DELIVERABLE

Project Acronym:	**E-ARK**
Grant Agreement Number:	**620998**
Project Title:	**European Archival Records and Knowledge Preservation**

DELIVERABLE DETAILS

DELIVERABLE REFERENCE NO.	**D3.2**
DELIVERABLE TITLE	E-ARK SIP Draft Specification
REVISION	**1.0**

AUTHOR(S)	
Name(s)	Organisation(s)
Tarvo Kärberg	National Archives of Estonia (NAE)
Karin Bredenberg	National Archives of Sweden / ES Solutions (ESS)
Björn Skog	ES Solutions (ESS)
Anders Bo Nielsen	Danish National Archives (DNA)
Kathrine Hougaard Edsen Johansen	Danish National Archives (DNA)
Hélder Silva	KEEP Solutions (KEEPS)

REVIEWER(S)	
Name(s)	Organisation(s)
Kuldar Aas	National Archives of Estonia (NAE)
Angela Dappert	DLM Forum / University of Portsmouth (DLM/UPHEC)
David Anderson	University of Portsmouth (UPHEC)

Project co-funded by the European Commission within the ICT Policy Support Programme Dissemination Level		
P	**Public**	x
C	**Confidential, only for members of the Consortium and the Commission Services**	

REVISION HISTORY AND STATEMENT OF ORIGINALITY

Submitted Revisions History

Revision No.	Date	Authors(s)	Organisation	Description
0.1	20.10.2014	Tarvo Kärberg	NAE	First draft.
0.2	13.11.2014	Tarvo Kärberg	NAE	Updating content.
0.3	02.12.2014	Tarvo Kärberg	NAE	Updating content.
0.4	17.01.2015	Tarvo Kärberg	NAE	Updating content.
0.5	21.01.2015	Karin Bredenberg	ESS	Updating content.
0.6	23.01.2015	Anders Bo Nielsen	DNA	Updating content.
0.7	23.01.2015	Kathrine Hougaard Edsen	DNA	Updating content.
0.71	26.01.2015	Björn Skog	ESS	Updating content
0.72	27.01.2015	Hélder Silva	KEEPS	Updating content
0.8	27.01.2015	Angela Dappert	DLM/UPHEC	Quality assurance and proof-reading.
0.9	29.01.2017	Kuldar Aas	NAE	Quality assurance and proof-reading.
0.91	30.01.2015	David Anderson	UPHEC	Quality assurance and proof-reading.
1.0	30.01.2015	Tarvo Kärberg	NAE	Final version.

Statement of originality:
This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

EXECUTIVE SUMMARY

A Submission Information Package (SIP) is defined in the OAIS standard[1] as an Information Package that is delivered by the Producer to the OAIS for use in the construction or update of one or more AIPs and/or the associated Descriptive Information. Many different SIP formats are used all over the world and unfortunately there is currently no central format for a SIP which would cover all individual national and business needs identified in the E-ARK Report on Available Best Practices.[2]

Therefore, the main objective of this deliverable is to describe a draft SIP specification for the E-ARK project – give an overview of the structure and main metadata elements for E-ARK SIP and provide initial input for the technical implementations of E-ARK ingest tools.[3]

The target group of this work are E-ARK project partners as well as all other archival institutions and software providers creating or updating their SIP format specifications.

This document provides an overview of:

- The general structure for submission information packages.
  This chapter explains how the E-ARK SIP is constructed by following the common rules for all other (archival, dissemination) information packages.
- The SIP METS Profile.
  This chapter provides a detailed overview of metadata sections and the metadata elements in these sections. The table with all metadata elements could possibly be of interest to technical stakeholders who wish to continue with the more detailed work of the E-ARK SIP implementation later.

Two examples with different kinds of content (MoReq2010, SIARD-E) following the common structure for E-ARK submission information package can be found in the appendixes to this document.

[1] Reference Model for an Open Archival Information System (OAIS), 2012, public.ccsds.org/publications/archive/650x0m2.pdf

[2] Deliverable D3.1 E-ARK Report on Available Best Practices, 2014, http://eark-project.com/resources/project-deliverables/6-d31-e-ark-report-on-available-best-practices

[3] This deliverable should be considered a draft E-ARK SIP specification as more detailed information for implementation will follow in the next phase (in deliverable D.3.3 in the E-ARK project).

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENT

The authors of this deliverable would like to thank all national archives, tool developers and other stakeholders who provided valuable knowledge about their submission information packages. A special gratitude goes to the National Archives of Sweden whose FGS (Förvaltningsgemensam Specifikation) structure influenced the E-ARK SIP METS profile development significantly.

1. INTRODUCTION

The purpose of this document is to describe the draft specification for the E-ARK SIP and, therefore, this work will be important mainly for E-ARK project partners; including partners who develop pre-ingest and ingest solutions for the E-ARK pilots later in this project. However, to gain also further insights beyond the project partners the draft specification will be disseminated and discussed among other organisations, mainly archival institutions and archival software providers.

This deliverable contains:

- GENERAL STRUCTURE AND DATA MODEL FOR SUBMISSION INFORMATION PACKAGES
- SIP METS PROFILE
- CONCLUSIONS
- REFERENCES
- Appendix A: Criteria for a submission information package

Appendix A: Criteria for a submission information package

Every SIP should follow the quality critery described below:

1. The structure and content of the SIP must be able to be validated:
    a. integrity, fixity (are all folders and files intact and at the right place)
    b. syntax (is the format of the information content correct)
    c. semantics (manual or semi-automatically)
2. The SIP format should allow efficient processing.
    a. huge information content must be able to be split and be spanned over many SIPs
    b. The SIP size is limited due to limitations on:
        i. capacity on removable media (also applies for archival storage on single media)
        ii. efficient network transfer
        iii. filesystem limitations, especially for number of files per folder
3. The SIP format should be expandable for the addition of new metadata standards or data formats.
4. The SIP format should be suitable for allowing conversion of the information content into new data formats.
5. The SIP format should allow including metadata for the formats for information content.
    a. shall be able to be extended to support any content type a digital repository needs to ingest
    b. shall allow separate content type descriptions to be specified for different types of submissions
6. The SIP format must be as simple as possible.

7. Data and metadata could be separated to ease the change of data formats or metadata formats.
8. Metadata could be separated into several files to ease the change of metadata formats and to limit the time spend on handling metadata.

- Appendix B: MoReq2010 records management metadata example
- Appendix C: SIARD SIP catalogue structure example
- Appendix D: Metadata for creating profiles
- Appendix E: Terminology

The deliverable D3.2 of the E-ARK project relies on the OAIS model[4] particularly focusing on the format of submission information package (SIP) as seen in

Figure 1.

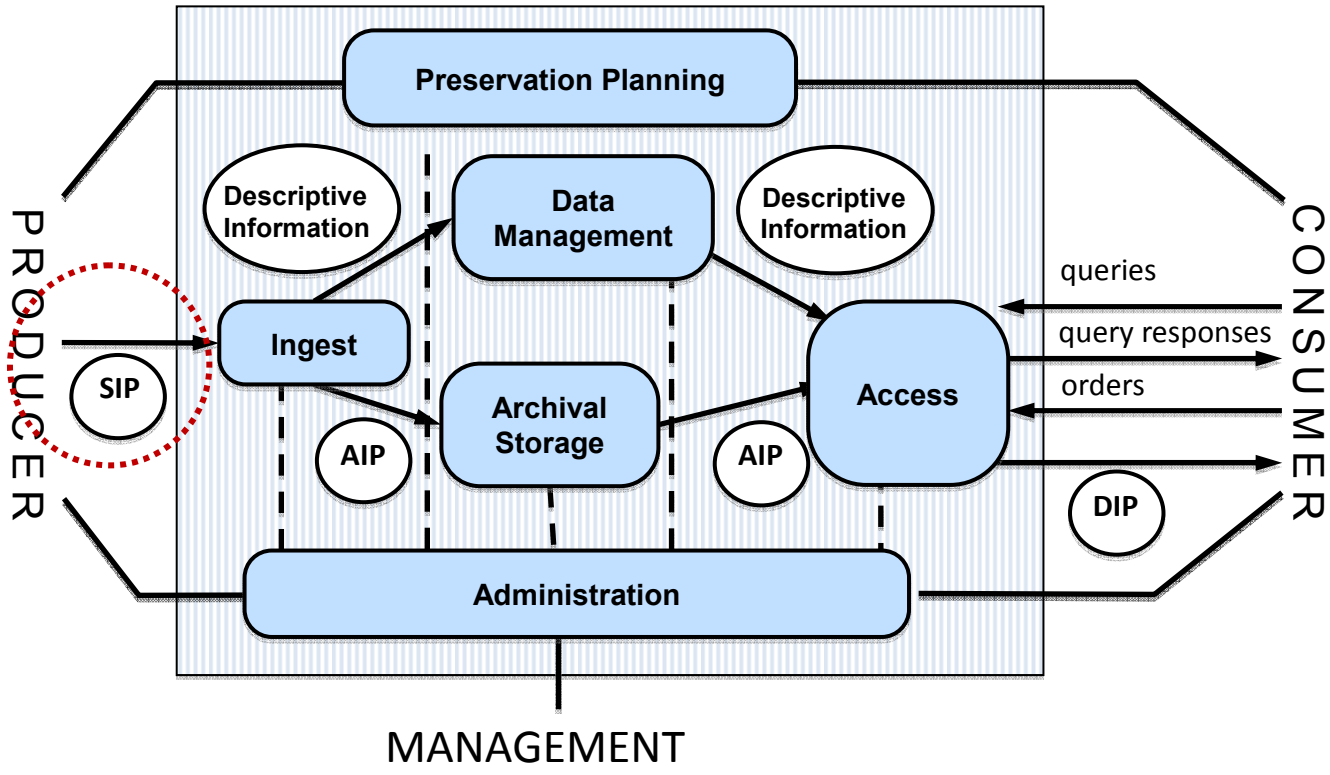

Figure 1: OAIS Functional Entities

The current deliverable does not define functional entities or workflows of a digital archive, but rather describes the draft SIP package structure and minimum set of required metadata for SIP delivery to the archive.[5]

Furthermore, this deliverable does not aim to provide information about archival information packages (AIPs) or dissemination packages (DIPs).[6]

[4] Reference Model for an Open Archival Information System (OAIS), 2012, public.ccsds.org/publications/archive/650x0m2.pdf

[5] Note that this deliverable should be considered a draft version of E-ARK SIP specification as more detailed information for implementation will follow in next phase.

