

Spezifikation Matterhorn METS

Stand 11.7.2016

Einleitung	2
Umsetzung des OAIS Informationsmodells	3
Was wird durch das Matterhorn METS-Modell beschrieben?	5
Struktur des Objekts	6
Verwendung von METS.....	7
Abbildung der Objektstruktur in METS	7
METS Header <metsHdr>	7
Descriptive Metadata Section <dmdSec>.....	8
Administrative Metadata Section <amdSec>.....	8
File Section <fileSec>	9
Structure Map <structMap>	9
Verwendung von PREMIS.....	11
PREMIS and the package structure of digital objects	11
PREMIS Identifiers.....	11
PREMIS object.....	12
PREMIS event	14

Einleitung

Das OAIS-Informationsmodell (ISO 14721)¹ beschreibt, welche Daten und Metadaten für die langfristige Nutzbarhaltung von digitalen Objekten zu führen sind. Das OAIS-Modell macht jedoch keine Aussage zur technischen Umsetzung des Informationsmodells.

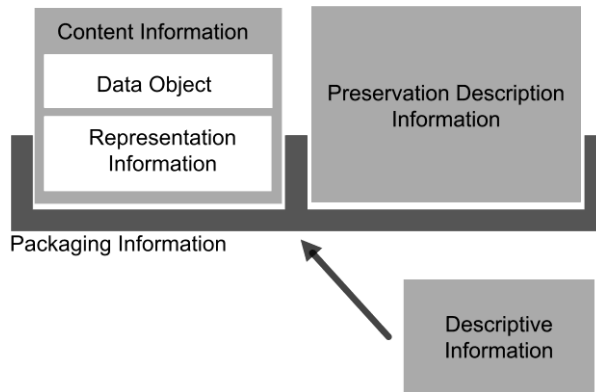


Abbildung: OAIS Informationsmodell

Matterhorn METS ist eine konkrete technische Umsetzung dieses Informationsmodells. Das Format kann sowohl für SIP, AIP und DIP verwendet werden. Es basiert auf den drei Metadatenstandards METS, Premis und EAD. Matterhorn METS wird in Form eines «METS Profils» beschrieben. METS-Profile beschreiben die konkrete Verwendung des METS-Standards in einem bestimmten Kontext, so dass darauf aufbauend die Verarbeitung der METS-Objekte technisch umgesetzt werden kann.² Mit Hilfe dieser Profile ist es beispielsweise möglich, METS-Dateien zwischen verschiedenen Werkzeugen übergeben werden müssen. Das Matterhorn METS-Profil ist bei der Library of Congress unter folgendem Link registriert: <http://www.loc.gov/standards/mets/profiles/00000041.xml>.

Matterhorn METS ist ein generisches Modell. Es ist nicht an einen bestimmten Verwendungskontext gebunden und lässt sich für die Archivierung digitaler Objekte ganz unterschiedlicher Art verwenden. In der Praxis wird das Objektmodell heute in Historischen Archiven, in Museen und Bibliotheken, für Forschungsdaten und in der Geschäftsdatenarchivierung verwendet.

Matterhorn METS wurde gemeinsam durch docuteam und das Staatsarchiv Wallis entwickelt und wird durch diese Organisationen gepflegt und weiterentwickelt.

¹ <http://public.ccsds.org/publications/archive/650x0m2.pdf>

² <http://www.loc.gov/standards/mets/mets-profiles.html>

Umsetzung des OAIS Informationsmodells

Zur Abbildung des OAIS Informationsmodells werden die drei folgenden Metadatenstandards verwendet:

- METS, Metadata Encoding and Transmission Standard als Container:
<http://www.loc.gov/standards/mets/>
- Premis Data Dictionary für technische und administrative Metadaten:
<http://www.loc.gov/standards/premis/>
- EAD, Encoded Archival Description
<https://www.loc.gov/ead/>

Matterhorn METS ist in Form eines METS-Profiles formuliert:

- METS Profiles:
<http://www.loc.gov/standards/mets/mets-profiles.html>
- Matterhorn METS:
<http://www.loc.gov/standards/mets/profiles/00000041.xml>

Die Abbildung der einzelnen Sektionen des OAIS-Informationsmodells geschieht durch die drei Standards wie folgt:

