</meta>

    <dc:source id="src-id">urn:isbn:9780375704024</dc:source>
    <meta refines="#src-id" property="identifier-type" scheme="onix:codelist5">22</meta>
```

NOTE There may be 1-n sources.

Identifier in PREMIS record:

```
<premis:objectIdentifier>
  <premis:objectIdentifierType>urn</premis:objectIdentifierType>
  <premis:objectIdentifierValue>
    URN:ISBN:978-952-222-272-5
  </premis:objectIdentifierValue>
</premis:objectIdentifier>

<mets:fileSec>
  <mets:file ID="filee01" OWNERID="URN:ISBN:978-952-222-272-5" ...>
    [...]
  </mets:file>
</mets:fileSec>
```

Release identifier (unique identifier and the last modification date):

```
<metadata xmlns:dc="http://purl.org/dc/elements/1.1/">
    <dc:identifier id="pub-id">urn:isbn:951-0-18434-9</dc:identifier>
    <meta property="dcterms:modified">2016-05-03T12:00:00Z</meta>
    …
</metadata>

Release identifier is 951-0-18434-9@2016-05-03T12:00:00Z
```

If an outdated EPUB publication is migrated during ingest to a more modern EPUB version or another e-book format, the OAIS archive SHALL acquire a new identifier for the migrated publication. The identifier type SHOULD not change; meaning that if the original e-book had an ISBN, the migrated one in another format should receive an ISBN too (ISBN system requires that each manifestation of a book has its own identifier). The archive MAY either request the new ISBN from the publisher, or assign its own identifier, depending on the agreement made with the publisher / producer.

The metadata in an AIP SHALL contain both identifiers, even if the AIP only contained the migrated document.

6.5 Core media type resource identifiers

Identifiers for core media type resources within an EPUB publication SHOULD be unique within the publication, and persist as long as the publication. These identifiers do not have to be globally unique or based on international standards. These identifiers SHALL be included in the manifest file of the EPUB publication as specified by the EPUB publications specification.

Best practice for digital preservation is to have all information (documents and metadata) in standardized and widely used formats. If there is a core media type resource which has been specified as non-archivable (for instance, a GIF image) in the submission agreement, it SHALL be migrated during ingest, and the manifest file of the migrated publication SHALL be updated accordingly. The metadata in the manifest within the EPUB in the AIP SHOULD contain identifiers for both the original and migrated resource even if the AIP only contained the latter.

The EPUB remote resources property SHOULD NOT be allowed for core media type resources in submission agreements because retrieval of these resources can fail during ingest. This would mean the failure of the entire ingest process, because the archived EPUB publication would be incomplete.

NOTE Modern repository systems allow ingest and storage of non-archivable file formats.

Example

Core media type resource identifiers in an EPUB manifest file:

```
<manifest>
    <item id="nav"
          href="nav.xhtml"
          properties="nav"
          media-type="application/xhtml+xml"/>
    <item id="intro"
          href="intro.xhtml"
          media-type="application/xhtml+xml"/>
    <item id="c1"
          href="chap1.xhtml"
          media-type="application/xhtml+xml"/>
    <item id="c1-answerkey"
          href="chap1-answerkey.xhtml"
          media-type="application/xhtml+xml"/>
    <item id="c2"
          href="chap2.xhtml"
          media-type="application/xhtml+xml"/>
    <item id="c2-answerkey"
          href="chap2-answerkey.xhtml"
          media-type="application/xhtml+xml"/>
    <item id="c3"
          href="chap3.xhtml"
          media-type="application/xhtml+xml"/>
    <item id="c3-answerkey"
          href="chap3-answerkey.xhtml"
          media-type="application/xhtml+xml"/>
    <item id="notes"
          href="notes.xhtml"
          media-type="application/xhtml+xml"/>
    <item id="cover"
          href="./images/cover.svg"
          properties="cover-image"
          media-type="image/svg+xml"/>
    <item id="f1"
          href="./images/fig1.jpg"
          media-type="image/jpeg"/>
    <item id="gif-f1"
          href="./images/fig1.gif"
          media-type="image/gif"/>
    <item id="css"
```

```
          href="./style/book.css"
          media-type="text/css"/>
    <item id="pls"
          href="./speech/dict.pls"
          media-type="application/pls+xml"/>
</manifest>
```

NOTE These identifiers are only valid within a single EPUB publication and SHALL only be used in that context. Therefore there is no requirement for global uniqueness.

6.6 Foreign resource identifiers

Identifiers for embedded foreign resources SHOULD be unique within the publication, but there is no requirement for global uniqueness. These identifiers SHALL be included in the manifest file of the EPUB publication as specified by the EPUB publications specification.

If there is a foreign resource which has been specified as non-archivable in the submission agreement, it SHALL be migrated during ingest, and the manifest file of the migrated publication SHALL be updated accordingly. The metadata in the manifest within the EPUB in the AIP SHOULD contain identifiers for both the original and migrated resource even if the AIP only contained the latter.

The EPUB remote resources property SHOULD NOT be allowed in submission agreements because retrieval of these resources can fail during ingest. This would mean the failure of the entire ingest process, because the archived EPUB publication would be incomplete.

Example

Foreign resource identifiers in an EPUB manifest file. Note that it is obligatory to specify the media type of these resources:

```
<manifest>
    <item id="item1"
          href="chap1_docbook.xml"
          media-type="application/docbook+xml"
          fallback="fall1"/>
    <item id="fall1"
          href="chap1.xml"
          media-type="application/z3986-auth+xml"
          fallback="fall2" />
    <item id="fall2"
          href="chap1.xhtml"
          media-type="application/xhtml+xml"/>
    …
</manifest>
```

NOTE 1 The fallback chain should terminate with a core media type (xhtml).

NOTE 2 The fallback mechanism is relevant for the preservation of EPUB resources in the long run. When an EPUB publication is preserved, the old and the new representation can be linked via the fallback chain. Those users who are still able to render the original publication can still use that, while others can use the latest version.

6.7 Identifiers for metadata records

A metadata record can be for instance a Dublin Core record or a PREMIS record embedded or linked to a SIP.

There SHALL be a unique and persistent identifier for each metadata record in a SIP. If possible, the identifier SHOULD be embedded in the identified record, using an appropriate metadata element (e.g. record identifier). This approach is not applicable for Dublin Core metadata records, since the format does not have a metadata identifier element. Private Dublin Core extensions that allow encoding of record identifiers SHOULD NOT be used.

There are at least three encoding options for metadata records, one of which SHALL be selected:

- Metadata records are embedded in a METS document within SIP using METS mdWrap elements.
- Metadata records are embedded in a SIP, with mdRef links from the METS file.
- Metadata records are external, linked to SIP using METS mdRef element.

If metadata is external, the repository system SHALL be able to retrieve the metadata records during ingest.

NOTE E-ARK common specification[3] requires embedded metadata in SIP, but not in METS file. This approach was chosen since compared with METS option it is more flexible, easier for producers and diminishes the risk of the METS file becoming too large to manage and use, especially if SIPs may contain several publications.

The metadata wrapper element <mdWrap> provides a wrapper around metadata within a METS document. Such metadata can be in one of two forms:

1) XML encoded metadata, where the XML encoding is identified as belonging to a namespace other than the METS document namespace.
2) Any arbitrary binary or textual form[16].

The metadata reference element <mdRef> element is a generic element used throughout the METS schema to provide an indicator to metadata residing outside the METS document. The location of the metadata SHALL be recorded in the xlink:href attribute[17].

Many metadata formats support metadata record identifiers such as LCCN (Library of Congress Card Number). If a metadata format is migrated during ingest, these identifiers SHALL be encoded so that there is no risk of mixing the publication identifiers and the metadata identifiers with one another.

In a METS document this is easy since both the entire administrative metadata section (<amdSec>) and all its parts (technical metadata, <techMD>; intellectual property rights metadata, <rightsMD>; source metadata, <sourceMD>; and digital provenance metadata, digiprovMD>) can have identifiers of their own.

If a PREMIS LINK is used to associate a metadata record with the rendition of an EPUB publication, the following syntax MAY be used:

```
<package … prefix="premis: http://www.loc.gov/standards/premis/v3/index.html">

