PREMIS PREservation Metadata Implementation Strategies

PREMIS Tutorial

We have only 90 minutes
this is going to be an introduction
to PREMIS!

Karin Bredenberg
Kommunalförbundet
Sydarkivera

Eld Zierau
Royal Danish Library

Micky Lindlar
TIB – Leibniz Information
Centre for Science and
Technology

Purpose of the Introduction (Tutorial)

- Introduce PREMIS
- Set a baseline of common understanding
- Introduce you to the PREMIS community

Who are we?

From the PREMIS Editorial Committee, who are the international team of experts maintaining PREMIS:

- Eld
- Karin
- Micky

We know that you in the audience will answer yes or no:

- Have you heard of PREMIS?
- Do you know the PREMIS data model?
- Have you used PREMIS in practice?
- Have you modelled environments?
- Are you from
  - a library?
  - an archive?
  - a university?
  - Something else?

Agenda

20:00-20:10 **Introduction to PREMIS**

Welcome
Background (brief history and rationale)
Benefits of implementing PREMIS
Website, PIG, id.loc.gov

20:10-21:20 **Introduction to PREMIS**

Outline of main Entities
Data Dictionary

Conformance and Interoperability

PREMIS Conformance
Repository interoperability

21:20-21.30 **Wrap Up**

Introduction to exercise (Objects, Events, Agents, Rights) for home
Answers to questions

PREMIS PREservation Metadata Implementation Strategies

Background (brief history and rationale)

Karin Bredenberg
Kommunalförbundet Sydarkivera

DIGITAL PRESERVATION METADATA - WHY IS IT NEEDED AND WHAT DOES IT LOOK LIKE?

What is digital preservation metadata?

- Digital preservation metadata =

  Metadata to ensure long-term accessibility of digital resources

- Digital objects must be self-descriptive

- Must be able to describe, manage and discover independently from the systems that were used to create them

  XML (machine and human readable)

- Often bundled with the content files in an information package

Domain

Born digital

Digitized

DP metadata supports preservation goals

Preservation Pyramid
(from Priscilla Caplan)

PREMIS PREservation Metadata Implementation Strategies

Background (brief history and rationale)

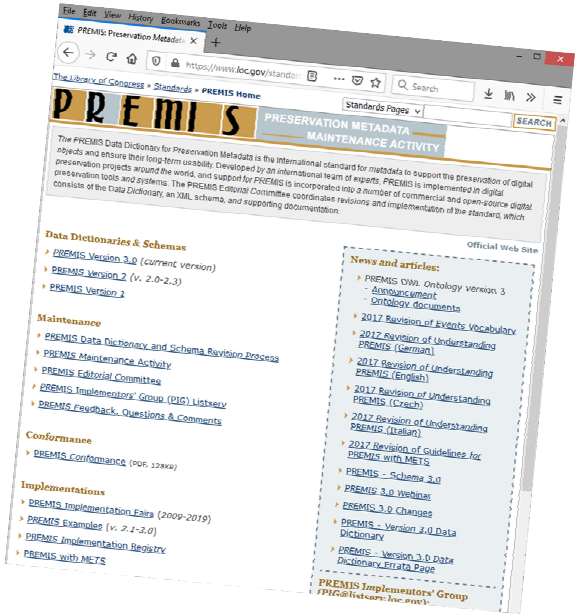
Karin Bredenberg
Kommunalförbundet Sydarkivera

WHAT IS PREMIS?

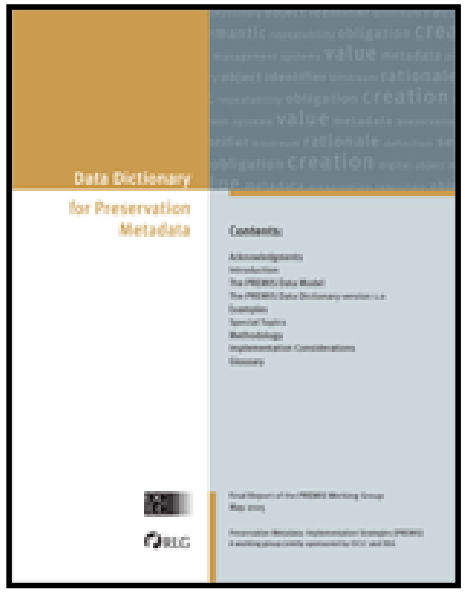
De-facto standard for preservation metadata

PREMIS = "things that most working preservation repositories are likely to need to know in order to support digital preservation" – *PREMIS Data Dictionary*

- valuable resource to know what you need to capture
- a method to model the information you need
- a data model with MANY implementations in different institutions
- a community of users with good resources



The PREMIS standard



- Data Dictionary (PREMIS 3.0)
  - http://www.loc.gov/standards/premis/v3/premis-3-0-final.pdf
  - Version 3 – major release
- XML schema v3.0
  - http://www.loc.gov/standards/premis/premis.xsd
- OWL ontology
- Supporting documentation

Scope

- What PREMIS DD is:
  - Common data model for organizing/thinking about preservation metadata
  - Implementable, Technically neutral and Core metadata

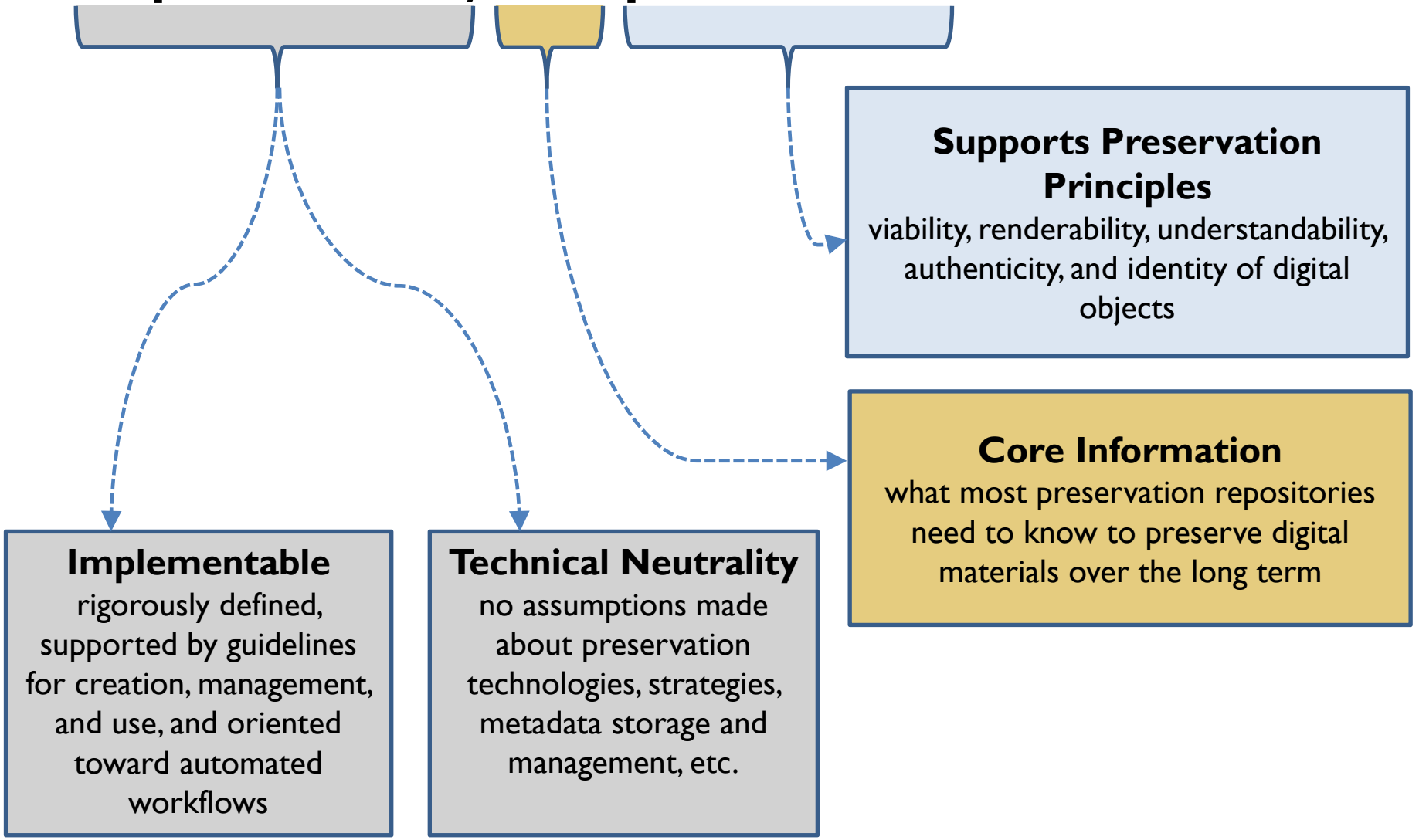
- What PREMIS DD is not:
  - Out-of-the-box solution
  - All needed metadata
  - Limited to just one use case

PREMIS PREservation Metadata Implementation Strategies

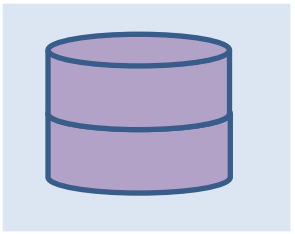
Karin Bredenberg
Kommunalförbundet Sydarkivera

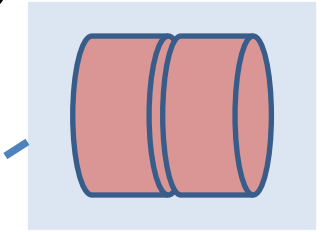
BENEFITS OF IMPLEMENTING PREMIS

Implementable, core preservation metadata



Benefits of implementing PREMIS (1)





IDobj="hdl.galapagos.6754.1"

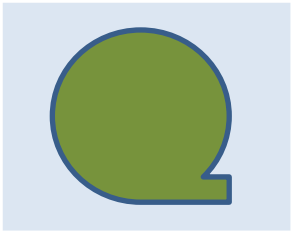
hdl="galapagos.6754.1"

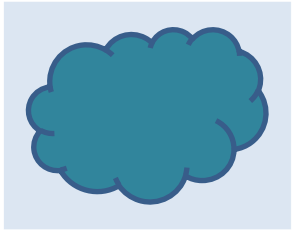
exchange consistency

?

