Seminar C
PREMIS Metadata

International Conference on Dublin Core and Metadata Applications

DC-2008 BERLIN

Markus Enders
British Library

Olaf Brandt
The Office of the Federal Commissioner for the Records of the Ministry for State Security of the former GDR

Data Dictionary for Preservation Metadata: PREMIS Seminar
26 September 2008
DC2008, Berlin, Germany

Presenters

Markus Enders (British Library)
mailto:Markus.Enders@BL.UK

Olaf Brandt (BstU)
mailto:brandt.lib@gmail.com
Agenda

9:00-9:45	Background and Context
	The PREMIS Data Model
	General introduction to the PREMIS Data Dictionary
9:45-10:30	Object entity
10:30-11:00	Break
11:00-11:45	Objects example, Q&A
11:45-12:30	Events, Events Example, Rights, Agents
12:30-13:30	Lunch
1:30-2:15	Implementation: general and in XML
2:15-3:00	Implementation in METS and METS/XML example
3:00-3:30	Break
3:30-5:00	Implementation panel and discussion:
	Karin Bredenberg (Statensarkiv, Sweden)
	Karsten Huth (Bundesarchiv)
	Markus Enders (British Library)

PREMIS PREservation Metadata Implementation Strategies

PREMIS Tutorial:
Understanding & Implementing the PREMIS Data Dictionary for Preservation Metadata

Rebecca Guenther, Library of Congress

PREMIS Tutorial
University of California/San Diego
February 11, 2008

PREMIS PREservation Metadata Implementation Strategies

GOALS

- Background and context of PREMIS Data Dictionary

- Discuss PREMIS data model, identifiers, and relationships

- Discuss semantic units defined in the Dictionary

- Discuss major implementation issues

- Show ways of representing PREMIS in XML
  - PREMIS and METS

- Discuss institutional experiences in working with the PREMIS Data Dictionary

PREMIS PREservation Metadata Implementation Strategies

INTRODUCTION: BACKGROUND AND CONTEXT

THE DECLARATION COMMITTEE.

PREMIS PREservation Metadata Implementation Strategies

Digital preservation: imperative and challenge

- More and more of scholarly and cultural record exists in digital form; steps must be taken to secure its long-term future

- Significant progress has been made in raising awareness about digital preservation imperative

- Shift in focus from **articulating** problem to **solving** it …
  - Not so much *"Why is digital preservation important",* but *"What must be done to achieve preservation objectives?"*

- Many practical challenges in implementing reliable, sustainable digital preservation programs
  - One key challenge: preservation metadata

PREMIS PREservation Metadata Implementation Strategies

Some background ...

- **Pre-2002:** various preservation metadata element sets released
  - Different scopes, purposes, underlying models/assumptions
  - No international standard; little consolidation of expertise/best practice

- **June 2002:** Preservation Metadata Framework
  - International working group (jointly sponsored by OCLC, RLG)
  - Comprehensive, high-level description of types of information constituting preservation metadata
  - Used OAIS reference model as starting point
  - Set of "prototype" preservation metadata elements
  - Consensus-based foundation for developing formal preservation metadata specifications … but not an "off-the-shelf, ready to implement" solution

- **Post-2002:** Need implementable preservation metadata, with guidelines for application and use, relevant to a wide range of digital preservation systems and contexts
  - Motivated formation of **PREMIS Working Group**

PREMIS PREservation Metadata Implementation Strategies

PREMIS Working Group

- June 2003: OCLC, RLG sponsored new international working group:
  - *PREMIS: **Pre**servation **M**etadata: **I**mplementation **S**trategies*

- Membership:
  - **> 30 experts from 5 countries, representing libraries, museums, archives, government agencies, and the private sector**
  - Co-Chairs: Priscilla Caplan (FCLA), Rebecca Guenther (LC)

- Objective 1: Identify and evaluate alternative strategies for encoding, storing, managing, and exchanging preservation metadata
  - PREMIS Survey Report (September 2004)
  - Snapshot of current practices/emerging trends related to managing and using preservation metadata in digital archiving systems
  - http://www.oclc.org/research/projects/pmwg/surveyreport.pdf

- Objective 2: Define implementable, core preservation metadata, with guidelines/recommendations for management and use

PREMIS PREservation Metadata Implementation Strategies

PREMIS Data Dictionary

- **May 2005:** *Data Dictionary for Preservation Metadata: Final Report of the PREMIS Working Group*

- **March 2008:** *PREMIS Data Dictionary for Preservation Metadata, version 2.0*

- Includes PREMIS Data Dictionary, context/assumptions, data model, usage examples

- XML schema to support implementation

- **Data Dictionary:**
  - Comprehensive view of information needed to support digital preservation
    - Guidelines/recommendations to support creation, use, management
  - Based on deep pool of institutional experiences in setting up and managing operational capacity for digital preservation
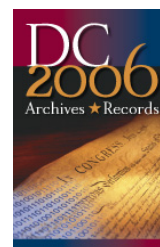
http://www.loc.gov/standards/premis/premis-2-0.pdf

PREMIS PREservation Metadata Implementation Strategies

2005 British Conservation Awards: Digital Preservation Award

2006 Society of American Archivists Preservation Publication Award

PREMIS PREservation Metadata Implementation Strategies

Some guiding principles …

- "Implementable, core, preservation metadata":
  - "Preservation metadata": maintain viability, renderability, understandability, authenticity, identity in a preservation context
  - "Core": What most preservation repositories need to know to preserve digital materials over the long-term
  - "Implementable": rigorously defined; supported by usage guidelines/recommendations; emphasis on automated workflows

- "Technical neutrality":
  - Digital archiving system: no assumptions about specific archiving technology, system/DB architectures, preservation strategy
  - Metadata management: no assumptions about whether metadata is stored locally or in external registry; recorded explicitly or known implicitly; instantiated in one metadata element or multiple elements
  - Promotes flexibility, applicability in wide range of contexts

---

PREMIS PREservation Metadata Implementation Strategies

Scope

- What PREMIS DD **is**:
  - Common data model for organizing/thinking about preservation metadata
  - Guidance for local implementations
  - Standard for exchanging information packages between repositories

- What PREMIS DD **is not**:
  - Out-of-the-box solution: need to instantiate as metadata elements in repository system
  - All needed metadata: excludes business rules, format-specific technical metadata, descriptive metadata for access, non-core preservation metadata
  - Lifecycle management of objects outside repository
  - Rights management: limited to permissions regarding actions taken within repository

PREMIS PREservation Metadata Implementation Strategies

PREMIS Maintenance Activity

- Web site:
  - Permanent Web presence, hosted by Library of Congress
  - Central destination for PREMIS-related info, announcements, resources
  - Home of the PREMIS Implementers' Group (PIG) discussion list

- PREMIS Editorial Committee:
  - Set directions/priorities for PREMIS development
  - Coordinate future revisions of Data Dictionary and XML schema
  - Membership: Library of Congress, OCLC, FCLA, British Library, Library Archives Canada, BStU (Germany)

http://www.loc.gov/standards/premis/

PREMIS PREservation Metadata Implementation Strategies
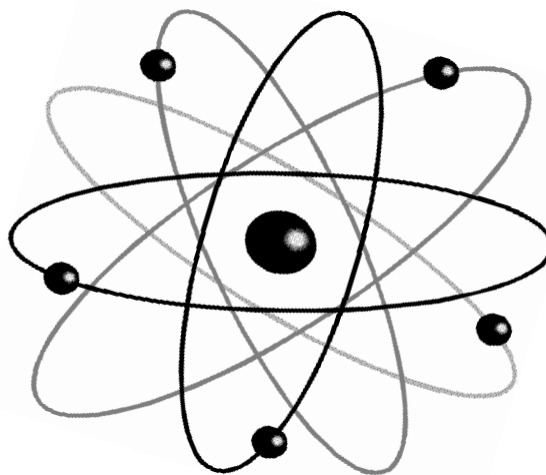
Activities

- First revision of Data Dictionary (PREMIS 2.0)
  - Revised by EC and released in late March
  - Changes based on extensive discussions with implementers and on Editorial Committee
  - http://www.loc.gov/premis/changes.html
- Guidelines for using PREMIS with METS (draft available at:)
  - http://www.loc.gov/premis/guidelines-premismets.html
- PREMIS Implementers' Registry
  - http://www.loc.gov/premis/premis-registry.html
- Previous PREMIS Tutorials:
  - Glasgow, Boston, Stockholm, Albuquerque, Washington , San Diego
- Consultancies:
  - "PREMIS for Dummies" document
  - Tool for converting PREMIS to METS/PREMIS and vice versa

PREMIS PREservation Metadata Implementation Strategies

Changes in PREMIS revision version 2.0

- Clarifications of ambiguous semantic units
- Added additional structure to preservationLevel and significantProperties
- Revised data model to make it more consistent with the data dictionary, allowing for links in both directions between PREMIS entities
- Rights statements revised to broaden scope; distinction made between rights imposed by copyright, license or statute
- Allow for extensibility for specified semantic units
- Add conventions for indicating different types of dates
- Introduced mechanism for controlled vocabularies
- Numerous other changes (obligation, application, etc.)

PREMIS PREservation Metadata Implementation Strategies

DATA MODEL

**PREMIS** PREservation Metadata Implementation Strategies

The PREMIS Data Model

- Data model includes:
  - Entities: "things" relevant to digital preservation that are described by preservation metadata (Intellectual Entities, Objects, Events, Rights, Agents)
  - Properties of Entities (semantic units)
  - Relationships between Entities

- Why have data model?
  - Organizational convenience (for development and use)
  - Useful framework for distinguishing applicability of semantic units across different types of Entities and different types of Objects
  - But: not a formal entity-relationship model; not sufficient to design databases

---

PREMIS PREservation Metadata Implementation Strategies

PREMIS Data Model

PREMIS PREservation Metadata Implementation Strategies
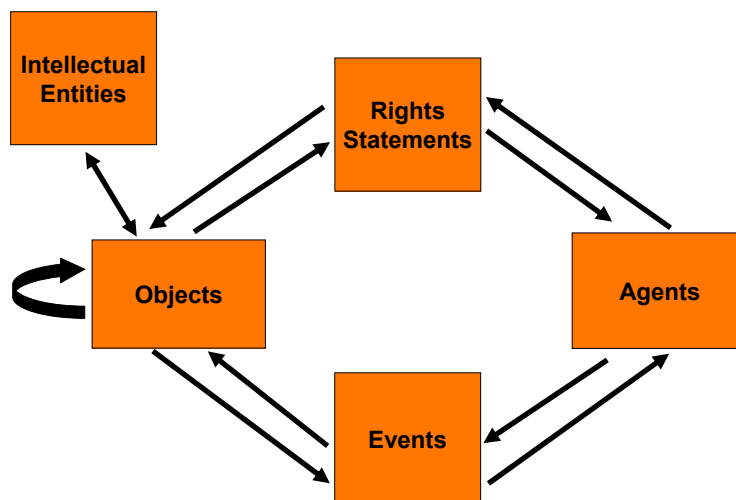
Intellectual Entities
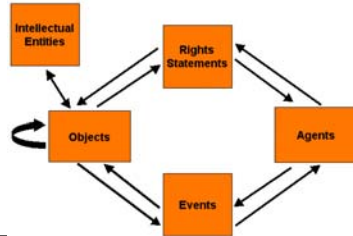
Examples:
- Rabbit Run by John Updike (a book)
- "Maggie at the beach" (a photograph)
- The Library of Congress Website (a website)
- The Library of Congress: American Memory Home page (a web page)

- Set of content that is considered a single intellectual unit for purposes of management and description (e.g., a book, a photograph, a map, a database)
- May include other Intellectual Entities (e.g. a website that includes a web page)
- **Has one or more digital representations**
- Not fully described in PREMIS DD, but can be linked to in metadata describing digital representation

---

PREMIS PREservation Metadata Implementation Strategies
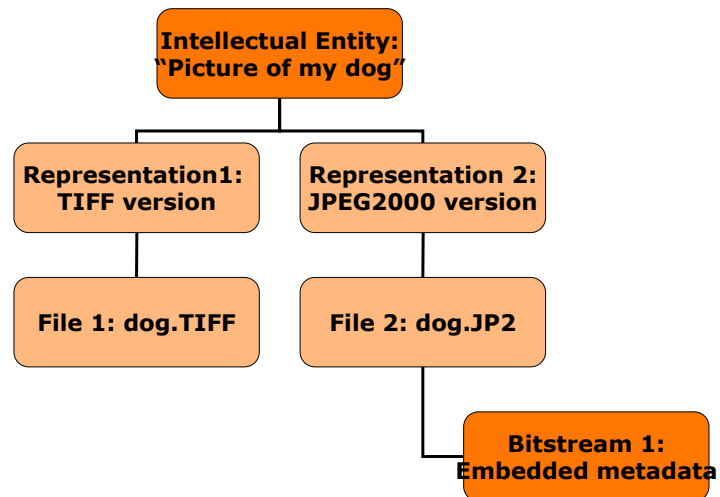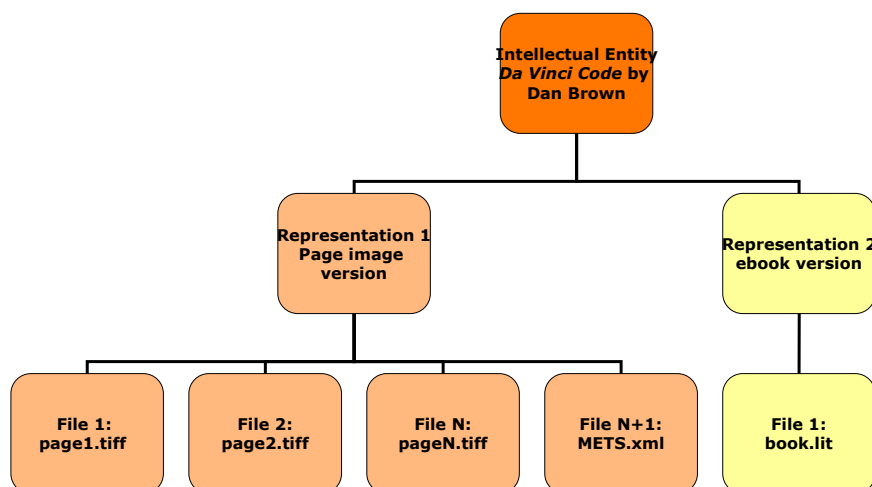
Objects
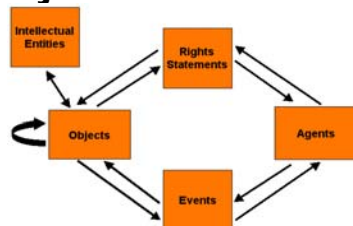
Examples:
- chapter1.pdf (a file)
- chapter1.pdf + chapter2.pdf + chapter3.pdf (representation of a book w/3 chapters)
- TIFF file containing header and 2 images (2 bitstreams (images), each with own set of properties (semantic units): e.g., identifiers, technical metadata, inhibitors, …)

- Discrete unit of information in digital form
- **Objects are what repository actually preserves**
- Three types of Object:
  - **FILE:** named and ordered sequence of bytes that is known by an operating system
  - **REPRESENTATION:** set of files, including structural metadata, that, taken together, constitute a complete rendering of an Intellectual Entity
  - **BITSTREAM:** data within a file with properties relevant for preservation purposes (but needs additional structure or reformatting to be stand-alone file)

PREMIS PREservation Metadata Implementation Strategies

An important aside about Objects

- Repository does NOT have to control Objects at all levels
  - E.g., repository may only manage files, not representations or bit streams.

- The PREMIS DD tells you:
  - **IF** you control at the representation level, these are the semantic units (properties) that pertain to representations;
  - **IF** you control at the file level, these are the semantic units (properties) that pertain to files;
  - **IF** you control at the bitstream level, these are the semantic units (properties) that pertain to bit streams;
  - **AND IF** you control at multiple levels, you need to record relationships between them (more on this soon).