    <metadata>
        …
        <link rel="textMD-record" href="http://example.org/textmd/12389347"/>
        …
    </metadata>
    …
</package>
```

Metadata record identifiers SHALL be used whenever there is a possibility that either the producer or the OAIS archive updates descriptive or administrative metadata during the ingest process or long-term preservation.

[16] http://www.loc.gov/standards/mets/docs/mets.v1-9.html#mdWrap
[17] http://www.loc.gov/standards/mets/docs/mets.v1-9.html#mdRef

When an OAIS archive creates a new representation of an EPUB publication via migration (for instance, from EPUB 2 to EPUB 3.0.1), there are two representations of the same intellectual object, which means these representations SHALL have different ISBNs[18].

Since migration can be a complex process, the producer and the archive SHALL specify either in the submission agreement or elsewhere how to manage the migrations. They are complex processes since not only is the document itself modified; metadata changes as well. Each rendition of the document shall have its own technical metadata. Descriptive metadata will remain the same except for the changes made in the technical metadata elements. The access rights metadata should not change at all, since any changes in the copyright or licensing are likely to have the same impact on both representations. The preservation metadata record will be enriched with the migration event information and information about the agents (human and software) related to it. The updated metadata record applies only to the latest representation of the EPUB publication.

The descriptive metadata record of a migrated document SHALL include the identifier of the original publication in an appropriate metadata element such as the Dublin Core element Source in order to enable linking between different manifestations of the resource, both in the repository system (in case these manifestations are in different AIPs), and in the producer's information systems. Producers can delete the original version of the document from the production systems and only keep the migrated version, because if needed the original can be retrieved from the repository system as a DIP, except if the repository system deletes the original version too. The best practice – that SHOULD be documented in the submission agreement - is to keep all of the versions of the resource in the OAIS archive if file size is not an issue to the capacity of the repository system.

Examples

Identifier for a preservation metadata record in a PREMIS format:

```
<mets:digiprovMD ID=”file2345AMDdprov01M”>
   mets:mdWrap MIMETYPE=”text/xml” MDType=”PREMIS” LABEL=”PREMIS preservation metadata”>
      <mets:xmlData>
         ….
     </mets:xmlData>
   </mets:mdWrap>
```

Identifier for a technical metadata record in a TextMD format embedded within a METS document:

```
<mets:techMD ID=”AMDTech01M”>
   mets:mdWrap MIMETYPE=”text/xml” MDType=”TextMD” LABEL=”Technical Metadata for Text”>
      <mets:xmlData>
         ….
     </mets:xmlData>
   </mets:mdWrap>
```

Link to an external ONIX metadata record from an EPUB publication:

```
<link rel="onix-record" href="http://example.org/meta/records/onix/121099"/>
```

6.8 Dates

There are many dates that may be relevant for EPUB publications in general. For instance, ONIX codelist issue 38 has 18 codes just for publishing dates (list 163, publishing date role)[19] including publication

[18] The process of acquiring new standard identifiers, such as ISBN is usually specified in standards and user guides. Archives have to follow the appropriate procedures when obtaining identifiers for migrated documents, or request new identifiers from the producer.

[19] http://www.editeur.org/files/ONIX%20for%20books%20-%20code%20lists/ONIX_BookProduct_Codelists_Issue_38.html

date, public announcement date, date of first publication, last reprint date and so on. From a digital preservation point of view, publishing date is important but there are also other important dates, including SIP creation and update dates, which SHALL be expressed in a machine understandable format and encoded in such a manner that there is no risk of confusion with other date information.

In order to guarantee machine understandability, all dates and times SHALL be expressed using ISO 8601[20]. The date or time given SHOULD be as accurate as possible and the time zone SHOULD be provided if needed (e.g. when the producer and the OAIS archive are on different time zones).

6.8.1 Creation date of a Submission Information Package

The SIP creation date SHALL be present in the metadata. The date SHALL be provided in the package header (<mets:metsHdr>) using the CREATEDATE attribute.

Example

```
<mets:metsHdr CREATEDATE="2011-02-15T15:41:12">
[...]
</mets:metsHdr>
```

6.8.2 Modification date of a Submission Information Package

If there are severe problems in the SIP, such as missing mandatory metadata or unknown file formats, ingest will usually fail. When the revised SIP is re-submitted to the repository system, the last modification date SHOULD be provided using the LASTMODDATE attribute alongside the original CREATEDATE attribute. In the initial package the RECORDSTATUS attribute is NEW, but the status of the resubmitted package SHOULD be MODIFIED. REPLACEMENT SHOULD be used only when EPUB publication that has already been archived is replaced by a more modern edition. MODIFIED explains why LASTMODDATE is used.

The LASTMODDATE attribute MAY also be included if the SIP has not been submitted before, but the package has been under construction for a long time (at least several days).

NOTE Publishers have built robust systems to deliver content for end users, but they might not be able to service third parties equally well. If the publisher is not able to meet the requirements, producer SHOULD fill in the gaps.

Example

SIP creation date and modification date:

```
<mets:metsHdr CREATEDATE="2011-02-15T15:41:12" LASTMODDATE="2016-02-29T10:54:30+02:00">
[...]
</mets:metsHdr>
```

6.8.3 Creation/modification date of an EPUB publication

According to [Kasdorf],

> *dc:date element SHALL be used to provide the date of the EPUB publication (not the publication date of a source publication, such as the print book from which the EPUB has been derived.*

Publication date SHALL be provided in the ISO 8601 format:

[20] In October 2018 the latest version of the standard was ISO 8601:2004. It will be replaced by a two-part standard which has reached FDIS stage. ISO 8601-1 will cover the same features than the current standard. Although these features will be sufficient in most situations, sometimes the extensions specified in 8601-2 may be needed.

```
YYYY-MM-DDThh:mm:ssZ
```

The precision of the date information varies; often just publication year is needed.

The last modification date of each rendition in an EPUB container is also a mandatory metadata element.

If there are two or more EPUB publications, or two or more renditions of an EPUB publication in one SIP, the dates SHALL be provided separately for each EPUB publication and rendition.

The last modification date is normally the publication date. But if it is necessary to specify both the publication date and last modification date because the producer has modified the publication to meet the ingest requirements, the last modification date SHOULD be provided as a PREMIS dateCreatedByApplication element within the publication's PREMIS metadata record.

If one or more of the underlying core media files are migrated during ingest, the archived EPUB publication SHOULD get a new last modification date as described above.

Examples

EPUB publication date in Dublin Core:

```
<dc:date>2016-01</dc:date>
```

EPUB publication modification date in PREMIS:

```
<premis:objectCharacteristics>
[...]
  <premis:creatingApplication>
    <premis:dateCreatedByApplication>2016-02-15T15:43:03
    </premis:dateCreatedByApplication>
  </premis:creatingApplication>
</premis:objectCharacteristics>
```

If the publication date of the source document is included, it SHALL be encoded in such a way that it is not confused with the EPUB publication date.

Example

Source document publication date in Dublin Core (encoding is just an example of what may be done; the actual encoding used may vary):

```
<dc:date.source>1923</dc:date.source>
```

6.8.4 Creation/modification of a metadata record

The date and time when a metadata record embedded in a SIP was created or last modified SHOULD be provided using a CREATED attribute in the appropriate METS metadata section (techMD etc.). The attribute requires precision down to a second; if the data is not accurate enough, the date can be padded with zeros if required. If even the specific date is unknown, the first of January (01-01) can be used instead.

The METS CREATED attribute SHOULD also be used when a metadata record has been modified.

Some metadata formats allow expression of creation and modification dates in the metadata record itself. For instance, a MARC record contains the date the record was created in a fixed length field 008, positions 00-05, format YYMMDD. This information is never changed. Date and time of the last transaction (the time the record was last modified) is stored in the field 005, in a format yyyymmddhhmmss.ff, where the ff represents the decimal fractions of a second.

Producers and OAIS archives MAY agree to use these metadata record creation and modification dates.

Example

Metadata record creation/modification date:

```
<mets:dmdSec ID="dmd-dc" CREATED="2015-02-15T00:00:00">
[...]
</mets:dmdSec>
```

6.9 Metadata format and its versions

This section is based on Bill Kasdorf's EPUB 3 Packaging and Metadata, which provides guidelines for supplying metadata in an EPUB 3.x publication.

<mdref> attribute MDTYPE allows indication of the format of the referenced metadata. The list of valid values of this attribute include MARC, EAD, DC (Dublin Core) and OTHER. The version of the format MAY be expressed using an MDTYPEVERSION attribute.

All EPUB 3 versions use the Dublin Core Metadata Element Set for much of its required and optional metadata. There are three mandatory (Dublin Core) metadata elements (title, identifier, language), which SHALL be embedded in all current versions of EPUB 3 publications.

The terms META, LINK, ITEM, and ITEMREF MAY be used to describe properties of key elements. Dublin Core metadata in an EPUB can be either simple or qualified; since the syntax for qualification is EPUB specific, it is possible that applications parsing the embedded Dublin Core records will "dumb" them down.

Example