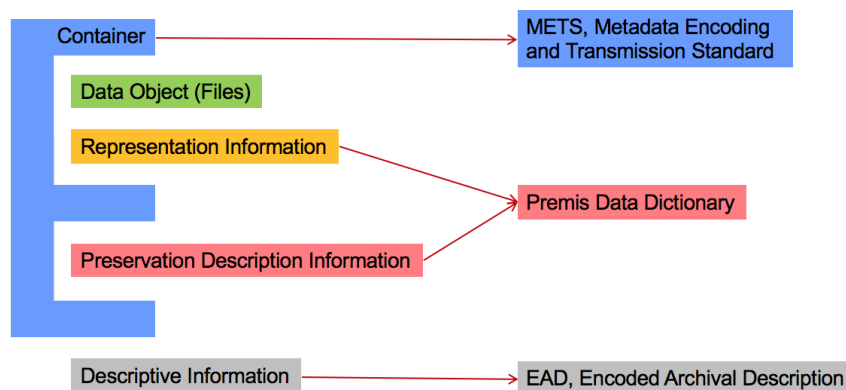
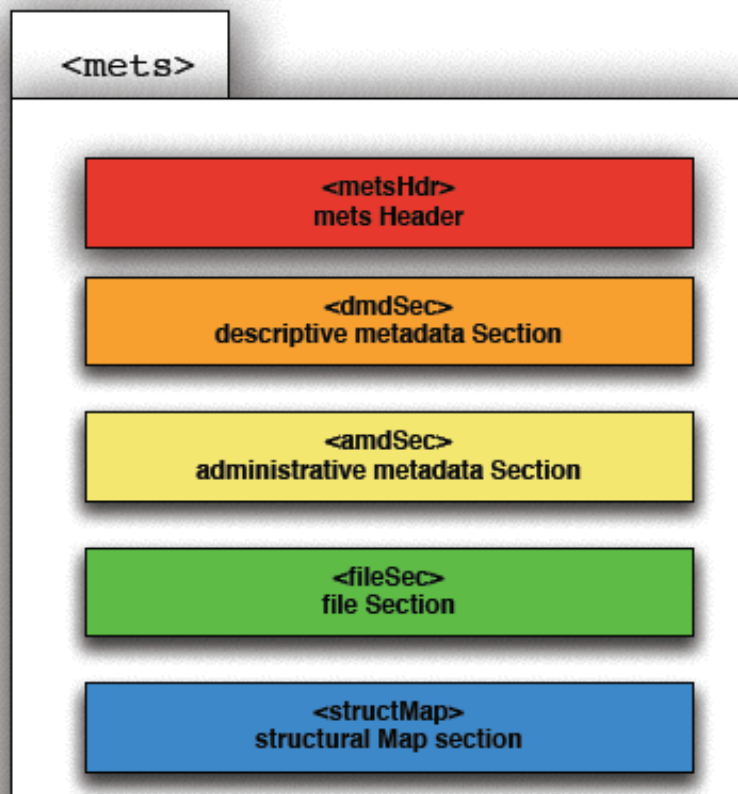


Abbildung: Umsetzung des OAIS-Informationsmodells mit Hilfe der Standards METS, Premis und EAD

Der METS-Standard setzt sich aus mehreren Sektionen zusammen:



Bei Matterhorn METS dient METS primär als Containerformat. Die verschiedenen Sektionen werden wie folgt verwendet:

- Der «**METS header**» umfasst Grundlageninformationen zum gesamten Objekt.
- Die «**fileSec**» enthält eine Liste mit allen Pfaden der Dateien die zum Objekt gehören. Jede Datei hat einen innerhalb des Objekts einen eindeutigen Identifier.
- In der «**structural Map section <structMap>**» wird die Hierarchie des Objekts abgebildet (Ordner, Unterordner, Files). Für jeden Knoten (Ordner, Datei) wird ein Identifier vergeben und es wird auf die beschreibenden Metadaten und die technisch / administrativen Metadaten (Premis) verwiesen. Die structMap bildet die Struktur (Hierarchie) des Objekts ab. Für die Tiefe der Struktur gibt es keine Einschränkung.
- Die «**descriptive metadata section <dmdSec>**» beinhaltet beschreibende Metadaten (EAD). In der Regel wird für jeden Ordner, Unterordner und Datei eine eigene `<dmdSec>` angelegt, in welcher die Metadaten für das jeweilige Objekt liegen.
- Die «**administrative metadata section <amdSec>**» beinhaltet technische und administrative Metadaten im Premis-Format.