ID="6754.1"
GID="galapagos"

ID="1"
TYPE="hdl"
GROUPID="galapagos.6754"

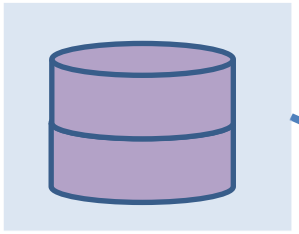


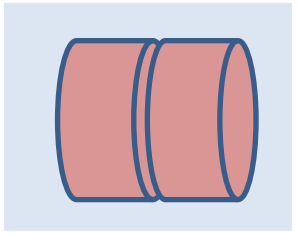


Interoperability through implementable preservation metadata

IDobj="hdl.galapagos.6754.1"

hdl="galapagos.6754.1"

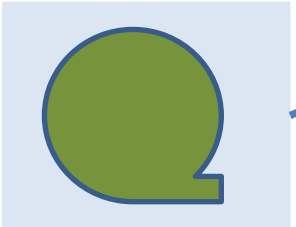




premis:objectIdentifier
premis:objectIdentifierType="hdl"
premis:objectIdentifierValue="galapagos.6754.1"

PREMIS

<galapagos.6754.1> a **premisOwl:IntellectualEntity** ;
premisOwl:identifier
<http://hdl.handle.net/galapagos.6754.1> .
<http://hdl.handle.net/galapagos.6754.1> a
http://id.loc.gov/vocabulary/identifiers/hdl .



ID="6754.1"
GID="galapagos"

ID="1"
TYPE="hdl"
GROUPID="galapagos.6754"

Benefits of implementing PREMIS (2)

- What do I need to know about a digital object I am preserving?
  - … today
  - … in the future
- How can I best capture that information?
- Is there a good practice?

PREMIS PREservation Metadata Implementation Strategies

Karin Bredenberg
Kommunalförbundet Sydarkivera

ON-LINE RESOURCES

The on-line resources

- Webpage
  - https://www.loc.gov/standards/premis/
- PREMIS Implementors' Group forum (pig@listserv.loc.gov)
  - Email message to LISTSERV@listserv.loc.gov :
    Subject:
    Text message: subscribe pig <your name>
- Vocabularies
  - https://id.loc.gov/vocabulary/preservation.html
- Ontology
  - https://id.loc.gov/ontologies/premis.html
- Tools at COPTR
  - https://coptr.digipres.org/index.php/PREMIS_(Preservation_Metadata_Implementation_Strategies)
- Zenodo
  - https://zenodo.org/communities/premis

PREMIS PREservation Metadata Implementation Strategies

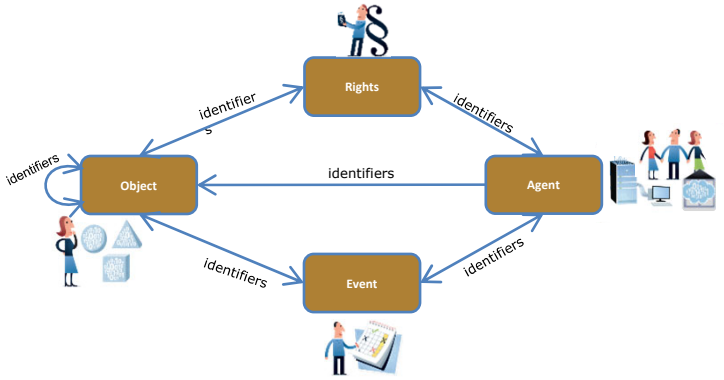
Eld Zierau
Royal Danish Library

OUTLINE OF MAIN ENTITIES

THE DATA MODEL & KEY CONCEPTS

DATA DICTIONARY

DESCRIPTION OF DATA MODEL



The PREMIS Data Model

- Data model includes:
  - Entities: "things" relevant to digital preservation
  - Semantic units: Properties of Entities
  - Relationships between Entities

- Why have a data model?
  - Organizational convenience (for development and use)
  - But: not a formal entity-relationship model; not sufficient to design databases

PREMIS 3 - Data model includes:

Entities
"things" relevant to digital preservation

Rights

Semantic Units
detailed information of entities

identifiers

identifiers

identifiers

Object

identifiers

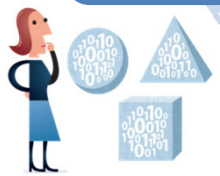
Agent

agentIdentifier

agentname

agentType

...



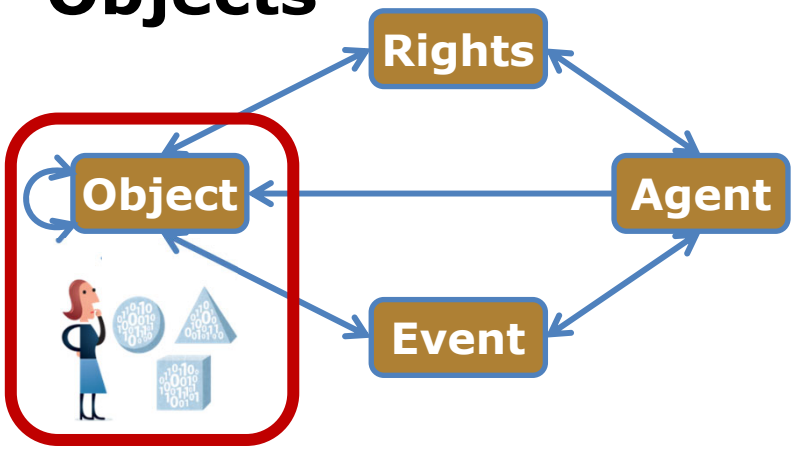
identifiers

identifiers

Event

Relationships ⟷
specifies relations between entities

Objects



- It is the objects that you preserve
- Objects can be intellectual entities, representations or bitstreams (more later)

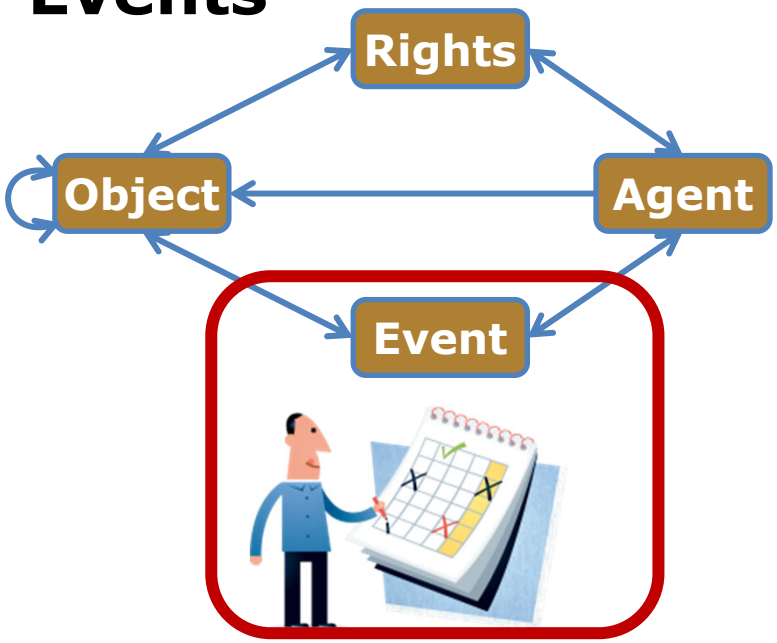
Examples:

- A PDF file
- An image within a file
- A book
- A book representation

Implementation choices :

- A repository does NOT have to manage all types of Objects

Events



- An action that involves or impacts at least one Object or Agent
- Can document digital provenance, needed to track history of Object

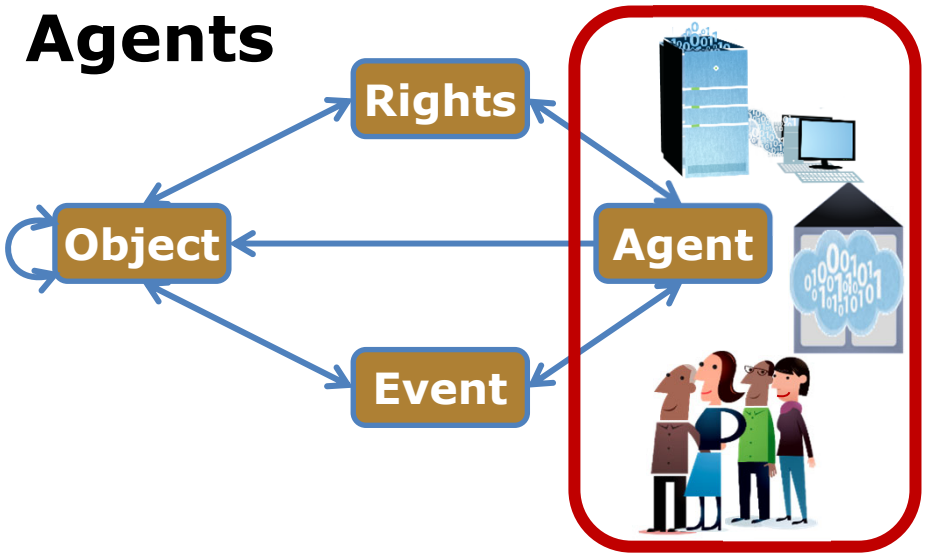
Examples:

- Validation Event e.g. using JHOVE
- Ingest Event

Implementation choices:

- Determining which Events are in scope
- Determining which Events should be recorded, as well as level of granularity

Agents



- Person, organization, or software program/system
- Intended only to identify the agent, and to allow linking from other entity types.

