Digital Preservation Metadata and Improvements to PREMIS in Version 3.0

Angela Dappert
University of Portsmouth
Agenda

- Digital preservation metadata
  - Why is it needed and what does it look like?
- PREMIS
  - What is it?
  - Data model
  - How to use it
- From V2 to V3
Agenda

- Digital preservation metadata
  - Why is it needed and what does it look like?
- PREMIS
  - What is it?
  - Data model
  - How to use it
- From V2 to V3
What is digital preservation metadata?

- Digital preservation metadata = metadata that is considered necessary to ensure long-term accessibility of digital resources

- Digital objects must be self-descriptive

- Must be able to exist independently from the systems which were used to create them
  
  XML (machine and human readable)
DP metadata supports preservation goals

Preservation Pyramid
(from Priscilla Caplan)
Domain
Technology dependence

- Digital dependence
- No direct access
- Complex environments

- Not self-descriptive
- Complex formats
Technology dependence

**Metadata:**
- Format information
- Rendering information
  - Software
  - Hardware
  - Other dependencies: schemas, style sheets, encodings, etc.
Complex structures

Metadata

- Physical structural relationships
  - Embedded files
  - File sequence
- Logical structural relationships
Supporting features

Metadata:
Semantic information for the designated community
Supporting features

**Metadata:** Semantic information for the designated community
Obsolescence

-> object transformations

Support

- Pre-emptive preservation actions
  - Bit migration
  - Content migration
- Forensic transformation actions
Obsolescence / object transformations

**Goals**

- Avoid rights violations
- Prove authenticity

**Metadata**

- Rights information for preservation actions during copyright / license period
- Provenance metadata:
  - History of all actions performed on the resource
  - History of custodianship

- Events
- Changes and decisions
- Agents (decision maker + tools used)
- Dates
Obsolescence / object transformations

**Goals**

- Demonstrate degree of authenticity
- Manage potential loss of object characteristics
- Explain decisions
  - Documentation

**Metadata**

- Significant characteristics = business requirement
- Technical and content characteristics of objects before and after preservation actions
- Business rules guiding preservation actions
Mutability

- Intentional or accidental change
- Decay: rapid and potentially complete

Goals

- Viability: the object is readable
- Fixity: the object is unchanged

Metadata

- Data carrier metadata
  - Type of medium
  - Its preservation characteristics
  - Age of medium
  - Date of recording
  - Usage patterns
- Checksums, message digests
- Event creating them
  - Hash algorithms creating them
  - Date/time
  - Originator
## Mutability

- Intentional or accidental change
- Decay: rapid and potentially complete

## Goals

- **Integrity**: the object is whole and unimpaired
- **Authenticity**: the object is what it purports to be

## Metadata

- Event information for format identification and validation events (= provenance)
- Structural metadata
- Provenance metadata
- Digital signatures
- Access rights
Context descriptions

Metadata:
Context descriptions

- Original source
- Related items (e.g. migration source)
Agenda

- Digital preservation metadata
  - Why is it needed and what does it look like?
- PREMIS
  - What is it?
  - Data model
  - How to use it
- From V2 to V3
The PREMIS standard

- International standard for metadata to support the preservation of digital objects and ensure their long-term usability.
  - Information you need to know for preserving digital documents

*Preservation Metadata: Implementation Strategies*

- Developed by an international team of experts.
- Implemented in digital preservation projects around the world.
- Incorporated into commercial and open-source digital preservation tools and systems.
The PREMIS standard

- Data Dictionary (PREMIS 2.2)
  - Version 3 will be released this summer – major release

- XML schema

- OWL ontology

- Supporting documentation
Activities

- The PREMIS Editorial Committee
  - Coordinates revisions and implementation of the standard
- PREMIS Implementors' Group forum (pig@loc.gov)
  - Email message to listserv@loc.gov:
    Text: subscribe pig <your name>
- PREMIS Implementation Fair (PIF)
  - User group meetings (@iPres)
Scope

What PREMIS DD is:

- Common data model for organizing/thinking about preservation metadata
- Implementable
- Standard for exchanging information packages between repositories
- Technically neutral
- Core metadata
Scope

- What PREMIS DD is not:
  - Out-of-the-box solution
  - All needed metadata
  - Lifecycle management of objects outside repository
  - Rights management
Agenda

- Digital preservation metadata
  - Why is it needed and what does it look like?
- PREMIS
  - What is it?
    - Data model
  - How to use it
- From V2 to V3
Data Model in PREMIS Version 2