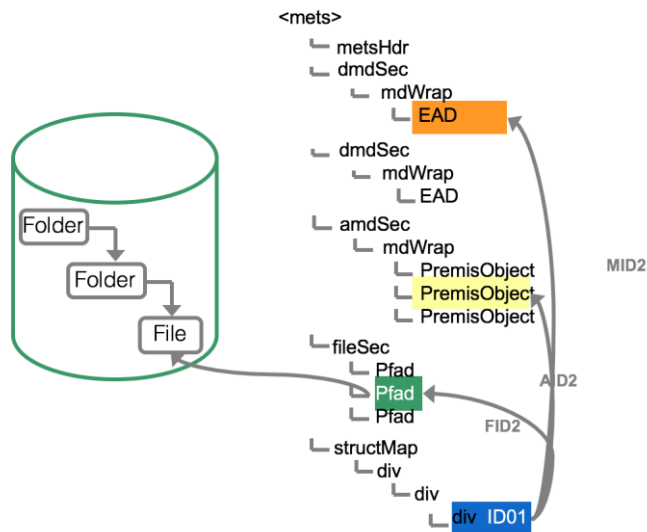


Abbildung: Struktur eines Matterhorn METS-Objekts

Was wird durch das Matterhorn METS-Modell beschrieben?

Matterhorn METS

- beschreibt die interne Struktur (Hierarchie) eines Informationspaketes und gibt an, welche Ordner, Unterordner und Dateien zum Objekt gehören.
- bindet die beschreibenden und die technisch/administrativen Metadaten ein und bringt sie mit den dazugehörigen Ordnern, Unterordnern und Dateien in Verbindung.

Matterhorn METS

- beschreibt nicht, wie Ablieferungs- und Archivierungsprozesse abgebildet werden müssen. Der Ablauf solcher Prozesse kann in Matterhorn METS jedoch dokumentiert werden.
- gibt nicht vor, mit welchen Werkzeugen die Pakete verarbeitet werden müssen.
- macht keine Vorgaben bezüglich Granularität des Informationspakets. Ein Paket kann eine einzelne Datei, aber auch eine umfangreiche Ablieferung mit tausenden von Dateien und vielen Hierarchiestufen umfassen.

Struktur des Objekts

Ein Objekt wird entweder in einem ZIP-Container oder in einem Ordner abgelegt.

Auf oberster Stufe des Zip-Containers oder des Ordners liegt eine Datei „mets.xml“. Diese Datei beinhaltet sämtliche Metadaten zum Objekt.

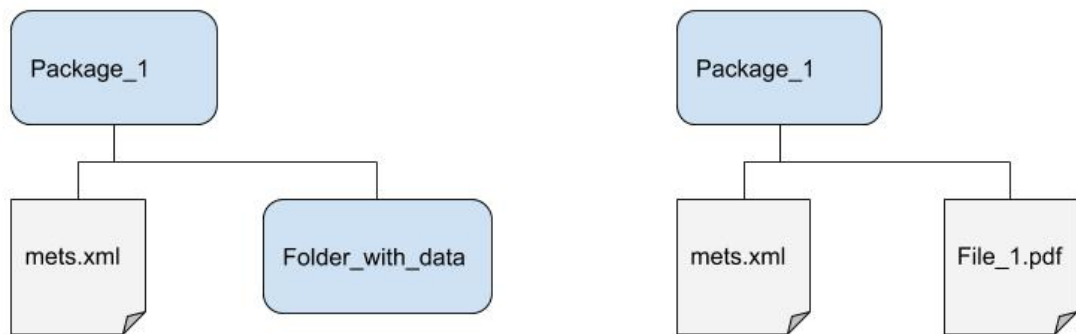
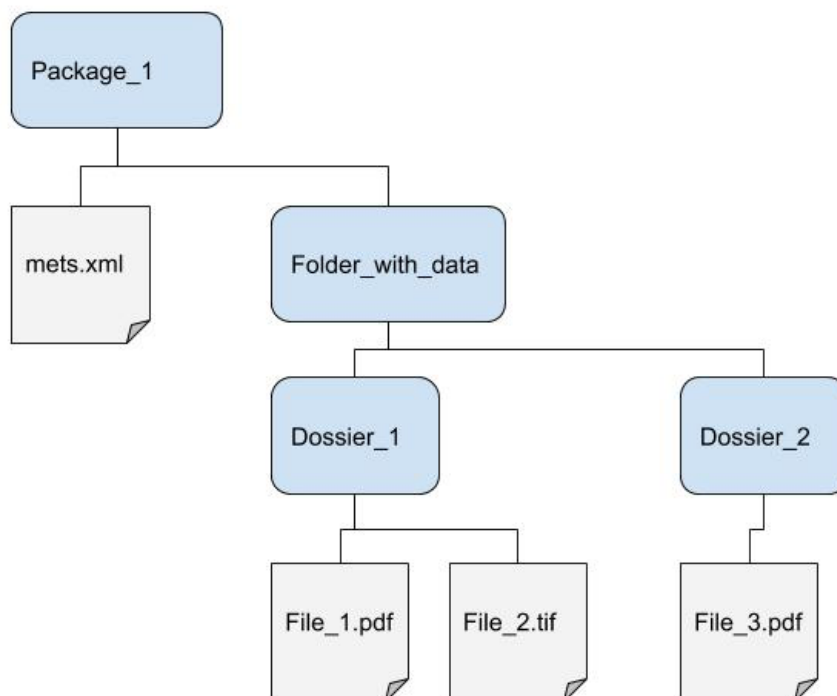