This deliverable is based on the following documents:

- Deliverable D3.1 E-ARK Report on Available Best Practices, 2014, http://eark-project.com/resources/project-deliverables/6-d31-e-ark-report-on-available-best-practices
- Deliverable D2.1 General pilot model and use case definition, 2014, http://eark-project.com/resources/project-deliverables/5-d21-e-ark-general-pilot-model-and-use-case-definition
- FGS package structure, 2013, https://riksarkivet.se/Media/pdf-filer/Projekt/FGS_Earkiv_Paket.pdf
- Reference Model for an Open Archival Information System (OAIS), 2012, public.ccsds.org/publications/archive/650x0m2.pdf
- Producer-Archive Interface Methodology Abstract Standard (PAIMAS), 2004, public.ccsds.org/publications/archive/651x0m1.pdf
- Producer-Archive Interface Specification (PAIS) – CCSDS, 2014, public.ccsds.org/publications/archive/651x1b1.pdf
- e-SENS (Electronic Simple European Networked Services) project, http://www.esens.eu/

Additional technical documentation regarding this deliverable can be found in the following documents:

- E-ARK METS Profile
  http://www.ra.ee/METS/v01/SIP.xml
- XML Schema for the E-ARK METS Profile
  http://www.ra.ee/METS/v01/SIP.xsd
- XLink Schema
  http://www.ra.ee/METS/v01/xlink.xsd
- Extension Schema
  http://www.ra.ee/METS/v01/ExtensionMETS.xsd

[6] These are provided respectively in deliverables 4.2 and 5.2 of the E-ARK project.

2. GENERAL STRUCTURE AND DATA MODEL FOR SUBMISSION INFORMATION PACKAGES

The draft SIP specification for E-ARK follows the general structure which is common for all information packages in the E-ARK project. The SIP data model describes the package submitted to the archive, which consists of submitted content and metadata (root level) as seen in Figure 2.[7]

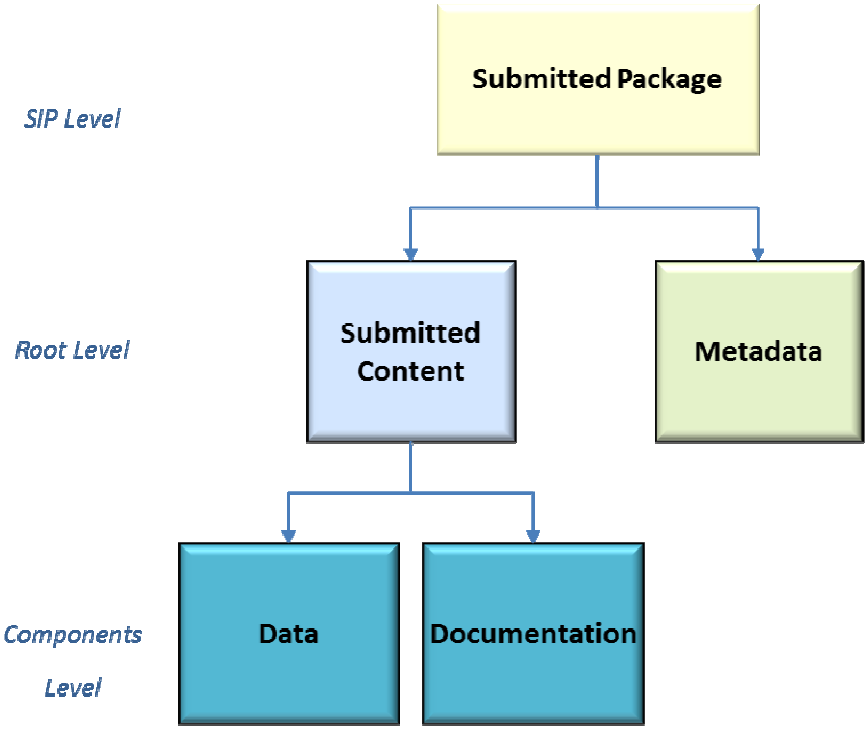

Figure 2: E-ARK SIP data model

The metadata on the root level describes the SIP package itself rather than submitted content. The submitted content consists of data and metadata (components). The metadata in the submitted content describes the submitted data.

This SIP data model also follows the logic described in E-ARK deliverable D3.1[8] , and requires that content and metadata should be separated in the SIP (as seen in Figure 2 as well – submitted package consists of submitted content and metadata), and that submitted content requires some additional metadata (documentation in Figure 2) to accompany the SIP.

[7] This is a conceptual model and does not describe the actual implementation structure.
[8] E-ARK Report on Available Best Practices, 2014, http://eark-project.com/resources/project-deliverables/6-d31-e-ark-report-on-available-best-practices

The E-ARK Report on Available Best Practices[9] showed that METS (Metadata Encoding and Transmission Standard)[10], EAD (Encoded Archival Description)[11] and PREMIS (PREservation Metadata: Implementation Strategies)[12] standards are most commonly used in SIPs. They all have different purposes:

- PREMIS provides metadata to support the long-term preservation of digital objects and their usability.
- EAD is a non-proprietary de facto standard and it is meant for the encoding of finding aids in an online environment.
- METS standard provides a means of associating the metadata related to an object and describes the relationships with other objects.

Mindful of the quality criteria (see Appendix A: Criteria for a submission information package) set for the E-ARK SIP it was decided to use the METS for the high-level descriptions of an information package and EAD, EAC-CPF[13] and PREMIS for the administrative and descriptive metadata as illustrated in

Figure 3.

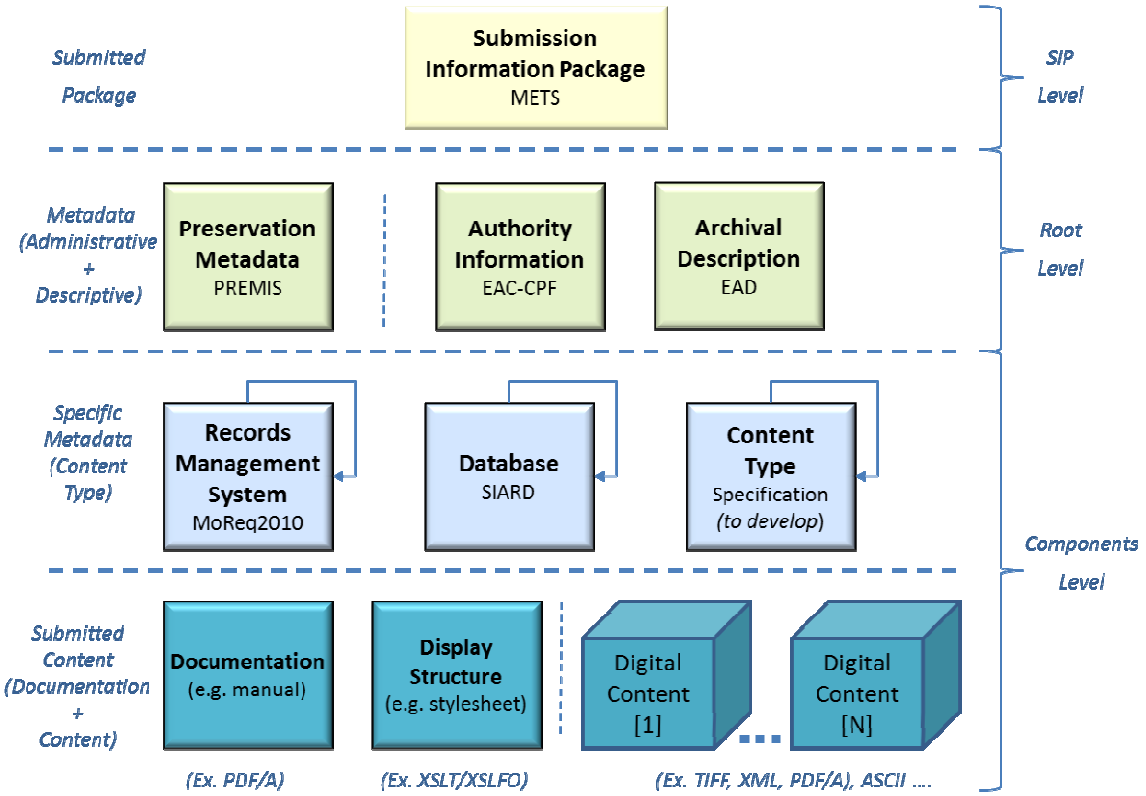

Figure 3: Common structure for submission information packages

[9] E-ARK Report on Available Best Practices, 2014, URL: http://eark-project.com/resources/project-deliverables/6-d31-e-ark-report-on-available-best-practices
[10] METS, 2015, http://www.loc.gov/standards/mets/
[11] Encoded Archival Description, 2015, http://www.loc.gov/ead/
[12] PREservation Metadata: Implementation Strategies , 2015, http://www.loc.gov/standards/premis/
[13] Encoded Archival Context for Corporate Bodies, Persons, and Families, 2015, http://eac.staatsbibliothek-berlin.de/

- Submitted Package – common rules for submission information packages.
- Metadata – the E-ARK project will focus on EAD (Encoded Archival Description), EAC-CPF (Encoded Archival Context – Corporate bodies, Persons, and Families) and PREMIS metadata standards, but the SIP model allows to incorporate any other descriptive metadata standards as well on the root level.
- Specific Metadata / Content Type – delivery-specific content descriptions. For example MoReq2010 for data delivered from records management systems or SIARD for databases.
  One of the core requirements for the E-ARK SIP specification is that it shall be able to be extended to support any content type a digital repository needs to ingest - separate content type descriptions can be specified and developed for different types of submissions.
- Content – digital content and additional information (additional documentation about digital content).

3. SIP METS PROFILE

The current E-ARK SIP METS Profile follows the general structure for information packages as described in chapter 2. "METS Profiles are intended to describe a class of METS documents in sufficient detail to provide both document authors and programmers with the guidance they require to create and process METS documents conforming to a particular profile."[14]

Creating a METS profile requires a good understanding of the METS Profile components. An overview of these components can be found in the METS online documentation[15] and in Table 5 of the Appendices of this document.