```
<metadata xmlns:dc="http://purl.org/dc/elements/1.1/">
[…]
dc:identifier id="pub-identifier">urn:isbn:9781449325299</dc:identifier>
<dc:title id="pub-title">EPUB 3 Best Practices</dc:title>
<dc:title id="t2">First Edition</dc:title>
   <meta refines="#t2" property="title-type">edition</meta>
<dc:language id="pub-language">en</dc:language>
</metadata>
```

This metadata SHOULD be copied into a METS document, because some long-term preservation applications may not be able to retrieve this metadata from an EPUB publication.

In a METS container, descriptive metadata (in the Dublin Core format) SHALL be expressed in a <mets:dmdSec> element using MDTYPE value "DC".

Example

Original metadata record in an EPUB container:

```
<metadata xmlns:dc="http://purl.org/dc/elements/1.1/">
        <dc:identifier id="pub-id">urn:uuid:A1B0D67E-2E81-4DF5-9E67-
A64CBE366809</dc:identifier>
        <dc:title>Norwegian Wood</dc:title>
        <dc:language>en</dc:language>
        <meta property="dcterms:modified">2011-01-01T12:00:00Z</meta>
    </metadata>
```

Simple Dublin Core metadata record embedded in a METS document. Note that the modification date of the record is included, encoded with an EPUB META term:

```
<mets:dmdSec ID="dmd-dc" CREATED="2016-05-03T14:00:00">
  <mets:mdWrap MIMETYPE="text/xml"
        MDTYPE="DC"
        MDTYPEVERSION="1.1"
        LABEL="Bibliographic metadata">
    <mets:xmlData>
      <dc:record>
        <dc:identifier id="pub-id">urn:uuid:A1B0D67E-2E81-4DF5-9E67-
A64CBE366809</dc:identifier>
        <dc:title>Norwegian Wood</dc:title>
        <dc:language>en</dc:language>
        <meta property="dcterms:modified">2011-01-01T12:00:00Z</meta>
      </dc:record>
    </mets:xmlData>
  </mets:mdWrap>
```

An EPUB publication can contain one or more links to external metadata records which describe the resource. If these records are not just alternative representations of embedded metadata (that is, if there is no descriptive metadata in the publication or if the embedded metadata is abridged), linked metadata record(s) SHOULD be retrieved and embedded into the SIP as part of the pre-ingest process. Another option is for the OAIS archive to retrieve the linked metadata during ingest to ensure the AIP is complete. The submission agreement SHALL specify if external metadata is allowed.

The process described above applies to all kinds of metadata. If essential metadata, be it descriptive or administrative, is not a part of the SIP but just linked to it, this metadata SHALL be retrieved either during a pre-ingest by the producer or during ingest by the OAIS archive.

7 Administrative metadata

This document is incomplete since it does not cover the administrative metadata elements needed to preserve EPUB core media type resources[21]. The technical metadata required for the preservation has to be media type specific, and covering the mandatory metadata elements needed for text, still images, and audio files in a single document is not feasible. The metadata elements listed here are media type neutral and are always needed in long-term preservation, no matter what the media type or file format is.

The preservation metadata data dictionary PREMIS[22] is used for preservation metadata. The Library of Congress has published guidelines for using PREMIS with METS[23] with the intention to "suggest common practices for encoding METS documents with PREMIS metadata for exchange purposes" [Guidelines]. You can find an example of a METS document using the profile at http://www.loc.gov/standards/premis/louis-2-0.xml.

OAIS archives follow their own practices when they create AIPs, but common guidelines SHOULD be applied for SIPs and DIPs in order to guarantee interoperability. It is the responsibility of both producers and OAIS archives to apply recommended practices whenever possible. If preservation metadata is included into the SIP, then for interoperability reasons it SHOULD follow common guidelines.

Preservation metadata requirements depend a lot on the complexity of the EPUB publications to be archived. An EPUB publication containing just text is easier to deal with than one containing for instance MPEG audio linked into the text. Reflowable EPUB publications are probably easier to preserve

[21] https://idpf.github.io/epub-cmt/v3/
[22] https://www.loc.gov/standards/premis/
[23] https://www.loc.gov/standards/premis/guidelines-premismets.pdf

in the long-term than fixed layout EPUB publications, since preserving the original look and feel is more challenging than preserving just the intellectual content. If the layout of an EPUB publication has an impact on its meaning, emulation is likely to be the best preservation method, and submission information package should contain metadata supporting it, such as the name or names of appropriate rendering applications.

Submission agreements covering EPUB publications SHALL list not only ingestible media types (file formats), but also the EPUB properties that MAY be used. These lists are not static; they SHALL be maintained in cooperation with the producer and the OAIS archive, since:

- EPUB core media type list is updated frequently, and the impact of changes (new media types, deprecated old ones) has to be checked.
- EPUB core media type list only covers file formats (such as image/jpeg), not their different versions. If a new version (like TIFF 7.0) is introduced, it is necessary to decide if it is ingestible or archivable and can therefore be submitted.
- An EPUB core media type may become non-archivable, or vice versa – a core media type previously regarded as non-archivable may become “acceptable”.
- When new versions of the EPUB core specification and related documents are published, it is important to check the impact they have on long-term preservation.

7.1 Technical metadata

7.1.1 File formats and their versions

EPUB version used SHALL be specified in the <package> element of the EPUB publication’s content.opf file.

NOTE ePubCheck does not provide accurate EPUB versioning beyond the major number.

Example

```
<package xmlns="http://www.idpf.org/2007/opf" unique-identifier="ean" version="3.0.1">
. . .
<package>
```

File formats present in an EPUB container embedded in a SIP, including both core media types and foreign resources, SHALL be indicated using PREMIS <premis:formatName> element. From the PREMIS encoding point of view, it makes no difference whether the resource is a core media type or not.

Versions of file formats SHALL be expressed in the <premis:formatVersion> element if they are known. Reference to a file format registry such as PRONOM[24] MAY be added if it is necessary to provide access to the full details of the file format.

File formats (but not the version) MAY also be expressed in METS using the MIMETYPE attribute of the <mets:file> element. The attribute requires the use of IANA MIME[25] media types.

The composition level SHALL be encoded using PREMIS compositionLevel element. Its value SHALL be 1 if the EPUB publication is considered to be a container. However, if the EPUB publication is seen as a file only, composition level 0 MAY be used.

NOTE Any application capable of rendering EPUB 3.x publications SHOULD be able to deal with all core media type resources. Whether foreign resources, included with a fallback

[24] https://www.nationalarchives.gov.uk/PRONOM/Default.aspx

[25] https://www.iana.org/assignments/media-types/media-types.xhtml

mechanism, will also be preserved beyond bit level, depends on what the producer and the OAIS archive have agreed on. The submission agreement SHOULD specify all ingestible and archivable file formats a publisher will submit. If other file formats are included in SIPs, they SHALL be encoded so that they are not validated during ingest (otherwise the ingest process will fail) and the OAIS archive's preservation responsibility is limited to bit level.

If core media type resources and other resources are preserved via migration, migrating even a single file means the entire EPUB 3.x publication SHALL be updated and a new AIP created, with updated descriptive and administrative metadata. If emulation is the chosen preservation method, the EPUB publication itself is not modified, but each time hardware or software environment changes, preservation metadata changes and a new AIP SHALL be formed. Such metadata (EPUB specific things that must be recorded to facilitate emulation) and its modifications are beyond the scope of this document.

Examples

The EPUB version:

```
<mets:amdSec>
  <mets:techMD ID="fileepub01-techmd" CREATED="2015-05-31T09:54:43">
    <mets:mdWrap MDTYPE="PREMIS:OBJECT">
      <mets:xmlData>
        <premis:object xsi:type="premis:file">
          <premis:objectIdentifier>
            [...]
          </premis:objectIdentifier>
          <premis:objectCharacteristics>
          <premis:compositionLevel>1</premis:compositionLevel>
          <premis:format>
            <premis:formatDesignation>
              <premis:formatName>application/epub+zip</premis:formatName>
              <premis:formatVersion>3.1</premis:formatVersion>
            </premis:formatDesignation>
          </premis:format>

          [...]
         </premis:objectCharacteristics>
       </premis:object>
     </mets:xmlData>
    </mets:mdWrap>
  </mets:techMD>
</mets:amdSec>

<mets:fileGrp>
  <mets:file ID="fileepub01" ADMID="fileepub01-techmd">
    [...]
  </mets:file>
</mets:fileGrp>
```

An EPUB Core media type resource:

```
<mets:amdSec>
  <mets:techMD ID="filee01-techmd" CREATED="2015-04-30T019:22:43">
    <mets:mdWrap MDTYPE="PREMIS:OBJECT">
      <mets:xmlData>
        <premis:object xsi:type="premis:file">
          <premis:objectIdentifier>
            [...]
          <premis:objectIdentifier>
          <premis:objectCharacteristics>
          <premis:compositionLevel>0</premis:compositionLevel>
          <premis:format>
            <premis:formatDesignation>
```