- Entities: “things” relevant to digital preservation that are described by preservation metadata
- Relationships between Entities
- Properties of Entities (semantic units)
Example: Object Entity semantic units

- 1.1 object Identifier
- 1.2 object Category
- 1.3 preservation Level
- 1.4 significant Properties
- 1.5 object Characteristics
  - 1.5.1 compositionLevel
  - 1.5.2 fixity
  - 1.5.3 size
  - 1.5.4 format
  - 1.5.5 creatingApplication
  - 1.5.6 inhibitors
- 1.6 original Name
- 1.7 storage
- 1.8 environment
- 1.9 signature Information
- 1.10 relationship
- 1.11 linkingEventIdentifier
- 1.13 linkingRightsStatementIdentifier
### Sample Data Dictionary Entry

<table>
<thead>
<tr>
<th>Semantic unit</th>
<th>size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semantic components</td>
<td>None</td>
</tr>
<tr>
<td>Definition</td>
<td>The size in bytes of the file or bitstream stored in the repository.</td>
</tr>
<tr>
<td>Rationale</td>
<td>Size is useful for ensuring the correct number of bytes from storage have been retrieved and that an application has enough room to move or process files. It might also be used when billing for storage.</td>
</tr>
<tr>
<td>Data constraint</td>
<td>Integer</td>
</tr>
<tr>
<td>Object category</td>
<td>Representation</td>
</tr>
<tr>
<td>Applicability</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Examples</td>
<td>2038927</td>
</tr>
<tr>
<td>Repeatability</td>
<td>Not repeatable</td>
</tr>
<tr>
<td>Obligation</td>
<td>Optional</td>
</tr>
<tr>
<td>Creation/Maintenance notes</td>
<td>Automatically obtained by the repository.</td>
</tr>
<tr>
<td>Usage notes</td>
<td>Defining this semantic unit as size in bytes makes it unnecessary to record a unit of measurement. However, for the purpose of data exchange the unit of measurement should be stated or understood by both partners.</td>
</tr>
</tbody>
</table>
Agenda

- Digital preservation metadata
  - Why is it needed and what does it look like?
- PREMIS
  - What is it?
  - Data model
    - How to use it
- From V2 to V3
Tayloring PREMIS to needs

- Increasing experience ensuring the longevity of digital objects
- Changing future technical possibilities
- Changing future legal framework

Tayloring solutions from core metadata

- Varying needs
  - Content-types
  - Institutional policies
  - Intended use
From here to an implementation ...
Example: Document in 3 representations

Intellectual Entity
2015-05-27-Presentation

<descriptive metadata>

Structural / is represented as

Derivation/
has source

Representation 1

File 1
2015-05-27-Presentation.pptx

Structural / includes

Representation 2

File 2
2015-05-27-Presentation.pdf

Structural / includes

Representation 3

File 3
2015-05-27-Notes.docx

Structural / includes
Example: Data model

- Intellectual Entity
- Representation
- Derivation / is represented as
- Structural / includes
- Structural / has sibling
- Structural / has source
- Event
- Agent
- File
objectIdentifier
  objectIdentifierType: ARK
  objectIdentifierValue::ark:/12148/cb37367035f
objectCategory: intellectual entity

relationshipType: structural
relationshipSubType: is represented as

objectIdentifier
  objectIdentifierType: ARK
  objectIdentifierValue: ark:/9999/h1.version1
objectCategory: representation
objectIdentifier
objectIdentifierType: ARK
objectIdentifierValue::ark:/12148/cb37367035f
objectCategory: intellectual entity

relationshipType: structural
relationshipSubType: is represented as

objectIdentifier
objectIdentifierType: ARK
objectIdentifierValue: ark:/9999/h1.version1
objectCategory: representation

relationshipType: structural
relationshipSubType: includes

objectIdentifier
objectIdentifierType: ARK
objectIdentifierValue: ark:/9999/h1.version1
objectCategory: file
format
formatDesignation
formatName: application/pdf
Object Identifier
objectIdentifierType: ARK
objectIdentifierValue: ark:/12148/cb37367035f
objectCategory: intellectual entity

relationshipType: structural
relationshipSubType: is represented as

Object Identifier
objectIdentifierType: ARK
objectIdentifierValue: ark:/9999/h1.version0
objectCategory: representation