PREMIS PREservation Metadata Implementation Strategies
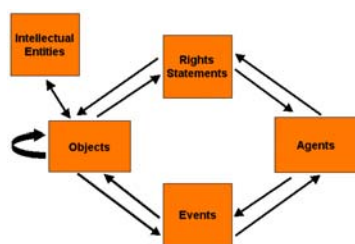
Events

Examples:
- Validation Event: use JHOVE tool to verify that chapter1.pdf is a valid PDF file
- Ingest Event: transform an OAIS SIP into an AIP (one Event or multiple Events?)
- Migration Event: create a new version of an Object in an up-to-date format

- An action that involves or impacts at least one Object or Agent associated with or known by the preservation repository
- Helps document digital provenance. Can track history of Object through the chain of Events that occur during the Objects lifecycle
- Determining which Events are in scope is up to the repository (e.g., Events which occur before ingest, or after de-accession)
- Determining which Events should be recorded, and at what level of granularity is up to the repository

## PREMIS · PREservation Metadata Implementation Strategies

Agents
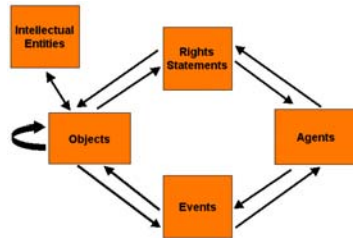
Examples:
- Priscilla Caplan (a person)
- Florida Center for Library Automation (an organization)
- Dark Archive in the Sunshine State implementation (a system)
- JHOVE version 1.0 (a software program)

- Person, organization, or software program/system associated with an Event or a Right (permission statement)
- Agents are associated only indirectly to Objects through Events or Rights
- Not defined in detail in PREMIS DD; not considered core preservation metadata beyond identification

PREMIS · PREservation Metadata Implementation Strategies

Rights Statements
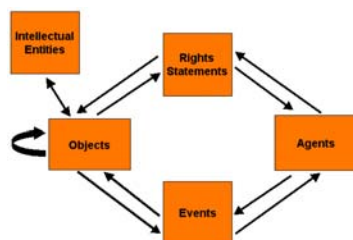
Example:
- Priscilla Caplan grants FCLA digital repository permission to make three copies of metadata_fundamentals.pdf for preservation purposes.

- An agreement with a rights holder that grants permission for the repository to undertake an action(s) associated with an Object(s) in the repository.
- Not a full rights expression language; focuses exclusively on permissions that take the form:
  - Agent X grants Permission Y to the repository in regard to Object Z.

PREMIS PREservation Metadata Implementation Strategies

Semantic units

- A semantic unit is a property of an Entity
  - Something you *need to know* about an Object, Event, Agent, Right
  - Piece of information most repositories need to know in order to carry out their digital preservation functions

- Two kinds of semantic unit:
  - Container: groups together related semantic units
  - Semantic components: semantic units grouped under the same container

- Example:

  ObjectIdentifier [container]

  ObjectIdentifierType [semantic component]

  ObjectIdentifierValue [semantic component]

PREMIS PREservation Metadata Implementation Strategies

Semantic units and metadata elements

- A semantic unit is *not* a metadata element
  - Metadata element is an implementation decision (how and whether a semantic unit is recorded in the system)

- Examples:
  - Semantic unit can be recorded in single metadata element, or multiple elements:
    - Example: significantProperties: break up into separate elements for content, "look and feel", and functionality, or record all in 1 element
  - Semantic unit can be recorded explicitly, or known implicitly
    - Example: IdentifierType: created/assigned internally by repository, assigned to all Objects, so no need to record

- However it is implemented/recorded, a semantic unit should be *recoverable* from archiving system (broadly defined)
  - PREMIS Data Dictionary describes semantic units relevant to most digital preservation activities and contexts

P R E M I S PREservation Metadata Implementation Strategies

IDENTIFIERS AND RELATIONSHIPS

P R E M I S PREservation Metadata Implementation Strategies

Identifiers

- Instances of Objects, Events, Agents and Rights statements are uniquely identified by **Identifiers**

Syntax
- [entity]Identifier
  - [entity]IdentifierType = a specification of the domain in which identifier is unique (e.g. URI, DOI, PURL)
  - [entity]IdentifierValue = the identifier string itself

Example
- ObjectIdentifier
  - ObjectIdentifierType = DRS
  - ObjectIdentifierValue =
    http://nrs.harvard.edu/urn-3:FHCL.Loeb:sa1

Example
- EventIdentifier
  - EventIdentifierType = DRS
  - EventIdentifierValue = 716593

Some notes on Identifiers

- "IdentifierType" optimally should contain sufficient information to indicate:
  - How to build the value
  - Who is the naming authority
  - Example from previous slide: ObjectIdentifierType = "DRS" (Harvard's Digital Repository Service). Could have also put "URL" (since identifier is unique in both domains) but "DRS" conveys more information.

- If all identifiers are local to repository system, it is unlikely that IdentifierType would be recorded for each identifier in the system
  - BUT should be supplied when exchanging data with others

- Identifiers can be created inside or outside the repository
  - Example: PURLs

Relationships

- **Many different types of information relevant to preservation can be expressed as relationships:**
  - e.g., "A is part of B", "A is scanned from B", "A is a version of B"

- **PREMIS Data Dictionary supports expression of relationships between:**
  - Different Objects
    - Across same level or different levels
    - Structural: relationships between parts of a whole
    - Derivation: relationships resulting from replication or transformation of an Object
  - Different Entities

- **Relationships are established through reference to Identifiers of other Objects or Entities**

PREMIS PREservation Metadata Implementation Strategies

Relationships between Objects: Which, How, Why

- **WHICH Objects are related?**
  - relatedObjectIdentification: type, value
  - relatedObjectSequence: documents "ordered" relationships: e.g., pages, chapters, slide #

- **HOW are the Objects related?**
  - relationshipType: structural, derivation
  - relationshipSubType: "is part of", "is source of", "is derived from"

- **WHY are the Objects related?**
  - Was relationship result of an Event? (e.g., "migration", "replication")
  - relatedEventIdentification: type, value
  - relatedEventSequence: ordered sequence of Events
    - Event 1: Convert Excel spreadsheet to ASCII tab-delimited file
    - Event 2: Convert ASCII file to new spreadsheet format
    - Avoids numerous bilateral format-to-format conversions

PREMIS PREservation Metadata Implementation Strategies

Example: Structural relationship
File "is part of" Representation

relationship [part of the description of File]
    relationshipType = structural
    relationshipSubType = is part of
    relatedObjectIdentification [the Web page]
        relatedObjectIdentifierType = repositoryID
        relatedObjectIdentifierValue = 0385503954
        relatedObjectSequence = 0
    relatedEventIdentification [none]

File: graphic.gif

is part of

Representation: Web Page

16

PREMIS PREservation Metadata Implementation Strategies
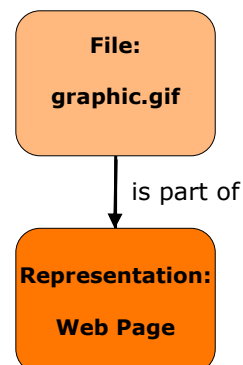
Example: Derivation relationship
File 1 "is source of" File 2 through Migration Event

is source of

| File 1 (original) | → | File 2 (migrated) |

relationship [part of description of File 1]
    relationshipType = derivation
    relationshipSubType = is source of
    relatedObjectIdentification [identifier of File 2]
        relatedObjectIdentifierType = repositoryID
        relatedObjectIdentifierValue = F004400
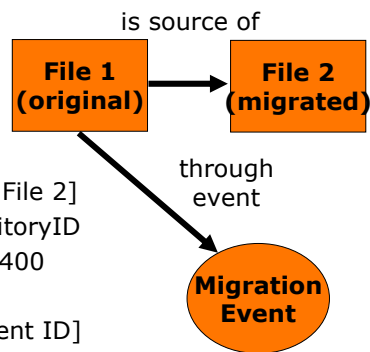   relatedObjectSequence [none]
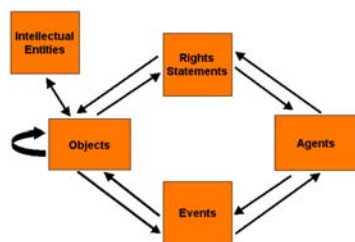   relatedEventIdentification [Migration Event ID]
        relatedEventIdentifierType = repEventID
        relatedEventIdentifierValue = E0192
        relatedEventSequence [none]

through event

Migration Event

PREMIS PREservation Metadata Implementation Strategies

Relationships between different Entities

- **Identifiers are used to link related Entities together**
- **For example, an Object can link to one or more Intellectual Entities, Rights statements, and Events via "linking" semantic units**

linkingIntellectualEntityIdentifier
    linkingIntellectualEntityIdentifierType
    linkingIntellectualEntityIdentifierValue

linkingPermissionStatementIdentifier
    linkingPermissionStatementIdentifierType
    linkingPermissionStatementIdentifierValue

linkingEventIdentifier – can you guess the two sub-elements? ☺

PREMIS — PREservation Metadata Implementation Strategies

Data dictionary descriptions

Semantic unit	Name that is descriptive and unique. Use externally aids interoperability. Need not be used internally in repository.
Semantic components	If a container, lists its sub-elements. Each component has own entry.
Definition	Meaning of semantic unit
Rationale	Why the unit is needed (if not obvious)
Data constraint	How it should be encoded; "Container": an umbrella for two or more; no values given "None": can take any form "Value should be taken from a controlled vocabulary"
Object category	Representation File Bit stream
Applicability	Whether it applies to the category of object
Examples	Illustrative examples of values
Repeatability	Whether it can take multiple values
Obligation	Whether values must be given. "Mandatory": something the repository must know independent of how or whether the repository records it. Means mandatory if applicable. If not explicitly recorded, it must be provided in exchange.
Creation /maintenance notes	Information about how values may be obtained or updated.
Usage notes	Information about intended use.

For each level of Object (applies to Object category through Obligation rows)

PREMIS — PREservation Metadata Implementation Strategies

Sample Data Dictionary entry

Semantic unit	size		
Semantic components	None		
Definition	The size in bytes of the file or bitstream stored in the repository.		
Rationale	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.		
Data constraint	Integer		
Object category	**Representation**	**File**	**Bitstream**
Applicability	Not applicable	Applicable	Applicable
Examples		2038927	
Repeatability		Not repeatable	Not repeatable
Obligation		Optional	Optional
Creation/ Maintenance notes	Automatically obtained by the repository.		
Usage notes	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.		

PREMIS PREservation Metadata Implementation Strategies

SEMANTIC UNITS PERTAINING TO OBJECTS

PREMIS PREservation Metadata Implementation Strategies

Object entity

- Aggregates characteristics relevant to preservation management that are properties of the object
- Semantic units may not all be applicable to each type of object (representation, file, bitstream)
- Main types of information
  - identifier
  - object characteristics
  - creation information
  - software and hardware environment
  - digital signatures
  - relationships to other objects
  - links to other types of entity

preservationLevel and objectCategory

- **objectCategory**
  - Values: representation, file, bitstream

- **preservationLevel**
  - **What preservation treatment/strategy the repository plans for this object**
  - Varying preservation options dependent on factors such as value, uniqueness, preservability of format
  - A business rule only relevant in a given repository
  - Examples: full, bit-level
  - Optional for representation and file

preservationLevel

- **preservationLevel**
  - Additional (optional) semantic units:
    - Role: requirement or capability
    - Rationale: important, when preservationLevelValue differs from usual repository policy, e.g. in case of a defective file.
    - Date: Date and Time when the preservationLevel was assigned to the object

P R E M I S PREservation Metadata Implementation Strategies

significantProperties

- Applicable to representation, file and bitstream
- Characteristics subjectively considered important
  – e.g embedded JavaScript in PDF might be considered as important while Links in PDF are considered as unimportant and need not be preserved
- May help to measure preservation success
- Container for subunits:
  - significantPropertiesType
  - significantPropertiesValue
  - significantPropertiesExtension

P R E M I S PREservation Metadata Implementation Strategies

significantProperties

- May apply to all objects of a certain class or may be unique to each individual object
- May be determined by business rules of the repository
- Not an intrinsic property of an object; a particular archive's assessment of which of the object's properties need to persist over time
- Related to the preservation strategy chosen by the archive
- Listing significant properties implies that the repository plans to preserve those properties and would note any modifications to them in eventOutcome
- Further work is needed in determining and describing significant properties

3

PREMIS PREservation Metadata Implementation Strategies

Examples of significantProperties

- For a PDF with embedded links that are not essential use: "Content only"
- For a TIFF file: "Color accuracy (Adobe RGB 1998)"
- For a Web page: "One of two embedded FLASH files for splash page"
- Revision in v. 2:

Example 1:
significantPropertiesType = "behavior"
significantPropertiesValue = "editable"

Example 2:
significantPropertiesType = "page width"
significantPropertiesValue = "210 mm"

PREMIS PREservation Metadata Implementation Strategies

Extension containers (general)
(e.g. significantPropertiesExtension, creatingApplicationExtension…)

- New in Premis 2.0
- Contains externally defined semantical units
- Allows to extend PREMIS with metadata elements which are more granular, non-core or out of scope of the PREMIS data dictionary
- Data in the container may replace, refine or be additional to the appropriate PREMIS semantical unit
- One schema per extension; if more schema are needed, the extension element needs to repeated

objectCharacteristics

- Applicable only to file and bitstream (although some have needed it for representation)
- Technical properties common to all/most file formats, not format specific
- Container for subunits:
  - compositionLevel
  - fixity
  - size
  - format
  - creatingApplication
  - Inhibitors
  - objectCharacteristicsExtension

fixity

- **Information used to verify whether an object has been altered; compare message digests ("checksums") calculated at different times**
- Container for messageDigestAlgorithm, messageDigest, messageDigestOriginator
- Automatically calculated and recorded by repository
- **messageDigestAlgorithm**: controlled vocabulary, example: SHA-1
- **messageDigest**: output of message digest algorithm
- **messageDigestOriginator**: agent that created original message digest; could be a string or a pointer

Example:
fixity
    messageDigestAlgorithm=Adler-32
    messageDigest=7c9b35da
    messageDigestOriginator=OCLC

PREMIS PREservation Metadata Implementation Strategies

format

- **Identifies the format of a file or bitstream**
- Container semantic unit
- Preservation activities depend on detailed and accurate knowledge about formats
- Should be ascertained by repository on ingest (for example, using JHOVE)
- May be a format name (**formatDesignation**) or a pointer into a registry (**formatRegistry**)
- Will be changed to repeatable in v. 2 to associate a format designation with a particular format registry)

PREMIS PREservation Metadata Implementation Strategies

formatDesignation and formatRegistry

- **formatDesignation**
  - **Identifies the format of an object by name and version**
  - Format may be a matter of opinion: Is it text, xml, or METS?
  - MIME type is most widely used authority list
  - May need more granularity; may be multipart (tiff 6.0/geotiff)

- **formatRegistry**
  - **Identifies format by reference to an entry in a format registry**
  - Detailed specifications on formats may be contained in a future format registry
  - formatRegistryName, formatRegistryKey, formatRegistryRole
  - Role includes purpose or expected use
- **formatNote** – free text

PREMIS PREservation Metadata Implementation Strategies

Examples of format

formatDesignation
    formatName=.eps
    formatVersion=2.0
formatRegistry
    formatRegistryName=PRONOM
    formatRegistryKey=eps
    formatRegistryRole=Basic

formatDesignation
    formatName=PDF
    formatVersion=1.5
formatRegistry
    formatRegistryName=LC digital format descriptions
    formatRegistryKey=fdd000123
    formatRegistryRole=assessment

PREMIS PREservation Metadata Implementation Strategies

creatingApplication

- **Information about the application which created a file/bitstream**
- Software bugs are not uncommon and may affect the integrity of content or create artifacts. In a repository it might be useful to search for all files created by a certain version of the an application to fix them.
- **creatingApplicationName**
- **creatingApplicationVersion**
- **dateCreatedByApplication**
  - Actual or approximated date and time when the object was created
- **creatingApplicationExtension**
  - Specified metadata schema can be included instead or in addition to PREMIS defined semantic units
  - Additional schema might contain values from a controlled list, point to a registry….