Abbildung: Ein Matherhorn METS-Objekt beinhaltet immer eine mets.xml-Datei auf oberster Stufe. Die Nutzdaten bestehen aus einer einzelnen Datei oder aus einem Ordner mit einer Unterstruktur und Dateien.

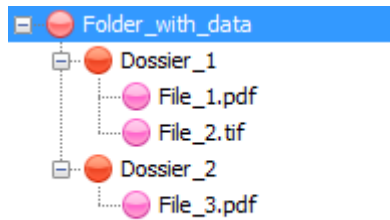
Die Primärdaten des Objekts liegen auf gleicher Stufe wie die mets.xml-Datei. Auf der obersten Stufe liegt eine Datei, falls das Objekt nur eine einzelne Datei umfasst. Ansonsten liegt an oberster Stufe ein Ordner, welcher wiederum Unterordner und Dateien enthalten kann. Die Bezeichnung für den Ordner mit Nutzdaten kann beliebig gewählt werden.



Verwendung von METS

Abbildung der Objektstruktur in METS

Die physische Struktur des Ordners mit den Primärdaten wird in der StructMap der METS-Datei abgebildet. Bei Matterhorn METS ist die Struktur der Primärdaten im Dateisystem (Ordner – Unterordner – Dateien) identisch mit der Struktur in der METS-StructMap. Es gibt also keine Unterscheidung zwischen der logischen und der physischen Struktur des Informationspakets.



```

<METS:structMap>
  <METS:div LABEL="Folder_with_data" TYPE="rootfolder" ADMID="_20160510172331013">
    <METS:div LABEL="EAD" TYPE="metadata" DMDID="_20160510172331060"/>
    <METS:div LABEL="Dossier_1" TYPE="folder" ADMID="_20160510172350248">
      <METS:div LABEL="EAD" TYPE="metadata" DMDID="_20160510172350294"/>
      <METS:div LABEL="File_2.tif" TYPE="file" ADMID="_20160510172529898">
        <METS:div LABEL="Content" TYPE="content">
          <METS:fptr FILEID="_20160510172529882"/>
        </METS:div>
      <METS:div LABEL="EAD" TYPE="metadata" DMDID="_20160510172529960"/>
    </METS:div>
    <METS:div LABEL="File_1.pdf" TYPE="file" ADMID="_20160510172531520">
      <METS:div LABEL="Content" TYPE="content">
        <METS:fptr FILEID="_20160510172531505"/>
      </METS:div>
      <METS:div LABEL="EAD" TYPE="metadata" DMDID="_20160510172531567"/>
    </METS:div>
  </METS:div>
  <METS:div LABEL="Dossier_2" TYPE="folder" ADMID="_20160510172423226">
    <METS:div LABEL="EAD" TYPE="metadata" DMDID="_20160510172423273"/>
    <METS:div LABEL="File_3.pdf" TYPE="file" ADMID="_20160510172550237">
      <METS:div LABEL="Content" TYPE="content">
        <METS:fptr FILEID="_20160510172550221"/>
      </METS:div>
      <METS:div LABEL="EAD" TYPE="metadata" DMDID="_20160510172550284"/>
    </METS:div>
  </METS:div>
</METS:structMap>
  