```
              <premis:formatName>image/png</premis:formatName>
              <premis:formatVersion>1.2</premis:formatVersion>
            </premis:formatDesignation>
          </premis:format>
          [...]
         </premis:objectCharacteristics>
       </premis:object>
     </mets:xmlData>
    </mets:mdWrap>
  </mets:techMD>
</mets:amdSec>

<mets:fileGrp>
  <mets:file ID="filee01" ADMID="filee01-techmd">
    [...]
  </mets:file>
</mets:fileGrp>
```

An Embedded foreign resource:

```
<mets:amdSec>
  <mets:techMD ID="filee01-techmd" CREATED="2015-06-30T015:12:00">
    <mets:mdWrap MDTYPE="PREMIS:OBJECT">
      <mets:xmlData>
        <premis:object xsi:type="premis:file">
          <premis:objectIdentifier>
            [...]
          <premis:objectIdentifier>
          <premis:objectCharacteristics>
          <premis:compositionLevel>0</premis:compositionLevel>
          <premis:format>
            <premis:formatDesignation>
              <premis:formatName>image/bmp</premis:formatName>
              <premis:formatVersion>1</premis:formatVersion>
            </premis:formatDesignation>
          </premis:format>
          [...]
         </premis:objectCharacteristics>
       </premis:object>
     </mets:xmlData>
    </mets:mdWrap>
  </mets:techMD>
</mets:amdSec>

<mets:fileGrp>
  <mets:file ID="filee01" ADMID="filee02-techmd">
    [...]
  </mets:file>
</mets:fileGrp>
```

EPUB publications in SIPs SHOULD contain resources in file formats not suitable for preservation if and only if the same resource is also included in an acceptable file format using a fallback mechanism.

If a file is migrated during pre-ingest to a format or a version suitable for preservation before submitting it to an archive, the migration SHOULD be documented in the SIP as a PREMIS EVENT. The original file MAY be included in the SIP with appropriate encoding (to bypass validation). If an archive migrates files during ingest, a PREMIS EVENT record is created. Migration related documentation SHALL be stored in AIPs regardless of who performed the migration, the producer, an archive, or a third-party.

The event encoding SHOULD contain the following metadata:

- ○ Event identifier
- ○ Timestamp: <eventDateTime>2016-04-05</eventDateTime>
- ○ Event type: <eventType>migration</eventType>

- Event outcome: <eventOutcome>success</eventOutcome>
- Link to the agent/agents: <premis:agentName>Word 2016</premis:agentName>
- Link to the PREMIS Object of the source file
  - Including the role in the event: <linkingObjectRole>source</linkingObjectRole>
- Link to the PREMIS object of the output file
  - Including the role in the event: <linkingObjectRole>outcome</linkingObjectRole>

NOTE PREMIS does not contain a standardized event vocabulary.

There MAY be several source and output files. A 1:1 relationship between them is not required.

Foreign files to be ignored during ingest SHALL be encoded using the METS <file> element with a USE attribute “no file-format-validation”. For instance:

```
<file USE=”no-file-format-validation” …>
```

The archive MAY choose to validate foreign files although there is no intention to preserve them except at bit level. If so, USE attribute “no-file-format-migration” may be used.

Example

An EPUB 2 document encoded for bit level preservation:

```
<mets:amdSec>
  <mets:techMD ID="fileepub201-techmd" CREATED="2016-05-03T09:54:43">
    <mets:mdWrap MDTYPE="PREMIS:OBJECT">
      <mets:xmlData>
        <premis:object xsi:type="premis:file">
          <premis:objectIdentifier>
            [...]
          </premis:objectIdentifier>
          <premis:objectCharacteristics>
          <premis:compositionLevel>1</premis:compositionLevel>
          <premis:format>
            <premis:formatDesignation>
              <premis:formatName>application/epub+zip</premis:formatName>
              <premis:formatVersion>2</premis:formatVersion>
            </premis:formatDesignation>
          </premis:format>

          [...]
         </premis:objectCharacteristics>
       </premis:object>
     </mets:xmlData>
    </mets:mdWrap>
  </mets:techMD>
</mets:amdSec>

<mets:fileGrp>
  <mets:file ID="fileepub201" ADMID="fileepub201-techmd" USE=”no-file-format-validation”>
    [...]
  </mets:file>
</mets:fileGrp>
```

7.1.2 Digital signatures and checksums

Archives can use digital signatures in various ways:

- For **submission** to an archive. A producer (publisher or a third party submitting the data) MAY sign an object, which enables the archive to guarantee that the submitting party is correct even if the transmission channel is not reliable.

- For **dissemination** from an archive. The archive MAY sign an object to assert that it truly is the source of the DIP.
- For **archival storage.** An archive MAY want to store signed objects so that it is possible for third-parties such as other archives or the data producer to confirm the origin and integrity of the data.

This document concentrates on the submission of objects to an archive, other uses are not discussed.

If there is no secure transmission channel, SIPs containing EPUB publications SHALL be digitally signed. A digital signature is not mandatory if there are other ways to make sure the SIP comes from the correct source.

Checksums SHOULD be calculated both for EPUB containers and their contents before the SIPs are sent to an archive in order to enable integrity checks. If the checksum is calculated to the container only, it is not possible to know which component has changed. If signatures are created immediately after the EPUB publication is created the producer can make sure that the content is not changed unintentionally before it is submitted to an archive.

Checksums SHOULD also be calculated for the core media resource files and for any foreign resources in the EPUB container.

The checksum SHALL be calculated by using an algorithm specified in the submission agreement. Recommended options include sha-224, sha-256, sha-384 and sha-512. md5 and sha-1 SHOULD NOT be used because they are no longer safe.

There are at least three different ways to embed checksums in metadata, one of which SHALL be selected in the submission agreement. Usage of METS File element is recommended, since then the METS file can be used to validate the integrity of the package.

- METS element File (<file>) has attributes CHECKSUM and CHECKSUMTYPE. The values allowed for the latter are HAVAL, MD5, SHA-256, SHA-384, SHA-512, TIGER and WHIRLPOOL. Recommended options are SHA-256, SHA-384, and SHA-512.
- A PREMIS element <premis:fixity> with attributes messageDigestAlgorithm and messageDigest. In PREMIS, running a fixity-check program on an object to detect unauthorized changes is called an EVENT.
- signatures.xml file in an EPUB container allows the encoding of signatures for EPUB publications, their renditions as a whole, or just their component parts.

Syntax example for signatures.xml file can be found in the EPUB Container Format[26].

Example

Checksum in a METS FILE element:

```
        <mets:file ID=”epi01m” CHECKSUMTYPE=”SHA-256”
        CHECKSUM=”a5d6ecbfc51f37b26778b24586dc15bfae8a0872275c39c2e19c63a5917650b5”>
        </mets:file>
```

Checksum in a PREMIS metadata record:

```
<mets:amdSec>
  <mets:techMD ID="fileepub01-techmd" CREATED=”2011-05-31T00:00:00”>
    <mets:mdWrap MDTYPE="PREMIS:OBJECT">
      <mets:xmlData>
```

[26] http://www.idpf.org/epub/301/spec/epub-ocf.html#sec-container-metainf-signatures.xml

```
        <premis:object xsi:type="premis:file">
          [...]
        <premis:objectCharacteristics>
          <premis:compositionLevel>2</premis:compositionLevel>
          <premis:fixity>
            <premis:messageDigestAlgorithm>
              SHA-256
            </premis:messageDigestAlgorithm>
            <premis:messageDigest>
              a5d6ecbfc51f37b26778b24586dc15bfae8a0872275c39c2e19c63a5917650b5
            </premis:messageDigest>
          </premis:fixity>
          [...]
         </premis:objectCharacteristics>
        </premis:object>
      </mets:xmlData>
    </mets:mdWrap>
  </mets:techMD>
</mets:amdSec>
```

7.2 Rights metadata

The copyright status of an EPUB publication SHOULD be expressed as rights metadata. If the embedded core media and foreign resources are copyrighted, their rights metadata SHOULD also be included if and when relevant. For instance, copyright owner for a foreign resource can be different than the copyright owner of the EPUB publication, even if access and use regulations were the same for all components of the publication.

If a copyrighted publication (with its component parts) is licensed for use, the rights metadata SHOULD provide basic information about the license. Details about the terms of the license MAY be provided by e.g. providing a link to the copy of the license on the web.

Any legal restrictions on the use of the document SHOULD be described in the embedded rights metadata.

If a SIP contains several renditions of an EPUB publication with different rights information, each publication SHALL have its own rights metadata record attached to the rendition to which the metadata applies.

There are at least three different methods for providing copyright status and license information. One of them SHALL be specified as mandatory in the submission agreement. The options are:

- PREMIS <rights> element.
- METS rightsMD element
- META-INF/rights.xml file as specified in the EPUB Open Container Format

Example

Rights metadata in PREMIS record.