PREMIS PREservation Metadata Implementation Strategies

inhibitors

- **Features of the object intended to inhibit access, use or migration**
- It is necessary to record the kind of encryption and the access key to allow future use of the object
- Applicable to file and bitstream
- **inhibitorType**
  - Inhibitor method employed, e.g. "DES", "password protection"
- **inhibitorTarget**
  - The content or function protected, e.g. "function: print"
- **inhibitorKey**
  - The decryption key or password

Example:

inhibitors
    inhibitorType=DES
    inhibitorTarget=all content
    inhibitorKey=[DES encryption key]

PREMIS PREservation Metadata Implementation Strategies

objectCharacteristicsExtension

- **Container to include externally defined semantic units – e.g. for more granularity.**
- Might contain format specific metadata for a file – e.g. technical metadata for still images (MIX)
- Not a replacement for units specified in PREMIS

compositionLevel

- **An indication of whether the object is subject to one or more processes of decoding or unbundling**
- How to describe layers of encodings so they can be correctly reversed?
  - Treat each layer as a "composition level"
  - Repeat description of object characteristics for each composition level
- A file with no compression and no encryption has compositionLevel 0 (zero)
- Each layer of encoding results in new format and incremented compositionLevel
- **Only applies if object is encrypted or compressed**
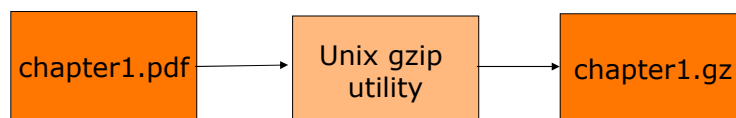- Value is an integer

Files again

- **FILE** = a named and ordered sequence of bytes that is known by an operating system.

  - chapter1.pdf
  - photo.tiff
  - mapofGlasgow.jp2

- Can be zero or more bytes
- Has a file format
- Has access permissions and file system statistics such as size and modification date

PREMIS PREservation Metadata Implementation Strategies

Bitstreams again

- **BITSTREAM** = contiguous or non-contiguous data within a file that has meaningful common properties for preservation purposes.

  - the video stream within an AVI file
  - an image within a TIFF file

- Not known to operating system
- Can be located by starting position within the file
- Can not stand alone as a file without the addition of a header, other structure, or reformatting

---

PREMIS PREservation Metadata Implementation Strategies

But some files aren't that simple

| chapter1.pdf | → | Unix gzip utility | → | chapter1.gz |

- format = PDF
- size = 500,000 bytes
- messageDigest = [something]

- format = gzip
- size = 324,876 bytes
- messageDigest = [something else]

compositionLevel

chapter1.pdf			
composition Level			0
fixity	message Digest Algorithm		SHA-1
fixity	message Digest		[big string]
fixity	message Digest Originato		Submitter
size			500000
format	format Designa-tion	format Name	PDF
format	format Designa-tion	format Version	1.2

chapter1.pdf.gz			
composition Level			1
fixity	message Digest Algorithm		SHA-1
fixity	message Digest		[another string]
fixity	message Digest Originator		Repository
size			324876
format	format Designa-tion	format Name	gzip
format	format Designa-tion	format Version	1.2.3

Ok, but what if you have this:

package.tar

file1.pdf

file2.pdf
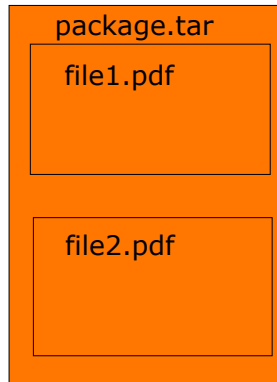
Inside the TAR file, file1 and file2 are simple PDF files.

Neither the containing TAR nor the contained PDFs are encrypted or compressed.

11

Then you have 3 objects!

```
package.tar
  file1.pdf

  file2.pdf
```

package.tar is a file object with compositionLevel 0 and a storageLocation in the file system

file1.pdf is a file object with compositionLevel 0 and a storageLocation as an offset in package.tar

file2.pdf is a file object with compositionLevel 0 and a storageLocation as an offset in package.tar

---

In conclusion

- Remember: Composition level increments only when you have a single file object with multiple successive encodings.

- Bonus question: why aren't the PDF files within package.tar considered bitstream objects?

- Because the PDFs inside the TAR are independently interpretable

Creation information

- **creatingApplication**
  - **Information about the application which created object**
  - Part of objectCharacteristics

- **originalName**
  - **Name of object as submitted to or harvested by repository**
  - Supplements repository supplied names
  - Only applicable to files (but may be extended to representations)

storage

- **How and where the object is stored**
- Container for contentLocation and storageMedium
- May be repeated if more than one identical copy in a different location
- **contentLocation**
  - Information needed to retrieve a file from a system or a bitstream from within a file
  - Subunits type and value
  - Could be fully qualified path or identifier used by storage system; for bitstream a byte offset
- **storageMedium**
  - Physical medium on which the object is stored
  - Useful for media management (e.g. media migration)
  - May be name of system that knows the medium
  - Examples: hard disk, TSM

13

PREMIS PREservation Metadata Implementation Strategies

Example of creation information and storage

creatingApplication
   creatingApplicationName=Adobe Acrobat
   creatingApplicationVersion=5.0
   dateCreatedByApplication=2004
originalName=main.pdf

storage
   contentLocation
      contentLocationType=FDA
      contentLocationValue=fda/prod/data/out/classa/ DF-2005-001002
   storageMedium=3590 [a type of tape unit]

PREMIS PREservation Metadata Implementation Strategies

Environment

- **What is needed to render or use an object**
  - Operating system
  - Application software
  - Computing resources
- Why is obligation optional?
  - Preservation strategies may differ in need for this information (e.g., may be unneeded for bit-level preservation)
  - We currently lack practical methods to collect and store this information
- Relevance to long-term preservation: Ability to render an object and interact with its content may depend on knowing these technical details
- Applies to all types of object (representation, file, bitstream)

PREMIS PREservation Metadata Implementation Strategies

Environment semantic units

- **environmentCharacteristic**
  - Multiple environments can support an object, but often not equally well
  - Suggested values: unspecified, known to work, minimum, recommended
  - Repository does not need to record all possible environments

- **environmentPurpose**
  - Use supported by the specified environment
  - Suggested values: render, edit
  - example: for x.pdf Adobe Acrobat (edit), Adobe Reader (render)

PREMIS PREservation Metadata Implementation Strategies

Environment semantic units (cont.)

- **software and hardware**
  - identify by name, version, type (broad category)
  - Many may apply; at least one should be recorded

- **dependency**
  - non-software component or file needed
  - dependency vs. swDependency
  - e.g. fonts, schemas, stylesheets
  - name and identifier

- **environmentNote**
  - Any additional information
  - Should not be used as substitute for more rigorous description

- **environmentExtension**
  - Replace or extend PREMIS semantical units
  - In an operation environment a link to an appropriate system/emulator can be stored.

15

PREMIS PREservation Metadata Implementation Strategies

Environment example: ETD (PDF file)

- environmentCharacteristic=known to work
- environmentPurpose=render
- software/swName= Mozilla Firefox
- software/swVersion= 1.0
- software/swType=renderer
- swOtherInformation=requires swDependencies as plug-ins
- software/swDependency= Adobe Acrobat Reader 7.0
- software/swDependency= RealPlayer 10
- software/swName= Windows NT
- software/swVersion=5.0
- software/swType=operatingSystem
- hardware/hwName=Intel Pentium II
- hardware/hwType=processor
- dependency/dependencyName=Mathematica 5.2 True Type math fonts

---

PREMIS PREservation Metadata Implementation Strategies

Environment registries

- Information may be complex and increasingly granular
- Information often applies to whole class of objects
- PREMIS does not assume the existence of an environment registry, but defines the information that would be needed in one
- PRONOM has some elements of environment registry
  - for any file extension, gives list of software that can
    - create
    - render
    - identify
    - validate
    - extract metadata from

Digital signatures

- In a transaction, verifies the identify of the sender and that the file was unchanged in transmission.

- Some archives sign stored objects for verification in the future.

- PREMIS digital signature semantic units are based on W3C's *XML Signature Syntax and Processing*
  - *de facto* standard for encoding signature information
  - PREMIS adopts structure/semantics where possible
  - Some departures: e.g., PREMIS permits a given signature to be a property of only 1 object.
  - Version 2 will use XML signatures for signature key

signatureInformation Container

- *Who signed it?*
  - signer (name or pointer to an Agent)

- *How was it signed?*
  - signatureInformationEncoding (e.g., Base64)
  - signatureMethod (e.g., DSA-SHA1)

- *How can we validate it?*
  - signatureValidationRules (could be a pointer to documentation for the validation procedure)
  - signatureProperties (additional information)
  - keyInformation: the signer's public key and other info
    - Type: e.g., DSA, RSA, PGP, etc.
    - Other info: e.g., certificate, revocation list, etc.

- *And of course, the signature itself*

PREMIS PREservation Metadata Implementation Strategies

signatureInformation example

signatureInformation
    signatureInformationEncoding=base64
    signer=Florida Digital Archive
    signatureMethod=RSA-SHA1
    signatureValue=MC0CFFrVLtRlkMc3Daon4BqqnkhCOTFEALE=
    signatureValidationRules=T1=C1
    signatureProperties=2003-03-19T12:25:14-05:00
    keyInformation
        keyType=x509v3-sign-rsa2
        keyValue=<DSAKeyValue>
             keyvalue
             </DSAKeyValue>

PREMIS PRESERVATION METADATA — MAINTENANCE ACTIVITY

Objects Example

This example shows semantic units pertaining to two TIFF files described as File Objects. The complete information package as held in the repository consists of 5 files: 2 TIFF masters, 2 JPEG derivatives, and a METS (XML) file that describes the relationships among them. The METS file is used to dynamically generate the display at http://lcweb2.loc.gov/diglib/ihas/loc.natlib.gottlieb.09601/default.html, which is considered the Intellectual Entity.

A complete description of this information package might consist of 5 File Objects and 2 Representation Objects (for the TIFF and JPEG versions). Only the 2 TIFFs are described in this example.

Objects Example (TIFF file 1)

semantic unit	semantic unit	semantic unit	semantic unit	Value
OBJECT				
objectIdentifier	objectIdentifierType			hdl
objectIdentifier	objectIdentifierValue			loc.music/gottlieb.09601
preservationLevel	preservationLevel Value			full
objectCategory				file
objectCharacteristics	compositionLevel			0
objectCharacteristics	fixity	messageDigestAlgorithm		MD5
objectCharacteristics	fixity	messageDigest		36b03197ad066cd719906c55eb68ab8d
objectCharacteristics	fixity	messageDigestOriginator		LocalDCMS
objectCharacteristics	size			20800896
objectCharacteristics	format	formatDesignation	formatName	image/tiff
objectCharacteristics	format	formatDesignation	formatVersion	6.0
objectCharacteristics	format	formatRegistry	formatRegistryName	PRONOM
objectCharacteristics	format	formatRegistry	formatRegistryKey	fmt/10
objectCharacteristics	format	formatRegistry	formatRegistryRole	specification
objectCharacteristics	significant properties			<NONE>

semantic unit	semantic unit	semantic unit	semantic unit	Value
objectCharacteristics	inhibitors	inhibitorType		<NONE>
objectCharacteristics	inhibitors	inhibitorTarget		<NONE>
objectCharacteristics	inhibitors	inhibitorKey		<NONE>
creatingApplication	creatingApplicationName			ScandAll 21
creatingApplication	creatingApplicationVersion			4.1.4
creatingApplication	DateCreatedByApplication			**1998-10-30**
creatingApplication	creatingApplicationName			Adobe Photoshop
creatingApplication	creatingApplicationVersion			CS2
creatingApplication	DateCreatedByApplication			**1998-10-30T08:29:02**
originalName				0001h.tif
storage	contentLocation	contentLocationType		filepath
storage	contentLocation	contentLocationValue		amserver/
storage	storageMedium			disk
environment	environmentCharacteristic			recommended
environment	environmentPurpose			render
environment	environmentPurpose			edit
environment	environmentNote			<NONE>
environment	dependency	dependencyName		<NONE>
environment	dependency	dependencyIdentifier	dependencyIdentifierType	<NONE>
environment	dependency	dependencyIdentifier	dependencyIdentifierValue	<NONE>
environment	software	swName		Adobe Acrobat
environment	software	swVersion		5.0
environment	software	swType		renderer
environment	software	swOtherInformation		<NONE>
environment	software	swDependency		<NONE>
environment	software	swName		Windows
environment	software	swVersion		XP
environment	software	swType		operatingSystem

semantic unit	semantic unit	semantic unit	semantic unit	Value
environment	software	swOtherInformation		<NONE>
environment	software	swDependency		<NONE>
environment	hardware	hwName		Intel x86
environment	hardware	hwType		processor
environment	hardware	hwOtherInformation		60 mhz minimum
environment	hardware	hwName		64 MB RAM
environment	hardware	hwType		memory
environment	hardware	hwOtherInformation		32 MB minimum
signatureInformation	signatureInformation Encoding			<NONE>
signatureInformation	signatureMethod			<NONE>
signatureInformation	signatureValue			<NONE>
signatureInformation	signatureValidation Rules			<NONE>
signatureInformation	signatureProperties			<NONE>
signatureInformation	keyInformation	keyType		<NONE>
signatureInformation	keyInformation	keyValue		<NONE>
signatureInformation	keyInformation	keyVerificationInformation		<NONE>
relationship	relationshipType			structural
relationship	relationshipSubType			is sibling
relationship	relatedObject Identification	relatedObjectIdentifierType		hdl
relationship	relatedObject Identification	relatedObjectIdentifierValue		loc.music/gottlieb.09602
relationship	relatedObject Identification	relatedObjectSequence		0
relationship	relatedEvent Identification	relatedEventIdentifierType		<NONE>
relationship	relatedEvent Identification	relatedEventIdentifierValue		<NONE>
relationship	relatedEvent Identification	relatedEventSequence		<NONE>
relationship	relationshipType			structural
relationship	relationshipSubType			is sibling
relationship	relatedObject Identification	relatedObjectIdentifierType		hdl

semantic unit	semantic unit	semantic unit	Value
relationship	relatedObject Identification	relatedObjectIdentifierValue	**lcweb2.loc.gov/coocoon/ ihas/loc.natlib.gottlieb.09601/mets.x ml**
relationship	relatedObject Identification	relatedObjectSequence	0
relationship	relatedEvent Identification	relatedEventIdentifierType	<NONE>
relationship	relatedEvent Identification	relatedEventIdentifierValue	<NONE>
relationship	relatedEvent Identification	relatedEventSequence	<NONE>
relationship	relationshipType		derivation
relationship	relationshipSubType		is source of
relationship	relatedObject Identification	relatedObjectIdentifierType	URL
relationship	relatedObject Identification	relatedObjectIdentifierValue	**http://lcweb2.loc.gov/natlib/ihas/servi ce/gottlieb/09601/ver01/0001v.jpg**
relationship	relatedObject Identification	relatedObjectSequence	0
relationship	relatedEvent Identification	relatedEventIdentifierType	LocalRepository
relationship	relatedEvent Identification	relatedEventIdentifierValue	E001.01
relationship	relatedEvent Identification	relatedEventSequence	1
linkingIntellectual EntityIdentifier	linkingIntellectual EntityIdentifierType		hdl
linkingIntellectual EntityIdentifier	linkingIntellectual EntityIdentifierValue		**loc.natlib.gottlieb.09601**
linkingIntellectual EntityIdentifier	linkingIntellectual EntityIdentifierType		URI
linkingIntellectual EntityIdentifier	linkingIntellectual EntityIdentifierValue		http://lcweb2.loc.gov/cocoon/ihas/loc.na tlib.gottlieb.09601/default.html
linkingPermission StatementIdentifier	linkingPermission StatementIdentifier		<NONE>

semantic unit	semantic unit	semantic unit	Value
linkingPermission StatementIdentifier	linkingPermission StatementIdentifier Type		<NONE>
	linkingPermission StatementIdentifier Value		
linkingEventIdentifier	linkingEventIdentifier Type		LocalRepository
linkingEventIdentifier	linkingEventIdentifier Value		E001.1
linkingEventIdentifier	linkingEventIdentifier Type		LocalRepository
linkingEventIdentifier	linkingEventIdentifier Value		E001.2