The graphical representation of the METS profile is seen in **Error! Reference source not found.**

[14] METS Profiles, http://www.loc.gov/standards/mets/mets-profiles.html
[15] METS Profile Components, http://www.loc.gov/standards/mets/profile_docs/components.html

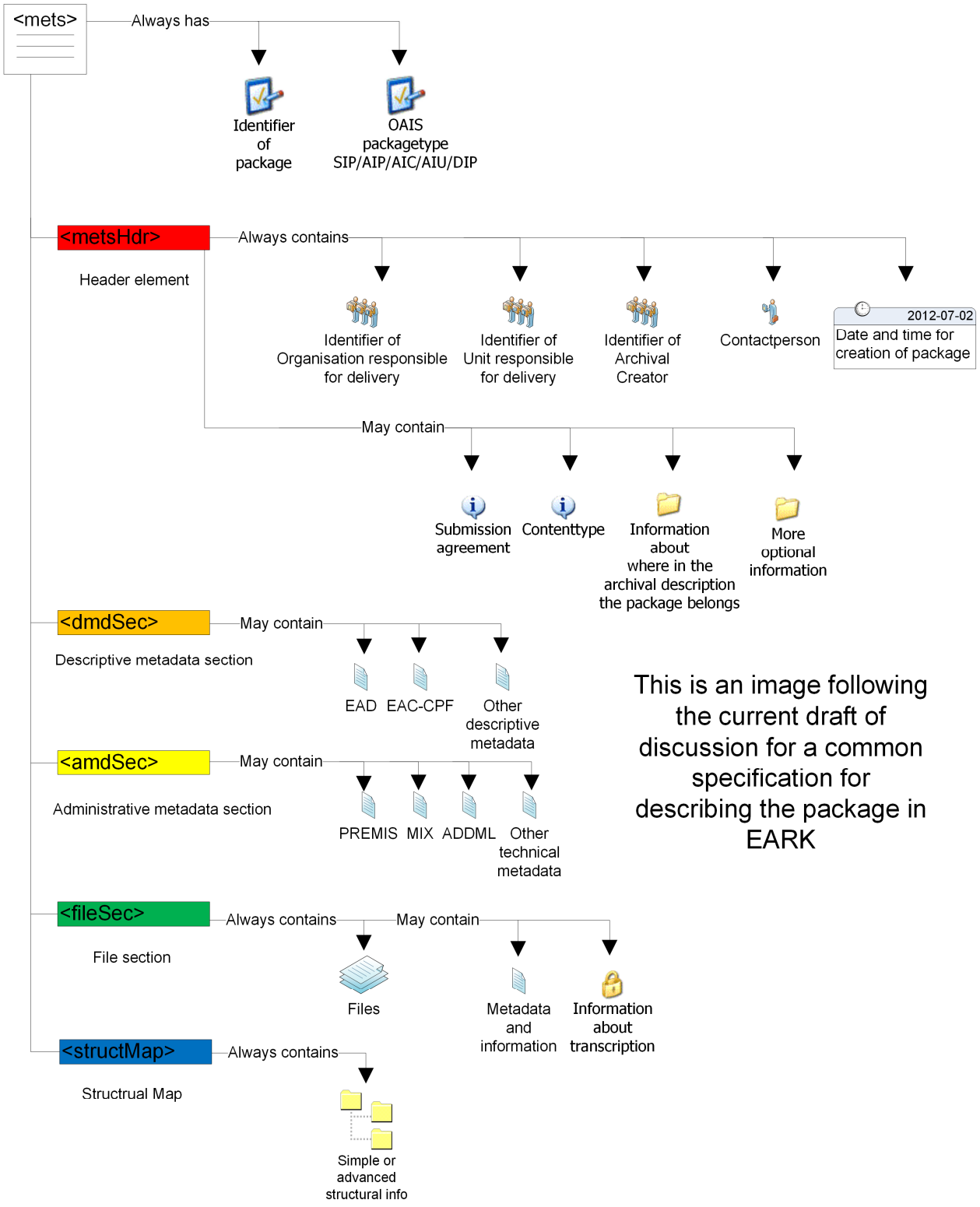

Figure 4: SIP METS Profile

There are 5 main sections in this profile:

- <metsHdr> - METS header (metadata about the creator, contact persons, etc.).
- <dmdSec> - descriptive metadata (EAD, EAC-CPF, etc.).
- <amdSec> - administrative metadata (how files were created and stored, intellectual property rights, etc.).
- <fileSec> - file section, lists all files containing content (may also contain metadata about files).
- <structMap> - structural map, describes the hierarchical structure for the digital object.

These sections will be described in more detail below by giving an overview of main metadata elements used for submission information packages.

The following terms are used in the header of tables regarding information package metadata:

- **Element** - The name of the element in plain text used in the accompanying schema for elements or attributes. For more information regarding elements and attributes in XML see WWW Consortium (http://www.w3.org/).
- **Definition** - Defines the functions of the element. Contains a description of the element and some example values.
- **Cardinality** – Represents the number of occurrences of an element (see below).
  - **0..1** – The element is optional and cannot be repeated.
  - **0..*** – The element is optional and can be repeated.
  - **1** – The element is mandatory and can only be stated once.
  - **1..*** – The element is mandatory and has one or more occurrences.
- **METS** - Defines the element in the METS standard used for designing the E-ARK element. The column uses XML-syntax. [ ] defines where the value is placed.

3.1 Metadata elements regarding the information package

These are the elements that give information regarding the submission of the SIP.

Table 1: Metadata about the information package

Element	Definition	Explanation	Card.	METS
Identity	Identification of the package	A code that uniquely identifies the SIP and the digital object/objects being submitted. A UUID or GUID should be used to create globally unique identifiers. Example: *"UUID:550e8400-e29b-41d4-a716-446655440004"*	1	<mets: OBJID="[Identity]" >

Element	Definition	Explanation	Card.	METS
Description	Description of the package	Short text describing the package. Example: *"Example of SIP for delivery of personnel information"*	0..1	<mets: LABEL=”[Description]”>
OAIS type of package	SIP/AIP/AIC/AIU/DIP	Where in the OAIS model this package is placed. According to vocabulary vcMETS_TYPE. Example: *”SIP”*	1	<mets: TYPE=”[OAIS type of package]”>
Profile	Profile name	Describes the used METS-profile. The name should contain the version number as captured in the profile file paths. Example: *” http://eark-project.com/resources/METS/v01/SIP.xml”*	1	<mets: PROFILE="[Profile]"
Date and time	Time of creation of package	Date and time for creation of the package described according to the XML-standard. This timestamp states when the package and the package file was created. Example: *"2012-04-26T12:45:00+01:00"*	1	<metsHdr: CREATEDATE=”[Date and time]”>

Element	Definition	Explanation	Card.	METS
Status	Package status	A way of giving the status of the package. According to vocabulary[16] vcRECORDSTATUS Example: ”*NEW*”	0..1	<metsHdr: RECORDSTATUS=”[Status]”>
DocID	METS document ID	A unique identifier for the METS document itself may be different from the Identity given. A suggestion is to use the file name given to the METS-document. Example: “*SIP20150127.xml*”	0..1	<metsDocumentID>[DocID]
Contenttype	The delivery-specific content type the package belongs to	Gives the content type the package belongs to. According to vocabulary vcCONTENTTYPE. Example: ”*ERMS*” Example: *”Personnel”*	0..1	<altrecordID: TYPE=”CONTENTTYPE”>[Contenttype]

[16] All vocabularies referred in this table are described in the E-ARK METS Profile, http://www.ra.ee/METS/v01/SIP.xml

Element	Definition	Explanation	Card.	METS
Content type specification	Content type specification used for the content type	Name or reference to the used content type specification which details how the content submitted is described. Example: *E-ARK SIP ERMS, version 1* Example (preferred usage): *" http://eark-project.com/resources/SIARD/v01/SIARDDK20150127.pdf"*	0..1	<altrecordID: TYPE="CONTENTTYPESPECIFICATION">[Content type specification]
Submission agreement[17]	Reference to the used submission agreement	Complete reference to the submission agreement for the submission of the package Example: *RA 13-2011/5329; 2012-04-12* Example: *"http://submissionagreement.kb.se/dnr331-1144-2011/20120711/"*	0..1	<altrecordID: TYPE="SUBMISSIONAGREEMENT">[Submission agreement]

[17] Reference Model for an Open Archival Information System (OAIS), 2012, public.ccsds.org/publications/archive/650x0m2.pdf

Element	Definition	Explanation	Card.	METS
System type	System type used for the information found in the package	The system type used for creating the information. In some cases the content type and system type differ. The content type may state the submitted content to be a Records Managements System (RMS) and the system type may be “Noark”. This means that the RMS is based upon the Norwegian Noark standard. Example: ”*Noark 3*”	0..1	<altrecordID: TYPE=”SYSTEMTYPE”>[System type]
Previous submission agreement	The previous submission agreement(s) the information belongs to in the case the information is recorded.	Reference to the previous submission agreement(s) which the information belongs is recorded. Example: *”FM 12-2387/12726, 2007-09-19”*	0..*	<altrecordID: TYPE=”PREVIOUSSUBMISSIONAGREEMENT”>[Previous submission agreement]
Data Submission Session	The number or name of the whole submission	A single SIP can be part of a larger submission. This element gives an opportunity to state the name of the whole submission. Example: *”Submission, 2012-04-15 15:00”*	0..1	<altrecordID: TYPE=”DATASUBMISSIONSESSION”>[Data Submission Session]
Number in submission	A running number for the IP in the data submission session	It may be necessary to record the package number within the data submission. Example: *”SIP Number 2938”*	0..1	<altrecordID: TYPE=”PACKAGENUMBER”> [Number in submission]

Element	Definition	Explanation	Card.	METS
Archival reference code	Reference code in the archival description	It is possible to give a reference code indicating where, in the archival description, the package shall be placed. Example: *”SE/RA/123456/24/P”*	0..1	<altrecordID: TYPE=”REFERENCE CODE”>[Archival reference code]
Archives name	The archives name in the archival description	It is possible to state the name of the archives (fond) the delivery forms is a part of. Not to be confused with the ”Receiving institution name”. Example: *“Business archive”*	0..1	<altrecordID: TYPE=”ARCHIVESNAME”>[Archives name]
Previous reference code	An earlier used reference code in the archival description	In case the SIP is originating from other institutions maintaining a reference code structure this element can be used to record these reference codes and therefore support the provenance of the package. Example: *”SE/FM/123/123.1/123.1.3”*	0..*	<altrecordID: TYPE=”PREVIOUSREFERENCECODE”>[Previous reference code]
Appraisal	Information in the package that may or may not be appraised.	States on a high level whether the package contains information that can or should be appraised. The information that can be appraised needs to be identified in a lower level with the help of the content type information. Example: *”Yes”*	0..1	<altrecordID: TYPE= ”APPRAISAL”>[Appraisal]

Element	Definition	Explanation	Card.	METS
Access restrict	Does the package contain information with access restrictions	States whether the package contains information with access restrictions. According to vocabulary vcACCESSRESTRICT Example: *”Yes”* Example: *”No”*	0..1	<altrecordID: TYPE= ”ACCESSRESTRICT”>[Access restrict]
Start date	Start date for the delivered information	Sometimes it is necessary to give quick access to the start date of the delivered content without looking into the content itself. This will be redundant information found in the delivered information. Example: *”1995-01-01”*	0..1	<altrecordID: TYPE= ”STARTDATE”>[Start date]
End date	End date for the delivered information	Sometimes it is necessary to give quick access to the end date of the delivered content without looking into the content itself. This will be redundant information found in the delivered information. Example: *”2001-01-01”*	0..1	<altrecordID: TYPE= ”ENDDATE”>[End date]

Element	Definition	Explanation	Card.	METS
Information classification	Information classification for the package as a whole	The highest used information classification[18] for the delivered information. A locally defined classification schema may be used. Example: *”Confidential”* *Example:* *“Level 5”*	0..1	<altrecordID: TYPE= ”INFORMATIONCLASS”>[Information classification]
Archival creator	Name of archival creator	Name of the original creator (organisation) of the data being transferred. Please note that this might be different from the organisation which has been charged with preparing and sending the SIP to the archives. Example: *”The Swedish health agency”*	0..1	<agent: ROLE=”ARCHIVIST” TYPE= ”ORGANIZATION”> <name>[Archival creator]
Archival creator identification code	A unique identification code for the archival creator	A unique identification code for the archival creator. The code uses a prefix followed by a ”:” Prefix[19] according to vcTypeOfIdentificationCode. Example: *”VAT:SE201345098701”*	0..1	<agent: ROLE=”ARCHIVIST” TYPE= ”ORGANIZATION”> <note>[Archival creator identification code]