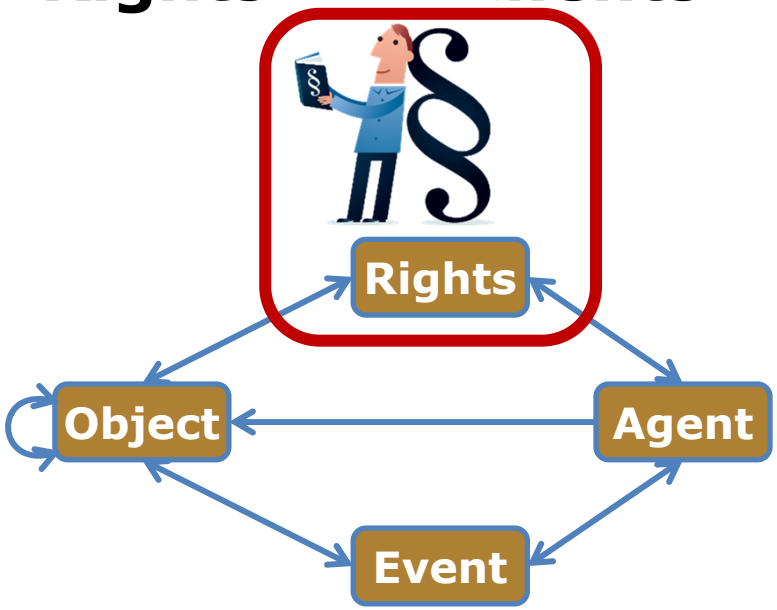
Examples:

- John Smith (a person)
- IIPC (an organization)
- JHOVE version 1.5 (a software program)

Implementation choices:

- use richer scheme that may be appropriate.

Rights Statements



- Rights to undertake an action(s) associated with an Object(s) in the repository.

Example:

- **Helen Smith** grants **FCLA digital repository permission** to the repository in regard to make three copies of **metadata_derived.pdf** for preservation purposes.

Implementation choices :

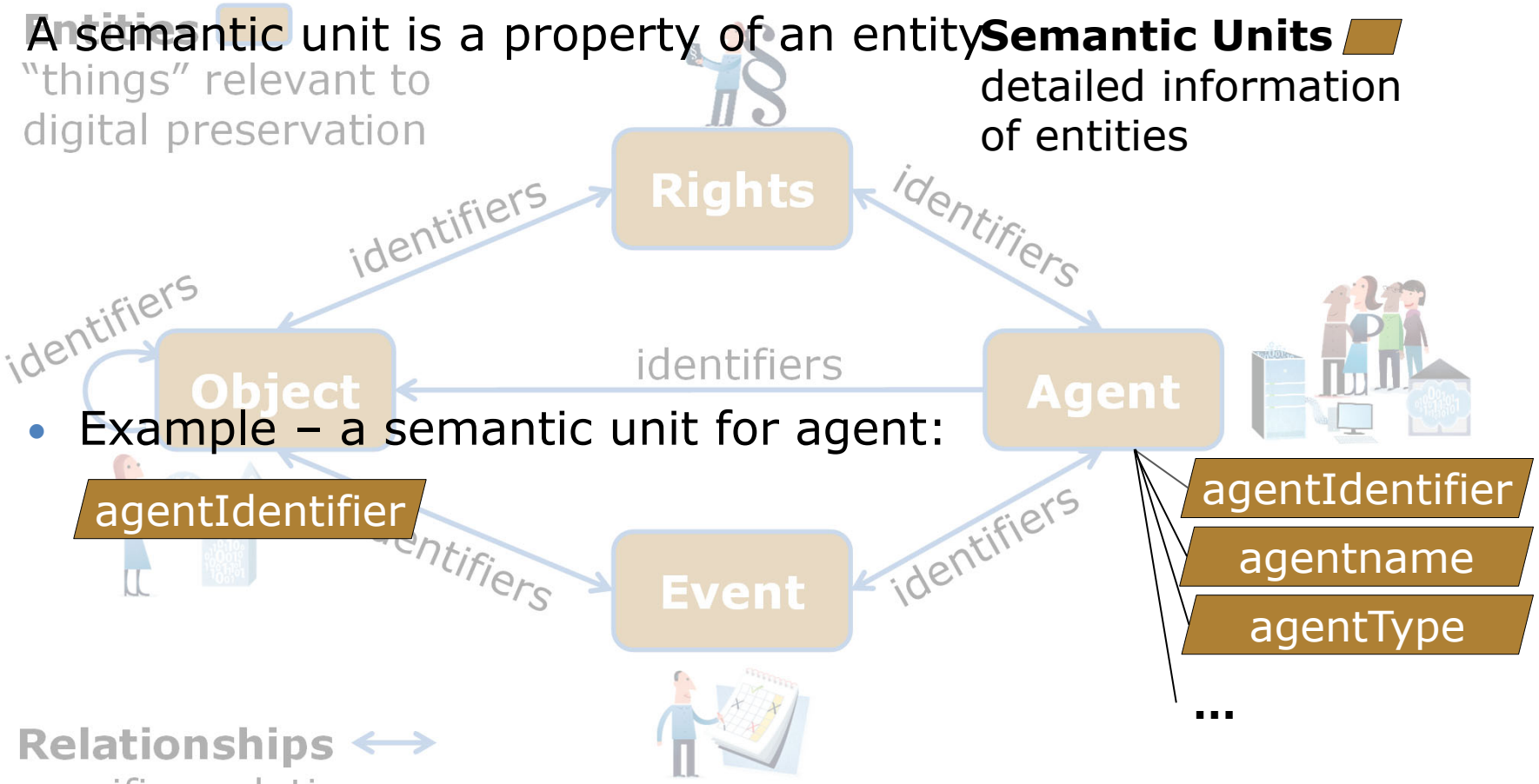
- Can use other schemes

  PREMIS form:
  Agent X
  grants **Permission Y**
  to the repository in regard to Object **Z**.

Semantic Units – Data model includes:

A semantic unit is a property of an entity

Semantic Units
detailed information of entities



- Example – a semantic unit for agent:

Semantic Units

A semantic unit is a property of an entity

A piece of information most repositories need to know in order to carry out their digital preservation functions

- Example – a semantic unit for agent:

agentIdentifier [container] groups together related semantic units

agentIdentifierType [semantic component]

agentIdentifierValue [semantic component]

Two kinds of semantic unit

Semantic units for entities

agentIdentifier

agentIdentifierType

agentIdentifierValue

Examplified by XML using XML schema v3.0:

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

```
<premis>
  <object … > … </object>
  <event> … </event>
  <agent> … </agent>
  <rights> … </rights>
</premis>
```

agentIdentifier

```
<agentIdentifier>
  <agentIdentifierType>
    UUID
  </agentIdentifierType>
  <agentIdentifierValue>
    41d10-099-1e2-9
  </agentIdentifierValue>
</agentIdentifier>
…
```

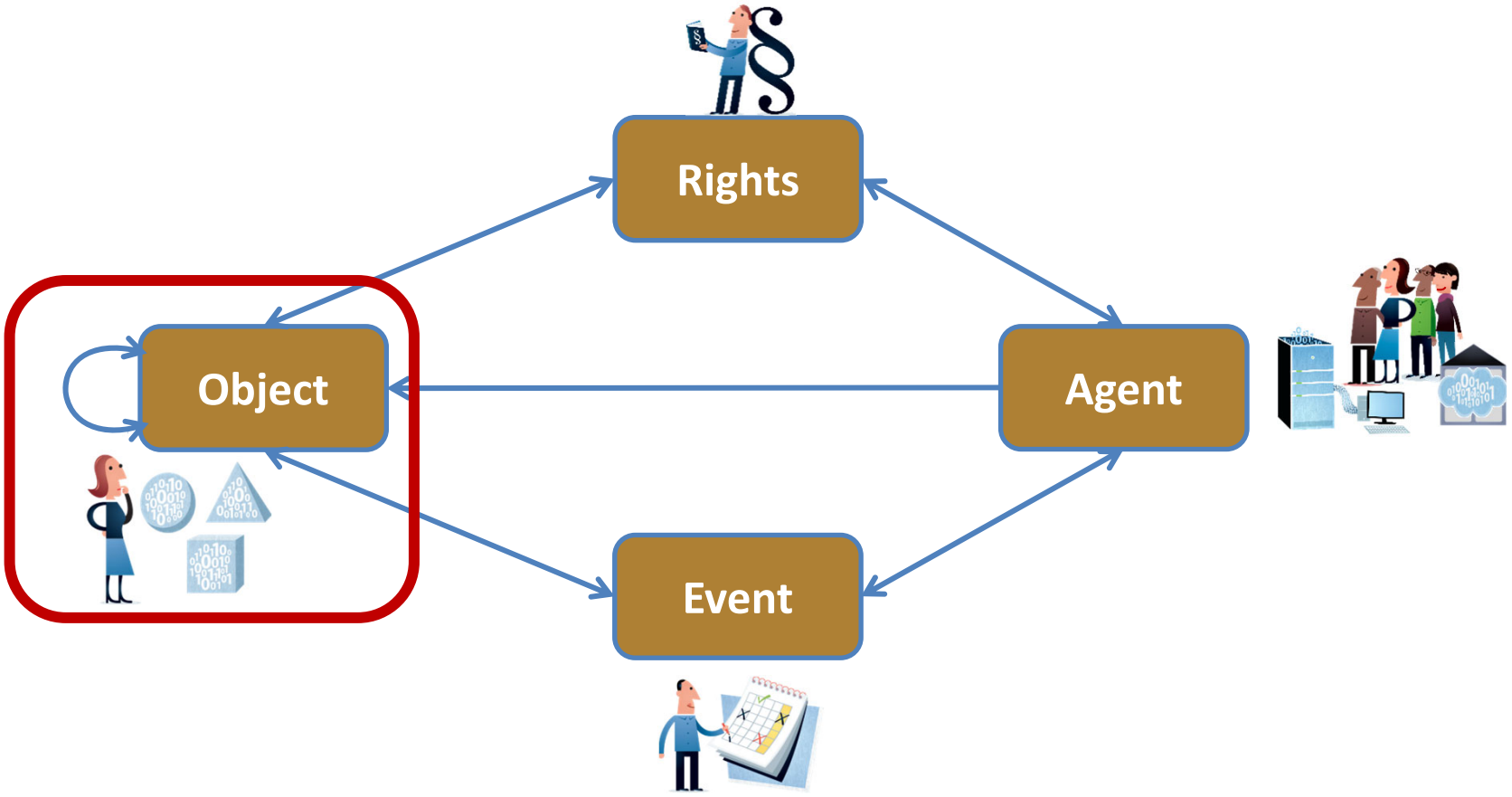
agentIdentifierType

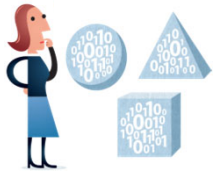
agentIdentifierValue

Can also use RDF

or just be inspired to use your own

Properties of Entities - Semantic units





High level semantic units for Objects

what technical information on it?

which object is it?