Objects Example (TIFF file 2)

semantic unit	semantic unit	semantic unit	Value
OBJECT			
objectIdentifier	objectIdentifierType		hdl
objectIdentifier	objectIdentifierValue		loc.music/gottlieb.09602
preservationLevel	preservationLevel Value		full
objectCategory			file
objectCharacteristics	compositionLevel		0
objectCharacteristics	fixity	messageDigestAlgorithm	MD5
objectCharacteristics	fixity	messageDigest	ceb3dbc5dacd3883d0985174ef5df7db
objectCharacteristics	fixity	messageDigestOriginator	LocalDCMS
objectCharacteristics	size		58238300
objectCharacteristics	format	formatDesignation	image/tiff
objectCharacteristics	format	formatDesignation	6.0
objectCharacteristics	format	formatRegistry	PRONOM
objectCharacteristics	format	formatRegistry	fmt/10
objectCharacteristics	format	formatRegistry	specification
objectCharacteristics	significant properties		<NONE>

semantic unit	semantic unit	semantic unit	semantic unit	Value
objectCharacteristics	inhibitors	inhibitorType		<NONE>
objectCharacteristics	inhibitors	inhibitorTarget		<NONE>
objectCharacteristics	inhibitors	inhibitorKey		<NONE>
creatingApplication	creatingApplication Name			ScandAll 21
creatingApplication	creatingApplication Version			4.1.4
creatingApplication	DateCreatedBy Application			**1998-10-30**
creatingApplication	creatingApplication Name			Adobe Photoshop
creatingApplication	creatingApplication Version			CS2
creatingApplication	DateCreatedBy Application			**1998-10-30T09:02:02**
originalName				0002h.tif
storage	contentLocation	contentLocationType		filepath
storage	contentLocation	contentLocationValue		amserver/
storage	storageMedium			disk
environment	environment Characteristic			recommended
environment	environmentPurpose			render
environment	environmentPurpose			edit
environment	environmentNote			<NONE>
environment	dependency	dependencyName		<NONE>
environment	dependency	dependencyIdentifier	dependencyIdentifier Type	<NONE>
environment	dependency	dependencyIdentifier	dependencyIdentifier Value	<NONE>
environment	software	swName		Adobe Acrobat
environment	software	swVersion		5.0
environment	software	swType		renderer
environment	software	swOtherInformation		<NONE>
environment	software	swDependency		<NONE>
environment	software	swName		Windows
environment	software	swVersion		XP
environment	software	swType		operatingSystem

semantic unit	semantic unit	semantic unit	semantic unit	Value
environment	software	swOtherInformation		\<NONE\>
environment	software	swDependency		\<NONE\>
environment	hardware	hwName		Intel x86
environment	hardware	hwType		processor
environment	hardware	hwOtherInformation		60 mhz minimum
environment	hardware	hwName		64 MB RAM
environment	hardware	hwType		memory
environment	hardware	hwOtherInformation		32 MB minimum
signatureInformation	signatureInformationEncoding			\<NONE\>
signatureInformation	signatureMethod			\<NONE\>
signatureInformation	signatureValue			\<NONE\>
signatureInformation	signatureValidationRules			\<NONE\>
signatureInformation	signatureProperties			\<NONE\>
signatureInformation	keyInformation	keyType		\<NONE\>
signatureInformation	keyInformation	keyValue		\<NONE\>
signatureInformation	keyInformation	keyVerificationInformation		\<NONE\>
relationship	relationshipType			structural
relationship	relationshipSubType			is sibling
relationship	relatedObjectIdentification	relatedObjectIdentifierType		hdl
relationship	relatedObjectIdentification	relatedObjectIdentifierValue		**loc.natlib.gottlieb.09601**
relationship	relatedObjectIdentification	relatedObjectSequence		0
relationship	relatedEventIdentification	relatedEventIdentifierType		\<NONE\>
relationship	relatedEventIdentification	relatedEventIdentifierValue		\<NONE\>
relationship	relatedEventIdentification	relatedEventSequence		\<NONE\>
relationship	relationshipType			structural
relationship	relationshipSubType			is sibling
relationship	relatedObjectIdentification	relatedObjectIdentifierType		hdl

semantic unit	semantic unit	semantic unit	semantic unit	Value
relationship	relatedObject Identification	relatedObjectIdentifierValue		lcweb2.loc.gov/coocoon/ihas/loc.natlib.gottlieb.09601/mets.xml
relationship	relatedObject Identification	relatedObjectSequence		0
relationship	relatedEvent Identification	relatedEventIdentifierType		<NONE>
relationship	relatedEvent Identification	relatedEventIdentifierValue		<NONE>
relationship	relatedEvent Identification	relatedEventSequence		<NONE>
relationship	relationshipType			derivation
relationship	relationshipSubType			is source of
relationship	relatedObject Identification	relatedObjectIdentifierType		URL
relationship	relatedObject Identification	relatedObjectIdentifierValue		http://lcweb2.loc.gov/natlib/ihas/service/gottlieb/09601/ver02/0001v.jpg
relationship	relatedObject Identification	relatedObjectSequence		0
relationship	relatedEvent Identification	relatedEventIdentifierType		LocalDCMS
relationship	relatedEvent Identification	relatedEventIdentifierValue		E002.01
relationship	relatedEvent Identification	relatedEventSequence		1
linkingIntellectual EntityIdentifier	linkingIntellectual EntityIdentifierType			hdl
linkingIntellectual EntityIdentifier	linkingIntellectual EntityIdentifierValue			loc.natlib.gottlieb.09601
linkingIntellectual EntityIdentifier	linkingIntellectual EntityIdentifierType			URI
linkingIntellectual EntityIdentifier	linkingIntellectual EntityIdentifierValue			http://lcweb2.loc.gov/coocoon/ihas/loc.natlib.gottlieb.09601/default.html
linkingPermission StatementIdentifier	linkingPermission StatementIdentifier Type			<NONE>

semantic unit	semantic unit	semantic unit	semantic unit	Value
linkingPermission StatementIdentifier	linkingPermission StatementIdentifier Value			<NONE>
linkingEventIdentifier	linkingEventIdentifier Type			LocalRepository
linkingEventIdentifier	linkingEventIdentifier Value			E002

PREMIS | PREservation Metadata Implementation Strategies

AGENTS, RIGHTS, EVENTS

PREMIS | PREservation Metadata Implementation Strategies

Agents

- The Agent entity aggregates information about agents (persons, organizations or software) associated with rights management and/or preservation events in the life of an object.
- Intended only to identify the agent unambiguously, and to allow linking from other entity types.
- Repositories encouraged to use any richer scheme that may be appropriate.

agentIdentifier (mandatory)
    agentIdentifierType (mandatory)
    agentIdentifierValue (mandatory)
agentName (optional)
agentType (optional)

PREMIS PREservation Metadata Implementation Strategies

Examples of agents

agentIdentifier
    agentIdentifierType = lcnaf
    agentIdentifierValue = oca05896076
agentName = Caplan, Priscilla
agentType = person

agentIdentifier
    agentIdentifierType = repositoryX
    agentIdentifierValue = 57
agentName = Timberline Publishing Company
agentType = organization

agentIdentifier
    agentIdentifierType = fda
    agentIdentifierValue = daitss1.4.14
agentName =
agentType = software

PREMIS PREservation Metadata Implementation Strategies

Rights

- The Rights entity aggregates information about statements of rights and permissions

- PREMIS version 1 addressed only narrow scope: what *permissions* have been granted to the repository itself to carry out actions related to objects within the repository

- PREMIS v 2.0 was extended and internationalized

  - Enabled for different legislations
  - Not bound to contracts alone, extended for different law and entitlement systems (e.g. moral rights or Urheberrecht)
  - Distinguishes between rights granted by copyright law, statute or license agreement
  - Extension for other right expressions added
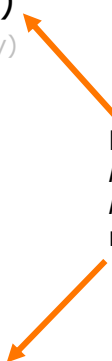
PREMIS PREservation Metadata Implementation Strategies

High level semantic units Rights

rightsStatement (optional)

- rightsStatementIdentifier (mandatory)
- rightsBasis (mandatory)
- copyrightInformation (optional)
- licenseInformation (optional)
- statuteInformation (optional)
- rightsGranted (optional)
- linkingObjectIdentifier (optional)
- linkingAgentIdentifier (optional)

rightsExtension (optional)

PREMIS PREservation Metadata Implementation Strategies

High level semantic units Rights

rightsStatement (optional)

- rightsStatementIdentifier (mandatory)
- rightsBasis (mandatory)
- copyrightInformation (optional)
- licenseInformation (optional)
- statuteInformation (optional)
- rightsGranted (optional)
- linkingObjectIdentifier (optional)
- linkingAgentIdentifier (optional)

Either *rightsStatement* or *rightsExtension* must be present!

rightsExtension (optional)

PREMIS PREservation Metadata Implementation Strategies

rightsBasis

- Designation of the basis for the right or permission
- Values should be taken from a *controlled vocabulary*
- Values: copyright, license, statute.
  - If *rightsBasis* is
  - "copyright", *copyrightInformation* should be provided.
  - "license", *licenseInformation* should be provided.
  - "statute", *statuteInformation* should be provided.

- If more than one basis applies, the entire rights entity should be *repeated*.

PREMIS PREservation Metadata Implementation Strategies

Example rightsBasis and statuteInformation

rightsBasis = statute

statuteInformation

    statuteJurisdiction = de (ISO 3166)

    statuteCitation = „Gesetz über die deutsche Nationalbibliothek vom 22. Juni 2006(DNBG)"
statuteInformationDeterminationDate = 20080901

    statuteNote = Legal Deposit Law in Germany (Applicability to web-published content)

4

PREMIS PREservation Metadata Implementation Strategies

Example rightsBasis and copyrightInformation

rightsBasis = copyright
copyrightInformation
  copyrightStatus = copyrighted
  copyrightJurisdiction = us (taken from ISO 3166)
  copyrightStatusDeterminationDate = 20080910
  copyrightNote = Copyright expiration expected in 2022

PREMIS PREservation Metadata Implementation Strategies

Example rightsBasis and licenseInformation

rightsBasis = license
licenseInformation
  licenseIdentifier
    licenseIdentifierType = fda
    licenseIdentifierValue = 3954
  licenseTerms = The actual license text and other license related information…
  licenseNote = „License embedded in XMP block in file header"

PREMIS PREservation Metadata Implementation Strategies

rightsGranted

rightsGranted (optional)
   act (mandatory)
   restriction (optional)
   termOfGrant (mandatory)
     startDate (mandatory)
     endDate (optional)
   rightsGrantedNote (optional)

rightsStatement
  rightsStatementIdentifier
  rightsBasis
  copyrightInformation
  licenseInformation
  statuteInformation
  rightsGranted
  linkingObjectIdentifier
  linkingAgentIdentifier

PREMIS PREservation Metadata Implementation Strategies

rightsGranted.act

- **The action the repository is granted permission to take**
- Suggested values:
  - replicate = make an exact copy
  - migrate = make a copy identical in content in a different file format
  - modify = make a version different in content
  - use = read without copying or modifying (e.g., to validate a file or run a program)
  - disseminate = create a DIP for use outside of the preservation repository
  - delete = remove from the repository

- Granularity up to the preserving institution
- Useful to employ the same controlled values as in *eventType*.

PREMIS PREservation Metadata Implementation Strategies

rightsGranted.restriction

- A condition or limitation on permissionGranted.act
- For example:

  - act = replicate
  - restriction = no more than 3 copies at any time

  - act = disseminate
  - restriction = rightsholder must be notified after the fact

- Repeatable if there are multiple conditions/limitations

PREMIS PREservation Metadata Implementation Strategies

rightsGranted.termOfGrant

- Beginning and ending dates of permission granted
- ISO 8601 format recommended, if possible
- Examples

      termOfGrant
          startDate = 20050101
          endDate = 20150101

      termOfGrant
           startDate = 1900
          endDate = OPEN

PREMIS PREservation Metadata Implementation Strategies

rightsGranted.rightsGrantedNote

- Defined as additional information about the rights granted

- Statement about risk assessment, for example, when a repository is not certain about what permissions have been granted

- Examples:

  - no contact information for rightsholder found
  - Unclear: restriction on dissemination possible

PREMIS PREservation Metadata Implementation Strategies

Other rightsStatement information

- rightsStatementIdentifier (mandatory)
  - rightsStatementIdentifierType (mandatory)
  - rightsStatementIdentifierValue (mandatory)

- linkingObjectIdentifier (optional)

- linkingAgentIdentifier (optional)

rightsExtension

- Container to include semantic units defined outside of PREMIS
- To replace or extend PREMIS (instead of or in addition)
- If you are using an extension schema, a reference to that schema must be provided!
- If *rightsExtension* container needs to be associated explicitly with any PREMIS subunit under *rights*, the container *rights* is repeated.
- If extensions from different external schemas are needed, *rights* should also be repeated.
  - E.g. with California Digital Library's copyrightMD schema www.cdlib.org/inside/projects/rights/schema/

Events

- The Events entity aggregates information about an action involving one or more Objects
- Recording events can be very important
  - to demonstrate digital provenance
  - to prove that rights have not been violated
  - as an audit trail
  - for problem solving if something goes wrong
  - for billing or reporting
- Judgement calls
  - what exactly are the boundaries of an Event?
  - what actions are worth recording as Events?

PREMIS PREservation Metadata Implementation Strategies

High level semantic units

- eventIdentifier (mandatory)
- eventType (mandatory)
- eventDateTime (mandatory)
- eventDetail (optional)
- eventOutcomeInformation (optional)
- linkingAgentIdentifier (optional)
- linkingObjectIdentifier (optional)

PREMIS PREservation Metadata Implementation Strategies

eventType

- Names the event
- From a controlled vocabulary
- Could use coded values
- Granularity is implementation-specific

Capture	Compression
Deaccession	Decompression
Decryption	Deletion
Dig. signature validation	Dissemination
Fixity check	Ingestion
Message digest calculation	Migration
Normalization	Replication
Validation	Virus check

PREMIS PREservation Metadata Implementation Strategies

eventDetail

- **Additional information about the event**
- Not necessarily intended to be machine-processable, but could be structured to allow this
- For example:

  eventType = dissemination

  eventDetail = A001923;WS;20060413T071530-0500

  [the agent requesting the dissemination; a dissemination type code; and the date/time of the request (which could be different from the time of the actual dissemination itself)]

PREMIS PREservation Metadata Implementation Strategies

eventOutcomeInformation

- Structure

eventOutcomeInformation
    eventOutcome = intended to be coded
    eventOutcomeDetail = more granular information
        eventOutcomeDetailNote = textual form
        eventOutcomeDetailExtension = container to include
                semantic units defined outside of PREMIS
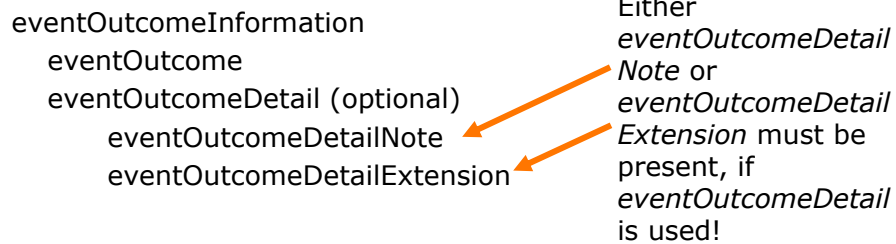
PREMIS PREservation Metadata Implementation Strategies

eventDetail

Structure:

eventOutcomeInformation
   eventOutcome
   eventOutcomeDetail (optional)
     eventOutcomeDetailNote
     eventOutcomeDetailExtension

Either *eventOutcomeDetail Note* or *eventOutcomeDetail Extension* must be present, if *eventOutcomeDetail* is used!