[18] Classified information, 2015, http://en.wikipedia.org/wiki/Classified_information
[19] All prefixes referred in this table are described in the E-ARK METS Profile, http://www.ra.ee/METS/v01/SIP.xml

Element	Definition	Explanation	Card.	METS
System name	Name of the system from which the information originated	Explains from which system the information originated. For example "Alfresco". If the system is commercial, the product name is given. Alternatively, internally developed programs may use the name employed within the organisation. Where no name is available, an explanatory text may be given instead. Example: *"Alfresco"*	0..1	<agent: ROLE="ARCHIVIST" TYPE="OTHER" OTHERTYPE= "SOFTWARE"> <name>[System name]
System Version	Version for the system	Gives the version of the system from which the information originates. Example: *"5.0.34"*	0..1	<agent: ROLE="ARCHIVIST" TYPE="OTHER" OTHERTYPE= "SOFTWARE"> <note>[System Version]
Delivering organisation name[20]	Name of the organisation delivering the package to the archive	Name of the organisation delivering the package to the archive. For stating and extending the information use of element "Producer organisation name" and "Submitting organisation name" is recommended. Example: *"The agency, Personnel*	0..1	<agent: ROLE="CREATOR" TYPE= "ORGANIZATION"> <name>[Delivering organisation name]

[20] All similar <agent> elements (the archival creator, delivering organisation, submitting organisation and producing organisation) may not be needed in the final SIP METS Profile. The decision will be done later in the E-ARK project

Element	Definition	Explanation	Card.	METS
Delivering organisation identification code	A unique identification code for the delivering organisation	A unique identification code for the delivering organisation. The code uses a prefix followed by a ":" Prefix according to vcTypeOfIdentificationCode. Example: *"HSA:SE2098109810-AF87"*	0..1	<agent: ROLE="CREATOR" TYPE= "ORGANIZATION"> <note>[Delivering organisation identification code]
Producer[21] organisation name	Name of the organisation producing the package	Name of the organisation or unit producing the submission package. Example: *"The agency, Records export function"*	0..1	<agent: ROLE="OTHER" OTHERROLE="PRODUCER" TYPE= "ORGANIZATION"> <name>[Producer organisation name]
Producer organisation identification code	A unique identification code for the producer organisation	A unique identification code for the producer organisation. The code uses a prefix followed by a ":" Prefix according to vcTypeOfIdentificationCode. Example: *"HSA:SE2098109810-AF87"*	0..1	<agent: ROLE="OTHER" OTHERROLE="PRODUCER" TYPE= "ORGANIZATION"> <note>[Producer organisation identification code]

[21] Please not to be confused with the "Archival creator organisation name" or "Submitting organisation name" – there may be cases when they are representing different organisations.

Element	Definition	Explanation	Card.	METS
Submitting organisation name	Name of the organisation submitting the package to the archive	Name of the organisation submitting the package to the archive. Extends the delivery information since it may be the case that the content of a creator is held by another part of the organisation. Example: *”The agency, Service function”*	0..1	<agent: ROLE=”OTHER” OTHERROLE=”SUBMITTER” TYPE= ”ORGANIZATION”> <name>[Submitting organisation name]
Submitting organisation identification code	A unique identification code for the submitting organisation	A unique identification code for the submitting organisation. The code uses a prefix followed by a ”:” Prefix according to vcTypeOfIdentificationCode. Example: *”HSA:SE2098109810-AF87”*	0..1	<agent: ROLE=”OTHER” OTHERROLE=”SUBMITTER” TYPE= ”ORGANIZATION”> <note>[Submitting organisation identification code]
Name of the organisation owning the intellectual rights to the information	Name of the organisation owning the intellectual rights to the submitted information	Name of the organisation owning the intellectual rights to the submitted information. Example: *”Ministry of Finance”*	0..1	<agent: ROLE=”IPOWNER” TYPE= ”ORGANIZATION”> <name>[Organisation owning the intellectual rights of the information]

Element	Definition	Explanation	Card.	METS
Identification code of the organisation owning the intellectual rights to the information	A unique identification code for the organisation owning the intellectual rights to the information	A unique identification code for the organisation owning the intellectual rights to the information. The code uses a prefix followed by a ”:” Prefix according to vcTypeOfIdentificationCode. Example: *”HSA:SE2098109810-AF87”*	0..1	<agent: ROLE=”IPOWNER” TYPE= ”ORGANIZATION”> <note>[Identification code of the organisation owning the intellectual rights to the information
Consultant name	Name of consultant who has helped in the creation of the package	Sometimes help is needed from a consultant in creating the package. Example: *The submission package creation advisor*	0..*	<agent: ROLE=”EDITOR” TYPE= ”ORGANIZATION”> <name>[Consultant name]
Consultant identification code	A unique identification code for the consultant	A unique identification code for the consultant. The code uses a prefix followed by a ”:” Prefix according to vcTypeOfIdentificationCode. Example: ”VAT:SE999999999901”	0..*	<agent: ROLE=”EDITOR” TYPE= ”ORGANIZATION”> <note>[Consultant identification code]

Element	Definition	Explanation	Card.	METS
Submitting system name	Name of the system that has created the package	It may be useful to record the name of the program/system that has created the package Example: *”SIP Creator”*	0..1	<agent: ROLE=”CREATOR” TYPE=”OTHER” OTHERTYPE= “SOFTWARE”> <name>[Submitting system name]
Submitting system version	Version of the system creating the package	It may be useful to record the version of the program/system creating the package Example: *”1.0”*	0..1	<agent: ROLE=”CREATOR” TYPE=”OTHER” OTHERTYPE= “SOFTWARE”> <note>[Submitting system version]
Contact person name	Contact person for the submission	It may be useful to record a contact person for the submission at the time of the submission. Example: *”Sven Svensson”*	0..*	<agent ROLE=”CREATOR” TYPE= ”INDIVIDUAL”> <name>[Contact person name]
Contact person contact information	Contact information for the contact person	Phone number and e-mail for the contact person. Example: *”08-12 34 56, sven.svensson@fm.se”*	0..*	<agent ROLE=”CREATOR” TYPE= ”INDIVIDUAL”> <note>[Contact person contact information]

Element	Definition	Explanation	Card.	METS
Receiving organisation name	Name of organisation receiving the package	Name of the receiver of a preservation package. Example: *”National Archives of Hungary”*	0..1	<agent ROLE= ”PRESERVATION” TYPE= ”ORGANIZATION”> <name>[Receiving organisation name]
Archive identification code	Identification code of organisation receiving the package	A unique identification code for the archive. The code uses a prefix followed by a ”:” Prefix according to vcTypeOfIdentificationCode. Example: *”ORG:2010340987”*	0..1	<agent ROLE= ”PRESERVATION” TYPE= ”ORGANIZATION”> <note>[Archive identification code]

More information can be added. The elements <altrecordID> and <agent> are repeatable. Use of more of these elements should be recorded in an extended profile.

Sometimes information needs to be added that does not fit into any given element. The suggestion is to do this in the METS-element amdSec sub-element techMD by following this structure:

```
<techMD ID="ID550e8400-e29b-41d4-a716-446655440000">
    <mdWrap MDTYPE="OTHER" OTHERMDTYPE="COMMENT">
        <xmlData>
            <comment:comment xmlns:comment="commentnamespace">
                [Informationstext]
            </comment:comment>>
        </xmlData>
    </mdWrap>
</techMD>
```

3.2 Metadata elements regarding submitted files

All files found in the submission package should be referenced once and only once (cardinality=1) in the METS-document describing the submission. The elements and attributes are the same regardless of the content type submitted.
When describing the content and documentation files in METS they are placed in the fileSec element in one or more fileGrp elements. The fileGrp element can be used for grouping files together in different ways. In this profile we do not group files in different groups, we only use the mandatory one fileGrp. Use of more fileGrp's must be decided in every implementation.

Table 2: Files metadata

Element	Definition	Explanation	Card.	METS
Identification of the file	Identification of the file object	A code that uniquely identifies the file in the METS-document. The code is suggested to use a prefix "ID" directly followed by an UUID or GUID or own stated identification code. ID follows the rules of to XML:ID. Example: *"ID550e8400-e29b-41d4-a716-4466554400bg"*	1	<file ID="[Identification of the file]">
File name	Name of the file	Name of the file and the path to locate it in the package. The file name must use the prefix file://. The attribute LOCTYPE is mandatory to describe how to find the file and uses a value list present in METS. Example: *"file:personnelexport.xml"*	1	<file <flocat: LOCTYPE="URL" xlink:href="[File name]" xlink:type="simple">
File name in original	The original name for the file	Sometimes the name of the file needs to be changed. The original name still needs to be available. Example: *"personal_avd_ö.xml"*	0..1	<file ext:originalfilename="[File name in original]"

Element	Definition	Explanation	Card.	METS
Date and time	Timestamp for the file	The timestamp seen on the file and used for validating the file. In most cases this is the last modification date. Described using the xsd:datetime rules. Example: *"2012-04-20T13:30:00+01:00"*	1	<file CREATED=”[Date and time]”>
MIME type[22]	Simplest way of describing a file type	The simplest way of describing a file type. Example: *"text/xml"*	1	<file MIMETYPE=”[MIME type]”>
File format name	Name of file format	Sometimes a more detailed name needs to be given to the file format Example: *"Extensible Markup Language”* Example: *”PDF/A”* Example: *”ISO/IEC 26300:2006”*	0..1 1 If File format name is used	<file ext:FILEFORMATNAME=”[File format name]”
File format version	Version of file format	The version of the file format Example: *"1.0"*	0..1	<file ext:FILEFORMATVERSION=”[File format version]”

[22] Media Types, 2015, https://www.iana.org/assignments/media-types/media-types.xhtml

Element	Definition	Explanation	Card.	METS
Format registry name	Name of the format registry identifying the file format	Name of the format registry identifying the file format According to vocabulary[23] vcTypeOfFormatregistry Example: *" PRONOM""*	0..1	<file ext:FORMATREGISTRY=”[Format registry name]”
Format key	Key of the file format in the registry	Key of the file format in the registry Example: *“fmt/101"*	0..1 1 If Format registry name is used	<file ext:FORMATREGISTRYKEY=”[Format key]”
File size	Size of the file in bytes	Size of the file in bytes. Example: *”8765324”*	1	<file SIZE=”[File size]”>
Function	Identifies the function of the file	Sometimes information of the function of the file is useful to record. A vocabulary for stating the “Function” is recommended in every implementation. Example: *”Submission file”*	0..1	<file: USE=”[Function]”>
Checksum type	Used algorithm for creating the checksum	Used algorithm for creating the checksum. Values predefined in METS. Example: *”SHA-256”*	0..1	<file: CHECKSUMTYPE=”[Checksum type]”>