```
<premis>
  <rights>
    <rightsStatement>
      <rightsBasis>Copyright</rightsBasis>
      <copyrightInformation>
       <copyrightStatus>Under copyright<copyrightStatus>
       <copyrightJurisdiction>fi<copyrightJurisdiction>
       <copyrightNote>Copyright expires 2022<copyrightNote>
      </copyrightInformation>
    </rightsStatement>
  </rights>
</premis>
```

This PREMIS record SHALL cover both rights related metadata and license metadata, if license covers preservation actions as well (see below).

7.2.1 Preservation related rights

Preservation related rights cover things done in the archive, from ingest to preservation to dissemination. These rights SHALL be based on the submission agreement, if copyright and licensing terms do not apply to actions done within an archive.

Submission agreements MAY restrict preservation related actions the archive personnel is entitled to carry out. These restrictions are usually not described in individual SIPs, unless the publications in the SIP require special treatment. If there is preservation related rights metadata in a SIP, it SHOULD override the regulations in the submission agreement. For instance, if the archive staff is normally allowed to carry out migrations to the publications submitted by a certain producer, the producer SHOULD be able to prevent that with appropriate preservation metadata in a SIP. Submission agreements MAY specify that the archive is not allowed to migrate documents from a certain produce. Then migrations SHALL be done by the producer or a trusted third-party, and if they are no longer capable of the task, the OAIS archive SHALL be able to do it.

For instance, a national library MAY outsource long-term preservation of its legal deposit EPUB publications, but the library MAY still want to carry out critical preservation actions such as migrations itself. The responsibility of the OAIS archive would be limited to bit level preservation of this content.

These preservation related restrictions SHALL be specified in the preservation plan. For instance, the plan may state that the OAIS archive is not allowed to migrate any EPUB publications submitted by a certain producer (for instance, the national library). If this information is not present in SIPs, the archive SHOULD add it to AIPs as preservation metadata during ingest. If the preservation plan is revised, old guidelines are deprecated and the OAIS archive SHALL update the rights metadata in the relevant AIPs.

In information packages, restrictions for preservation actions SHALL be expressed using the PREMIS Rights metadata format and encoding in the METS <rightsMD> element. PREMIS rights metadata MAY also be included in EPUB publications.

A controlled vocabulary maintained by The Library of Congress available at http://id.loc.gov/vocabulary/preservation/actionsGranted.html SHALL be used to describe the preservation actions (as of 2016-07-21, these are delete, disseminate, migrate, modify, replicate, and use).

Example

Premis rights metadata record included in a METS file:

```
<amdSec ID="rights">
  <rightsMD ID="preservation-rights1">
    <mdWrap MIMETYPE="text/xml" MDTYPE="PREMIS" LABEL="PREMIS Rights Schema">
      <premis>
        <rights>
       <rightsStatement>
            <rightsBasis>Submission agreement</rightsBasis>
            <rightsGranted>
       <act>Disseminate</act>
       <restriction>Disallow</restriction>
       <termOfGrant>
             <startDate>2016-08-01</startDate>
```

```
                <endDate>open</endDate>
          <termOfGrant>
          <act>Modify</act>
          <restriction>Disallow</restriction>
          <act>Migrate</act>
          <restriction>Disallow</restriction>
              </rightsGranted>
            </rightsStatement>
          </rights>
        </premis>
      </mdWrap>
    </rightsMD>
  </amdSec>
```

7.3 Structural metadata

In this chapter it is necessary to discuss:

- internal structure of EPUB publication(s), and
- structure of the SIP, which may contain 0-n EPUB publications

EPUB Open Container Format (OCF) SHALL be used to describe the structure of EPUB publication, as specified in http://www.idpf.org/epub/31/spec/epub-ocf.html.

METS structMap SHALL be used to describe the structure of a SIP. If the SIP contains several EPUB publications, structMap SHALL specify them and the order in which they are to be presented.

Structural metadata in OCF and METS structMAP is complementary; the former does not need to be aware of the SIP, and the latter does not need to describe the internal structure of the EPUB publication. But if the same structural metadata is provided in both formats, it SHOULD NOT be contradictory.

The internal structure of an EPUB publication SHALL, according to the EPUB standard, be specified in an EPUB Navigation Document in both human and machine readable format[27]. This information does not need to be replicated in the METS document.

An OAIS archive capable of ingesting EPUB 3.x documents SHALL have an EPUB validator in order to make sure that the ingested publications are well formed. The archive SHOULD also have the reader application/applications a producer recommends for rendering the EPUB publications it has submitted, in order to be able to check when necessary that it is possible to render the ingested publications correctly. However, some OAIS archives MAY just ingest the EPUB publications and leave it to the users to find an appropriate EPUB reader or readers.

Example

A SIP containing 6 versions of the same EPUB publication, arranged hierarchically in different folders according to the nature of the versions.

Example has been adapted from Rutgers university's METS structural map guidelines document[28]:

```
<structMap TYPE="logical">
<div ID="div1" LABEL="EPUB-SIP" ORDER="1" TYPE="folder">
<fptr FILEID="FILE001" CONTENTIDS="ID1"/>
<div ID="div1.1" LABEL="Folder A" ORDER="1" TYPE="folder">
<fptr FILEID="FILE002" CONTENTIDS="IDH2"/>
<fptr FILEID="FILE003" CONTENTIDS="ID3"/>
<div ID="div1.1.1" LABEL="Folder A.1" ORDER="1" TYPE="folder">
```

[27] http://www.idpf.org/epub/301/spec/epub-contentdocs.html#sec-xhtml-nav

[28] https://rucore.libraries.rutgers.edu/collab/ref/spc_sawg_r7_0_file_hierarchy.pdf

```
<fptr FILEID="FILE004" CONTENTIDS="ID4"/>
</div>
</div>
<div ID="div1.2" LABEL="Folder B" ORDER="2" TYPE="folder">
<fptr FILEID="FILE005" CONTENTIDS="ID5"/>
</div>
<div ID="div1.3" LABEL="Folder C" ORDER="3" TYPE="folder">
<fptr FILEID="FILE006" CONTENTIDS="ID6"/>
</div>
</div>
</structMap>
```

Within an EPUB container the HTML 5 nav element provides structural information:

```
<nav epub:type="lot">
    <h2>List of tables, broken down into individual groups, one per major section of the
publication content</h2>
    <ol>
        <li><span>Tables in Chapter 1</span>
            <ol>
                <li><a href="chap1.xhtml#table-1.1">Table 1.1</a>
                </li>
                <li><a href="chap1.xhtml#table-1.2">Table 1.2</a></li>
            </ol>
        </li>
        <li><span>Tables in Chapter 2</span>
            <ol>
                <li><a href="chap2.xhtml#table-2.1">Table 2.1</a>
                </li>
                <li><a href="chap2.xhtml#table-2.2">Table 2.2</a></li>
                <li><a href="chap2.xhtml#table-2.3">Table 2.3</a></li>
            </ol>
        </li>
       …
        <li><span>Tables in Appendix</span>
            <ol>
                <li><a href="appendix.xhtml#table-a.1">Table A.1</a>
                </li>
                <li><a href="appendix.xhtml#table-a.2">Table B.2</a></li>
            </ol>
        </li>
    </ol>
</nav>
```

Each EPUB publication in a SIP SHALL contain the complete table of contents in the EPUB navigation document, covering all levels of the document hierarchy (see http://www.idpf.org/accessibility/guidelines/content/nav/toc.php). This information is important from an accessibility point of view, and although it is not as such relevant for preservation, it is required for the completeness of the SIP.

7.4 Preservation metadata

Preservation metadata is a means of describing all relevant events that have taken place during the document lifecycle prior to, during, and after the ingest to an archive. For instance, if a producer has migrated the submitted EPUB 3 publication from EPUB 2 or some other file format, the preservation metadata provides information about software (and if necessary, hardware) used, and changes in the content and layout of the original document. Often good quality preservation metadata in PREMIS format is not required within the SIP but only created during ingest at the archives.

However, if the submitter is able to provide preservation metadata about (migration) events occurring during pre-ingest or even earlier, it helps to demonstrate provenance and authenticity and could help archives, especially if some issues, such as migration problems, occur during ingest. If such producer-generated preservation metadata cannot be migrated to PREMIS, it MAY be included in the METS administrative metadata section in its native format.

Some preservation metadata elements such as checksums may be generated very early in the life span of a publication, but archive-quality preservation metadata is usually produced during pre-ingest or ingest. At this point, it SHALL be expressed in a PREMIS format. If there is preservation metadata about past events that cannot be migrated to PREMIS, such metadata MAY be included in the METS administrative metadata section in its native format.

If the original (un-archivable) version of the publication is also submitted to the OAIS archive, producer-generated preservation metadata in the SIP SHOULD specify the differences between the two renderings of the publication. Such metadata is useful when a customer is deciding which version of the publication would serve his needs better. If and when the publication is migrated during preservation, similar metadata about format migration SHOULD also be created and stored in the new AIP.

Preservation metadata in a SIP MAY contain local PREMIS event types created by the producer. The archive SHALL pass all this metadata into the AIPs unchanged during the ingest process, except if there is a need to normalize vocabulary used.

In order to simplify the ingest process, the producer or the submitting organization MAY check the validity of the EPUB publications before submitting them. If the publication has been validated, there SHOULD be a PREMIS validation event record in the SIP, documenting the outcome of the validation process and the validation tool used. Archival ingest workflow SHALL include validation as one of the steps. If a submitted publication has already been validated succesfully, the archive MAY choose to bypass the validation step.

NOTE OAIS archives tend to not trust producer-generated file validations. So even if a producer has validated EPUB publications it has submitted, and recorded appropriate metadata about these validation events in SIPs, the archive may decide to validate the submitted publications again.

Submission agreements MAY require producers to carry out validation. Validation tools to be used, such as EpubCheck[29] or Ace by DAISY[30], SHOULD be specified in the agreement as well. An example of PREMIS metadata for a successful validation event of an EPUB publication is provided below.

If validation fails, the publication SHOULD NOT be submitted before the problem is fixed, unless the submission agreement allows the submission of EPUB publications which have not passed validation. Such an allowance MAY apply to certain kinds of validation problems only. For instance, it is possible that the validation fails even though the problematic publication is rendered successfully with EPUB reading systems the document has been designed for. If the validation fails but the publication is nevertheless ingested as such, the negative validation result SHALL be included in the AIP preservation metadata. If the producer submits a corrected EPUB publication, there is no need to store the validation result of the original document.

Core media file resources and foreign resources in EPUB publications itself SHOULD be validated independently to make sure that they actually are in formats claimed. Once the OAIS archive is certain that these resources are in file formats suitable for either ingest or preservation, these resources are treated accordingly (ingestible resources are migrated to archivable formats, resources in archival formats are transferred to AIPs). Common validation tools like JHOVE[31] SHOULD be used whenever possible (that is, when tools the OAIS archive has do support the file formats to be processed).

If the validation fails, the SIP SHOULD NOT be submitted before the problem is fixed unless the submission agreement allows the producer to submit SIPs with non-archivable resources in them. Resources in these file formats SHALL be encoded in such a way that no attempt is made to validate or migrate them during ingest.

[29] https://github.com/IDPF/epubcheck/releases

[30] https://github.com/daisy/ace

[31] http://jhove.openpreservation.org/

The common validation tools do not cover all file formats and validation results can be less than perfect. Therefore it can be difficult to validate some resources beyond just rendering them.

Validation results SHALL be expressed using the <mets:digiprovMD> element with PREMIS events (<premis:event>). A list of event types and examples of their use are provided in Annex 2. The event type list is not complete, but it covers typical events that can occur during the lifecycle of preserved documents.

Examples

Validation:

```
<mets:digiprovMD ID="ev001" CREATED="2016-03-08T00:00:00">
  <mets:mdWrap MDTYPE="PREMIS:EVENT">
    <mets:xmlData>
      <premis:event>
        <premis:eventIdentifier>
          <premis:eventIdentifierType>local</premis:eventIdentifierType>
          <premis:eventIdentifierValue>
            epub3validation-001
            </premis:eventIdentifierValue>
          </premis:eventIdentifier>
          <premis:eventType>validation</premis:eventType>
          <premis:eventDateTime>2016-03-08T11:12:13</premis:eventDateTime>
          <premis:eventOutcomeInformation>
            <premis:eventOutcome>success</premis:eventOutcome>
             <premis:eventOutcomeDetail>
               format="EPUB"; version="3.0.1"; result="Well-formed and valid"
             </premis:eventOutcomeDetail>
          </premis:eventOutcomeInformation>
          <premis:linkingAgentIdentifier>
            <premis:linkingAgentIdentifierType>
              local
          </premis:agentIdentifierType>
          <premis:linkingAgentIdentifierValue>
            epubvalidator-4
          </premis:agentIdentifierValue>
        </premis:linkingAgentIdentifier>
      </premis:event>
    </mets:xmlData>
  </mets:mdWrap>
</mets:digiprovMD>

<mets:digiprovMD ID="ag001" CREATED="2016-03-08T00:00:00">
  <mets:mdWrap MDTYPE="PREMIS:AGENT">
    <mets:xmlData>
      <premis:agent>
        <premis:agentIdentifier>
          <premis:agentIdentifierType>local</premis:agentIdentifierType>
          <premis:agentIdentifierValue>
            epubvalidator-4
          </premis:agentIdentifierValue>
        </premis:agentIdentifier>
        <premis:agentName>EpubCheck 4.0.0</premis:agentName>
        <premis:agentType>software</premis:agentType>
      </premis:agent>
    </mets:xmlData>
  </mets:mdWrap>
</mets:digiprovMD>

Rendering an EPUB 3 file:

<mets:digiprovMD ID="ev002" CREATED="2016-03-10T00:00:00">
  <mets:mdWrap MDTYPE="PREMIS:EVENT">
    <mets:xmlData>
      <premis:event>
        <premis:eventIdentifier>
```

```
          <premis:eventIdentifierType>local</premis:eventIdentifierType>
          <premis:eventIdentifierValue>
            epub3rendering-001
            </premis:eventIdentifierValue>
          </premis:eventIdentifier>
          <premis:eventType>rendering</premis:eventType>
          <premis:eventDateTime>2016-03-10T14:12:05</premis:eventDateTime>
          <premis:eventOutcomeInformation>
            <premis:eventOutcome>success</premis:eventOutcome>
             <premis:eventOutcomeDetail>
               format="EPUB"; version="3.1"
             </premis:eventOutcomeDetail>
          </premis:eventOutcomeInformation>
          <premis:linkingAgentIdentifier>
            <premis:linkingAgentIdentifierType>
              local
          </premis:agentIdentifierType>
          <premis:linkingAgentIdentifierValue>
            epubrender-1
          </premis:agentIdentifierValue>
        </premis:linkingAgentIdentifier>
      </premis:event>
    </mets:xmlData>
  </mets:mdWrap>
</mets:digiprovMD>