`ark:/12148/btp6k102002g/f1`

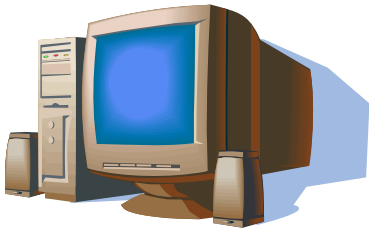
what is my preservation strategy for this object?

what kind of object?

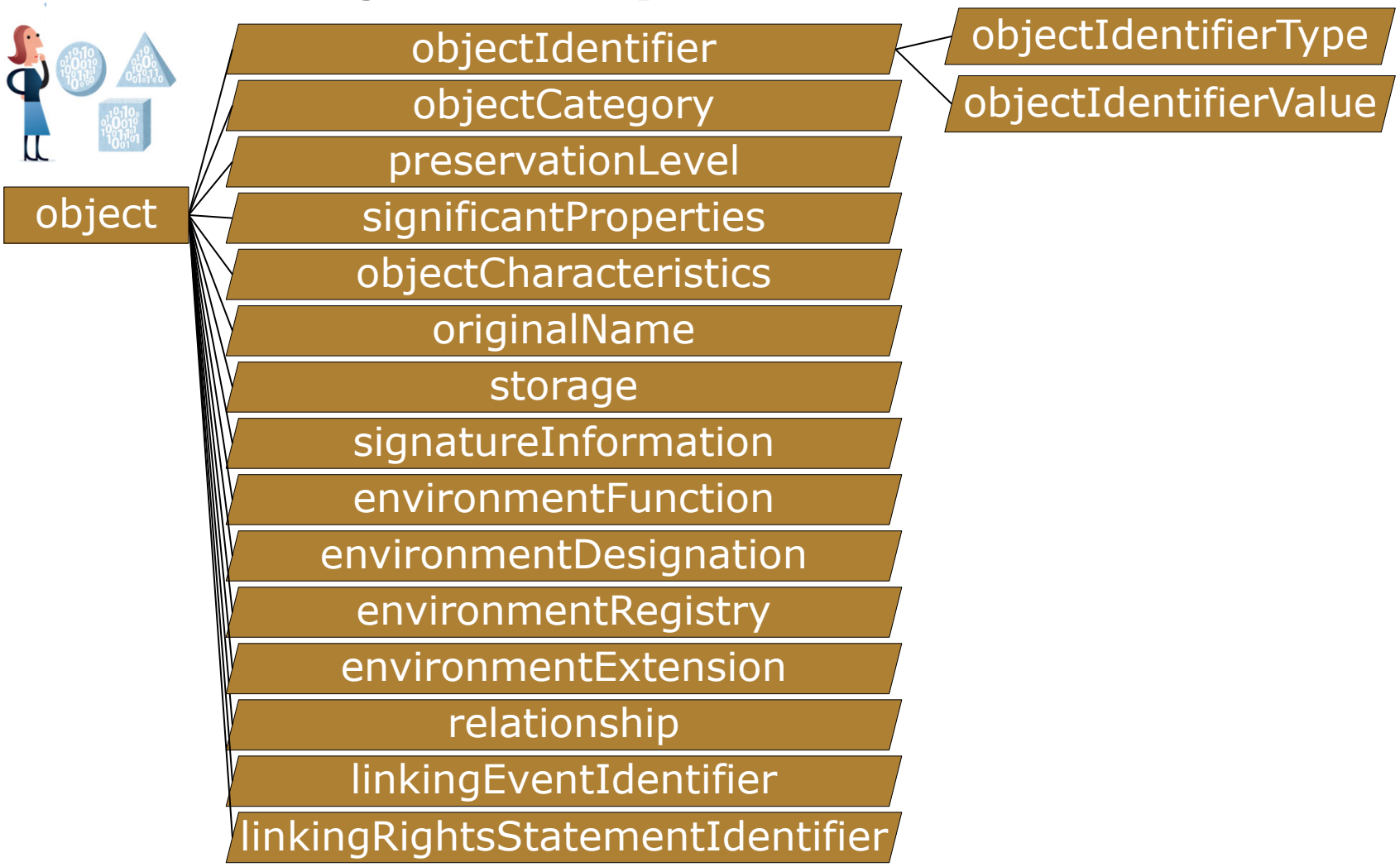
where is it stored?
on which media?

which of its characteristics do I want to preserve in it?

what software or hardware should be used to handle the object?

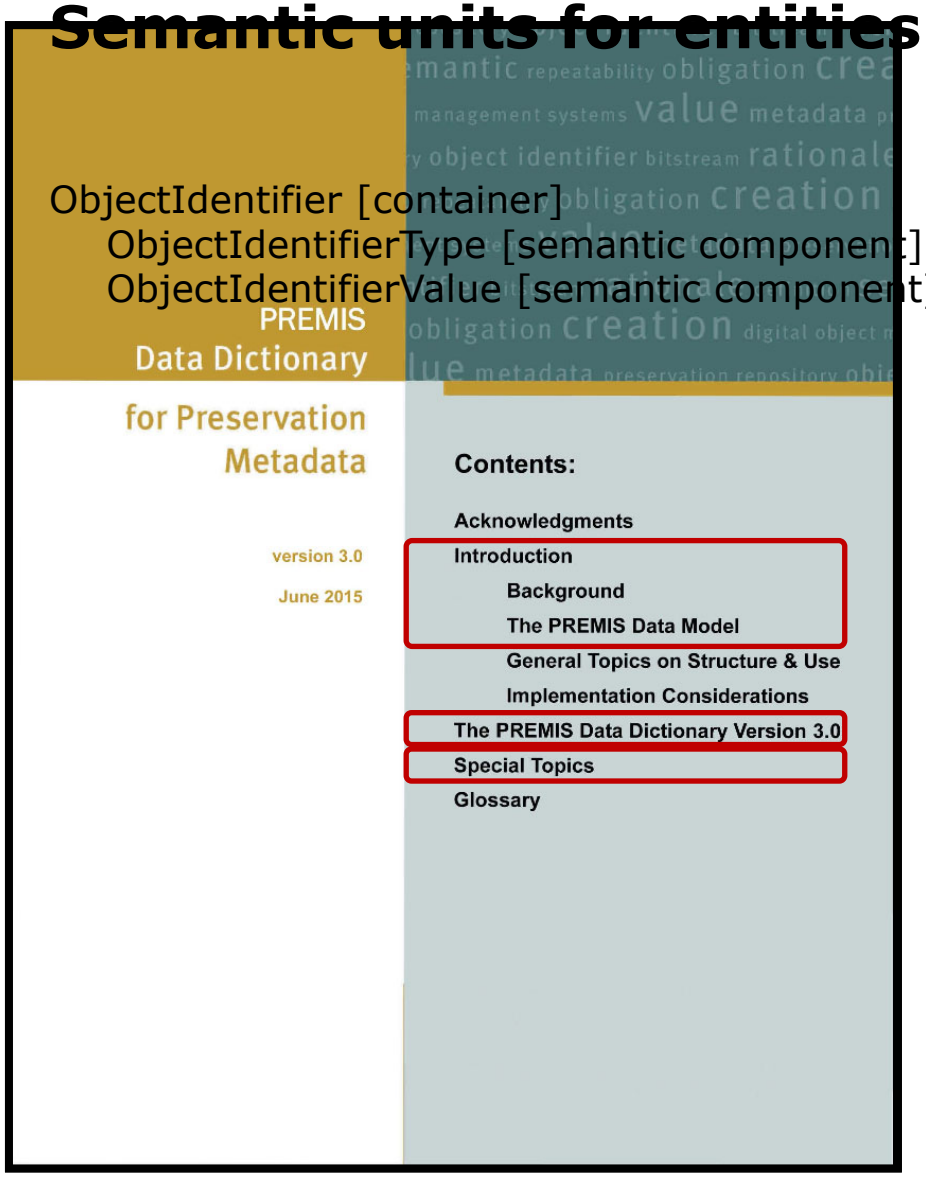


PREMIS Object Entity – Semantic Units



Semantic units for entities

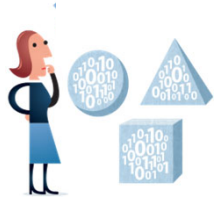
ObjectIdentifier [container]
ObjectIdentifierType [semantic component]
ObjectIdentifierValue [semantic component]



Data Dictionary (PREMIS 3.0)

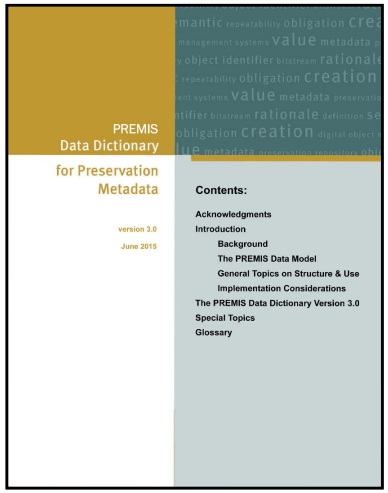
http://www.loc.gov/standards/premis/v3/premis-3-0-final.pdf

PREMIS Object Entity – Semantic Units



object

- 1.1 objectIdentifier
  - 1.1.1objectIdentifierType
  - 1.1.2 objectIdentifierValue
- 1.2 objectCategory
- 1.3 preservationLevel
- 1.4 significantProperties
- 1.5 objectCharacteristics
- 1.6 originalName
- 1.7 storage
- 1.8 signatureInformation
- 1.9 environmentFunction
- 1.10 environmentDesignation
- 1.11 environmentRegistry
- 1.12 environmentExtension
- 1.13 relationship
- 1.14 linkingEventIdentifier
- 1.15 linkingRightsStatementIdentifier



Sample Data Dictionary Entry

1.1 objectIdentifier

1.1.1objectIdentifierType

1.1.2 objectIdentifierValue

Object category (type)
Intellectual Entity
Representation
File
Bitstream

Repeatable (R)
Not Repeatable (NR)

Optional (O)
Mandatory (M)

Semantic unit	**1.1 objectIdentifier**		
Semantic components	1.1.1 objectIdentifierType 1.1.2 objectIdentifierValue		
Definition	A designation used to identify the Object uniquely within the preservation repository system in which it is stored.		
Rationale	Each Object held in the preservation repository must have a unique identifier to allow other entities to refer to it and to relate it to descriptive, technical, and other metadata unambiguously.		
Data constraint	Container		
Object category	**Intellectual Entity / Representation**	**File**	**Bitstream**
Applicability	Applicable	Applicable	Applicable
Repeatability	Repeatable	Repeatable	Repeatable
Obligation	Mandatory	Mandatory	Mandatory
Creation / Maintenance notes	An identifier may be created by the repository system at the time of ingest, or it may be created or assigned outside of the repository and submitted with an object as metadata. Similarly, identifiers can be generated automatically or manually.		
Usage notes	The *objectIdentifier* is mandatory for all Objects stored. The *objectIdentifier* is repeatable in order to allow both repository		

Sample Data Dictionary Entry

1.1 objectIdentifier

1.1.1objectIdentifierType

1.1.2 objectIdentifierValue

Object category (type)
Intellectual Entity
Representation
File
Bitstream

Repeatable (R)
Not Repeatable (NR)

Optional (O)
Mandatory (M)

Semantic unit	1.1 objectIdentifier
Semantic components	1.1.1 objectIdentifierType

Entity semantic units

NB: Semantic units are applicable for Intellectual Entities, Representations, Files and Bitstreams unless otherwise indicated.