[23] All vocabularies referred in this table are described in the E-ARK METS Profile, http://www.ra.ee/METS/v01/SIP.xml

Element	Definition	Explanation	Card.	METS
Checksum value	Calculated checksum for the file	Check sum for the file. Example: *”574b69cf71ceb5534c8a2547f5547d”*	0..1	<file: CHECKSUM=”[Checksum]”>
Transformation algorithm	Transformation algorithm used for transformation	Transformation algorithm used for transformation (decryption/encryption). Example: ”DES”	0..1	<file <transformFile TRANSFORM-TYPE=”decryption” TRANSFORM-ALGORITHM= ”[Transformation algorithm]”>
Transformation key	Transformation key for a transformed file	Transformation key for a transformed file. In attribute “TRANSFORMATIONTYPE” the transformation type according to predefined values are stated. Example: *”574b69cf71ceb5534c8a2547f5547d”*	0..1 1 If Transformation algorithm is used	<file <transformFile TRANSFORMTYPE=”decryption” TRANSFORMKEY= ”[Transformation key]”>

3.3 PREMIS metadata

Preservation metadata can be included in the METS package. It is recommended that PREMIS is used for preservation metadata. For further reading:

- More information about PREMIS can be found at: http://www.loc.gov/standards/premis/ .
- A guide on using PREMIS with METS may be found at: http://www.loc.gov/standards/premis/guidelines-premismets.pdf.
- Decisions made during the use of PREMIS can be recorded using this document: http://www.loc.gov/standards/premis/premis_mets_checklist.pdf

To include PREMIS in a METS profile the use of amdSec is to be preferred according to the guide referenced above. The rules for the amdSec are stated in the profile below. However, please note that preservation

metadata varies for different content types and therefore best practice guidelines should be intelligently interpreted and adapted as required. E-ARK will develop profiles for the key E-ARK content types (records systems, databases) and define how the submitted content-specific preservation metadata should be mapped to PREMIS.

Table 3: PREMIS metadata

Element	Definition	Explanation	Card.	METS
PREMIS metadata file	Metadata file in PREMIS format referenced in the METS document	Metadata file in PREMIS format referenced in the METS document	0..*	<amdSec> <digiProvMD> <mdRef: MDTYPE= ”PREMIS” xlink:href=”[PREMIS metadata file]” xlink:type=”simple”>
PREMIS metadata	Metadata in PREMIS format embedded in the METS document	Metadata in PREMIS format embedded in the METS document	0..*	<amdSec> <digiProvMD> <mdWrap: MDTYPE= ”PREMIS”> <xmlData> [PREMIS metadata in XML-format]

3.4 EAD and EAC-CPF metadata

Archival information can be included in the METS package. For the core interest groups in E-ARK – archival institutions – this information is delivered to the recipient in the formats EAD and EAC-CPF.
To include EAD and EAC-CPF in a METS profile the use of dmdSec is to be preferred according to the guide referenced above. The rules for the dmdSec are stated in the profile below.

Other metadata standards for description and administrative purposes can be used and referencing or embedding them must adhere to amdSec and dmdSec rules.

Table 4: EAD metadata

Element	Definition	Explanation	Card.	METS
EAD metadata file	Metadata file in EAD format referenced in the METS document	Metadata file in EAD format referenced in the METS document	0..*	<dmdSec> <mdRef: MDTYPE=”EAD” xlink:href=”[EAD-metadata file]” xlink:type=”simple”>
EAD metadata	Metadata in EAD format embedded in the METS document	Metadata in EAD format embedded[24] in the METS document	0..*	<dmdSec> <mdWrap: MDTYPE=”EAD”> <xmlData> [EAD metadata in XML-format]
EAC-CPF metadata file	Metadata file in EAC-CPF format referenced in the METS document	Metadata file in EAC-CPF format referenced in the METS document	0..*	<mdSec> <mdRef: MDTYPE= ”EAC-CPF” xlink:href=”[EAC-CPF metadata file]” xlink:type=”simple”>
EAC-CPF metadata	Metadata in EAC-CPF format	Metadata in EAC-CPF format	0..*	<dmdSec>

[24]Usually if EAD metadata has been embedded in the METS document once, it will be not referenced as EAD file again.

Element	Definition	Explanation	Card.	METS
	embedded in the METS document	embedded[25] in the METS document		<mdWrap: MDTYPE= ”EAC-CPF”> <xmlData> [EAC-CPF metadata in XML-format]

3.5 Structural map metadata

The element structMap is mandatory. In this SIP profile just a simple structMap is defined, but customised structMaps may be used and described in extending profiles of the archive.

```
<mets:structMap LABEL="Profilestructmap">
        <mets:div>
              <mets:fptr FILEID="ID550e8400-e29b-41d4-a716-4466554400ad"/>
        </mets:div>
</mets:structMap>
```

3.6 Content-type-specific content

As discussed above (Section 2), a SIP can contain content-type specific content. Types of data files and their structural relationships, and data models and metadata elements vary for different content-types. Metadata is submitted to an archive so that it can support functions in the archive. Metadata produced by a content-type specific system will variously be intended to support descriptive, structural, administrative, technical, preservation, provenance (relating to authenticity) and rights (relating to IP, retention and access) functions. The METS standard used in the E-ARK SIP specification does not offer *one* single structure in which the content-type specific metadata could be stored as a whole. In order to efficiently use the metadata to support archival functions the SIP defines separate SIP METS sections as containers for the various metadata functions, such as the METS header for package management, the dmdSec for EAD and other descriptive metadata standards, the amdSec for preservation (PREMIS), technical (MIX) and other functions. In order to use the submitted metadata it has to be extracted into the standards used in the SIP

[25] Usually if EAC-CPF metadata has been embedded in the METS document once, it will be not referenced as EAD file again.

METS sections.[26] To do this the content-type specific metadata elements need to be mapped to those containers and implemented in the agreed standards. Therefore, complementary E-ARK SIP metadata profiles will be defined for the key E-ARK content types (records systems, databases) that implement the E-ARK SIP for these specific content-types. They will define how the submitted content-specific preservation metadata should be mapped to the E-ARK SIP structure.

The content-type specific content can, for example, be exported from a MoReq2010 compliant records management system as XML file. One such case is presented in Appendix B: MoReq2010 records management metadata example. This example is based on MoReq2010 metadata originating from The National Archives of Hungary.

The second example represents a database encapsulated in the E-ARK SIP. This example approach is seen in Figure 6 in Appendix C: SIARD SIP catalogue structure example. This example structure presumes that the database is previously exported in the SIARD-E format (a harmonised format for database archiving based on SIARD).

[26] In the case of descriptive metadata it is even additionally integrated with external systems, such as the catalogue in order to support external access to the archive. One might also want to do that for other metadata (e.g. technical or preservation) in order to ease management of the archive.

4. CONCLUSIONS

This document describes the proposed general structure for submission information packages by explaining how the E-ARK SIP can be generally constructed by following the common rules developed for all other (archival, dissemination) information packages.

The central section describes the metadata sections and elements in these sections.

It also describes the example structure for the E-ARK SIP and shows how it can manage various content and metadata.

Additionally, preliminary example of using MoReq2010 metadata in the E-ARK SIP METS profile is introduced on page 39 and another example for databases on page 42.

The work introduced in this deliverable is a first draft – a fully mature E-ARK SIP specification focusing on pilot projects will be delivered in E-ARK deliverable D3.3.

5. REFERENCES

1. A Checklist for Documenting PREMIS-METS Decisions in a METS Profile, 2010, URL: http://www.loc.gov/standards/premis/premis_mets_checklist.pdf
2. Database Preservation Toolkit, 2015, URL: http://keeps.github.io/db-preservation-toolkit/
3. Classified information, 2015, URL: http://en.wikipedia.org/wiki/Classified_information
4. E-ARK Report on Available Best Practices, 2014, URL: http://eark-project.com/resources/project-deliverables/6-d31-e-ark-report-on-available-best-practices
5. e-SENS (Electronic Simple European Networked Services) project, 2015, URL: http://www.esens.eu/
6. Encoded Archival Context for Corporate Bodies, Persons, and Families, 2015, URL: http://eac.staatsbibliothek-berlin.de/
7. Executive_Order_on_Submission_Information_Packages, 2010, URL: http://www.sa.dk/media%283367,1033%29/Executive_Order_on_Submission_Information_Packages.pdf
8. FGS packet structure, 2013, URL:https://riksarkivet.se/Media/pdf-filer/Projekt/FGS_Earkiv_Paket.pdf
9. Guidelines for using PREMIS with METS for exchange, Revised September 17, 2008 URL: http://www.loc.gov/standards/premis/guidelines-premismets.pdf.
10. Media Types, 2015, URL: https://www.iana.org/assignments/media-types/media-types.xhtml
11. METS, 2015, URL: http://www.loc.gov/standards/mets/
12. METS Profile Components, 2011, URL: http://www.loc.gov/standards/mets/profile_docs/components.html
13. METS Profiles, 2012, URL: http://www.loc.gov/standards/mets/mets-profiles.html
14. Planets project, 2007, URL: http://www.planets-project.eu/
15. Producer, Submission Agreements: Glossary of Terms, 2015, URL: http://sites.tufts.edu/dca/about-us/research-initiatives/taper-tufts-accessioning-program-for-electronic-records/project-documentation/submission-agreements-glossary-of-terms/
16. Producer-Archive Interface Methodology Abstract Standard (PAIMAS), 2004, URL: public.ccsds.org/publications/archive/651x0m1.pdf
17. Producer-Archive Interface Specification (PAIS) – CCSDS, 2014, URL: public.ccsds.org/publications/archive/651x1b1.pdf
18. Records Creator, Submission Agreements: Glossary of Terms, 2015, URL: http://sites.tufts.edu/dca/about-us/research-initiatives/taper-tufts-accessioning-program-for-electronic-records/project-documentation/submission-agreements-glossary-of-terms/
19. Reference Model for an Open Archival Information System (OAIS), 2012, URL: public.ccsds.org/publications/archive/650x0m2.pdf
20. SIARD Specification, 2015, URL: http://www.bar.admin.ch/dienstleistungen/00823/01911/index.html

6. APPENDICIES

Appendix A: Criteria for a submission information package

Every SIP should follow the quality critery described below:

9. The structure and content of the SIP must be able to be validated:
    a. integrity, fixity (are all folders and files intact and at the right place)
    b. syntax (is the format of the information content correct)
    c. semantics (manual or semi-automatically)
10. The SIP format should allow efficient processing.
    a. huge information content must be able to be split and be spanned over many SIPs
    b. The SIP size is limited due to limitations on:
        i. capacity on removable media (also applies for archival storage on single media)
        ii. efficient network transfer
        iii. filesystem limitations, especially for number of files per folder
11. The SIP format should be expandable for the addition of new metadata standards or data formats.
12. The SIP format should be suitable for allowing conversion of the information content into new data formats.
13. The SIP format should allow including metadata for the formats for information content.
    a. shall be able to be extended to support any content type a digital repository needs to ingest
    b. shall allow separate content type descriptions to be specified for different types of submissions
14. The SIP format must be as simple as possible.
15. Data and metadata could be separated to ease the change of data formats or metadata formats.
16. Metadata could be separated into several files to ease the change of metadata formats and to limit the time spend on handling metadata.