Object Identifier
objectIdentifierType: ARK
objectIdentifierValue: ark:/9999/h1.version1
objectCategory: representation
objectIdentifier
    objectIdentifierType: ARK
    objectIdentifierValue: ark:/12148/cb37367035f
    objectCategory: intellectual entity

relationshipType: structural
relationshipSubType: is represented as

objectIdentifier
    objectIdentifierType: ARK
    objectIdentifierValue: ark:/9999/h1.version0
    objectCategory: representation

relationshipType: derivation
relationshipSubType: has source
relatedObjectIdentifier
    relatedObjectIdentifierType: ARK
    relatedObjectIdentifierValue: ark:/9999/h1.version0

objectIdentifier
    objectIdentifierType: ARK
    objectIdentifierValue: ark:/9999/h1.version1
    objectCategory: representation
Tayloring PREMIS to needs

- We cannot know for sure
  - Increasing experience ensuring the longevity of digital objects
  - Changing future technical possibilities
  - Changing future legal framework

- Tayloring solutions from core metadata
  - Varying needs
    - Content-types
    - Institutional policies
    - Intended use
  - Off-the-shelf (OS / commercial) or custom-built

- Off-the-shelf systems
  - Predefined metadata profiles
  - Out-of-the-box tools

- Configured, extended, adapted
  - Metadata profiles and tools

- Custom-built systems
  - Metadata profiles and tools
Agenda

- Digital preservation metadata
  - Why is it needed and what does it look like?
- PREMIS
  - What is it?
  - Data model
  - How to use it
- From V2 to V3
PREMIS: From V2 to V3

- Next major version of the PREMIS Data Dictionary
- Released by July 2015
- Still in proof-reading phase
PREMIS: From V2 to V3

- Improving PREMIS based on user needs
- Add preservationLevelType semantic unit
- Add agentVersion semantic unit
- Add eventDetailInformation semantic unit
- Add “unknown” values
- Add authority for controlled vocabulary
- Make Intellectual Entity an Object category
- Make Environments independent Objects
- Add physical Objects
- Update conformance statement
Approved Changes:
Add preservationLevelType semantic unit

- 1.3 preservationLevel
  - 1.3.1 preservationLevelValue
  - 1.3.2 preservationLevelRole
  - 1.3.3 preservationLevelRationale
  - 1.3.4 preservationLevelDateAssigned
Approved Changes:
Add preservationLevelType semantic unit

- 1.3 preservationLevel
- 1.3.1 preservationLevelType
- 1.3.2 preservationLevelValue
- 1.3.3 preservationLevelRole
- 1.3.4 preservationLevelRationale
- 1.3.5 preservationLevelDateAssigned

Associate type of preservation function with preservation level.
objectIdentifier
  objectIdentifierType: ARK
  objectIdentifierValue: ark:/9999/c1
objectCategory: file
  preservationLevel
    preservationLevelType: Bit preservation
    preservationLevelValue: medium
  preservationLevel
    preservationLevelType: Functional preservation
    preservationLevelValue: migration
objectCharacteristics
  compositionLevel: 0
  size: 726970368
  format
    formatDesignation
      format name: application/vnd.ms-excel
Approved Changes:
Add agentVersion semantic unit

- If agentType is software,
  - agentVersion can be used to refine agentName.

3.1 agentIdentifier
3.2 agentName
3.3 agentType

3.4 agentNote
3.5 agentExtension
3.6 linkingEventIdentifier
3.7 linkingRightsStatementIdentifier
Approved Changes: Add agentVersion semantic unit

- If agentType is software,
  - agentVersion can be used to refine agentName.

3.1 agentIdentifier
3.2 agentName
3.3 agentType
3.4 agentVersion
3.5 agentNote
3.6 agentExtension
3.7 linkingEventIdentifier
3.8 linkingRightsStatementIdentifier
3.9 linkingEnvironmentIdentifier
Approved Changes:
Add eventDetailInformation semantic unit.

- 2.1 eventIdentifier
- 2.2 eventType
- 2.3 eventDateTime
- 2.4 eventDetail
- 2.5 eventOutcomeInformation
- 2.6 linkingAgentIdentifier
- 2.7 linkingObjectIdentifier
Approved Changes:
Add eventDetailInformation semantic unit.