PREMIS PREservation Metadata Implementation Strategies

Examples for eventOutcomeInformation

eventOutcomeInformation
  eventOutcome = 00 [means ok]
    eventOutcomeDetail
    eventOutcomeDetailNote = new file successfully
     created

eventOutcomeInformation
  eventOutcome = FV-S
        [means file validation successful]
    eventOutcomeDetail
     eventOutcomeDetailNote = A4,A14,A19
        [coded list of validation errors found]

12

PREMIS PREservation Metadata Implementation Strategies

linking Events with Agents and Objects

- linkingAgentIdentifier
  - linkingAgentIdentifierType
  - linkingAgentIdentifierValue
  - linkingAgentRole = because the same Agent may
        have different Role in the digital Archive system
- linkingObjectIdentifier
  - linkingObjectIdentifierType
  - linkingObjectIdentifierValue
  - linkingObjectRole

PREMIS PREservation Metadata Implementation Strategies

Event Example:

The TIFF FILES described in the Objects example were deposited into the LC digital repository on July 6, 2006.

This process consists of two Events: validation and ingest.

According to the repository's business rules, the repository itself is considered the Agent of record for all Events.

Therefore, software used to carry out Event processes is recorded in eventDetail.

PR EMIS PREservation Metadata Implementation Strategies

First Event: Validation

eventIdentifier
   eventIdentifierType   LocalRepository
   eventIdentifierValue   E001.1
eventType   Validation
eventDateTime   2007-07-07T01:05:07.001
eventDetail   jhove_1e
eventOutcomeInformation
   eventOutcome   successful
   eventOutcomeDetail
      eventOutcomeDetailNote   well-formed and valid
linkingAgentIdentifier
   linkingAgentIdentifierType   AgentID
   linkingAgentIdentifierValue na12345
   linkingAgentIdentifierRole   Authorizer
linkingObjectIdentifier
   linkingObjectIdentifierType hdl
   linkingObjectIdentifierValue loc.music/gottlieb.09601
   linkingObjectRole   input

PR EMIS PREservation Metadata Implementation Strategies

Second Event: Ingest

eventIdentifier
   eventIdentifierType   LocalRepository
   eventIdentifierValue   E001.2
eventType   Ingest
eventDateTime   2007-07-07T07:07:23.002
eventDetail   ingester1_0.exe
eventOutcomeInformation
   eventOutcome   successful
   eventOutcomeDetail
      eventOutcomeDetailNote   Object transferred to storage
linkingAgentIdentifier
   linkingAgentIdentifierType   AgentID
   linkingAgentIdentifierValue   na12345
   linkingAgentIdentifierRole   Authorizer
linkingObjectIdentifier
   linkingObjectIdentifierType   hdl
   linkingObjectIdentifierValue   loc.music/gottlieb.09601
   linkingObjectRole   source

EVENTS EXAMPLE

The TIFF files described in the Objects example were deposited into the LC repository on July 6, 2006. This process consists of two Events: validation and ingest. According to repository business rules, the repository itself is considered the Agent of record for an Event, so software programs used in this process are documented in eventDetail.

SEMANTIC UNIT	SEMANTIC UNIT	VALUE
eventIdentifier	eventIdentifierType	LocalRepository
eventIdentifier	eventIdentifierValue	E001.1
eventType		validation
eventDateTime		2006-06-06T00:00:00.001
eventDetail		jhove1_1e
eventOutcomeInformation	eventOutcome	successful
eventOutcomeInformation	eventOutcomeDetail	Well-formed and valid
linkingAgentIdentifier	linkingAgentIdentifierType	AgentID
linkingAgentIdentifier	linkingAgentIdentifierValue	na12345
linkingAgentIdentifier	linkingAgentIdentifierRole	<NONE>
linkingObjectIdentifier	linkingObjectIdentifierType	hdl
linkingObjectIdentifier	linkingObjectIdentifierValue	loc.music/gottlieb.09601
linkingObjectIdentifier	linkingObjectRole	<NONE>

SEMANTIC UNIT	SEMANTIC UNIT	VALUE
eventIdentifier	eventIdentifierType	LocalRepository
eventIdentifier	eventIdentifierValue	E001.2
eventType		ingest
eventDateTime		2006-06-06T00:00:00.002
eventDetail		ingester1_0.exe
eventOutcomeInformation	eventOutcome	successful
eventOutcomeInformation	eventOutcomeDetail	<NONE>
linkingAgentIdentifier	linkingAgentIdentifierType	AgentID
linkingAgentIdentifier	linkingAgentIdentifierValue	na12345
linkingAgentIdentifier	linkingAgentIdentifierRole	<NONE>
linkingObjectIdentifier	linkingObjectIdentifierType	hdl
linkingObjectIdentifier	linkingObjectIdentifierValue	loc.music/gottlieb.09601

| linkingObjectIdentifier | linkingObjectRole | <NONE> |

P R E M I S PREservation Metadata Implementation Strategies

IMPLEMENTATION ISSUES

P R E M I S PREservation Metadata Implementation Strategies

How PREMIS can be used

- For systems in development
  - as a basis for metadata definition
- For existing repositories
  - as a checklist for evaluation

"It seems that often people say they aren't ready to implement PREMIS yet, but they don't seem to realise they are already collecting some of the same information that PREMIS describes. The metadata is the same because it is often common sense that it is needed in a repository system. PREMIS can be useful to point out a few extra areas they perhaps hadn't thought of yet."   Deborah Woodyard-Robinson

Implementation issues: models

- Reconciling data models
  - PREMIS data model is for convenience of aggregation
  - Many arbitrary decisions, e.g. is an anomaly discovered during validation a property of the object or an outcome of the validation event?
  - Other data models equally valid, e.g. NLNZ has Process, Object, File, Metadata
  - However: PREMIS encourages consistent application of preservation metadata across different categories of objects (representation, file, bitstream)

- Implementation in relational databases
  - PREMIS data model is not entity-relationship model

Implementation issues: obtaining values

- How to create or obtain metadata values?
  - Most can be populated by program but tools would help
    - JHOVE, NLNZ Metadata Extraction Tool
    - Tool page is available: http://www.loc.gov/standards/premis/tools.html
  - Need registries for format and environment information
    - Pronom, GDFR
- What values to use for controlled vocabularies?
  - In version 1, PREMIS has not had a semantic unit to indicate what controlled vocabulary is used
  - Version 2 introduces a mechanism to document controlled vocabularies (implementation coming)
  - LC will set up registries with starter lists (taken from "suggested values")

PREMIS PREservation Metadata Implementation Strategies

Controlled vocabularies databases

- Library of Congress is establishing databases with controlled vocabulary values for standards that it maintains
- Controlled lists are represented using SKOS as well as alternative syntaxes
- Lists currently in progress:
  - ISO 639-2 and MARC language code list
  - MARC geographic area codes
  - MARC country code list
  - MARC relators
  - PREMIS controlled value lists
  - Thesaurus of Graphic Material
- Other possibilities
  - Enumerated values in MODS schema
  - Coded and uncoded value lists in MARC

Controlled vocabularies in SKOS: example

```
<rdf:Description rdf:about=
   "http://www.loc.gov/standards/registry/vocabulary/pres
   ervationEvents/creation">
<rdf:type rdf:resource=
   "http://www.w3.org/2008/05/skos#Concept"/>
<skos:prefLabel xml:lang="en-latn">
   creation</skos:prefLabel>
<skos:narrower rdf:resource=
   "http://www.loc.gov/standards/registry/vocabulary/pres
   ervationEvents/migration"/>
<skos:narrower rdf:resource=
   "http://www.loc.gov/standards/registry/vocabulary/pres
   ervationEvents/normalization"/>
<skos:definition xml:lang= "en-latn">the act of creating a
   new object</skos:definition>
<skos:inScheme rdf:resource=
   "http://www.loc.gov/standards/registry/vocabulary
   /preservationEvents"/>
</rdf:Description>
```

3

PREMIS — PREservation Metadata Implementation Strategies

Using controlled vocabularies in PREMIS

- Semantic units that specify a controlled vocabulary will become a "concept scheme" in LC's registry
- Each value (now in the PREMIS DD as "suggested values") will be expressed as a SKOS instance in LC's concept database
- Other implementers will be encouraged to add their values within a concept scheme
- There will be a mechanism to import the values into the PREMIS XML schema to enable validation if desired
- If a concept is available in multiple standards, they may be established for broad usage in a concept scheme
  - Example is a scheme for cryptographic hash functions, which are used with different names in PREMIS (messageDigestAlgorithm, MIX and METS (CHECKSUMTYPE)
- LC is exploring an RDF version of PREMIS for semantic web applications

Those wishing to experiment:
http://www.loc.gov:8081/standards/registry/lists.html

PREMIS — PREservation Metadata Implementation Strategies

Implementation issues: conformance

- Conformance is defined in PREMIS Final Report
  - if you use the name, use the definition
  - local metadata can supplement but not modify PREMIS
  - can define more stringent repeatability and obligation but not more liberal

- Meaning of mandatory:
  - you have to know it, and you have to be able to supply it if exporting for exchange
  - you don't have to record it in repository

PREMIS PREservation Metadata Implementation Strategies

Implementation issues: need for additional metadata

- preservation metadata not considered core
  - core = all objects, all preservation strategies
  - example of non-core = installation requirements

- more detailed information on Rights and Agents (but in version 2 Rights are expanded)

- metadata describing Intellectual Entity

- format-specific technical metadata (version 2 supplies an extensible container to include within PREMIS)

- business rules of the repository

- information about the metadata itself (e.g., who obtained or recorded a value, when last changed...)

PREMIS PREservation Metadata Implementation Strategies

XML issues
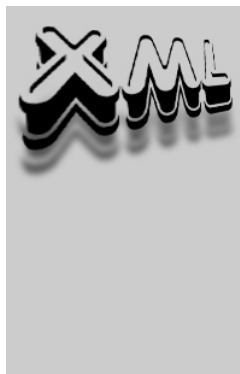
A Brief Introduction to XML

W3C®

Extensible Markup Language (XML) 1.0 (Fourth Edition)

W3C Recommendation 16 August 2006, edited in place 29 September 2006

This version:
http://www.w3.org/TR/2006/REC-xml-20060816
Latest version:
http://www.w3.org/TR/xml
Previous version:
http://www.w3.org/TR/2006/PER-xml-20060614
Editors:
Tim Bray, Textuality and Netscape <tbray@textuality.com>
Jean Paoli, Microsoft <jeanpa@microsoft.com>
C. M. Sperberg-McQueen, W3C <cmsmcq@w3.org>
Eve Maler, Sun Microsystems, Inc. <eve.maler@east.sun.com>
François Yergeau

Please refer to the **errata** for this document, which may include some normative corrections.

The previous errata for this document, are also available.

See also **translations**.

This document is also available in these non-normative formats: XML and XHTML with color-coded revision inc

---

XML Example: Data + Meaning = Information?

start-tag
Element
end-tag

```
<software>
  <swName>Windows</swName>
  <swVersion>
    2000
  </swVersion>
  <swType>Operating System</swType>
</software>
```

Markup

Element

P R E M I S PREservation Metadata Implementation Strategies
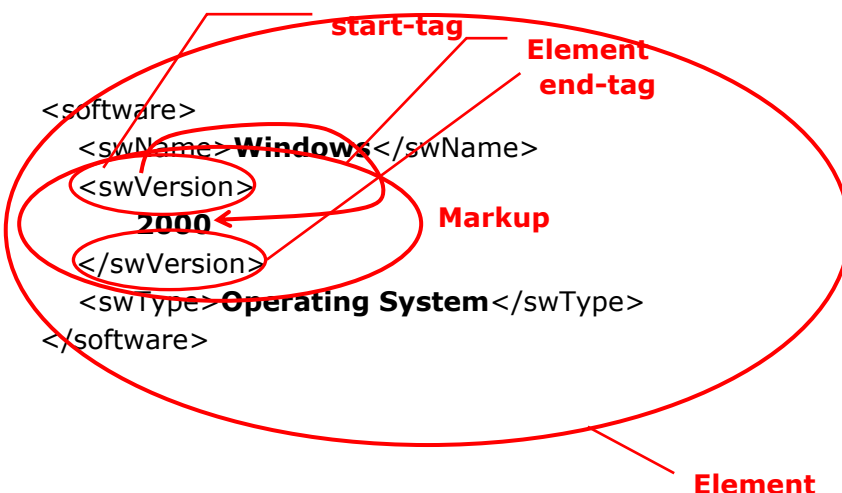
XML: Extensible Markup Language

- A technical approach to convey meaning with data
- Not a natural language, although uses natural languages
  - <姓名>Louis Armstrong</姓名>
  - <name>Louis Armstrong</name>
- Not a programming language
- Language in the sense of:
  - A limited set of tags defines the vocabularies that can be used to markup data
  - The set of tags and their relationships need to be explicitly defined (e.g., in XML schema)
  - We can build software that uses XML as input and process them in a meaningful way
- You can define your own markups and schemas

P R E M I S PREservation Metadata Implementation Strategies

XML Schema Defines:

- What elements may be used?
- Of which types?
- Any attributes?
- In which order?
- Optional or compulsory?
- Repeatable?
- Subelements?
- Logic?
- …

XML Validation

XML Instance

XML Schema

Validator

Valid

Invalid
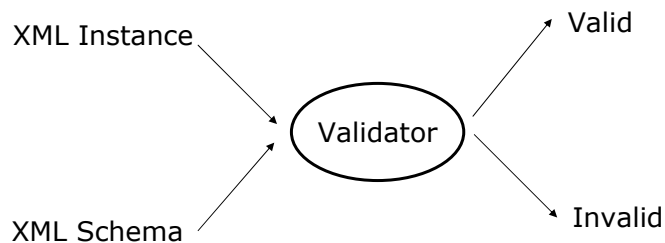
PREMIS Publishes official schemas for validating the XML implementations.