Appendix B: MoReq2010 records management metadata example

MoReq2010 SIP example catalogue structure consist of folders and computer files which are all grouped as TAR (Tape ARchive) file. The structure (without compression or packaging) is seen Figure 5.

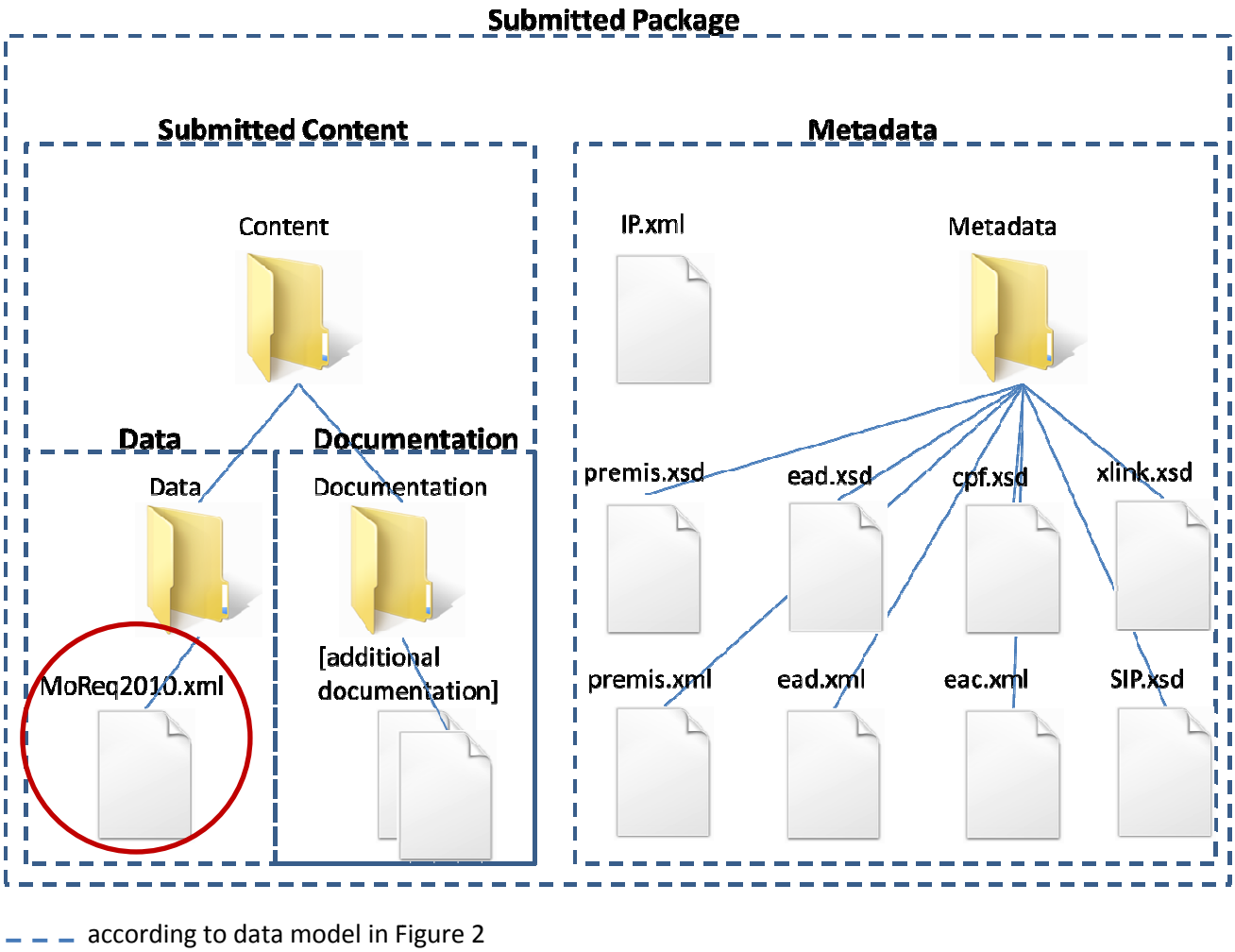

Figure 5: Catalogue structure of MoReq2010 SIP example

The Content folder contains the folders Data and Documentation.

The Data folder can contain delivery-specific content (e.g. MoReq2010) in XML format. Using the MoReq2010 content doesn't restrict extraction of some of the metadata elements from an XML file and incorporating them in ead.xml or in some other description file represented in the E-ARK SIP.

The Documentation folder can contain extra documentation (e.g. separate PDF/A file) about content.

The file ip.xml is the information package description based on the SIP METS profile.

The Metadata folder contains metadata (usually added during the SIP creation process) in XML format and related XML schemas. This folder contains also an XML schema (SIP.xsd) for an (submission) information package. Including schema to the SIP package allows validating the SIP even without Internet connection.

Appendix C: SIARD SIP catalogue structure example

The SIARD-E SIP example catalogue structure consists of folders and computer files which can be grouped in a container such as a TAR (Tape ARchive) file. The structure (without compression or packaging) is seen in Figure 6.

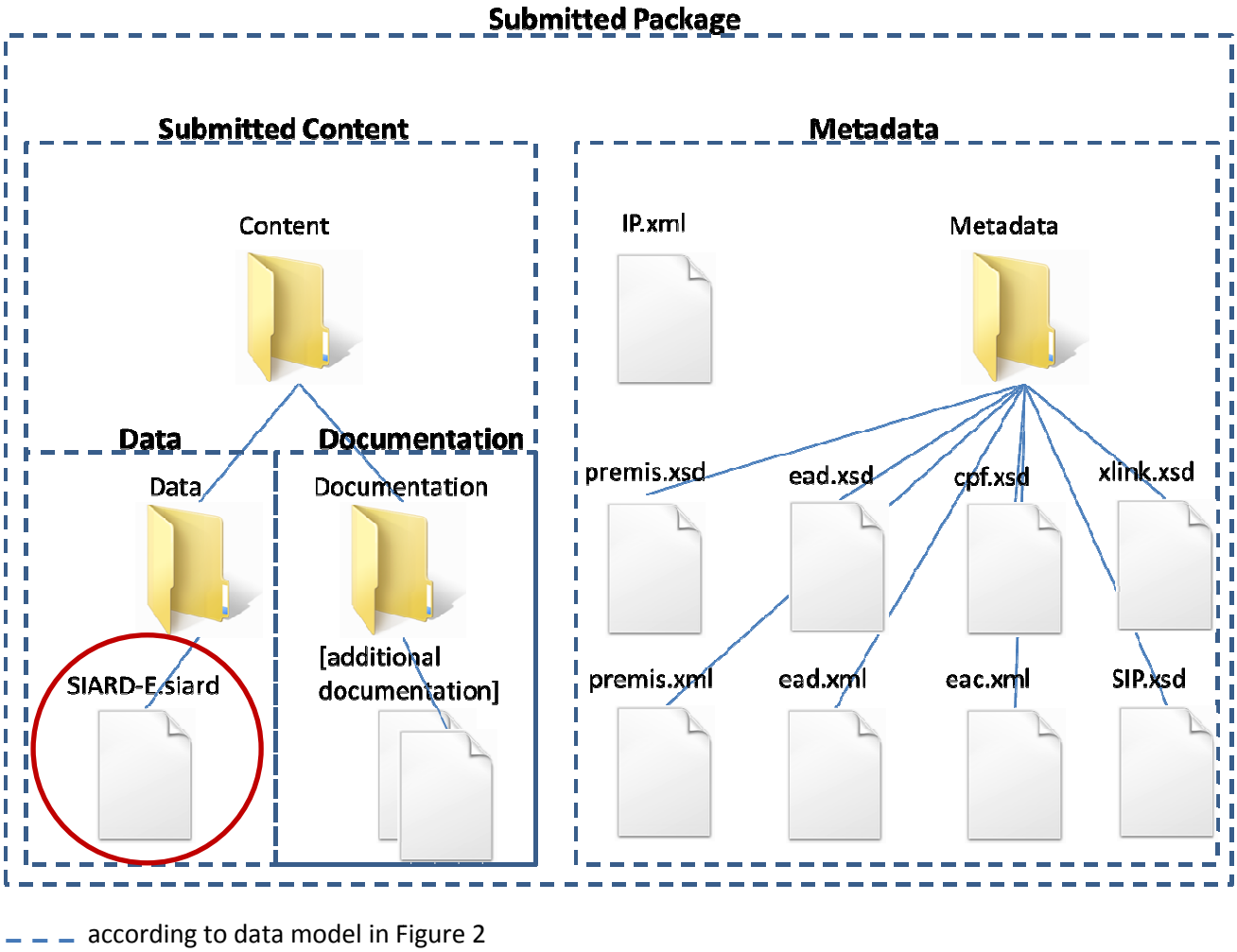

Figure 6: Catalogue structure of SIARD-E SIP example

The Content folder contains the folders Data and Documentation.

The Data folder can contain delivery-specific content (e.g. a database representation) in XML or a similar format.

For databases the E-ARK project will recommend and deliver extended SIARD (work name: SIARD-E) format, a common harmonised format for archiving databases. It is based on the SIARD[27] format recommended by

[27] SIARD Specification, http://www.bar.admin.ch/dienstleistungen/00823/01911/index.html

the EU project PLANETS[28] and harmonised with the SIARDDK[29] format used by the Danish National Archives and the DBML[30] format used by the Portuguese National Archives.

The SIARD-E.siard file (or subfolder structure for large databases) is in the extended SIARD format. Using the extended SIARD format doesn't restrict extraction of some of the metadata elements from an *.siard file and incorporating them in ead.xml or in some other description file represented in the E-ARK SIP.

The Documentation folder can contain extra documentation (e.g. a separate PDF/A file) about content data.

The file ip.xml is the information package description based on the SIP METS profile.

The Metadata folder contains metadata in XML format and related XML schemas. This folder contains also an XML schema (SIP.xsd) for an (submission) information package.

[28] Planets project, http://www.planets-project.eu/
[29] SIARDDK, The Executive Order on Submission Information Packages
http://www.sa.dk/media%283367,1033%29/Executive_Order_on_Submission_Information_Packages.pdf
[30] Database Preservation Toolkit, http://keeps.github.io/db-preservation-toolkit/

Appendix D: Metadata for creating profiles

The next table shows the elements which can be used for creating METS profiles.

Description marked with "*"are taken from Schema Documentation http://www.loc.gov/standards/mets/profile_docs/mets.profile.v2-0.html.