- 2.1 eventIdentifier
- 2.2 eventType
- 2.3 eventDateTime
- 2.4 eventDetailInformation
  - 2.4.1 eventDetail
  - 2.4.2 eventDetailExtension
- 2.5 eventOutcomeInformation
- 2.6 linkingAgentIdentifier
- 2.7 linkingObjectIdentifier
Approved Changes: Unknown compositionLevel and format

compositionLevel and format:
  - A value of *unknown* added if the information is not available.
Implementation specific change: Add authority for controlled vocabulary

- Record the name of the authoritative list used
- Expressed as a string or as a unique URI
  - http://id.loc.gov/vocabulary/preservation/eventType
  - UC San Diego Rights Basis Vocabulary
- Usable in semantic units that suggest use of controlled vocabularies under data constraint.

Changes only to the XML schema

complexType 'stringPlusAuthority' is xs:string with the three new attributes:

- **authority**: name of a controlled vocabulary as a string
- **authorityURI**: name of a controlled vocabulary as URI
- **valueURI**: value from a controlled vocabulary that is in the form of a URI.

Some controlled vocabularies as Linked Data at id.loc.gov: http://id.loc.gov/vocabulary/preservation
eventIdentifier: 
  eventIdentifierType: UUID 
  eventIdentifierValue: 908985d3-9600-4da4-a7e7-c6e9508bf24c

eventType: validation

authority="premisEventType"
authorityURI= "http://id.loc.gov/vocabulary/preservation/eventType.html"
valueURI= "http://id.loc.gov/vocabulary/preservation/eventType/val.html"

eventDateTime: 2014-07-03T23:18:19

eventDetailInformation:
  eventDetail: program="Jhove"; version="1.5"

eventOutcomeInformation:
  eventOutcome: fail
  eventOutcomeDetail:
    eventOutcomeDetailNote:
      format="JPEG"; version="1.02"; result="Not well-formed"
Approved Changes:
Make Intellectual Entity an Object category

- A set of content that is considered a single intellectual unit for purposes of management and description
  - For example, a particular book, map, photograph, or database.
  - An Intellectual Entity can include other Intellectual Entities; for example, a Web site can include a Web page; a Web page can include an image.

V2:
- Assumed to be held in a container metadata schema
- No Intellectual Entity semantic units
  - Exception: identifier to enable linking to a description
  - PREMIS Objects link to it.

V3:
- Possibility to describe intellectual entities
  - Same semantic units as Representations
  - Identifier to enable linking to describe intellectual entities
Approved Changes:
Make Intellectual Entity an Object category

- Relate to PREMIS Events and RightsStatements.
- Support structural and derivative relationships with Objects.
- Represent an aggregate, such as a collection, FRBR work, FRBR expression, fonds or series.
- Capture versioning information and metadata update events at the Intellectual Entity level as core provenance preservation metadata.
- Associate business requirements with them.
  - Significant characteristics, risk definitions, guidelines for preservation actions, etc..
Approved Changes: Make Intellectual Entity an Object category

Like before, an intellectual entity can be represented as a representation, or directly as a single file or a single bitstream, skipping the intermediate Object types.

An Intellectual Entity may have one or more digital representations.

- relationshipType: structural
- relationshipSubType: represents

An intellectual entity relates to other Objects through a relationship.
Approved Changes: Make Environments independent Objects

- What is needed to render or use an object
  - Operating system
  - Application software
  - Computing resources

- A high-level data model
- **No** detailed characteristics specific to an environment type
- A standardized way of treating environments
- Sharable and exchangeable
Example: Environment stack and dependency relationships

- Modularised environment aggregates as a network
- Re-usable and distributed environment descriptions
  - across different Objects
  - across repositories and registries

relationshipType: dependency
relationshipSubType: requires
Data Model in PREMIS V2
Data Model in PREMIS V3

Diagram showing the relationships between Object, Environment, Rights, Agent, and Event, with identifiers connecting them.
Example: An object and its rendering environment

- Intellectual Entity for content Object
- Intellectual Entity hardware
- Intellectual Entity operating system
- Intellectual Entity software application
- Content Object
- File Object ISO image
- File Object executable file

represents = relationshipType: structural relationshipSubType: represents
requires = relationshipType: dependency relationshipSubType: requires
1. **Object to environment** - specify computational context
2. **environment to Object** - documentation, specifications, surrogates
3. **environment to environment** - inclusion, dependency, derivation, other
4. **environment is an Object** – preserved software source code
5. **Agent to Environment** - role of an Agent
6. **environment to Event** - environment specific Events (provenance)
7. **environment to RightsStatement** - software license, policy

“Object”: here a traditional content Object
Expanded relationship types for environment Objects

- Dependency
  - Requires, is required by
  - Is deployed on

- Derivation
  - Is source of, has source

- Logical
  - generalises, is generalised by

- Reference
  - Documents, is documented in

- Replacements
  - Supercedes, is superceded by

- Structural
  - Includes, is included in
  - Represents, is represented as
Expanded relationship types for environment Objects

relationshipType: reference
relationshipSubType: is documented in
relatedObjectIdentifier
  relatedObjectIdentifierType: URL
  relatedObjectIdentifierValue: https://wiki.ubuntu.com/QuantalQuetzal/TechnicalOverview
Semantic units only applicable to intellectual entities.