1.1 objectIdentifier (M, R)
- 1.1.1 objectIdentifierType (M, NR)
- 1.1.2 objectIdentifierValue (M, NR)

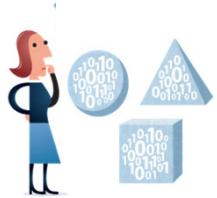
1.2 objectCategory (M, NR)

1.3 preservationLevel (O, R) [Intellectual Entity, Representation, File]
- 1.3.1 preservationLevelType (O, NR) [Intellectual Entity, Representation, File]
- 1.3.2 preservationLevelValue (M, NR) [Intellectual Entity, Representation, File]
- 1.3.3 preservationLevelRole (O, NR) [Intellectual Entity, Representation, File]
- 1.3.4 preservationLevelRationale (O, R) [Intellectual Entity, Representation, File]
- 1.3.5 preservationLevelDateAssigned (O, NR) [Intellectual Entity, Representation, File]

1.4 significantProperties (O, R)
- 1.4.1 significantPropertiesType (O, NR)
- 1.4.2 significantPropertiesValue (O, NR)
- 1.4.3 significantPropertiesExtension (O, R)

1.5 objectCharacteristics (M, R) [File, Bitstream]
- 1.5.1 compositionLevel (O, NR) [File, Bitstream]
- 1.5.2 fixity (O, R) [File, Bitstream]
  - 1.5.2.1 messageDigestAlgorithm (M, NR) [File, Bitstream]
  - 1.5.2.2 messageDigest (M, NR) [File, Bitstream]
  - 1.5.2.3 messageDigestOriginator (O, NR) [File, Bitstream]
- 1.5.3 size (O, NR) [File, Bitstream]

PREMIS Object Entity – Semantic Units



object

- objectIdentifier
- objectCategory

always mandatory

- objectCharacteristics
- significantProperties
- preservationLevel
- originalName
- storage
- signatureInformation
- environmentFunction
- environmentDesignation
- environmentRegistry
- environmentExtension
- relationship
- linkingEventIdentifier
- linkingRightsStatementIdentifier

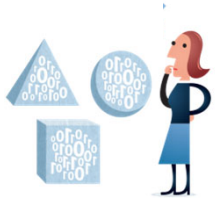
objectCategory

- Values:
  - intellectual entity
  - representation
  - file
  - bitstream

- Implemented as an xsi:type in PREMIS XML-schema so not explicitly recorded

```
<premis>
  <object xsi:type="file">
      ...
  </object>
  ...
</premis>
```

objectCategory (types of objects)



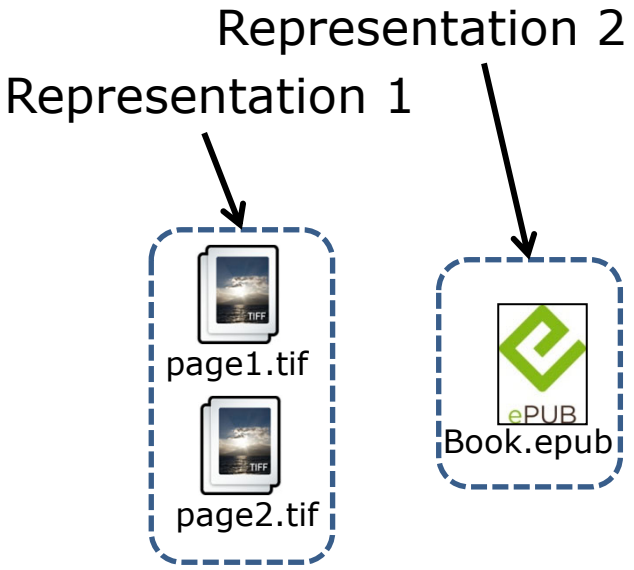
INTELLECTUAL ENTITY: a distinct intellectual or artistic creation that is considered relevant to a designated community in the context of digital preservation.

A literary "work"

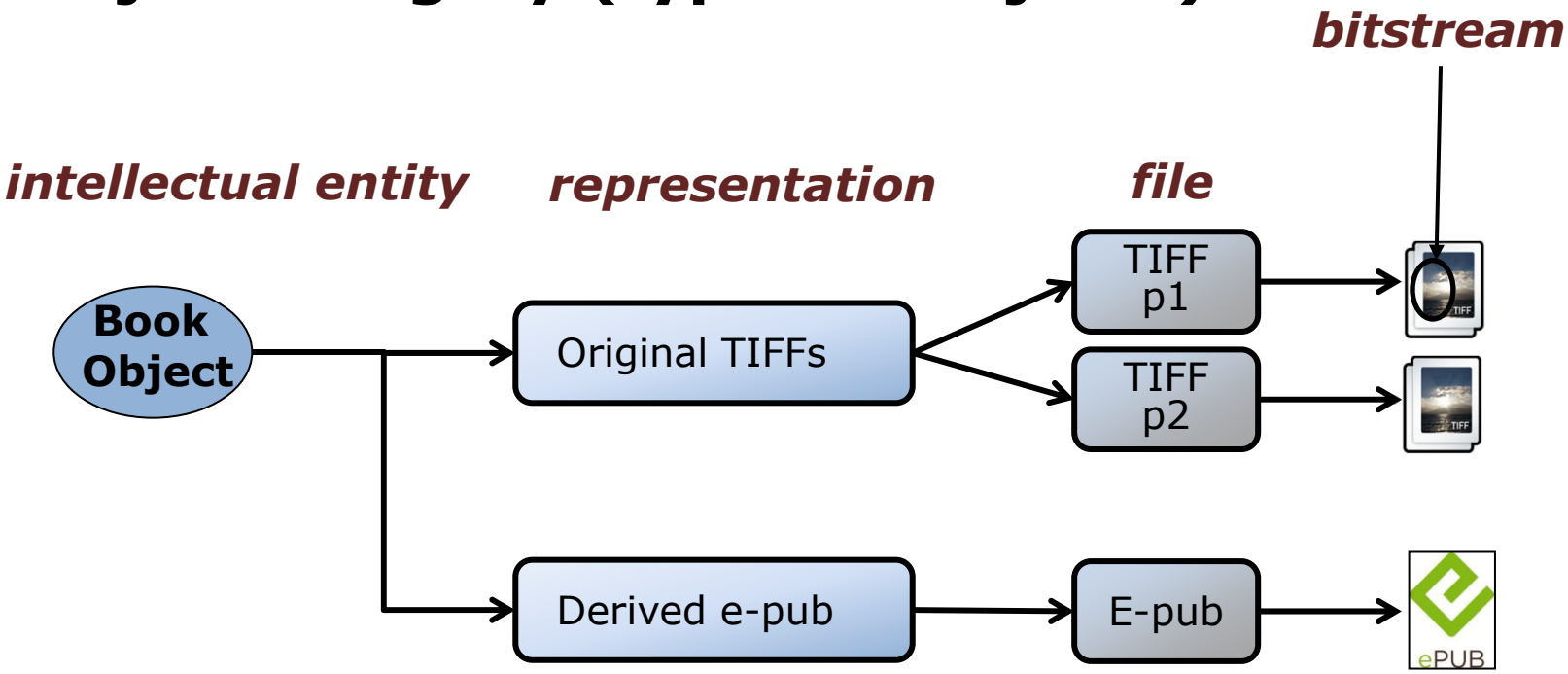
REPRESENTATION: set of files, including structural metadata, that, taken together, constitute a complete rendering of an Intellectual Entity.

FILE: named and ordered sequence of bytes that is known by an operating system (file-streams (files within files) are considered files since they can be rendered alone)

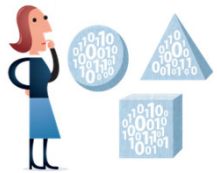
BITSTREAM: data within a file with properties relevant for preservation purposes (but needs additional structure or reformatting to be stand-alone file)



objectCategory (types of objects)



PREMIS Object Entity



object

- objectIdentifier
- objectCategory
- objectCharacteristics
- significantProperties
- preservationLevel
- originalName
- storage
- signatureInformation
- environmentFunction
- environmentDesignation
- environmentRegistry
- environmentExtension
- relationship
- linkingEventIdentifier
- linkingRightsStatementIdentifier

always mandatory

specific characteristic

objectCharacteristics

[mandatory for file or bitstream]