<mets:digiprovMD ID="ag001" epub3:CREATED="2016-03-10">
  <mets:mdWrap MDTYPE="PREMIS:AGENT">
    <mets:xmlData>
      <premis:agent>
        <premis:agentIdentifier>
          <premis:agentIdentifierType>local</premis:agentIdentifierType>
          <premis:agentIdentifierValue>
            epubrender-1
          </premis:agentIdentifierValue>
        </premis:agentIdentifier>
        <premis:agentName>EPUBReader 1.5.0.8</premis:agentName>
        <premis:agentType>software</premis:agentType>
      </premis:agent>
    </mets:xmlData>
  </mets:mdWrap>
</mets:digiprovMD>
```

8 Structure of Submission Information Packages

This document does not pose any requirements on the specific structure of the SIP.

A producer or other party submitting EPUB publications to an archive SHALL assemble a SIP containing 1-n EPUB publications or just descriptive or administrative metadata about them. Multiple EPUB publications in the same SIP are allowed if they are parts of the same work; for instance, chapters in a book.

A SIP MAY contain 0-n representations of the submitted EPUB publication in different formats such as PDF/A, and their associated metadata. Instead of a publication itself a SIP MAY contain just metadata about it.

Each representation of a publication in a SIP SHALL have its own metadata.

A SIP SHALL contain descriptive metadata in Dublin Core (as required by the EPUB 3 specification) and administrative metadata in PREMIS. Metadata in other formats MAY be included, as specified in the EPUB and METS standards, submission agreement, and other ingest related agreements between the producer and the OAIS archive (if any).

If SIPs are sent over a network and the transmission channel used is not secure, SIPs SHALL be signed electronically. Submission agreement SHALL specify secure means of transmission; they may include electronic (e.g. Secure File Transfer Protocol) and traditional channels (e.g. DHL).

The archive MAY use the same package structure when sending documents back to the producer or other consumer as dissemination information packages (DIPs). However, a DIP does not necessarily contain all the data and metadata present in a SIP. If the preserved EPUB 3 documents and metadata have been migrated during the preservation, the archive usually submits the latest version unless the customer demands an older version.

9 Content of Submission Information Packages

A SIP SHOULD contain a manifest file specifying the content of the package using METS structMap and fileSec.

The name and the location of the file depends on the submission agreement. If EPUB manifest.xml file is used, it SHALL be located in the META-INF directory at the root level of the EPUB container file system.

If METS manifest file is used and it is located in the METS container, its name SHOULD be mets.xml[32]. If other name is used, it SHALL be specified in the submission agreement.

The EPUB manifest file manifest.xml SHALL be compliant with the EPUB Open Container Format requirements.

The manifest file mets.xml SHALL use METS schema to encapsulate information of the files in the SIP. Supported METS version(s) SHOULD be specified in the submission agreement. As of this writing the latest version of the standard is 1.11.

The manifest file mets.xml SHALL be compliant with the EPUB 3 METS profile.

The character encoding of the mets.xml file SHALL be UTF-8, and the file SHALL be located in the root directory of a SIP.

The mets.xml SHALL contain in the structMap element structural metadata needed for locating the EPUB 3 publications and other documents such as PDF/A versions of the publication in the SIP. The SIP SHALL also contain descriptive and administrative metadata required for ingest and archival of the EPUB publications and other documents in the package. Depending on the submission agreement, the producer may have to copy mandatory metadata elements from the EPUB publications to the mets.xml file. Otherwise the archive harvests such metadata directly from the EPUB publications during ingest.

A SIP MAY contain additional metadata not required by this document. If so, such metadata elements SHALL be encoded in an appropriate manner if there is a risk of confusion. For instance, if there are additional MARC 21 identifier elements (for e.g. related publications or related versions of the submitted publication), it SHALL be possible to tell them apart from the identifier of the submitted EPUB publication. If the SIP contains several EPUB publications, or versions of the EPUB publication in other file formats, it SHALL be possible to attribute the metadata records in SIP to correct documents.

The internal structure of EPUB 3 containers does not need to be expressed in the mets.xml file (if present). Repository systems supporting EPUB 3 as the ingest and preservation format SHOULD have the tools with which to render EPUB 3 documents and embedded core media resources. Although the OAIS archive may not be able to view every EPUB publication, it SHOULD have appropriate viewer applications with which to check whether a certain EPUB publication is well formed. Such rendering implies that the structural metadata in the spine of the EPUB publication is used.

[32] Two options are provided to guarantee backwards compatibility with existing systems.

If a digital signature is needed to prove the identity of the submitting organization, there are two possible approaches, one of which SHOULD be selected in the submission agreement.

An EPUB publication MAY contain signatures.xml file within the META-INF directory at the root level of the container file system. This file, if present, contains the digital signatures of the EPUB container and its contents. Its schema is specified in the EPUB Open Container Format.

If the EPUB container does not have a signatures.xml file, before the SIP is sent the producer SHALL create a file signature.sig as described below, especially if the transmission channel is not secure. The signature.sig file SHALL contain all the required signatures, created with specific algorithms.

The archive SHOULD be able to check the authenticity of the publication or publications in a SIP by a checksum. Checksums algorithms to be used SHALL be specified in the submission agreement. Such algorithms SHALL be commonly used message digest algorithms such as SHA-256.

The producer SHOULD[33] calculate the checksum for every file and add it in the preservation metadata. If the SIP does not contain checksums, the OAIS archive SHALL calculate them and add them to the AIP.

The integrity of the submitted content is checked by comparing the checksum calculated in the archive to the existing checksums. If the checksums do not match, the SIP's integrity cannot be verified and the ingest fails. The archive sends an error report to the producer or submitting organization, who SHALL check the problematic file and resubmit it.

[33] This requirement ought to be SHALL. But in practice, producers often fail to provide checksums, and making them mandatory might mean that producers would not send their publications to the archive anymore. This situation can be improved by providing to the publishers tool sets which simplify the task of creating SIPs.

Annex A
(Informative)

Digital Signature

The data integrity and nonrepudiation can be verified by a digital signature. With the signature the data sender can be verified even if the transmission channel has been unreliable.

According to the EPUB documentation, creating digital signatures for the EPUB container and its contents is not mandatory, but if signatures have been created, they SHALL be placed into a signatures.xml file located in the META-INF directory.

If the archive is located in-house, or if a secure channel has been used during the submission, a digital signature is not mandatory, but the SIP SHALL contain checksums so that it is possible to detect corruption of data during transfer and initial storage at the producer's systems. Checksums SHOULD be created right after documents are completed; otherwise it is not possible to detect if an EPUB publication has changed unintentionally during the initial storage.

There are various ways in which digital signatures can be created. The submission agreement SHOULD specify the procedure. One possible approach is the following, adopted in the Finnish National Digital Library project:

The submitting organization digitally signs the manifest file with a PKCS#7[34] (Public Key Cryptography Standard no. 7) signature. PKCS#7 is used to sign messages in the Public Key Infrastructure[35].

The signature is formed as follows:

- A Checksum for the manifest.xml file is calculated by using one of the algorithms specified in the submission agreement. Common options include md5, sha1, sha224, sha256, sha384 and sha512. Checksums for the EPUB 3 files have to already be contained in the METS document so as to avoid the need to calculate them as well while forming the signature for manifest.xml.
- A Checksum is added to a text file by writing the following into it: checked file's (manifest.xml) path relative to the SIP's root, algorithm used and the checksum. These data elements are separated by semicolons.
- A File is signed by PKCS#7 signature using a certificate.
- A File is saved into the SIP package's root with the file name "signature.sig" in S/MIME format.

The archive verifies the checksum in the "signature.sig" file with the signer's public key and calculates the checksum using the same algorithm as the sender. If these checksums match, the integrity of the mets.xml file has been maintained during the transfer to the archive, and the archive can be certain that the file was signed by the correct producer.

Below is an example of a "signature.sig" file created in this way. In the example the checksum is calculated by SHA-1 algorithm that is signed by a PKCS7 signature.

```
MIME-Version: 1.0
Content-Type: multipart/signed; protocol="application/x-pkcs7-signature";
micalg=sha1; boundary="----57E5EFE5F87ADB48166F35F180BE72AC"

This is an S/MIME signed message

------57E5EFE5F87ADB48166F35F180BE72AC
```

[34] https://en.wikipedia.org/wiki/PKCS

[35] https://en.wikipedia.org/wiki/Public_key_infrastructure