1.9 environmentFunction
  - environmentFunctionType
  - environmentFunctionLevel

1.10 environmentDesignation
  - environmentName
  - environmentVersion
  - environmentOrigin
  - environmentDesignationNote
  - environmentDesignationExtension

1.11 environmentRegistry
  - environmentRegistryName
  - environmentRegistryKey
  - environmentRegistryRole

1.12 environmentExtension

1.13 relationship
  - relatedEnvironmentPurpose
  - relatedEnvironmentCharacteristic

relationshipType: structural
relationshipSubType: represents

objectIdentifier
  - objectIdentifierType: ARK
  - objectIdentifierValue: ark:/9999/b1

objectCategory: intellectual entity

environmentFunction
  - environmentFunctionType: software
  - environmentFunctionLevel: 1
  - environmentFunction: Ubuntu 32-bit, version 12.10

environmentFunction
  - environmentFunctionType: operating system
  - environmentFunctionLevel: 2
  - environmentFunction: Ubuntu 32-bit, version 12.10

objectIdentifier
  - objectIdentifierType: ARK
  - objectIdentifierValue: ark:/9999/c1

objectCategory: file

objectCharacteristics
  - compositionLevel: 0
  - size: 726970368
  - format
    - formatDesignation
      - format name: application/x-iso9660-image
Semantic units only applicable to environment Intellectual Entities

1.9 environmentFunction
- environmentFunctionType
- environmentFunctionLevel

1.10 environmentDesignation
- environmentName
- environmentVersion
- environmentOrigin
- environmentDesignationNote
- environmentDesignationExtension

objectCategory: intellectual entity
environmentFunction
  environmentFunctionType: software
  environmentFunctionLevel: 1
environmentFunction
  environmentFunctionType: operating system
  environmentFunctionLevel: 2

environmentDesignation
  environmentName: Ubuntu
  environmentVersion: Version: 12.10
  environmentDesignationNote: 32-bit version
  environmentDesignationNote: maintenance deadline: 2014-04

environmentDesignation
  environmentName: Ubuntu
  environmentVersion: Quantal Quetzal
Semantic units only applicable to environment Intellectual Entities

- 1.9 environmentFunction
  - environmentFunctionType
  - environmentFunctionLevel

- 1.10 environmentDesignation
  - environmentName
  - environmentVersion
  - environmentOrigin
  - environmentDesignationNote
  - environmentDesignationExtension

- 1.11 environmentRegistry
  - environmentRegistryName
  - environmentRegistryKey
  - environmentRegistryRole

- relationship
  - relatedEnvironmentPurpose
  - relatedEnvironmentCharacteristic

- objectCategory: intellectual entity
- environmentFunction
  - environmentFunctionType: software
  - environmentFunctionLevel: 1

- environmentFunction
  - environmentFunctionType: operating system
  - environmentFunctionLevel: 2

- environmentDesignation
  - environmentName: Windows XP Professional
  - environmentVersion: Service Pack 3

- environmentRegistry
  - environmentRegistryName: PRONOM
  - environmentRegistryKey: x-sfw/8
  - environmentRegistryRole: identity

- relationshipType: dependency
- relationshipSubType: requires
Semantic units only applicable to environment Intellectual Entities

- **1.9 environmentFunction**
  - environmentFunctionType
  - environmentFunctionLevel

- **1.10 environmentDesignation**
  - environmentName
  - environmentVersion
  - environmentOrigin
  - environmentDesignationNote
  - environmentDesignationExtension

- **1.11 environmentRegistry**
  - environmentRegistryName
  - environmentRegistryKey
  - environmentRegistryRole