```

METS Header <metsHdr>

Der metsHdr beinhaltet die folgenden Attribute

Attribute	Description	Requirement
CREATEDATE	contains the timestamp of this SIP's creation	M
LASTMODDATE	Datum der letzten Änderung des Objekts	K

RECORDSTATUS	It describes the record status in regard to the ingest process and contains a static text like "Submitted", "Validated", "Archived", "New", "Migrated", or "Confirmed". It may contain other texts as needed by the context.	M
METS:Agent	At least one "METS:agent" element with the attributes "ROLE='CREATOR'" and "TYPE='INDIVIDUAL'" is mandatory. It must contain the element "METS:name" with the name of the creator. It may contain the element "METS:note" with additional notes about the creator.	M

Descriptive Metadata Section <dmdSec>

Für eine Datei oder einen Ordner des Objekts kann entweder keine oder genau eine dmdSec vorhanden sein.

		Muss/Kann
ID	Die dmdSec muss als Attribut eine ID beinhalten. Die ID matches the corresponding object's "METS:div[@DMDID]" attribute in the structMap.	M

Administrative Metadata Section <amdSec>

The METS file may contain zero or one AMD sections. The AMD section contains, if it is present, zero or more "METS:digiprovMD" elements.

		Muss/Kann
METS:digiprovMD	A single "METS:digiprovMD" element must contain exactly one "METS:mdWrap[@MDTYPE='PREMIS']/METS:xmlData/PREMIS:premis[@version='2.2']" element, which in turn contains a possibly empty list of "PREMIS:object" elements. All "PREMIS:object" elements have to occur in the list before any "PREMIS:event" elements.	K
ID	The "METS:digiprovMD" element requires the "@ID" attribute to match the corresponding object's "METS:div[@ADMID]" attribute in the structMap.	M
PREMIS:objects	Die obligatorischen Elemente innerhalb PREMIS:objects sind: - "PREMIS:objectIdentifier/PREMIS:objectIdentifierType" - "PREMIS:objectIdentifier/PREMIS:objectIdentifierValue"	M
	Weiter werden verwendet um ein Objekt zu charakterisieren:	
PREMIS:event	Die obligatorischen Elemente eines PREMIS:events sind: - "PREMIS:eventIdentifier/PREMIS:eventIdentifierType" - "PREMIS:eventIdentifier/PREMIS:eventIdentifierValue" - "PREMIS:eventType" - "PREMIS:eventDateTime" - "PREMIS:eventOutcomeInformation/PREMIS:eventOutcome" - "PREMIS:linkingObjectIdentifier/PREMIS:linkingObjectIdentifierType"	

	- "PREMIS:linkingObjectIdentifier/PREMIS:linkingObjectIdentifierValue"	
PREMIS:agent	Die obligatorischen Elemente innerhalb PREMIS:agent sind: - "PREMIS:agentIdentifier/PREMIS:agentIdentifierType" - "PREMIS:agentIdentifier/PREMIS:agentIdentifierValue"	
PREMIS:rights	Die obligatorischen Elemente innerhalb PREMIS:rights sind: - "PREMIS:rightsStatement/PREMIS:rightsStatementIdentifier/PREMIS:rightsStatementIdentifierType" - "PREMIS:rightsStatement/PREMIS:rightsStatementIdentifier/PREMIS:rightsStatementIdentifierValue"	

File Section <fileSec>

Die File Section umfasst eine Liste mit allen Dateien, welche zum Objekt gehören. Die METS-Datei enthält genau eine File Section (oder keine, wenn das Objekt keine Dateien beinhaltet). Die fileSec, falls vorhanden, enthält genau METS:fileGrp-Element. Das METS:fileGrp-Element wiederum enthält METS:file-Elemente, in diesen werden die zum Objekt gehörenden Dateien aufgelistet.

		Muss/Kann
METS:file	Each "METS:file" element must contain an "ID"-attribute which must match the "METS:fptr[@FILEID]" attribute value in the corresponding object from the structMap.	
METS:FLocat	Each "METS:file" element must necessarily contain a "METS:FLocat" element with the following mandatory non-empty attributes: - "LOCTYPE" containing the static value "URL" - "href" containing the file path relative to the SIP root directory in URL format.	