XML Schema Examples

```
<xs:element name="software" minOccurs="0"
   maxOccurs="unbounded">
   <xs:complexType>
        <xs:sequence>
                <xs:element name="swName" minOccurs="1"
   maxOccurs="1" type="xs:string"></xs:element>
                <xs:element name="swVersion" minOccurs="0"
   maxOccurs="1" type="xs:string"></xs:element>
                <xs:element name="swType" minOccurs="1"
   maxOccurs="1" type="xs:string"></xs:element>
                <xs:element name="swOtherInformation"
   minOccurs="0" maxOccurs="unbounded" type="xs:string">
   </xs:element>
                <xs:element name="swDependency" minOccurs="0"
   maxOccurs="unbounded" type="xs:string"> </xs:element>
        </xs:sequence>
   </xs:complexType>
</xs:element>
```

PREMIS PREservation Metadata Implementation Strategies

Will the following XML validate?

```
<software>
    <swName>Windows</swName>
    <swVersion>2000</swVersion>
    <swType>Operating System</swType>
</software>
```

How about this?

```
<swVersion>2000</swVersion>
```

PREMIS PREservation Metadata Implementation Strategies

PREMIS XML schemas

- In version 1, PREMIS in XML was implemented as 5 schema for each PREMIS entity in data model and a container schema
  - Allowed user to choose which parts of PREMIS to use

- Version 2 consists of one schema with each PREMIS entity defined globally so that not all need to be used

- There is an optional PREMIS container if it is desirable to keep some or all PREMIS metadata together

- Semantic units in PREMIS schemas
  - XML is faithful to data dictionary
  - Semantic units for objects may be validated according to the level for which they are applicable (i.e. representation, file, bitstream)

  http://www.loc.gov/standards/premis/premis.xsd

PREMIS — PREservation Metadata Implementation Strategies

Significant changes in XML schema v 2.0

- An abstract object type allows for better validation of object category; objectCategory is not an element
- Defining main elements globally allow for reuse
- Extensibility mechanism is provided for further structure or for schemas from other namespaces
  - significantProperties
  - objectCharacteristics
  - creatingApplication
  - environment
  - signatureInformation
  - eventOutcomeDetail
  - Rights
- Includes definitions for types of date expressions not in W3CDTF, including ISO 8601 basic format (without hyphens) and conventions for special types of dates (e.g. open-ended or questionable dates)

Implementing PREMIS using XML in METS

METS — Metadata Encoding & Transmission Standard
Official Web Site

PREMIS PREservation Metadata Implementation Strategies

METS introduction

- METS records the (possibly hierarchical) structure of digital objects, the names and locations of the files that comprise those objects, and the associated metadata
- A METS document may be a unit of storage (e.g. OAIS AIP) or a transmission format (e.g. OAIS SIP or DIP)
- METS is extensible and modular
- METS uses extension "wrappers" or "sockets" where elements from other schemas can be plugged in
- METS uses the XML Schema facility for combining vocabularies from different Namespaces
- The METS Editorial Board has endorsed PREMIS as an extension schema
- Many institutions trying to use PREMIS within the METS context

---

The structure of a METS file

METS

- fileSec → file inventory
- dmdSec → descriptive metadata
- amdSec → administrative metadata
- behaviorSec → behaviour metadata
- structMap → structural map

PREMIS PREservation Metadata Implementation Strategies
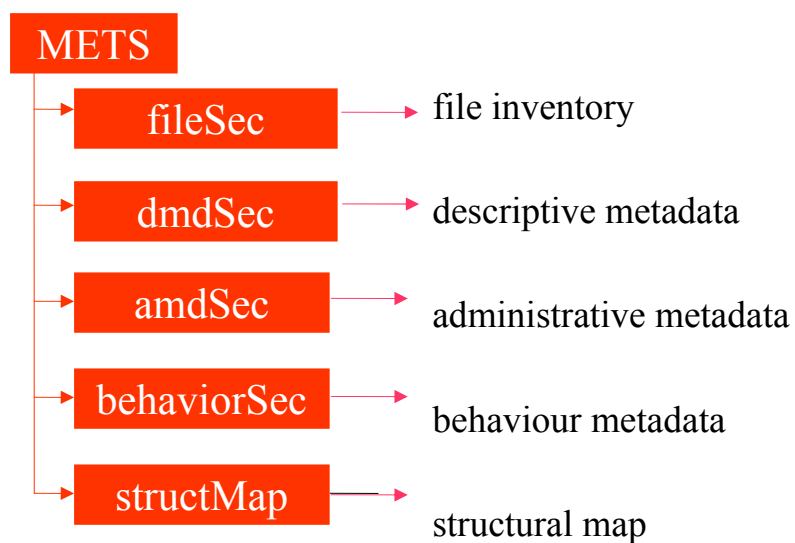
Inserting technical metadata in a METS Document

```
<mets>
  <amdSec>
    <techMD>
      <mdWrap>
        <xmlData>
            <!--   insert data from different namespace here  -->
        </xmlData>
      </mdWrap>
    </techMD>
  </amdSec>
  <fileSec />
  <structMap />
</mets>
```

Linking in METS Documents
(XML ID/IDREF links)

DescMD
　　　mods
　　　　　relatedItem
　　　　　relatedItem

AdminMD
　　techMD
　　sourceMD
　　digiprovMD
　　rightsMD

fileGrp
　　file
　　file

StructMap
　div
　　div
　　　fptr
　　　　div
　　　　　fptr

Linking in METS Documents
(XML ID/IDREF links)

DescMD
    mods
        relatedItem
        relatedItem

AdminMD
    techMD
    sourceMD
    digiprovMD
    rightsMD

fileGrp
    file
    file

StructMap
    div
        div
            fptr
        div
            fptr

Linking in METS Documents
(XML ID/IDREF links)

DescMD
    mods
        relatedItem
        relatedItem

AdminMD
    techMD
    sourceMD
    digiprovMD
    rightsMD

fileGrp
    file
    file

StructMap
    div
        div
            fptr
        div
            fptr

DescMD
mods
relatedItem
relatedItem

AdminMD
techMD
sourceMD
digiprovMD
rightsMD

fileGrp
file
file

StructMap
div
div
fptr

div
fptr

Linking in METS Documents
(XML ID/IDREF links)

DescMD
mods
relatedItem
relatedItem

AdminMD
techMD
sourceMD
digiprovMD
rightsMD

fileGrp
file
file

StructMap
div
div
fptr

div
fptr

PREMIS PREservation Metadata Implementation Strategies

METS extension schemas

- "wrappers" or "sockets" where elements from other schemas can be plugged in
- Provides extensibility
- Uses the XML Schema facility for combining vocabularies from different Namespaces
- Endorsed extension schemas:
  - Descriptive: MODS, DC, MARCXML
  - Technical metadata: MIX (image); textMD (text)
  - Preservation related: PREMIS

PREMIS PREservation Metadata Implementation Strategies

Issues in using PREMIS with METS

- Which METS sections to use and how many
- Whether to record elements redundantly in PREMIS that are defined explicitly in the METS schema
- How to record elements that are also part of a format specific technical metadata schema (e.g. MIX)
- Recording structural relationships
- How to deal with locally controlled vocabularies
- Whether to use the PREMIS container

PREMIS and METS sections

- Flexibility of METS requires implementation decisions
- You can't put all PREMIS metadata directly under amdSec
- What sections to use for PREMIS metadata?
  - Alternative 1
    - Object in techMD
    - Event in digiProvMD
    - Rights in rightsMD
    - Agent with event or rights
  - Alternative 2
    - Everything in digiProvMD
  - Alternative 3
    - Everything in techMD
- How many administrative MD sections to use?
- Experimentation will result in best practices

```
<fileSec><fileGrp>
<file ID="FID1" SIZE="184302" ADMID="TMD1PREMIS TMD1MIX DP1EVENT
   DP1AGENT"
   CHECKSUM="4638bc65c5b9715557d09ad373eefd147382ecbf"
   CHECKSUMTYPE="SHA-1">
<FLocat LOCTYPE="OTHER" xlink:href="BXF22.JPG" />
</file></fileGrp></fileSec>
<techMD ID="TMD1PREMIS">
 <mdWrap MDTYPE="PREMIS">
  <xmlData>
   <premis:object >
    <objectCharacteristics>
     <fixity>
      <messageDigestAlgorithm>SHA-1 </messageDigestAlgorithm>
      <messageDigest>4638bc65c5b9715557d09ad373eefd147382ecbf
       </messageDigest>
      <messageDigestOriginator>EchoDep/messageDigestOriginator>
     </fixity>
     <size>184302</size> </objectCharacteristics>
```

Elements defined in both METS and PREMIS:
- **METS: Checksum, Checksumtype**
  - attribute of <file>
  - not repeatable
- **PREMIS: fixity**
  - also includes messageDigestOriginator
  - allows multiples

16

PREMIS — PREservation Metadata Implementation Strategies

```
<fileSec><fileGrp>
<file ID="FID1" ADMID="TMD1PREMIS DP1EVENT DP1AGENT"
   MIMETYPE="image/jpeg"
<FLocat LOCTYPE="OTHER" xlink:href="BXF22.JPG"/>
</file></fileGrp></fileSec>

<techMD ID="TMD1PREMIS"
  <mdWrap MDTYPE="PREMIS">
   <xmlData>
     <premis:object>
       <objectCharacteristics>
         <format>
           <formatDesignation>
            <formatName>image/jpeg</formatName>
            <formatVersion>1.02 </formatVersion>
           </formatDesignation></format>
       </objectCharacteristics>
```

Elements defined both in METS and PREMIS:

- **METS: MIMETYPE**
  - attribute of <file>
  - optional
- **PREMIS: <format>**
  - more granular; includes name and version (although name may be MIMETYPE)
  - mandatory

PREMIS — PREservation Metadata Implementation Strategies

```
<fileSec> <fileGrp>
 <file ID="FID1" ADMID="TMD1PREMIS TMD1MIX DP1EVENT DP1AGENT">
<techMD ID="TMD1PREMIS">
  <linkingEventIdentifier>
   <linkingEventIdentifierType>ECHODEP Hub Event
   </linkingEventIdentifierType>
   <linkingEventIdentifierValue>echo12345</linkingEventIdentifierValue>
    </linkingEventIdentifier>
<digiprovMD ID="DP1EVENT">
  <premis:event>
    <eventIdentifier>
     <eventIdentifierType>ECHODEP Hub Event</eventIdentifierType>
     <eventIdentifierValue>echo12345 </eventIdentifierValue>
    </eventIdentifier>
    <eventType>ingestion</eventType>
    <eventDateTime>2006-05-02T15:12:53 </eventDateTime></event>
```

Elements defined both in METS and PREMIS

- **METS ID/Idref:** used to associate metadata in different sections and for different files
- **PREMIS identifiers**: explicit linking between entity types

```
<structMap TYPE="physical">
 <div ORDER="1" TYPE="text">
 <:fptr FILEID="FID9"/>
   <div ORDER="1" TYPE="page" LABEL=" Page [1]">
   <fptr FILEID="FID1"/></mets:div>
    <div ORDER="2" TYPE="page" LABEL=" Page [2]">
    <fptr FILEID="FID2"/></mets:div>
  </div>

<relationship>
  <relationshipType>structural</relationshipType>
  <relationshipSubType>is sibling of </relationshipSubType>
 <relatedObjectIdentification>
   <relatedObjectIdentifierType>UCB</relatedObjectIdentifierType>
   <relatedObjectIdentifierValue>FID2</relatedObjectIdentifierValue>
   <relatedObjectSequence>1</relatedObjectSequence>
```

Elements defined both in METS and PREMIS:
- **METS: structMap**
  - details structural relationships and is the heart of the METS document
  - hierarchical, so may be more expressive than PREMIS semantic units
  - links the elements of the structure to content files and metadata
- **PREMIS: <relationship>**
  - details all kinds of relationships, including structural
  - data dictionary says that implementations may record by other means

---

Should semantic units be recorded redundantly?

- Various options are possible when there is overlap between PREMIS and METS or PREMIS and other technical metadata schemas
  - Record only in METS
  - Record only in PREMIS
  - Record in both
- Are there advantages in using PREMIS semantic units?
- Is it important to keep PREMIS metadata together as a unit? There may be an advantage for reuse and maintenance purposes

18

PREMIS PREservation Metadata Implementation Strategies

How to record elements from 2 different technical metadata schemas

- Format specific metadata may be included in addition to PREMIS general technical metadata
- Use multiple techMD sections and specify source in MDType attribute and/or namespace declaration
  - e.g. MDTYPE="NISOIMG" or "PREMIS"
  - Give MIX schema declaration in METS document
- MIX was recently revised to correspond with the revision of the Z39.87 technical metadata for digital still images standard; names harmonized with corresponding PREMIS semantic units
- For digital still images, best practice may be to use PREMIS for general semantic units defined in PREMIS and MIX for format specific units without redundancy

---

PREMIS PREservation Metadata Implementation Strategies

Examples of PREMIS in XML

- **PREMIS in METS:**
  - Portrait of Louis Armstrong (XML) (Library of Congress) Web Presentation of this object
  - Peoria County, Illinois aerial photograph (ECHO Depository, UIUC Grainger Engineering Library)

- **MATHARC implementation:**
  http://pigpen.lib.uchicago.edu:8888/pigpen/uploads/13/asset_descr_mets_premis_02v2.xml

- **UC examples using PREMIS**
  - **Stanford (geospatial and "transfer manifest")**
  - **UCSD (complex object)**
  - **UCB (general METS profile)**

PREMIS PREservation Metadata Implementation Strategies

MPEG-21 Digital Item Declaration (DID)

- ISO/IEC 21000-2: Digital Item Declaration
- A promising alternative to represent Digital Objects
- Starting to get supported by some repositories, e.g., aDORe, DSpace, Fedora
- A flexible and expressive model that easily represents compound objects (recursive "item")
- Attach well-formed XML from persistent namespaces as metadata

---

PREMIS PREservation Metadata Implementation Strategies

Abstract Model for MPEG-21 DID

container: grouping of *items* and descriptor/statement constructs pertaining to the container

item: represents a Digital Item aka Digital Object aka asset. *Descriptor/statement* constructs convey information about the Digital Item

component: binding of *descriptor/statements* to datastreams

resource: datastream
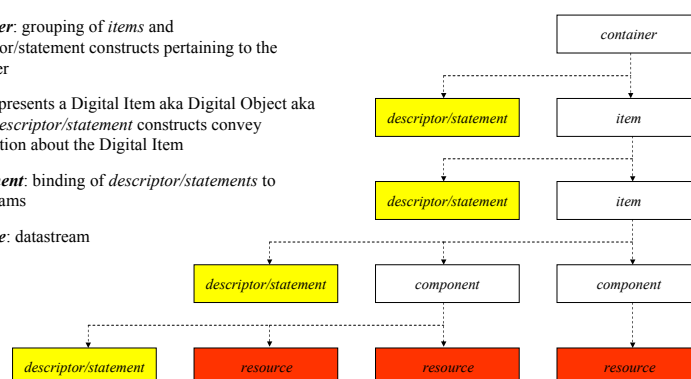
PREMIS PREservation Metadata Implementation Strategies
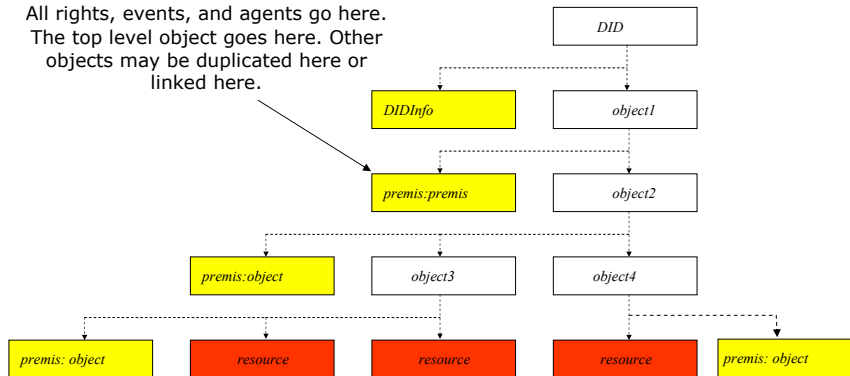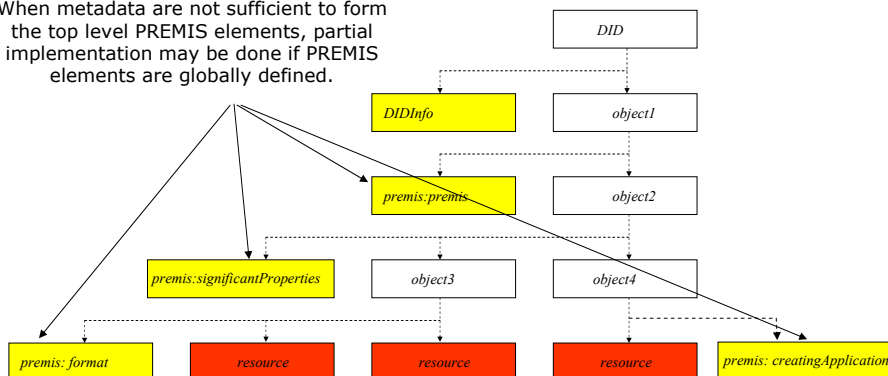
Example of PREMIS in MPEG DID

- **PREMIS in MPEG DID:**
  - **aDORe example** **(LANL)**

PREMIS PREservation Metadata Implementation Strategies

Summary: container formats

- A container format is needed to package together all forms of metadata (of which PREMIS is one) and digital content
- Use of a container is compatible with and an implementation of the OAIS information package concept
- Co-existence with other types of metadata requires best practices for both approaches; redundancy seems to be preferred
- Changes to the next version of the PREMIS XML schemas will facilitate a phased approach to full PREMIS implementation
- Development of registries for controlled vocabularies will benefit implementation
- Tools are being developed to facilitate implementation

PREMIS PREservation Metadata Implementation Strategies

Summary: METS vs. MPEG 21 DID

- METS and MPEG DID are similar types of container formats in that both are expressed in XML, both represent the structure of digital objects, and both include metadata
- MPEG DID doesn't have the segmentation in metadata sections that METS does, so this implementation decision need not be made in DID
- METS is open source and developed by open discussion, mainly cultural heritage community
- MPEG DID is an ISO standard and has industry support, but is often implemented in a proprietary way and standards development is closed
- It would be possible to transform a METS container to a MPEG DID and vice versa; development of stylesheets will enable transformations

PREMIS PREservation Metadata Implementation Strategies

Implementers' panel

- What types of objects are you preserving?
- Has your institution implemented a preservation repository?
- What preservation metadata are you recording?
- How are you recording it, e.g. database, METS/XML, other
- Do you plan to exchange preservation metadata with other repositories?
- Are you planning to or already using PREMIS?
- Which semantic units are most useful?
- Which semantic units are least useful?
- What difficulties have you had applying PREMIS units?