Alternative:
Link to an external registry

- x-sfw/8
  - Description of Windows XP Professional in PRONOM

relationshipType: dependency
relationshipSubType: requires
relatedEnvironmentPurpose: render
relatedEnvironmentCharacteristic: recommended
relatedObjectIdentifier
  - relatedObjectIdentifierType: PUID
  - relatedObjectIdentifierValue: x-sfw/8

Content Object
Semantic units only applicable to environment Intellectual Entities

- 1.9 environmentFunction
  - environmentFunctionType
  - environmentFunctionLevel

- 1.10 environmentDesignation
  - environmentName
  - environmentVersion
  - environmentOrigin
  - environmentDesignationNote
  - environmentDesignationExtension

- 1.11 environmentRegistry
  - environmentRegistryName
  - environmentRegistryKey
  - environmentRegistryRole

- 1.12 environmentExtension

- 1.13 relationship
  - relatedEnvironmentPurpose
  - relatedEnvironmentCharacteristic
objectCategory: intellectual entity
environmentFunction
environmentFunctionType: software
environmentFunctionLevel: 1
environmentFunction
environmentFunctionType: software application
environmentFunctionLevel: 2

Firefox 10.0

relationshipType: dependency
relationshipSubType: requires
relatedEnvironmentPurpose: render
relatedEnvironmentCharacteristic: known to work

relationshipType: dependency
relationshipSubType: requires
relatedEnvironmentPurpose: create

objectCategory: file
size: 72943
format
formatDesignation: text/html

BlueGriffon 1.6

1.13 relationship
...
Approved Changes:
Make Environments independent Objects

- Environments: i.e. hardware and software needed to use digital objects
- Described and preserved reusing the Object entity,
- Described as Intellectual Entities, preserved as Representation, File or Bitstream.
- Semantic units that are specific to Environment descriptions
  - Capture the function and designation of the Environment,
  - Link to descriptions in external registries.
- Represented as
  - Aggregate environments or as
  - Individual components of an environment (e.g. an executable file, a stylesheet);
- Relationships become important.
- Direct relationship between Agents and Objects to capture the Environment that acted as the Agent in an Event.
Approved Changes: Add physical Objects

- A physical Object is
  - A content Object, such as a manuscript, or printed document
  - An environment Object, such as a physical hardware device.

- Representation: A digital or physical Object
- Either one instantiates or embodies an Intellectual Entity
- Digital and non-digital Objects can be captured uniformly.
- Physical Objects can relate to digital Objects and other physical Objects.
- In V3 storage is applicable to Representations. For physical Representations: the physical location, e.g. a shelf location.
Envirnons can be physical (e.g. hardware) or digital (e.g. binary or source code).
Approved Changes: Add physical Objects

- [Physical representation]

- relationshipType: derivation
  relationshipSubType: has source
  relatedObjectIdentifier
    relatedObjectIdentifierType: Internal call number
    relatedObjectIdentifierValue: Rés. Ye-3535

- relationshipType: derivation
  relationshipSubType: has source
  relatedObjectIdentifier
    relatedObjectIdentifierType: ARK
    relatedObjectIdentifierValue: ark:/9999/h1.version0

- objectIdentifier
  objectIdentifierType: ARK
  objectIdentifierValue: ark:/12148/cb37367035f
  objectCategory: intellectual entity
  format
    formatDesignation
      formatName: image/tiff
      formatVersion: 6.0

- relationshipType: structural
  relationshipSubType: is represented as

- objectIdentifier
  objectIdentifierType: ARK
  objectIdentifierValue: ark:/9999/h1.version1
  objectCategory: file
  format
    formatDesignation
      formatName: image/jp2

- objectIdentifier
  objectIdentifierType: ARK
  objectIdentifierValue: ark:/12148/cb37367035f
  objectCategory: file
  format
    formatDesignation
      formatName: image/tiff
      formatVersion: 6.0

- relatedObjectIdentifier
  relatedObjectIdentifierType: Internal call number
  relatedObjectIdentifierValue: Rés. Ye-3535

- relatedObjectIdentifier
  relatedObjectIdentifierType: ARK
  relatedObjectIdentifierValue: ark:/9999/h1.version0

Angela Dappert - Digital Preservation Metadata and Improvements to PREMIS in Version 3.0
Approved Changes:
Update conformance statement

- Conformance statement
Thank you!

- Resources: http://www.loc.gov/standards/premis/
- PREMIS Implementors Group Forum: PIG@listserv.loc.gov
55 minutes:

120 minutes: What is DP METADATA

15 minutes: what is PREMIS + examples of use

20 minutes: new features