- Technical properties common to all/most file formats, not format specific

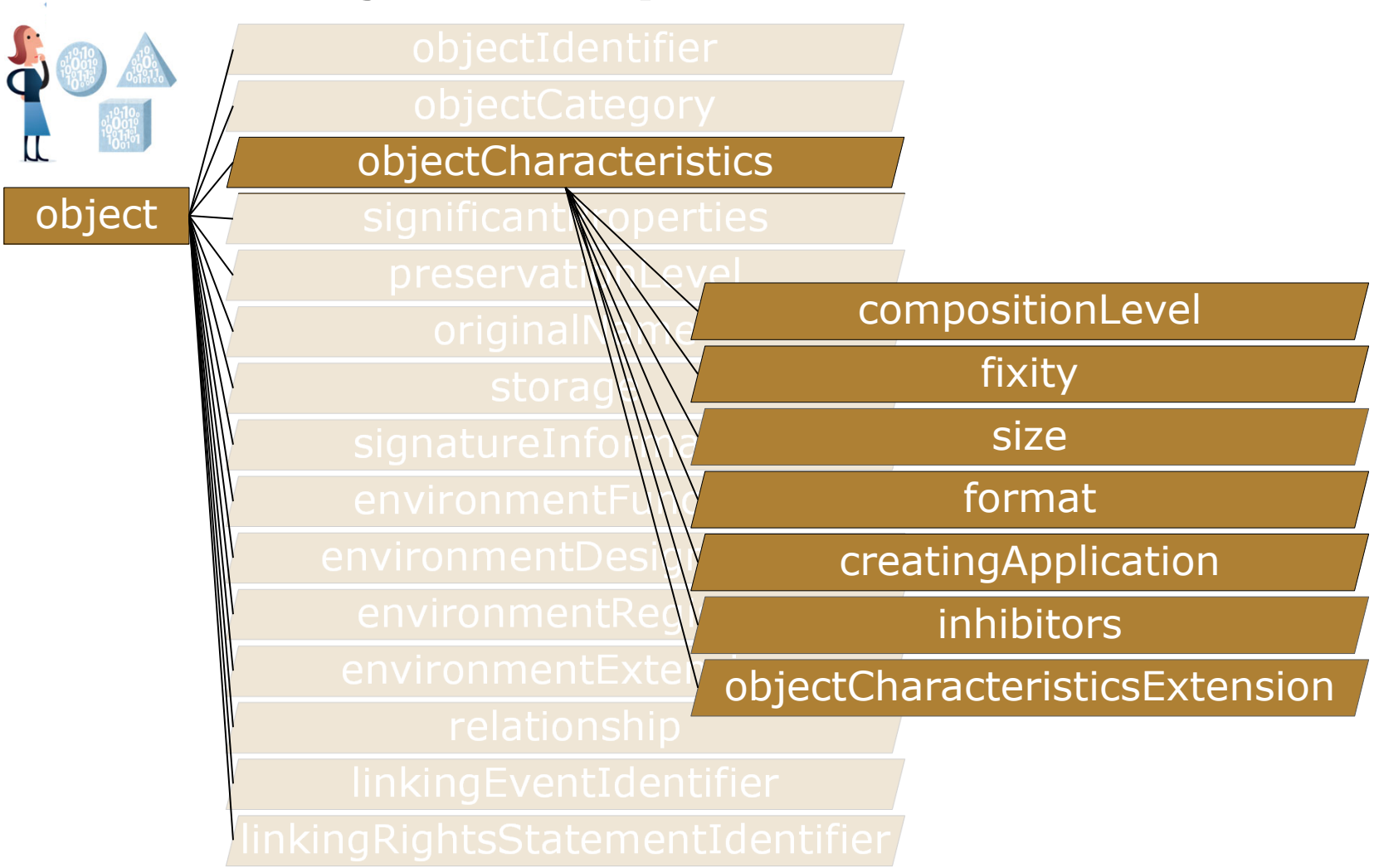
Looking at files

FILE = a named sequence of bytes

- chapter1.pdf
- photo.tiff
- mapofBerlin.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 Object Entity



Composition Level

sometimes there is more than one layer of characteristics

chapter1.pdf

Composition Level 0

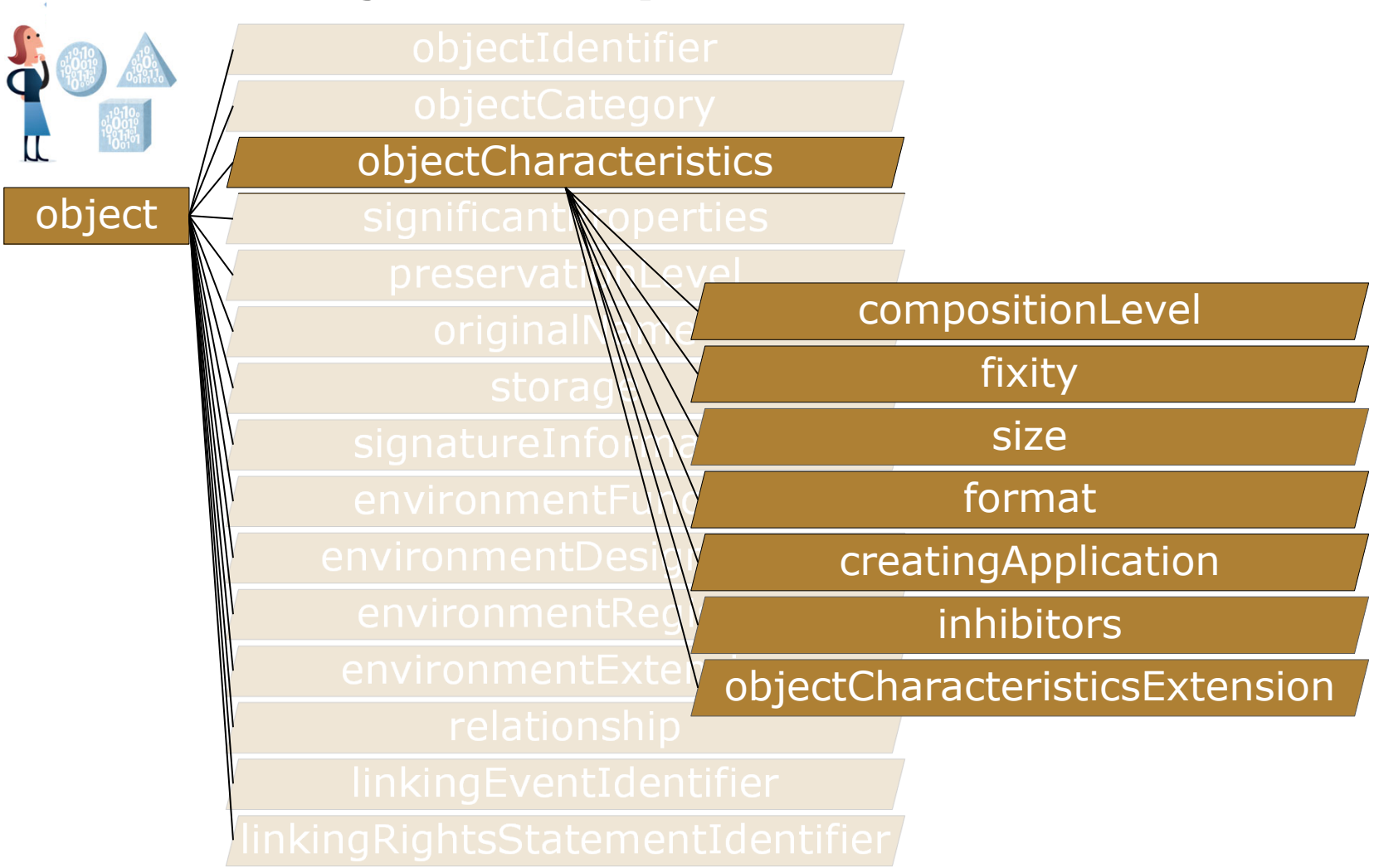
chapter1.pdf.gz

Composition Level 1

Or they be part of other files e.g.

- Mail attachments
- Images in PDF's etc

PREMIS Object Entity



objectCharacteristicsExtension

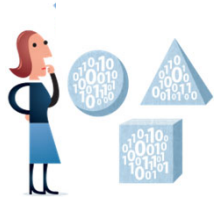
Container to include externally defined semantic units – e.g. for more granularity

Might contain format specific metadata for a file – e.g. technical metadata for still images (MIX)

objectCharacteristicsExtension - example

```
<premis> ...
 <object xsi:type="file">
  <objectCharacteristics>  ...
   <objectCharacteristicsExtension>
    <mix:mix xsi:schemaLocation=
        "... http://www.loc.gov/standards/mix/mix20/mix20.xsd">
      ...
      <mix:BasicImageInformation>
       <mix:BasicImageCharacteristics>
        <mix:imageWidth>5894</mix:imageWidth>
        <mix:imageHeight>7768</mix:imageHeight>
        ...
       </mix:BasicImageCharacteristics>
      </mix:BasicImageInformation>
      ...
     <mix:mix>
   </objectCharacteristicsExtension>   ...
  <objectCharacteristics>   ...
 </object>  ...
</premis>
```

PREMIS Object Entity – Semantic Units



object

- objectIdentifier
- objectCategory
- objectCharacteristics
- significantProperties
- preservationLevel
- originalName
- storage
- signatureInformation
- environmentFunction
- environmentDesignation
- environmentRegistry
- environmentExtension
- relationship
- linkingEventIdentifier
- linkingRightsStatementIdentifier

always mandatory

specific characteristic

more general info

preservationLevel

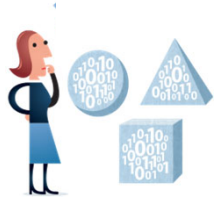
What preservation treatment/strategy the repository plans for this object

- Varying preservation options dependent on factors such as value, uniqueness, preservability of format
- A business rule only relevant in a given repository

Contains

- **preservationLevelType, e.g.** logicalStrategy or BitSafety
- **preservationLevelValue, e.g.** migration or High
- **preservationLevelRole** (context), e.g. intention or requirement
- **preservationLevelRationale**, when differs from policy
- **preservationLevelDateAssigned** when Level was assigned

PREMIS Object Entity – Semantic Units



object

- objectIdentifier
- objectCategory
- objectCharacteristics
- significantProperties
- preservationLevel
- originalName
- storage
- signatureInformation
- environmentFunction
- environmentDesignation
- environmentRegistry
- environmentExtension
- relationship
- linkingEventIdentifier
- linkingRightsStatementIdentifier