Structure Map <structMap>

The structMap contains exactly one "METS:div" element as the root node. Other "METS:div" elements are recursively nested within each other and describe the hierarchical structure of the SIP. Each "METS:div" element represents a file or folder.

		Muss/Kann
METS:div	Jedes METS:div-Element hat drei obligatorische Attribute: - "LABEL" containing the file or folder name - "TYPE" containing one of the static values "rootfolder", "folder", "rootfile" or "file" - "ADMID" must match the corresponding METS:digiprovMD[@ID] attribute in the amdSec section There are four different types of "METS:div" elements, distinguished by their "TYPE" attribute. The topmost element can be one of "rootfolder"	

	<p>or "rootfile". All other elements nested below can be one of "folder" or "file".</p> <p>Folder and rootfolder type "METS:div" elements contain zero or more "METS:div" elements, one for each file or folder within this folder in the file system.</p> <p>File type "METS:div" elements contain exactly one more "METS:div" element with its "LABEL" attribute containing the static value "Content", its "TYPE" attribute containing the static value "content", and a "METS:fptr" element with its "FILEID" attribute value matching the corresponding "METS:file[@ID]" attribute in the fileSec section.</p>	
--	---	--

Verwendung von PREMIS

Docuteam uses PREMIS to store and structure administrative and technical metadata within digital repositories. The following paragraphs show the details of this implementation, i.e. to which degree the standard is used and which values are valid for certain elements.

PREMIS and the package structure of digital objects

Digital objects managed by Docuteam are structured using METS. Administrative metadata as part of the whole package will be located within the respective METS section (amdSec) as inline XML. Instead of having one large PREMIS block for all the folders/files of a digital object, Docuteam creates a separate PREMIS block, wrapped within a digiproVMD element, for each part of the digital object:

```
<METS:amdSec ID="_1282902012679">
  <METS:digiproVMD ID="_1282902012689">
    <METS:mdWrap MDTYPE="PREMIS">
      <METS:xmlData>
        <PREMIS:premis version="2.0">
          [PREMIS objects and events]
        </PREMIS:premis>
      </METS:xmlData>
    </METS:mdWrap>
  </METS:digiproVMD>
  [More digiproVMD blocks as needed]
</METS:amdSec>
```

Each div element within the METS structMap will link to a digiproVMD element by using the ADMID attribute which is a reference to the digiproVMD's ID attribute:

```
<METS:structMap>
  <METS:div LABEL="bilder" TYPE="rootfolder" ADMID="_1282902012689">
    [More div elements as needed]
  </METS:structMap>
```

Each PREMIS section contains 1...n PREMIS objects (created at the time of submission) and 0...n PREMIS events. Each event has to be linked to at least one of the objects.

PREMIS Identifiers

Both for objects as well as events, the identifiers are made up of the objectIdentifierType/EventIdentifierType which is always "Docuteam" and the objectIdentifierValue/eventIdentifierValue which is a timestamp in milliseconds with an underline prefix:

```
<PREMIS:objectIdentifier>
  <PREMIS:objectIdentifierType>Docuteam</PREMIS:objectIdentifierType>
  <PREMIS:objectIdentifierValue>_1273497046765</PREMIS:objectIdentifierValue>
</PREMIS:objectIdentifier>
```

PREMIS object

There are three types of PREMIS objects: representation, file and bitstream.

Only "representation" and "file" are being used by Docu team. They are synonymous to "rootfolder/folder" and "file" of the structMap respectively. The type of a PREMIS object is given as the xsi:type attribute of the PREMIS object element and equivalent to the objectCategory element which can be omitted.

Representation

The only mandatory information for representations is an objectIdentifier element so that events during ingest can be linked to it.