```
Content-Type: text/plain

./mets.xml:sha1:effdb5f96a28acd2eb19dcb15d8f43af762bd0ae

------57E5EFE5F87ADB48166F35F180BE72AC
Content-Type: application/x-pkcs7-signature; name="smime.p7s"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="smime.p7s"
MIIFKwYJKoZIhvcNAQcCoIIFHDCCBRgCAQExCzAJBgUrDgMCGgUAMAsGCSqGSIb3
...
UY/I9QDibkW5qAUY00rN7Q1F+uAGB+twEg6un1SVdg==
------57E5EFE5F87ADB48166F35F180BE72AC--
```

Annex B
(Informative)

Events

Long-term preservation requires actions made by human or software agents. To preserve the archived documents, they are ingested, refreshed, repackaged, replicated, and migrated in the archive, and disseminated to users. These actions are PREMIS events.

Taken together, PREMIS events targeting an archived EPUB publication form a transaction log: a history of what has been done to the publication, and by what/whom. Such history can persist longer than the archived documents themselves. Therefore, digital archives SHALL keep track of everything that is done to their contents, and investigate all exceptions. An unauthorized copying of an archived publication is an example of an event that SHALL be scrutinized by the archive management. The likely result of such scrutiny is deletion of the extra copy.

Some preservation actions may have an impact on the look and feel and/or even the intellectual content of the preserved documents. If and when such changes do take place, they SHALL be documented properly as part of an event description. That way archive users can decide which version of the relevant document suits their information needs best. Migrated version may be the easiest to use, though it may not be authentic enough for a critical user. But since migrating fixed layout publications or interactive publications can cause serious loss of semantics or functionality, emulation is the preferred preservation method for such EPUB publications. Moreover, emulation-based access to the original, non-migrated publication may serve best those users who prefer to use the most authentic version of an archived resource.

PREMIS is the de facto tool with which events are documented in long-term preservation. The standard itself does not contain a list of event types, but such lists have been created or are maintained by e.g. The Library of Congress and Archivematica:

http://id.loc.gov/vocabulary/preservation/eventType.html
https://wiki.archivematica.org/PREMIS_metadata:_events

Some libraries have created their own event lists. For instance, Bibliothèque nationale de France (BnF) uses packageDelivery (delivery of an information package) event described above, packageCreation (SIP ready), ingestCompletion (AIP ready) and disseminationCompletion (DIP ready) events.

See also

http://premis-event-service.readthedocs.org/en/latest/technical_overview.html

for an explanation of PREMIS events and their use.

The list below is provided as an example of how diverse PREMIS events can be. It contains events from Archivematica and Library of Congress lists and some other sources. The <premis:eventDetail> syntax is from the Archivematica list, but it is also possible (and perhaps preferable) to describe all applications as PREMIS agents; that way the eventDetail would just refer to the agent.

disseminationCompletion	Dissemination Information Package (DIP) ready
packageDelivery	Delivery of an information package
packageCreation	Creation of a new information package (SIP)

ingestCompletion	Archival Information Package (AIP) ready
capture	The process whereby a repository actively obtains an object.
compression	The process of coding data to save storage space or transmission time.

creation	The act of creating a new EPUB publication. If this event is included, event outcome (usually success) SHALL be provided, and the name and version of the tool used SHOULD be specified in the PREMIS eventDetail element. For example:
	`<premis:eventDetail>`
	`program=”AdobeInDesign”;version=”CC (2015)”`
	`</premis:eventDetail>`

deaccession	The process of removing an object from the inventory of a repository.
decompression	The process of reversing the effects of compression.
decryption	The process of converting encrypted data to plain text.
deletion	The process of removing an object from repository storage.
digital signature validation	The process of determining that a decrypted digital signature matches an expected value.
fixity check	The process of verifying that an object has not been changed in a given period.
ingestion	The process of adding objects to a preservation repository.
message digest calculation	The process by which a message digest ("hash") is created.

migration	A transformation of an object creating a version in a more contemporary or more widely used file format (for instance, migrating a DITA file into EPUB 3). If this event is included, the outcome SHALL be provided in eventOutcome, and the name and the version (when applicable) of the tool used SHOULD be added into PREMIS eventDetail element. For example:
	`<premis:eventOutcomeInformation>success`
	`</premis:eventOutcomeInformation>`
	`<premis:eventDetail>`
	`program=”DITA for Publishers EPUB plugin” version=”1.0.0RC16”`
	`</premis:eventDetail>`
	`<premis:eventOutcomeDetail>`

	format="EPUB";version="3.0.1";
	result="success"
	</premis:eventOutcomeDetail>

processing	The process of doing something for the object, to be used when there is no dedicated event type for an operation. A description of the process SHALL be added into the PREMIS eventDetail element. For example: <premis:eventDetail> Deskew: 0.49 CW; Crop: left 24px, top 9px, right 29px, bottom 12px; </premis:eventDetail>
rendering	The process of rendering the document; for instance, reading an EPUB 3 document.
replication	The process of creating a copy of an object that is, bit-wise, identical to the original.
validation	The process of validating an EPUB file. If this event is included, the name and the version (when applicable) of the tool used SHOULD be added into the PREMIS eventDetail element and the result of the process SHALL be provided as an eventOutcomeDetail element. For example: <premis:eventDetail> program=”EpubCheck”version=”4.0.2” </premis:eventDetail> <premis:eventOutcomeInformation>pass </premis:eventOutcomeInformation> <premis:eventOutcomeDetail> format="EPUB";version="3.0.1"; result="Well-formed and valid" </premis:eventOutcomeDetail>
virus check	The process of scanning a file for malicious programs.

Usually the event outcome is a success (pass), failure, or unknown.

Below is another example from the BnF, describing how delivery of an information package can be expressed as a PREMIS event.

The producer is referred to with digiprovMD (ID=”AMD.10”), which makes it possible to describe the producer as a PREMIS agent below. Note that in the BnF implementation, producers also have UUIDs.

```
<digiprovMD ID="AMD.9"  ADMID="AMD.10">
  <mdWrap MIMETYPE="text/xml" MDTYPE="PREMIS:EVENT">
    <xmlData>
      <premis:event>
        <premis:eventIdentifier>
          <premis:eventIdentifierType>UUID</premis:eventIdentifierType>
          <premis:eventIdentifierValue>f082af00-85b4-11e2-98c4-00144f80ca6b
</premis:eventIdentifierValue>
        </premis:eventIdentifier>
        <premis:eventType>packageDelivery</premis:eventType>
        <premis:eventDateTime>2016-01-25T11:22:16+01:00</premis:eventDateTime>
        <premis:eventDetail>Channel 215: initial delivery</premis:eventDetail>

        <premis:linkingAgentIdentifier>
          <premis:linkingAgentIdentifierType>UUID</premis:linkingAgentIdentifierType>
          <premis:linkingAgentIdentifierValue>2ba05b98-a4c3-11e5-bf7f-feff819cdc9f
</premis:linkingAgentIdentifierValue>
          <premis:linkingAgentRole>issuer</premis:linkingAgentRole>
        </premis:linkingAgentIdentifier>

        <premis:linkingObjectIdentifier>
          <premis:linkingObjectIdentifierType>Order form
</premis:linkingObjectIdentifierType>
          <premis:linkingObjectIdentifierValue>16-SI-0001
</premis:linkingObjectIdentifierValue>
          <premis:linkingObjectRole>request</premis:linkingObjectRole>
        </premis:linkingObjectIdentifier>

        <!-- fileGroup uses that are delivered  -->
        <premis:linkingObjectIdentifier>
          <premis:linkingObjectIdentifierType>USE</premis:linkingObjectIdentifierType>
          <premis:linkingObjectIdentifierValue>master</premis:linkingObjectIdentifierValue
>
          <premis:linkingObjectRole>outcome</premis:linkingObjectRole>
        </premis:linkingObjectIdentifier>
        <premis:linkingObjectIdentifier>
          <premis:linkingObjectIdentifierType>USE</premis:linkingObjectIdentifierType>
          <premis:linkingObjectIdentifierValue>colorProfile</premis:linkingObjectIdentifie
rValue>
          <premis:linkingObjectRole>outcome</premis:linkingObjectRole>
        </premis:linkingObjectIdentifier>
      </premis:event>
    </xmlData>
  </mdWrap>
</digiprovMD>
<digiprovMD ID="AMD.10">
  <mdWrap MIMETYPE="text/xml" MDTYPE="PREMIS:AGENT">
    <xmlData>
      <premis:agent>
        <premis:agentIdentifier>
          <premis:agentIdentifierType>UUID</premis:agentIdentifierType>
          <premis:agentIdentifierValue>2ba05b98-a4c3-11e5-bf7f-
feff819cdc9f</premis:agentIdentifierValue>
        </premis:agentIdentifier>
        <premis:agentName>Producer name</premis:agentName>
        <premis:agentType>organization</premis:agentType>
      </premis:agent>
    </xmlData>
  </mdWrap>
</digiprovMD>
```

Bibliography

[1] *CCSDS 650.0-M-2 Reference model for an Open Archival Information System (OAIS)*. [online]. [viewed 2016-05-03]. Available from http://public.ccsds.org/publications/archive/650x0m2.pdf.

[2] *Digital Preservation Handbook*. 2nd, revised ed. [online]. Digital Preservation Coalition, 2017. Available from: http://www.dpconline.org/handbook. [viewed 2017-07-25].

[3] E-ARK: *Common specification for information packages. Version 1.0* [online]. E-ARK Project, 2016. Available from: http://www.dasboard.eu/specifications/common-specification. [viewed 2017-08-23].

[4] *EPUB Open Container Format (OCF) 3.0.1.* International Digital Publishing Forum, 2014. [online]. Available from: http://www.idpf.org/epub/301/spec/epub-ocf.html. [viewed 2018-10-25].

[5] *EPUB 3.0.1.* International Digital Publishing Forum, 2014. [online] . Available from: http://idpf.org/epub/301. [viewed 2018-10-30].

[6] Guidelines for using PREMIS with METS for exchange. Revised September 17, 2008 [online].

[7] KASDORF, Bill: *EPUB 3 packaging and metadata*. [online]. [s.l.]: EPUBZone, 2013. Available from: http://epubzone.org/news/epub-3-packaging-and-metadata [viewed 2017-07-26]. Archived version available at: https://web.archive.org/web/20140701083126/http://epubzone.org/news/epub-3-packaging-and-metadata

[8] LAVOIE, Brian. *Meeting the challenges of digital preservation: the OAIS reference model*. [online]. Columbus, OH: OCLC, 2000. Available from: http://www.oclc.org/research/publications/library/2000/lavoie-oais.html. [viewed 2016-07-15].

[9] *Metadata requirements and preparing content for digital preservation*. [online]. The National Digital Library Project, 2018. Available from: http://digitalpreservation.fi/files/Metadata-1.7.0-en.pdf. [viewed 2018-09-10].

[10] *Understanding metadata*. [online]. Bethesda, MD.: NISO, 2004. Available from: http://www.niso.org/publications/press/UnderstandingMetadata.pdf. [viewed 2016-07-20].