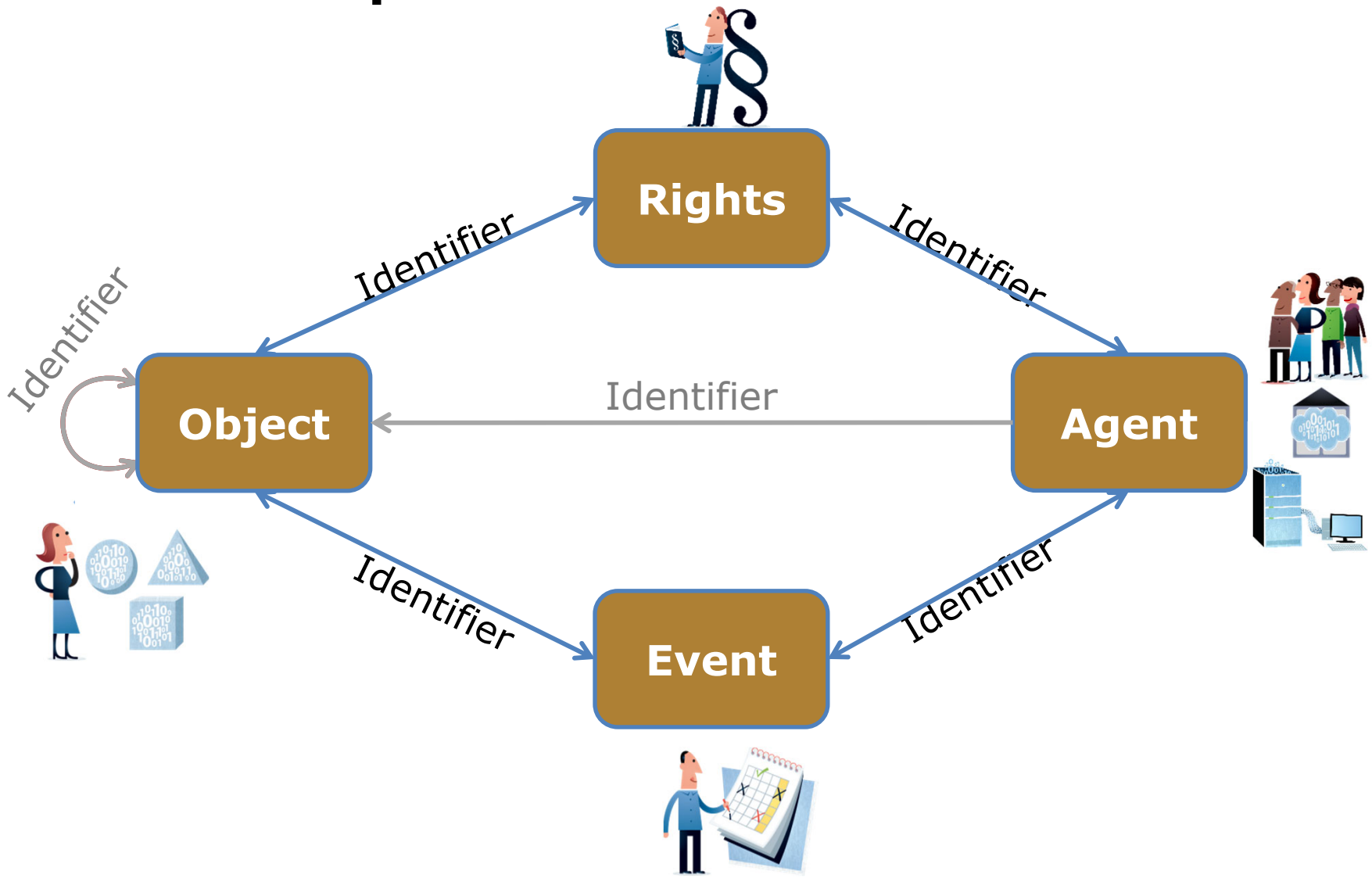
always mandatory

specific characteristic

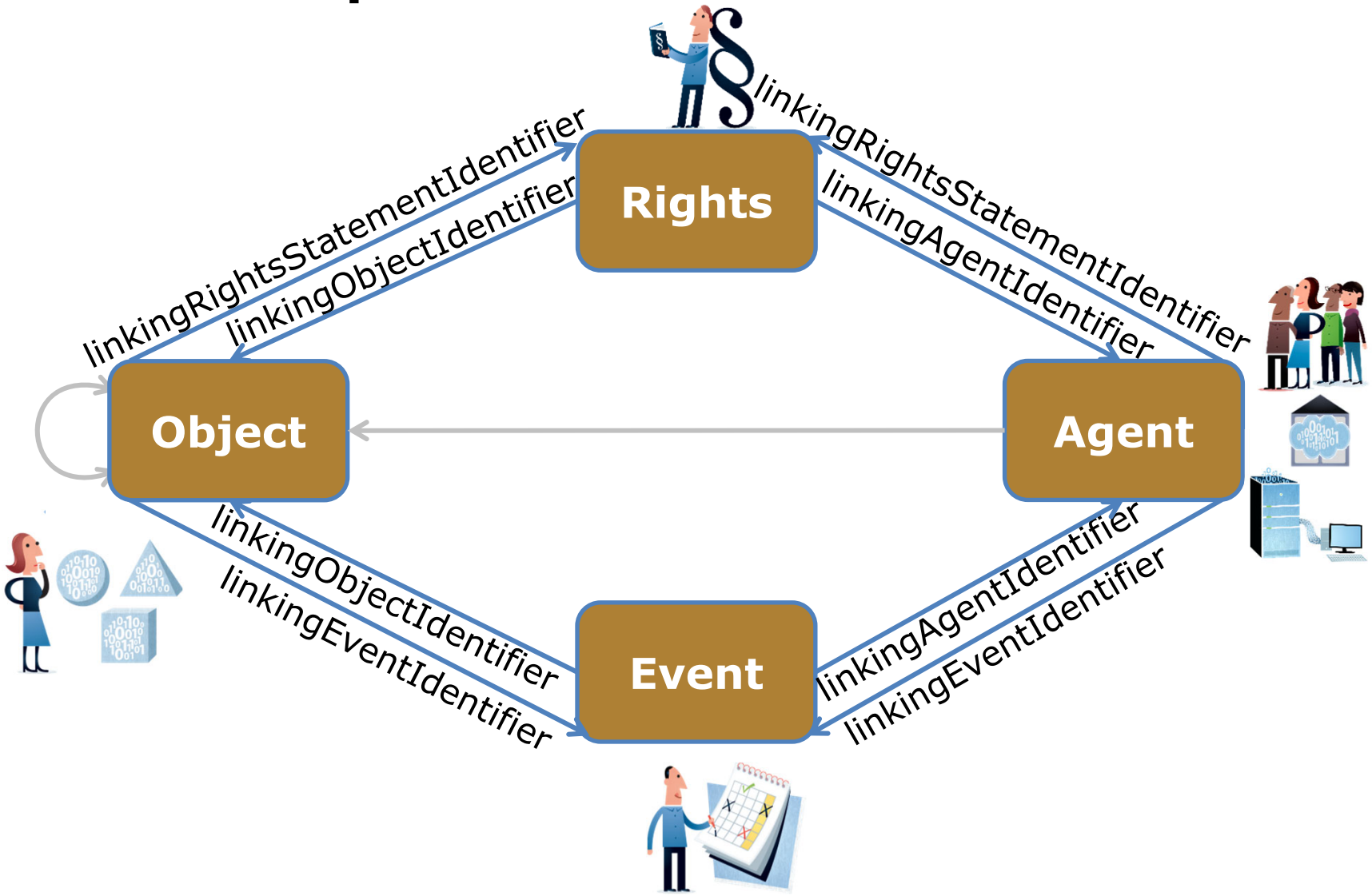
more general info

entity linking

Relationships: Semantic Unit Identifiers



Relationships: Semantic Unit Identifiers



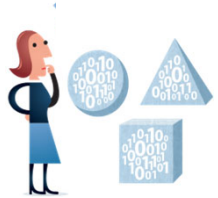
Linking Objects with Agents and Events

- **linkingRightsStatementIdentifier**
  - **linkingRightsStatementIdentifierType**
  - **linkingRightsStatementIdentifierValue**
- **linkingEventIdentifier**
  - **linkingEventIdentifierType**
  - **linkingEventIdentifierValue**

Likewise for the other entities

except there may be possibility of specifying roles, e.g. for Event because the same Agent may have a different Role in the digital Archive system

PREMIS Object Entity – Semantic Units



object

- objectIdentifier
- objectCategory
- objectCharacteristics
- significantProperties
- preservationLevel
- originalName
- storage
- signatureInformation
- environmentFunction
- environmentDesignation
- environmentRegistry
- environmentExtension
- relationship
- linkingEventIdentifier
- linkingRightsStatementIdentifier