Table 5: Metadata elements

No	Name	Description	Minimal occurrence	Maximum occurrence
1.	**<URI>**	Every METS profile must be assigned a unique URI RFC 2396 by the institution responsible for its creation. This URI must have the attribute ASSIGNEDBY="local". If the profile is to be publicly registered at the Library of Congress Network and MARC Standards Office, the URI must not duplicate any URI assigned to any currently registered profile. A second URI element will be added if and when a profile is registered at the Library of Congress. This URI will have the attribute ASSIGNEDBY="metsboard".*	1	∞
2.	**<title>**	The profile must contain a short, human readable string describing the class of METS objects being profiled (e.g., "NYU Monograph", "Stereographic Image", "SCORM object", etc.). This title must not exceed 256 characters.*	1	∞
3.	**<abstract>**	The profile must contain a one-paragraph description of the profile's nature and purpose. This abstract must not exceed 2048 characters in length.*	1	∞
4.	**<date>**	The profile must include the date and time the profile was created by the responsible institution.*	1	1
5.	**<contact>**	The profile must contain contact information for an individual or entity responsible for the profile's	1	∞

No	Name	Description	Minimal occurrence	Maximum occurrence
		creation and who may be contacted for clarification of the contents of the profile. Contact information must include a mailing address that can be used to contact someone regarding the profile, and may also include the name of a specific individual to contact for information regarding the profile, the name of the institution responsible for creating the schema, a phone number for a contact individual and an e-mail address.*		
6.	<contact> **<name>**	The name of an individual who may be contacted regarding the profile.*	1	1
7.	<contact> **<institution>**	The name of an institution which may be contacted regarding the profile.*	1	1
8.	<contact> **<address>**	The address for the individual and/or institution which may be contacted regarding the profile.*	1	1
9.	<contact> **<email>**	An e-mail address which people may contact for information regarding the profile.*	1	1
10.	**registration_info**	A date when the profile was registered with the Library of Congress, and a URI for obtaining the registered profile from the Library (this will be provided by the registrars and need not be included in profiles being submitted). *	0	1
11.	**<related_profile>**	A profile may indicate its relationship with other METS profiles. METS profiles are not explicitly versioned, as implementations may exist that use older editions of METS profiles. Therefore a new version of a profile must be registered as a new profile. In this case, the RELATIONSHIP attribute should be used to indicate that a profile supersedes a profile already registered with the Library of Congress Network Development and MARC Standards Office. For each related profile,	1	∞

No	Name	Description	Minimal occurrence	Maximum occurrence
		the profile should specify a URI for the related profile and the nature of the relationship between the current profile and the related profile.*		
12.	**<profile_content>**	A profile may document the context in which it operates through the inclusion of a Resource Model and statement of Use Cases.*	1	1
13.	<profile_content> **<resource_model>**	This section is made up of simple text describing what the objects are that the Profile deals with, and could include information such as whether the profile is intended for an OAIS SIP, AIP, or DIP. Available are an optional head sub-element and some available text formatting elements from the xhtml namespace. Available xhtml elements include p, various list/item elements, text formatting such as b and i, a, and img.*	0	1
14.	**<external_schema>**	A profile which will be registered with the Network Development and MARC Standards Office must identify all external schema which may be used in constructing a METS document conforming with the profile. External schema for registered profiles MUST be publicly available. The schema must be identified in sufficient detail to allow a document author previously unfamiliar with the schema to unambiguously identify and retrieve it. Those registering profiles with the Network Development and MARC Standards Office are strongly encouraged to include a URI for each identified external schema which may be used to retrieve that schema from any Internet workstation, and may wish to include the complete text of any required external schema as an appendix to their profile. A single external_schema element should	1	∞

No	Name	Description	Minimal occurrence	Maximum occurrence
		be used for each external schema listed in the profile. Multiple external_schema elements may be used within a single profile. For each external schema described, you may provide a name of the schema, a URI assigned to the schema, a context description indicating where in a conforming METS document the schema may be used, and an additional note. The note element has an optional head sub-element and some available text formatting elements from the xhtml namespace. Available xhtml elements include p, various list/item elements, text formatting such as b and i, a, and img.*		
15.	<external_schema> **<name>**	A name of the external schema.	0	1
16.	<external_schema> **<URL>**	A URL of the external schema.	0	∞
17.	<external_schema> **<context>**	A content description of the external schema.	0	∞
18.	<external_schema> **<note>**	A note for recording any additional information regarding the external schema.	0	1
19.	<external_schema> <note> **<head>**	A heading of the additional information section for the external schema.	0	1
20.	**<description_rules>**	An institution can employ particular rules of description when encoding text within elements and attributes of a METS document. For example, a library might decide that descriptive metadata within a <dmdSec><mdWrap> section will be encoded using the Library of Congress' MARC 21 XML Schema MARCXML, and that the content of all elements and attributes within the MARC 21 XML	1	1

No	Name	Description	Minimal occurrence	Maximum occurrence
		sections must be prepared in accordance with the Anglo-American Cataloguing Rules 2nd Edition AACR2. The Rules of Description portion of a METS profile for that institution's METS objects should indicate that AACR2 must be applied to all content within a MARC 21 XML Schema portion of a METS document conforming to that profile. The description_rules element has an optional head sub-element and some available text formatting elements from the xhtml namespace. Available xhtml elements include p, various list/item elements, text formatting such as b and i, a, and img.*		
21.	**<controlled_vocabularies>**	An institution may choose to employ certain controlled vocabularies, such as the Library of Congress Subject Headings or the Getty Thesaurus of Geographic Names, for the content of elements within portions of a METS document. If use of a particular controlled vocabulary is mandatory in any section of a conforming METS document, that controlled vocabulary must be listed in this section of a profile. For all such controlled vocabularies, you should provide a name for the controlled vocabulary, an agency responsible for the vocabulary's maintenance, and a URI assigned to the vocabulary. If you desire, you may also include the individual values/terms within the controlled vocabulary, although it is anticipated that this will only be done when you wish to publicize the contents of a locally-produced controlled vocabulary to others who wish to produce conforming METS documents; there is no need to	1	1

No	Name	Description	Minimal occurrence	Maximum occurrence
		itemize the contents of well-known controlled vocabularies such as LCSH. For all controlled vocabularies, you should provide contextual information indicating where within a conforming METS document the vocabulary may be used, and if desired brief description of the vocabulary and its purpose. The context and description sub-elements of vocabulary have an optional head sub-element and some available text formatting elements from the xhtml namespace. Available xhtml elements include p, various list/item elements, text formatting such as b and i, a, and img.*		
22.	<controlled_vocabularies> **<vocabulary>**	Selected lists of keywords and phrases, which are used to tag units of information packages.	0	∞
23.	<controlled_vocabularies> <vocabulary> **<name>**	Name of the controlled vocabulary.	0	1
24.	<controlled_vocabularies> <vocabulary> **<maintenance_agency>**	Name of an agency who maintains the vocabulary.	0	1
25.	<controlled_vocabularies> <vocabulary> **<values>**	Individual values/terms within the controlled vocabulary.	0	1
26.	<controlled_vocabularies> <vocabulary> **<context>**	Contextual information indicating where within a conforming METS document the vocabulary may be used.	0	∞
27.	<controlled_vocabularies> <vocabulary> **<description>**	A brief description of the vocabulary and its purpose.	0	1
28.	<controlled_vocabularies> <vocabulary>	Optional element that designates the heading for a section of text.	0	1

No	Name	Description	Minimal occurrence	Maximum occurrence
	<description> **<head>**			
29.	**<structural_requirements>**	The METS document structure is extraordinarily flexible; that flexibility may be problematic inasmuch as creating software to process any arbitrary METS document in any but the most rudimentary way (XML parsing and validation) is a non-trivial task. This task can be simplified to some degree if those creating software to process METS documents know that there are further constraints on the structure of a METS document beyond those of the METS schema itself. The structural requirements portion of a METS profile allows an institution to delineate additional restrictions on the structure of a conforming METS document beyond those specified by the METS format itself. It is permissible to specify restrictions on the structure of a conforming METS document which cannot be validated by standard XML validation tools. For example, it would be a permissible restriction to state that master still images within a METS document should be contained within a separate file group from derivative images.*	1	1
30.	<structural_requirements> **<metsRootElement>**	Root element of METS.	0	1
31.	<structural_requirements> <metsRootElement> **<requirement>**	The requirement element has four attributes: 1. an optional XML ID attribute, 2. an optional IDREFS attribute called RELATEDMAT, which you may use to indicate other portions of the profile document where this particular requirement is relevant. Requirement elements are in turn composed of a sequence of	1	∞

No	Name	Description	Minimal occurrence	Maximum occurrence
		paragraph <p> elements, 3. an optional IDREFS attribute called EXAMPLES, which you may use to point to examples in the <Examples> section that demonstrate the requirement, and 4. an optional REQLEVEL attribute with values drawn from RFC 2119 (http://www.ietf.org/rfc/rfc2119.txt) . • MUST: This word means that the definition is an absolute requirement. • SHOULD: This word means that there may exist valid reasons in particular circumstances to ignore the requirement, but the full implications must be understood and carefully weighed before choosing a different course. • MUST NOT: This phrase means that the prohibition described in the requirement is an absolute prohibition of the profile. • SHOULD NOT: This phrase means that there may exist valid reasons in particular circumstances when violating the prohibition described in the requirement is acceptable or even useful, but the full implications should be understood and the case carefully weighed before doing so. The requirement text should clarify such circumstances. • MAY: This word means that an item is not prohibited but fully optional. The requirement element has two children: description and tests. The required description		

No	Name	Description	Minimal occurrence	Maximum occurrence
		element has an optional head sub-element and some available text formatting elements from the xhtml namespace. Available xhtml elements include p, various list/item elements, text formatting such as b and i, a, and img.*		
32.	<structural_requirements> <metsRootElement> <requirement> **<description>**	A brief description of the requirement.	0	1
33.	<structural_requirements> <metsRootElement> <requirement> <description> **<head>**	Optional element that designates the heading for a section of text.	0	1
34.	<structural_requirements> <metsRootElement> <requirement> **<tests>**	The optional tests subelement of description exists to supplement a textual description of a requirement with one that can be used to eventually provide machine validation. It includes one or more testGrp elements, which in turn contain testWrap or testRep elements. It is assumed each test's desired outcome is to evaluate to "true". Implementers must ensure that the requirement level indicated and any formal test evaluation pattern are aligned so that the test indicates conformance rather than an error.*	0	1
35.	<structural_requirements> **<metsHdr>**	METS header element.	0	1
36.	<structural_requirements> <metsHdr> <requirement> **<description>**	See the description of the <metsRootElement> element.	0	1
37.	<structural_requirements>	See the description of the <metsRootElement>	1	∞