PREMIS at the Swedish National Archives

Karin Bredenberg 2008-09-26

STATENS ARKIV
Riksarkivet och Landsarkiven

Two different approaches

- In our own archival database (in use)

- For our suppliers (in progress)

In our database

- Incorporated in the archival database ARKIS2

- Connected with the archival object of type born digital

- Also used for image files from our scanning projects

ARKIS2

- A Swedish archival management system

- Own development

- All metadata about archives regardless of storage media. (paper as well as digital)

Principal datamodel for ARKIS II

PREMIS implementation

Premis Implementation in Arkis II

Storage Architecture
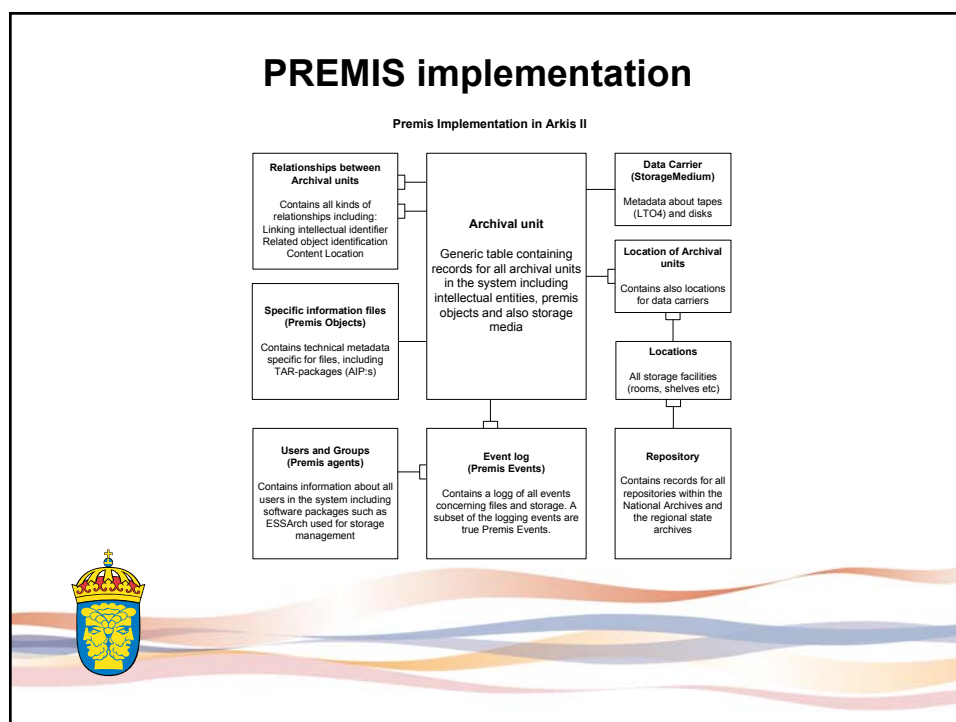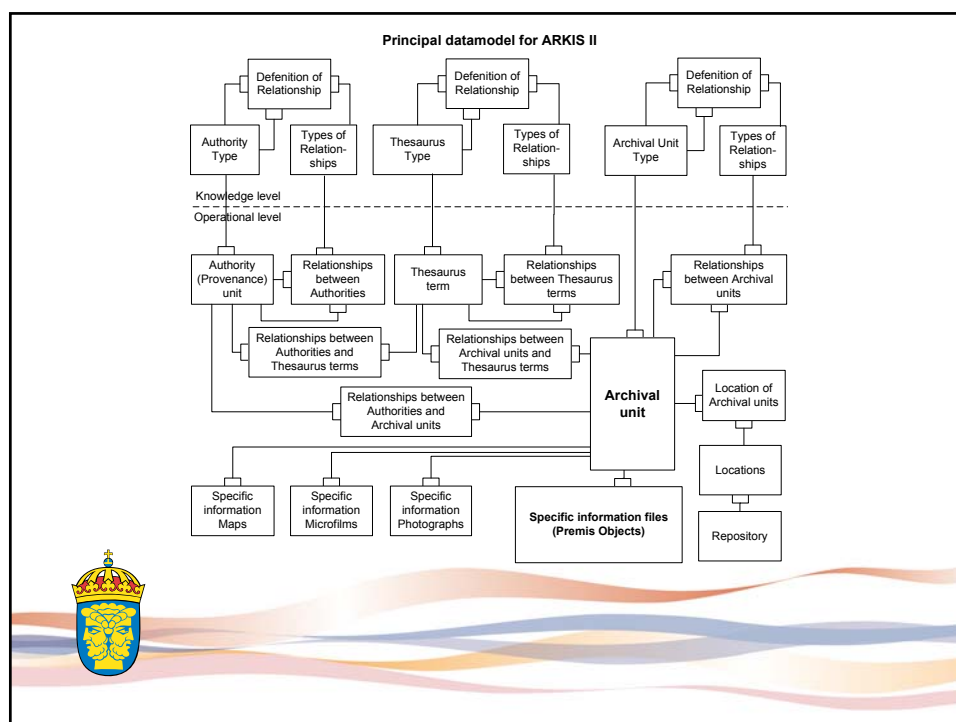
National Archives
Storage Architecture
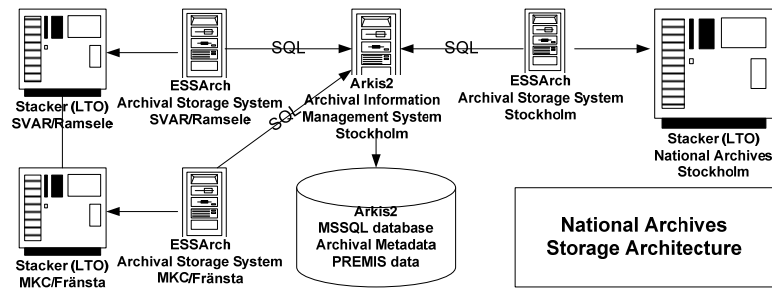
For our suppliers

- A way of letting them describe the digital objects history before we receive it.

- XML included in the AIP.

- A predefined set of taggs that we require.

- Eventually a full PREMIS-file

Suppliers cont.

- Level of **accuracy** depends on decisions of destruction of the data history (e.g. is data about creation, upgrading and so on going to be saved)

- METS to be used as package information. PREMIS inside METS or own file not yet decided

```xml
              TYPE="MODS">
    si:schemaLocation="http://www.loc.gov/mods/v3 http://www.loc.gov/standards/mods/v3/mods-3-3.xsd" ID="ver1">
    fo>
    e>
    of Louis Armstrong, between 1938 and 1948]
    e>
    nfo>
    type="personal">
    mePart>Gottlieb, William P.</mods:namePart>
    mePart type="date">1917-</mods:namePart>
    e>
    oleTerm authority="marcrelator" type="text">creator</mods:roleTerm>
    e>
    e>
    oleTerm type="text">photographer.</mods:roleTerm>
    e>
    e>
    fResource>still image</mods:typeOfResource>
    e authority="marc">photograph</mods:genre>
    e authority="gmgpc">Portrait photographs-1930-1950.</mods:genre>
    e authority="gmgpc">Film negatives-1930-1950.</mods:genre>
    nInfo>
    ce>
    laceTerm type="code" authority="marccountry">xxu</mods:placeTerm>
    ce>
    elssued encoding="marc" point="start">1938</mods:dateIssued>
    elssued encoding="marc" point="end">1948</mods:dateIssued>
    elssued encoding="marc" point="start" qualifier="questionable">1938</mods:dateIssued>
    elssued encoding="marc" point="end" qualifier="questionable">1948</mods:dateIssued>
    uance>monographic</mods:issuance>
    nInfo>
    calDescription>
    n authority="gmd">graphic</mods:form>
    ent>1 negative : b&w ; 3 1/4 x 4 1/4 in.</mods:extent>
    icalDescription>
    Gottlieb Collection Assignment No. 040</mods:note>
    Original negative and contact print not served.</mods:note>
    Purchase William P. Gottlieb</mods:note>
    ype="version">original negative</mods:note>
    ct authority="lcsh">
    ne type="personal">
    Part>Armstrong, Louis, 1900-1971</mods:namePart>
```

on>
sicalLocation displayLabel="negative">
Congress Prints & Photographs Division Washington D.C. 20540 USA
ysicalLocation>
on>
sicalLocation displayLabel="contact print">
Congress Prints & Photographs Division Washington D.C. 20540 USA
ysicalLocation>
ion>
ifier type="stock number">LC-GLB13-0960 DLC</mods:identifier>
sCondition type="restrictionOnAccess">Original negative and contact print not served. </mods:accessCondition>
dInfo>
ordContentSource authority="marcorg">DLC</mods:recordContentSource>
ordCreationDate encoding="marc">990119</mods:recordCreationDate>
ordChangeDate encoding="iso8601">19990520104721.0</mods:recordChangeDate>
ordIdentifier source="DLC">got99000960</mods:recordIdentifier>
rdInfo>
dItem type="otherVersion" ID="ver2">
e type="version">contact print with annotations</mods:note>
dItem>

bject1">
DTYPE="PREMIS">
>
ct xsi:type="premis:file" xsi:schemaLocation="info:lc/xmlns/premis-v2 http://www.loc.gov/standards/premis/v2/premis-v2-0.xsd">
objectIdentifier>
objectIdentifierType>hdl</premis:objectIdentifierType>
objectIdentifierValue>loc.music/gottlieb.09601</premis:objectIdentifierValue>
objectIdentifier>
eservationLevel>
preservationLevelValue>full</premis:preservationLevelValue>
preservationLevelDateAssigned>20070529</premis:preservationLevelDateAssigned>
reservationLevel>
gnificantProperties>
significantPropertiesType> behavior </premis:significantPropertiesType>
significantPropertiesValue> hyperlinks traversable </premis:significantPropertiesValue>
gnificantProperties>
jectCharacteristics>

```xml
      <s:formatRegistry>
        <premis:formatRegistryName>PRONOM</premis:formatRegistryName>
        <premis:formatRegistryKey>fmt/10</premis:formatRegistryKey>
        <premis:formatRegistryRole>specification</premis:formatRegistryRole>
      </premis:formatRegistry>
    </premis:format>
    <premis:creatingApplication>
      <premis:creatingApplicationName>ScandAll 21</premis:creatingApplicationName>
      <premis:creatingApplicationVersion>4.1.4</premis:creatingApplicationVersion>
      <premis:dateCreatedByApplication>1998-10-30</premis:dateCreatedByApplication>
    </premis:creatingApplication>
    <premis:creatingApplication>
      <premis:creatingApplicationName>Adobe Photoshop</premis:creatingApplicationName>
      <premis:creatingApplicationVersion>CS2</premis:creatingApplicationVersion>
      <premis:dateCreatedByApplication>1998-10-30T08:29:02</premis:dateCreatedByApplication>
    </premis:creatingApplication>
    <premis:objectCharacteristicsExtension>
      <mix:mix xsi:schemaLocation="http://www.loc.gov/mix/v20 http://www.loc.gov/standards/mix/mix20/mix20.xsd">
        <mix:BasicDigitalObjectInformation>
          <mix:byteOrder>little endian</mix:byteOrder>
          <mix:Compression>
            <mix:compressionScheme>1</mix:compressionScheme>
          </mix:Compression>
        </mix:BasicDigitalObjectInformation>
        <mix:BasicImageInformation>
          <mix:BasicImageCharacteristics>
            <mix:imageWidth>3982</mix:imageWidth>
            <mix:imageHeight>5223</mix:imageHeight>
            <mix:PhotometricInterpretation>
              <mix:colorSpace>1</mix:colorSpace>
            </mix:PhotometricInterpretation>
          </mix:BasicImageCharacteristics>
        </mix:BasicImageInformation>
        <mix:ImageCaptureMetadata>
          <mix:GeneralCaptureInformation>
            <mix:dateTimeCreated>1998-10-03T08:25:28</mix:dateTimeCreated>
            <mix:imageProducer>Library of Congress</mix:imageProducer>
          </mix:GeneralCaptureInformation>
          <mix:orientation>normal*</mix:orientation>
        </mix:ImageCaptureMetadata>
        <mix:ImageAssessmentMetadata>
          <mix:SpatialMetrics>
            <mix:samplingFrequencyUnit>no absolute unit of measurement</mix:samplingFrequencyUnit>
```