File

A PREMIS object of the type "file" must, apart from the above mentioned objectIdentifier, include the following information:

- objectCharacteristics
 - compositionLevel
 - static value of "0"*
 - fixity
 - messageDigestAlgorithm
 - "MD5" or "SHA-512"*
 - messageDigest
 - [the md5 or SHA-512 checksum of the respective file]*
 - size
 - [the file size in bytes]*
 - format
 - formatDesignation
 - formatName
 - [textual representation of the file format]*
 - formatVersion
 - [textual representation of the file format version]*
 - formatRegistry
 - formatRegistryName
 - static value of "PRONOM"*
 - formatRegistryKey
 - [the PUID (Pronom-Identifier) of the respective file format]*
 - originalName
 - [the name of the file (without path) when it was submitted by the records producer]*

Derived Objects

A special case is the migration of objects and the documentation of these as they will effect the creation of new files and hence also PREMIS objects. However, as the migrated files are only new versions of the original logical piece of information, the respective PREMIS object and successive events will be stored within the same digiprovMD section as the original file's. This will work regardless of the decision to keep or destroy the original file.

While the event documenting the migration will be linking to the original PREMIS object, the PREMIS object of the newly created file will be extended with a relationship element to document its origination:

- relationship
 - relationshipType
static value of "derivation"
 - relationshipSubType
static value of "derived from"
 - relatedObjectIdentification
 - relatedObjectIdentifierType
static value of "Docuteam"
 - relatedObjectIdentifierValue
the system time in milliseconds with an underline prefix, referencing the original object within the same PREMIS block which the current object is derived from.
 - relatedEventIdentification
 - relatedEventIdentifierType
static value of "Docuteam"
 - relatedEventIdentifierValue
the system time in milliseconds with an underline prefix, referencing the event within the same PREMIS block that documents the migration action.

PREMIS event

A PREMIS event has the following structure with all elements being mandatory except the eventOutcomeDetail:

- eventType
Docuteam manages its own controlled vocabulary for event types which is, however, based on the PREMIS propositions: Renaming, Migration, Submission, Fixity Check, Creation, Relocation, Deletion.
- eventDateTime
[the date and time when the event took place]
- eventDetail
[a short textual description of the action(s) that were executed and the tools used during the event]
- eventOutcomeInformation
 - eventOutcome
The result of an event should be recorded and given using a controlled vocabulary: success, failure
 - eventOutcomeDetail
Optional
 - eventOutcomeDetailNote
The result of an event can be described in more detail if needed.
- linkingObjectIdentifier
 - linkingObjectIdentifierType
static value of "Docuteam"
 - linkingObjectIdentifierValue
a timestamp in milliseconds with an underline prefix, referencing the object within the same PREMIS block.

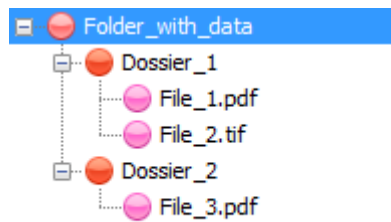
Beispiel

Beispiel-Paket:



Folder_with_data.zip

Struktur des Pakets:



Übersicht über die mets.xml-Datei, welche alle Metadaten enthält:

```

<?xml version="1.0" encoding="UTF-8"?>
<METS:mets xmlns:METS="http://www.loc.gov/METS/" xmlns:EAD="urn:isbn:1-931666-22-9
  xmlns:PREMIS="info:lc/xmlns/premis-v2"
  xmlns:OAI_DC="http://www.openarchives.org/OAI/2.0/oai_dc/"
  xmlns:DC="http://purl.org/dc/elements/1.1/" xmlns:xlink="http://www.w3.org/199
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation=" http://www.loc.gov/METS/ http://www.loc.gov/standa
  PROFILE="http://www.docuteam.ch/xmlns/sip-profile.xml" TYPE="sa_all-formats-01
  <METS:metsHdr CREATEDATE="2016-07-11T10:57:20" LASTMODDATE="2016-07-11T10:58:2
    RECORDSTATUS="New" ADMID="_20160711105720176">
  <METS:agent ROLE="CREATOR" TYPE="INDIVIDUAL">
    <METS:name>peter</METS:name>
  </METS:agent>
</METS:metsHdr>
  <METS:dmdSec ID="_20160711105720333"> [29 lines]
  <METS:dmdSec ID="_20160711105720395"> [27 lines]
  <METS:dmdSec ID="_20160711105723880"> [27 lines]
  <METS:dmdSec ID="_20160711105723989"> [27 lines]
  <METS:dmdSec ID="_20160711105724051"> [27 lines]
  <METS:dmdSec ID="_20160711105724161"> [27 lines]
  <METS:amdSec> [265 lines]
  <METS:fileSec> [15 lines]
  <METS:structMap> [29 lines]
</METS:mets>
  
```