No	Name	Description	Minimal occurrence	Maximum occurrence
	<metsHdr> **<requirement>**	element.		
38.	<structural_requirements> <metsHdr> <requirement> <description> **<head>**	See the description of the <metsRootElement> element.	0	1
39.	<structural_requirements> <metsHdr> <requirement> **<tests>**	See the description of the <metsRootElement> element.	0	1
40.	<structural_requirements> **<dmdSec>**	Descriptive metadata section.	0	1
41.	<structural_requirements> <dmdSec> **<requirement>**	See the description of the <metsRootElement> element.	1	∞
42.	<structural_requirements> <dmdSec> <requirement> **<description>**	See the description of the <metsRootElement> element.	0	1
43.	<structural_requirements> <dmdSec> <requirement> <description> **<head>**	See the description of the <metsRootElement> element.	0	1
44.	<structural_requirements> **<dmdSec>** <requirement> **<tests>**	See the description of the <metsRootElement> element.	0	1
45.	<structural_requirements> **<amdSec>**	Administrative metadata section.	0	1
46.	<structural_requirements>	See the description of the <metsRootElement>	1	∞

No	Name	Description	Minimal occurrence	Maximum occurrence
	<amdSec> **<requirement>**	element.		
47.	<structural_requirements> <amdSec> <requirement> **<description>**	See the description of the <metsRootElement> element.	0	1
48.	<structural_requirements> <amdSec> <requirement> <description> **<head>**	See the description of the <metsRootElement> element.	0	1
49.	<structural_requirements> <amdSec> <requirement> **<tests>**	See the description of the <metsRootElement> element.	0	1
50.	<structural_requirements> **<fileSec>**	File section.	0	1
51.	<structural_requirements> <fileSec> **<requirement>**	See the description of the <metsRootElement> element.	1	∞
52.	<structural_requirements> <fileSec> <requirement> **<description>**	See the description of the <metsRootElement> element.	0	1
53.	<structural_requirements> <fileSec> <requirement> <description> **<head>**	See the description of the <metsRootElement> element.	0	1
54.	<structural_requirements> **<fileSec>** <requirement>	See the description of the <metsRootElement> element.	0	1

No	Name	Description	Minimal occurrence	Maximum occurrence
	<tests>			
55.	<structural_requirements> **<structMap>**	Structural map section.	0	1
56.	<structural_requirements> <structMap> **<requirement>**	See the description of the <metsRootElement> element.	1	∞
57.	<structural_requirements> <structMap> <requirement> **<description>**	See the description of the <metsRootElement> element.	0	1
58.	<structural_requirements> <structMap> <requirement> <description> **<head>**	See the description of the <metsRootElement> element.	0	1
59.	<structural_requirements> <structMap> <requirement> **<tests>**	See the description of the <metsRootElement> element.	0	1
60.	<structural_requirements> **<structLink>**	Structural linking section.	0	1
61.	<structural_requirements> <structLink> **<requirement>**	See the description of the <metsRootElement> element.	1	∞
62.	<structural_requirements> <structLink> <requirement> **<description>**	See the description of the <metsRootElement> element.	0	1
63.	<structural_requirements> <structLink> <requirement> <description>	See the description of the <metsRootElement> element.	0	1

No	Name	Description	Minimal occurrence	Maximum occurrence
	<head>			
64.	<structural_requirements> <structLink> <requirement> **<tests>**	See the description of the <metsRootElement> element.	0	1
65.	<structural_requirements> **<structSec>**	Structural section.	0	1
66.	<structural_requirements> <structSec> **<requirement>**	See the description of the <metsRootElement> element.	1	∞
67.	<structural_requirements> <structSec> <requirement> **<description>**	See the description of the <metsRootElement> element.	0	1
68.	<structural_requirements> <structSec> <requirement> <description> **<head>**	See the description of the <metsRootElement> element.	0	1
69.	<structural_requirements> **<structSec>** <requirement> **<tests>**	See the description of the <metsRootElement> element.	0	1
70.	<structural_requirements> **<multiSection>**	Element for describing multiple sections.	0	1
71.	<structural_requirements> <multiSection> **<requirement>**	See the description of the <metsRootElement> element.	1	∞
72.	<structural_requirements> <multiSection> <requirement> **<description>**	See the description of the <metsRootElement> element.	0	1

No	Name	Description	Minimal occurrence	Maximum occurrence
73.	<structural_requirements> <multiSection> <requirement> <description> **<head>**	See the description of the <metsRootElement> element.	0	1
74.	<structural_requirements> <multiSection> <requirement> **<tests>**	See the description of the <metsRootElement> element.	0	1
75.	**<technical_requirements>**	A METS document may reference a variety of external files, including the content files for the METS object (via <FLocat>elements), executable behaviors (via the <mechanism> element), and external metadata files (via <mdRef> elements). Non-XML content and metadata files may also be embedded within a METS instance, if they have been Base64 encoded. Institutions may wish to place restrictions on the nature of these external and non-XML files, such as insisting that all image files be in the TIFF 6.0 format and have a bit-depth between 16 and 32 bits, or that references to external metadata identified as being of type "MARC" via the MDTYPE attribute will point to MARC records conforming to the MARC 21 standard (or alternatively, to an HTML display of a MARC 21 record). The Technical Requirements section of a profile allows institutions to set forth the full set of restrictions on the technical nature of files which may be referenced from a conforming METS document. It is subdivided into sections for restrictions on content files, restrictions on behavior files, and restrictions on	1	1

No	Name	Description	Minimal occurrence	Maximum occurrence
		metadata files. Profile authors should bear in mind that one of the primary purposes of the Technical Requirements section is to allow software developers to anticipate what types of content will be accessible via links from the METS objects, and hence what software is needed to process that content. Every sub-element within the technical_requirements section is composed of a sequence of individual requirement elements.*		
76.	<technical_requirements> **<content_files>**	Content files associated with that division.	0	1
77.	<technical_requirements> <content_files> **<requirement>**	See the description of the <metsRootElement> element.	1	∞
78.	<technical_requirements> <content_files> <requirement> **<description>**	A description of the requirement.	0	1
79.	<technical_requirements> <metsRootElement> <requirement> <description> **<tests>**	See the description of the <metsRootElement> element.	0	1
80.	<technical_requirements> **<behavior_files>**	Associate executable behaviors with content.	0	1
81.	<technical_requirements> <behavior_files> **<requirement>**	See the description of the <metsRootElement> element.	1	∞
82.	<technical_requirements> <behavior_files> <requirement> **<description>**	A description of the requirement.	0	1

No	Name	Description	Minimal occurrence	Maximum occurrence
83.	<technical_requirements> **<metadata_files>**	Requirements about metadata files.	0	1
84.	<technical_requirements> <metadata_files> **<requirements>**	See the description of the <metsRootElement> element.	1	∞
85.	<technical_requirements> <metadata_files> <requirements> **<description>**	A description of the requirement.	0	1
86.	<technical_requirements> <metadata_files> <requirements> <description> **<tests>**	See the description of the <metsRootElement> element.	0	1
87.	**<tool>**	A profile should provide a description of any affiliated tools, including validators, stylesheets, authoring tools, rendering applications, which can or should be used with METS documents conforming to the profile. The description should provide a name for the tool, the agency responsible for its development, a description of the tool, and a URI for obtaining the tool or further information regarding it. The description and note sub-elments of tool have an optional head sub-element and some available text formatting elements from the xhtml namespace. Available xhtml elements include p, various list/item elements, text formatting such as b and i, a, and img.*	1	∞
88.	<tool> **<name>**	A name of the tool.	0	1
89.	<tool>	A name of an agency which is responsible for the	0	1

No	Name	Description	Minimal occurrence	Maximum occurrence
	<agency>	development of this tool.		
90.	<tool> **<URI>**	A URI of the tool.	0	∞
91.	<tool> **<description>**	A description of the tool.	0	1
92.	<tool> **<note>**	A note about the tool.	0	1
93.	**<Example>**	A METS profile may contain one or more Example elements. Unlike Appendix elements, which are expected to contain complete METS encodings, an Example element would contain just a code snippet intended to demonstrate some requirement of the profile. The data type of the Examples element allows for partial, well-formed encodings that still might not satisfy all of the requirements for a full METS encoding. A requirement can link to one or more Example elements by referencing the ID value of the pertinent Example elements in its EXAMPLES attribute. The Example element has two attributes: 1. ID. This attribute is required, as every Example should be linked to from the requirement of which it is an example. 2. LABEL. This optional attribute allows for the descriptive labeling of an example.*	0	∞
94.	**<Appendix>**	A profile must contain an appendix containing an example METS document which conforms to the requirements set out in the profile. Profile authors should note that in order to insure that the completed profile document is valid, any namespace and schemaLocation declarations contained in the root <mets> element should be	1	∞

No	Name	Description	Minimal occurrence	Maximum occurrence
		moved to the root <METS_Profile> element.*		

Appendix E: Terminology

Archive*	An Organisation that intends to preserve information for Access and use by a Designated Community.
Content type	The delivery-specific content submission type.
ERMS	Is a type of content management system and refers to the electronic records management system.
Information Package*	A logical container composed of optional Content Information and optional associated Preservation Description Information. Associated with this Information Package is Packaging Information used to delimit and identify the Content Information and Package Description information used to facilitate searches for the Content Information.
Ingest Functional Entity*	The OAIS functional entity that contains the services and functions that accept Submission Information Packages from Producers, prepares Archival Information Packages for storage, and ensures that Archival Information Packages and their supporting Descriptive Information become established within the OAIS.
OAIS*	The Open Archival Information System is an archive (and a standard: ISO 14721:2003), consisting of an organisation of people and systems that has accepted the responsibility to preserve information and make it available for a Designated Community.
Submission Information Package (SIP)*	An Information Package that is delivered by the Producer to the OAIS for use in the construction or update of one or more AIPs and/or the associated Descriptive Information.
Submitting organisation	Name of the organisation submitting the package to the archive. Extends the delivery information since it may be the case that the content of a creator is held by another part of the organisation.
Delivering organisation	The organisation delivering the package to the archive. For stating and extending the information use of element “Producer organisation name” and “Submitting organisation name” is recommended.
Archival creator[31]	An organization unit or individual that creates records and/or manages those records during their active use.
Producing organisation[32]	The organizational unit or individual that has the authority to transfer records to an archive. Usually the producer is also the records creator, the organizational unit or

[31] Records Creator, Submission Agreements: Glossary of Terms, 2015, http://sites.tufts.edu/dca/about-us/research-initiatives/taper-tufts-accessioning-program-for-electronic-records/project-documentation/submission-agreements-glossary-of-terms/

	individual that created and managed the records during their active use. This is not always the case, sometimes the producer is different from the records creator. For example: An author dies and her literary executor gains the authority to transfer her papers to an archive. The author is the records creator and the literary executor is the producer. For example: Department X gets reorganized out of existence and Department Y, which takes over the functional responsibilities of Department X, gains the authority to transfer the records of Department X to the archive. Department X is the records creator and Department Y is the producer. Counter example: The Department of Widget Science transfers some of its own records to the archive. The Department of Widget Science is the records creator and the producer.

***** Reference Model for an Open Archival Information System (OAIS), 2012, http://public.ccsds.org/publications/archive/650x0m2.pdf

[32] Producer, Submission Agreements: Glossary of Terms, 2015, http://sites.tufts.edu/dca/about-us/research-initiatives/taper-tufts-accessioning-program-for-electronic-records/project-documentation/submission-agreements-glossary-of-terms/