      <mix:bitsPerSampleValue>8</mix:bitsPerSampleValue>
    </mix:BitsPerSample>
    <mix:samplesPerPixel>1</mix:samplesPerPixel>
  <mix:ImageColorEncoding>
  </mix:ImageAssessmentMetadata>
</mix>
    </objectCharacteristicsExtension>
  </objectCharacteristics>
  <originalName>0001h.tif</premis:originalName>
  </storage>
  <contentLocation>
    <premis:contentLocationType>filepath</premis:contentLocationType>
    <premis:contentLocationValue>amserver</premis:contentLocationValue>
  </contentLocation>
    <storageMedium>disk</premis:storageMedium>
  </storage>
  <environment>
    <environmentCharacteristic>recommended</premis:environmentCharacteristic>
    <environmentPurpose>render</premis:environmentPurpose>
    <environmentPurpose>edit</premis:environmentPurpose>
    <software>
      <premis:swName>Adobe Acrobat</premis:swName>
      <premis:swVersion>5.0</premis:swVersion>
      <premis:swType>renderer</premis:swType>
    </software>
    <software>
      <premis:swName>Windows</premis:swName>
      <premis:swVersion>XP</premis:swVersion>
      <premis:swType>operatingSystem</premis:swType>
    </software>
    <hardware>
      <premis:hwName>Intel x86</premis:hwName>
      <premis:hwType>processor</premis:hwType>
      <premis:hwOtherInformation>60 mhz minimum</premis:hwOtherInformation>
    </hardware>
    <hardware>
      <premis:hwName>64 MB RAM</premis:hwName>
      <premis:hwType>memory</premis:hwType>
      <premis:hwOtherInformation>32 MB minimum</premis:hwOtherInformation>
    </hardware>
    <environmentExtension>
      <hardwareInformation/>
      <softwareInformation/>

```
lationship>
relationshipType>structural</premis:relationshipType>
relationshipSubType>is sibling</premis:relationshipSubType>
relatedObjectIdentification>
s:relatedObjectIdentifierType>URI</premis:relatedObjectIdentifierType>
is:relatedObjectIdentifierValue>
"lcweb2.loc.gov/cocoon/ihas/loc.natlib.gottlieb.09601/mets.xml
is:relatedObjectSequence>0</premis:relatedObjectSequence>
:relatedObjectIdentification>
lationship>
lationship>
relationshipType>derivation</premis:relationshipType>
relationshipSubType>is source of</premis:relationshipSubType>
relatedObjectIdentification>
s:relatedObjectIdentifierType>URL</premis:relatedObjectIdentifierType>
s:relatedObjectIdentifierValue>
"lcweb2.loc.gov/natlib/ihas/service/gottlieb/09601/ver01/0001v.jpg
is:relatedObjectIdentifierValue>
s:relatedObjectSequence>0</premis:relatedObjectSequence>
:relatedObjectIdentification>
relatedEventIdentification>
s:relatedEventIdentifierType>LocalDCMS</premis:relatedEventIdentifierType>
s:relatedEventIdentifierValue>E001.01</premis:relatedEventIdentifierValue>
s:relatedEventSequence>1</premis:relatedEventSequence>
relatedEventIdentification>
lationship>
nkingEventIdentifier>
linkingEventIdentifierType>Local Repository</premis:linkingEventIdentifierType>
linkingEventIdentifierValue>E001</premis:linkingEventIdentifierValue>
nkingEventIdentifier>
kingIntellectualEntityIdentifier>
linkingIntellectualEntityIdentifierType>hdl</premis:linkingIntellectualEntityIdentifierType>
linkingIntellectualEntityIdentifierValue>loc.natlib.gottlieb.09601</premis:linkingIntellectualEntityIdentifierValue>
kingIntellectualEntityIdentifier>
kingIntellectualEntityIdentifier>
linkingIntellectualEntityIdentifierType>URI</premis:linkingIntellectualEntityIdentifierType>
linkingIntellectualEntityIdentifierValue>
web2.loc.gov/cocoon/ihas/loc.natlib.gottlieb.09601/default.html
:linkingIntellectualEntityIdentifierValue>
nkingIntellectualEntityIdentifier>
ect>
>
```

```xml
     preservationLevelDateAssigned>20070529</premis:preservationLevelDateAssigned>
   reservationLevel>
 gnificantProperties>
   significantPropertiesType> behavior </premis:significantPropertiesType>
   significantPropertiesValue> hyperlinks traversable </premis:significantPropertiesValue>
 gnificantProperties>
 bjectCharacteristics>
   compositionLevel>0</premis:compositionLevel>
 fixity>
   s:messageDigestAlgorithm>MD5</premis:messageDigestAlgorithm>
   s:messageDigest>ceb3dbc5dacd3883d0985174ef5df7db</premis:messageDigest>
   s:messageDigestOriginator>LocalDCMS</premis:messageDigestOriginator>
 :fixity>
   size>58238300</premis:size>
 format>
   s:formatDesignation>
   mis:formatName>image/tiff</premis:formatName>
   mis:formatVersion>6.0</premis:formatVersion>
   is:formatDesignation>
   s:formatRegistry>
   mis:formatRegistryName>PRONOM</premis:formatRegistryName>
   mis:formatRegistryKey>fmt/10</premis:formatRegistryKey>
   mis:formatRegistryRole>specification</premis:formatRegistryRole>
   is:formatRegistry>
 :format>
   creatingApplication>
   s:creatingApplicationName>ScandAll 21</premis:creatingApplicationName>
   s:creatingApplicationVersion>4.1.4</premis:creatingApplicationVersion>
   s:dateCreatedByApplication>1998-10-30</premis:dateCreatedByApplication>
   :creatingApplication>
   creatingApplication>
   s:creatingApplicationName>Adobe Photoshop</premis:creatingApplicationName>
   s:creatingApplicationVersion>CS2</premis:creatingApplicationVersion>
   s:dateCreatedByApplication>1998-10-30T08:29:02</premis:dateCreatedByApplication>
   :creatingApplication>
   objectCharacteristicsExtension>
mix xsi:schemaLocation="http://www.loc.gov/mix/v20 http://www.loc.gov/standards/mix/mix20/mix20.xsd">
   :BasicDigitalObjectInformation>
   nix:byteOrder>little endian</mix:byteOrder>
   nix:Compression>
   <mix:compressionScheme>1</mix:compressionScheme>
   nix:Compression>
   c:BasicDigitalObjectInformation>
```

```xml
                <mix:denominator>1</mix:denominator>
              </mix:footroom>
            - <mix:headroom>
                <mix:numerator>0</mix:numerator>
                <mix:denominator>1</mix:denominator>
              </mix:headroom>
            </mix:Component>
          - <mix:Component>
              <mix:componentPhotometricInterpretation>G</mix:componentPhotometricInterpretation>
            - <mix:footroom>
                <mix:numerator>255</mix:numerator>
                <mix:denominator>1</mix:denominator>
              </mix:footroom>
            - <mix:headroom>
                <mix:numerator>0</mix:numerator>
                <mix:denominator>1</mix:denominator>
              </mix:headroom>
            </mix:Component>
          - <mix:Component>
              <mix:componentPhotometricInterpretation>B</mix:componentPhotometricInterpretation>
            - <mix:footroom>
                <mix:numerator>255</mix:numerator>
                <mix:denominator>1</mix:denominator>
              </mix:footroom>
            - <mix:headroom>
                <mix:numerator>0</mix:numerator>
                <mix:denominator>1</mix:denominator>
              </mix:headroom>
            </mix:Component>
          </mix:ReferenceBlackWhite>
        </mix:PhotometricInterpretation>
      </mix:BasicImageCharacteristics>
    </mix:BasicImageInformation>
  - <mix:ImageCaptureMetadata>
    - <mix:GeneralCaptureInformation>
        <mix:dateTimeCreated>1998-10-30T08:29:02</mix:dateTimeCreated>
        <mix:imageProducer>Library of Congress</mix:imageProducer>
      </mix:GeneralCaptureInformation>
      <mix:orientation>normal*</mix:orientation>
    </mix:ImageCaptureMetadata>
  - <mix:ImageAssessmentMetadata>
    - <mix:SpatialMetrics>
        <mix:samplingFrequencyUnit>no absolute unit of measurement</mix:samplingFrequencyUnit>
```

```xml
<mix:bitsPerSampleValue>8</mix:bitsPerSampleValue>
</mix:BitsPerSample>
<mix:samplesPerPixel>1</mix:samplesPerPixel>
nix:ImageColorEncoding>
x:ImageAssessmentMetadata>
nix>
:objectCharacteristicsExtension>
bjectCharacteristics>
iginalName>0002h.tif</premis:originalName>
orage>
contentLocation>
s:contentLocationType>filepath</premis:contentLocationType>
s:contentLocationValue>amserver/</premis:contentLocationValue>
:contentLocation>
storageMedium>disk</premis:storageMedium>
torage>
vironment>
environmentCharacteristic>recommended</premis:environmentCharacteristic>
environmentPurpose>render</premis:environmentPurpose>
environmentPurpose>edit</premis:environmentPurpose>
software>
s:swName>Adobe Acrobat</premis:swName>
s:swVersion>5.0</premis:swVersion>
s:swType>renderer</premis:swType>
:software>
software>
s:swName>Windows</premis:swName>
s:swVersion>XP</premis:swVersion>
s:swType>operatingSystem</premis:swType>
:software>
hardware>
s:hwName>Intel x86</premis:hwName>
s:hwType>processor</premis:hwType>
s:hwOtherInformation>60 mhz minimum</premis:hwOtherInformation>
:hardware>
hardware>
s:hwName>64 MB RAM</premis:hwName>
s:hwType>memory</premis:hwType>
s:hwOtherInformation>32 MB minimum</premis:hwOtherInformation>
:hardware>
environmentExtension>
varelnformation/>
aelnformation/>
```

lationship>
  relationshipType>structural</premis:relationshipType>
  relationshipSubType>is sibling</premis:relationshipSubType>
  relatedObjectIdentification>
    s:relatedObjectIdentifierType>URI</premis:relatedObjectIdentifierType>
    s:relatedObjectIdentifierValue>
    "lcweb2.loc.gov/cocoon/ihas/loc.natlib.gottlieb.09601/mets.xml
    is:relatedObjectIdentifierValue>
    s:relatedObjectSequence>0</premis:relatedObjectSequence>
  :relatedObjectIdentification>
lationship>
lationship>
  relationshipType>derivation</premis:relationshipType>
  relationshipSubType>is source of</premis:relationshipSubType>
  relatedObjectIdentification>
    s:relatedObjectIdentifierType>URL</premis:relatedObjectIdentifierType>
    s:relatedObjectIdentifierValue>
    "lcweb2.loc.gov/natlib/ihas/service/gottlieb/09601/ver01/0001v.jpg
    is:relatedObjectIdentifierValue>
    s:relatedObjectSequence>0</premis:relatedObjectSequence>
  :relatedObjectIdentification>
  relatedEventIdentification>
    s:relatedEventIdentifierType>LocalDCMS</premis:relatedEventIdentifierType>
    s:relatedEventIdentifierValue>E001.01</premis:relatedEventIdentifierValue>
    s:relatedEventSequence>1</premis:relatedEventSequence>
  :relatedEventIdentification>
lationship>
nkingEventIdentifier>
  linkingEventIdentifierType>Local Repository</premis:linkingEventIdentifierType>
  linkingEventIdentifierValue>E001</premis:linkingEventIdentifierValue>
nkingEventIdentifier>
kingIntellectualEntityIdentifier>
  linkingIntellectualEntityIdentifierType>hdl</premis:linkingIntellectualEntityIdentifierType>
  linkingIntellectualEntityIdentifierValue>loc.natlib.gottlieb.09601</premis:linkingIntellectualEntityIdentifierValue>
nkingIntellectualEntityIdentifier>
kingIntellectualEntityIdentifier>
  linkingIntellectualEntityIdentifierType>URI</premis:linkingIntellectualEntityIdentifierType>
  linkingIntellectualEntityIdentifierValue>
  web2.loc.gov/cocoon/ihas/loc.natlib.gottlieb.09601/default.html
  :linkingIntellectualEntityIdentifierValue>
nkingIntellectualEntityIdentifier>
ect>
>

```xml
            entDetail>jhove1_1e</premis:eventDetail>
         entOutcomeInformation>
            eventOutcome>successful</premis:eventOutcome>
            eventOutcomeDetail>
               s:eventOutcomeDetailNote>Well-formed and valid</premis:eventOutcomeDetailNote>
               s:eventOutcomeDetailExtension>
                  fileInfo>
                     n/>
                     ut/>
                  fileInfo>
               is:eventOutcomeDetailExtension>
            :eventOutcomeDetail>
         ventOutcomeInformation>
         kingAgentIdentifier>
            linkingAgentIdentifierType>AgentID</premis:linkingAgentIdentifierType>
            linkingAgentIdentifierValue>na12345</premis:linkingAgentIdentifierValue>
         nkingAgentIdentifier>
         kingObjectIdentifier>
            linkingObjectIdentifierType>hdl</premis:linkingObjectIdentifierType>
            linkingObjectIdentifierValue>loc.music/gottlieb.09601</premis:linkingObjectIdentifierValue>
         kingObjectIdentifier>
      nt>
   e>

      >
      ID="event2">
      DTYPE="PREMIS">
      >
   nt>
      entIdentifier>
         eventIdentifierType>LocalRepository</premis:eventIdentifierType>
         eventIdentifierValue>E001.2</premis:eventIdentifierValue>
      entIdentifier>
      ventType>ingestion</premis:eventType>
      ventDateTime>2006-06-06T00:00:00.002</premis:eventDateTime>
      entDetail>ingester1_0.exe</premis:eventDetail>
      ventOutcomeInformation>
         eventOutcome>successful</premis:eventOutcome>
      ventOutcomeInformation>
      kingAgentIdentifier>
         linkingAgentIdentifierType>AgentID</premis:linkingAgentIdentifierType>
         linkingAgentIdentifierValue>na12345</premis:linkingAgentIdentifierValue>
      k1ngAgentIdentifier>
```

t>
entIdentifier>
eventIdentifierType>LocalRepository</premis:eventIdentifierType>
eventIdentifierValue>E001.3</premis:eventIdentifierValue>
ventIdentifier>
entType>validation</premis:eventType>
entDateTime>2006-06-06T00:00:00.005</premis:eventDateTime>
entDetail>jhove1_1e</premis:eventDetail>
entOutcomeInformation>
eventOutcome>successful</premis:eventOutcome>
eventOutcomeDetail>
s:eventOutcomeDetailNote>Well-formed and valid</premis:eventOutcomeDetailNote>
:eventOutcomeDetail>
ventOutcomeInformation>
nkingAgentIdentifier>
linkingAgentIdentifierType>AgentID</premis:linkingAgentIdentifierType>
linkingAgentIdentifierValue>na12345</premis:linkingAgentIdentifierValue>
nkingAgentIdentifier>
kingObjectIdentifier>
linkingObjectIdentifierType>hdl</premis:linkingObjectIdentifierType>
linkingObjectIdentifierValue>loc.music/gottlieb.09602</premis:linkingObjectIdentifierValue>
nkingObjectIdentifier>
nt>
>

ID="event4">
DTYPE="PREMIS">
>
t>
entIdentifier>
eventIdentifierType>LocalRepository</premis:eventIdentifierType>
eventIdentifierValue>E001.4</premis:eventIdentifierValue>
ventIdentifier>
entType>ingestion</premis:eventType>
entDateTime>2006-06-06T00:00:00.006</premis:eventDateTime>
entDetail>ingester1_0.exe</premis:eventDetail>
entOutcomeInformation>
eventOutcome>successful</premis:eventOutcome>
ventOutcomeInformation>
kingAgentIdentifier>
linkingAgentIdentifierType>AgentID</premis:linkingAgentIdentifierType>
kingAgentIdentifierValue>na12345</premis:linkingAgentIdentifierValue>

```
t>
entIdentifier>
agentIdentifierType>AgentID</premis:agentIdentifierType>
agentIdentifierValue>na12345</premis:agentIdentifierValue>
gentIdentifier>
ntName>LC Repository</premis:agentName>
entType>organization</premis:agentType>
nt>
>
>

="MASTER">
YPE="image/tiff" GROUPID="G1" ID="masterd1e102963" ADMID="object1 agent1 event1 event2">
.OCTYPE="URL" xlink:href="http://lcweb2.loc.gov/natlib/ihas/warehouse/gottlieb/09601/ver01/0001.tif"/>

YPE="image/tiff" GROUPID="G1" ID="masterd1e102965" ADMID="object2 agent1 event3 event4">
.OCTYPE="URL" xlink:href="http://lcweb2.loc.gov/natlib/ihas/warehouse/gottlieb/09601/ver02/0001.tif"/>

="SERVICE">
YPE="image/jpeg" GROUPID="G1" ID="serviced1e102963">
.OCTYPE="URL" xlink:href="http://lcweb2.loc.gov/natlib/ihas/service/gottlieb/09601/ver01/0001v.jpg"/>

YPE="image/jpeg" GROUPID="G1" ID="serviced1e102965">
.OCTYPE="URL" xlink:href="http://lcweb2.loc.gov/natlib/ihas/service/gottlieb/09601/ver02/0001v.jpg"/>

MODS" TYPE="photo:photoObject">
"photo:version" DMDID="ver1">
="photo:image">
LEID="masterd1e102963"/>
LEID="serviced1e102963"/>

"photo:version" DMDID="ver2">
="photo:image">
LEID="masterd1e102965"/>
EID="serviced1e102965"/>
```

PREMIS
PRESERVATION METADATA — MAINTENANCE ACTIVITY

Data Dictionary for Preservation Metadata: PREMIS Seminar at DC2008
26 September 2008

Evaluation

Do you work in a:
      library   archives   data center   other _____

Is your institution implementing a preservation repository now or in the future?
      yes   no   maybe   don't know

Coming into this Tutorial, were you familiar with preservation metadata?
      very   moderately   a little   not at all

Please rate this tutorial in how effective it was in the following areas, using the key:
    E = Excellent, G = Good, F = Fair, P = Poor, N = Not applicable

1. Communicating the purpose and uses of PREMIS.	E	G	F	P	N
2. Presenting an overview of the PREMIS data model.	E	G	F	P	N
3. Communicating a basic understanding of the semantic units pertaining to Objects.	E	G	F	P	N
4. Communicating a basic understanding of the semantic units pertaining to Agents, Rights and Events.	E	G	F	P	N
5. Presenting an overview of implementation issues.	E	G	F	P	N
6. Describing how PREMIS semantic units can be expressed in XML.	E	G	F	P	N
7. Responding to the questions of the participants.	E	G	F	P	N

8. The examples were:	very helpful	helpful	not helpful
8. The implementers' panel was:	very useful	useful	not useful
9. The overall pace of instruction was:	too fast	about right	too slow
10. The level of detail was:	too much	about right	not enough

Please write any comments or suggestions for improvement on the back of this sheet.
Thank you!