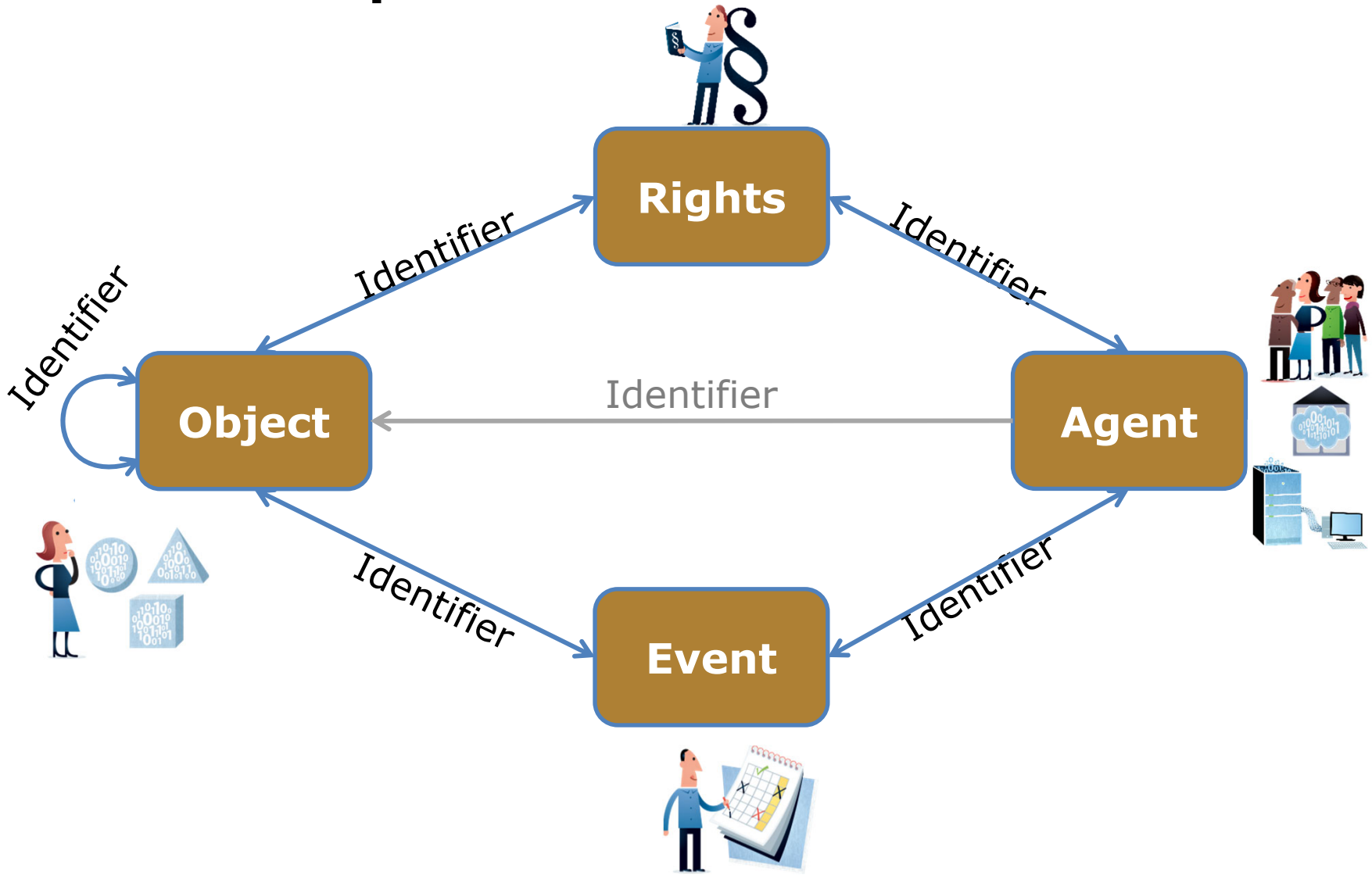
always mandatory

specific characteristic

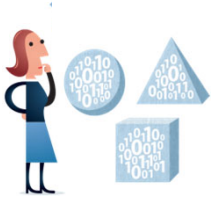
more general info

entity linking

Relationships: Semantic Unit Identifiers



Objects and their interrelations



Object

is part of **Intellectual Entity**

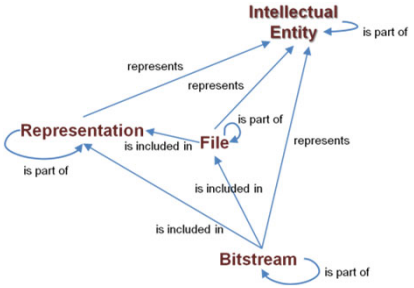
is part of **Representation**

represents

is included in

is part of **File**

is part of **Bitstream**



PREMIS Object Entity – Semantic Units

relationship

- relationshipType
- relationshipSubType
- relatedObjectIdentifier
  (type, value, sequence)
- relatedEventIdentifier
  (type, value, sequence)
- relatedEnvironmentPurpose
- relatedEnvironmentCharacteristic

environments - later

Example:
Representation pointing to an Intellectual Entity

structural

represents

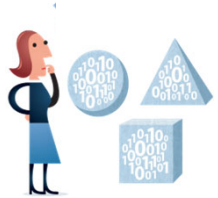
Identifier of Intellectual Entity

relation through (because of) an Event – e.g. for migration

Objects can be associated with Events in two ways:

- If the Object has an associated Event with ***relationship***
- If the Object has an associated Event with ***no relationship*** to a second Object, e.g. ingest: use **linkingEventIdentifier**

PREMIS Object Entity – Semantic Units



object

- objectIdentifier
- objectCategory
- objectCharacteristics
- significantProperties
- preservationLevel
- originalName
- storage
- signatureInformation
- environmentFunction
- environmentDesignation
- environmentRegistry
- environmentExtension
- relationship
- linkingEventIdentifier
- linkingRightsStatementIdentifier

always mandatory

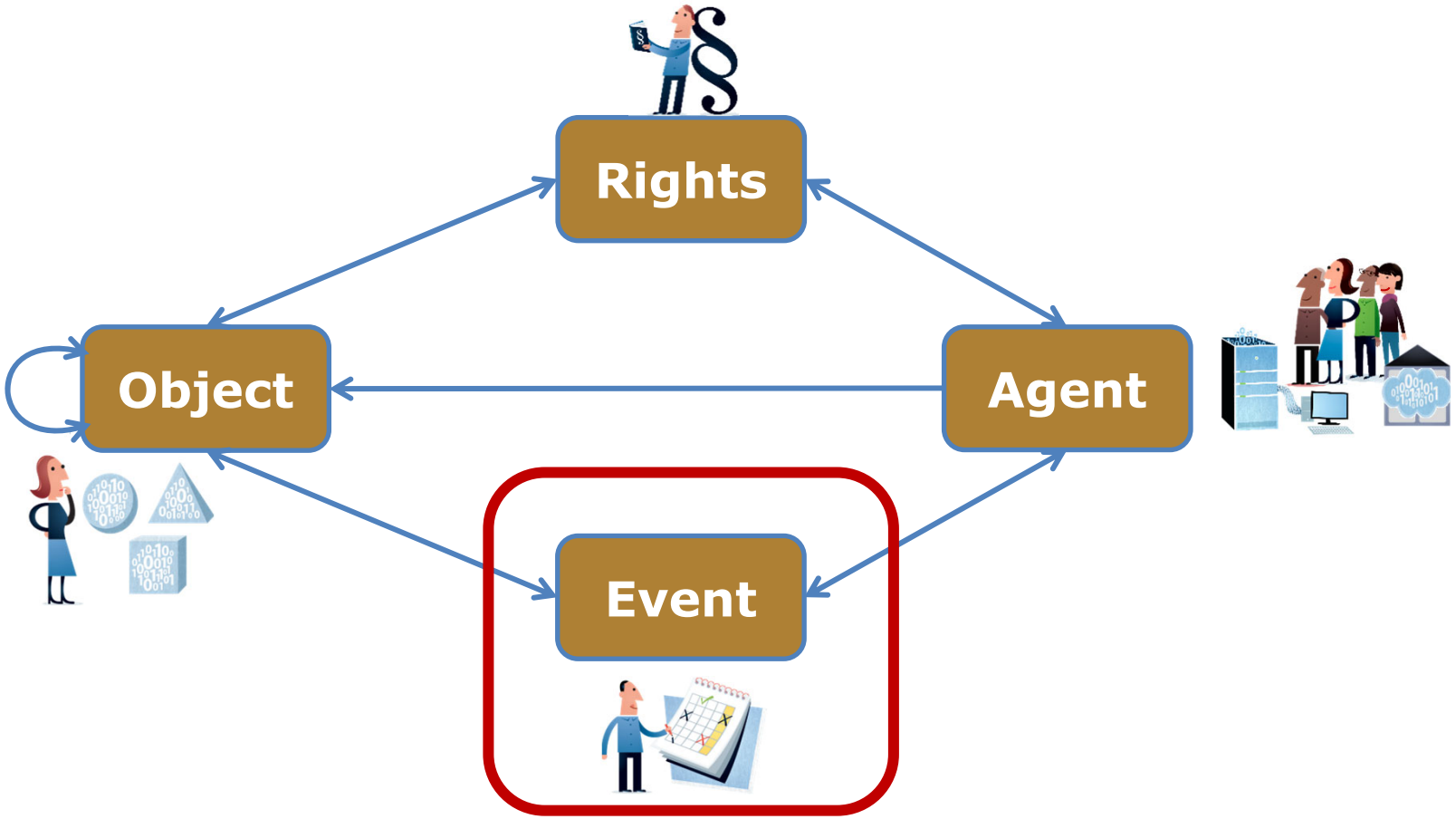
specific characteristic

more general info

environments - later

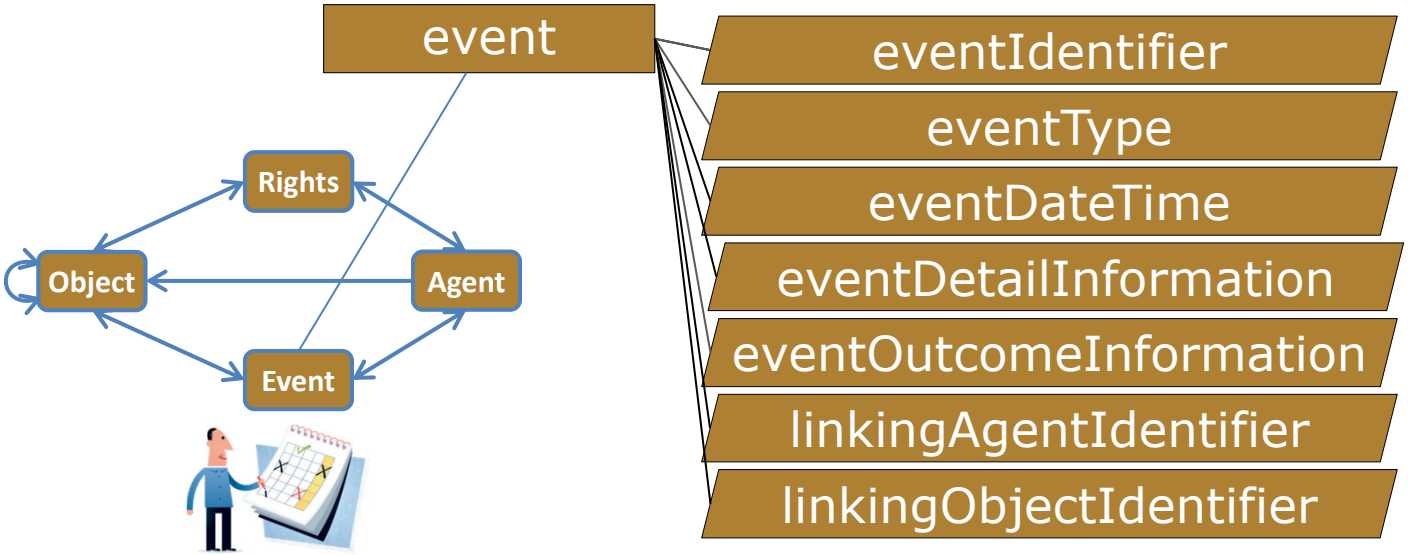
entity linking

Properties of Entities - Semantic units



PREMIS Event Entity

- Mandatory semantic units are: *eventIdentifier, eventType, and eventDateTime.*
- Must be related to one or more Objects.
- Can be related to one or more Agents.



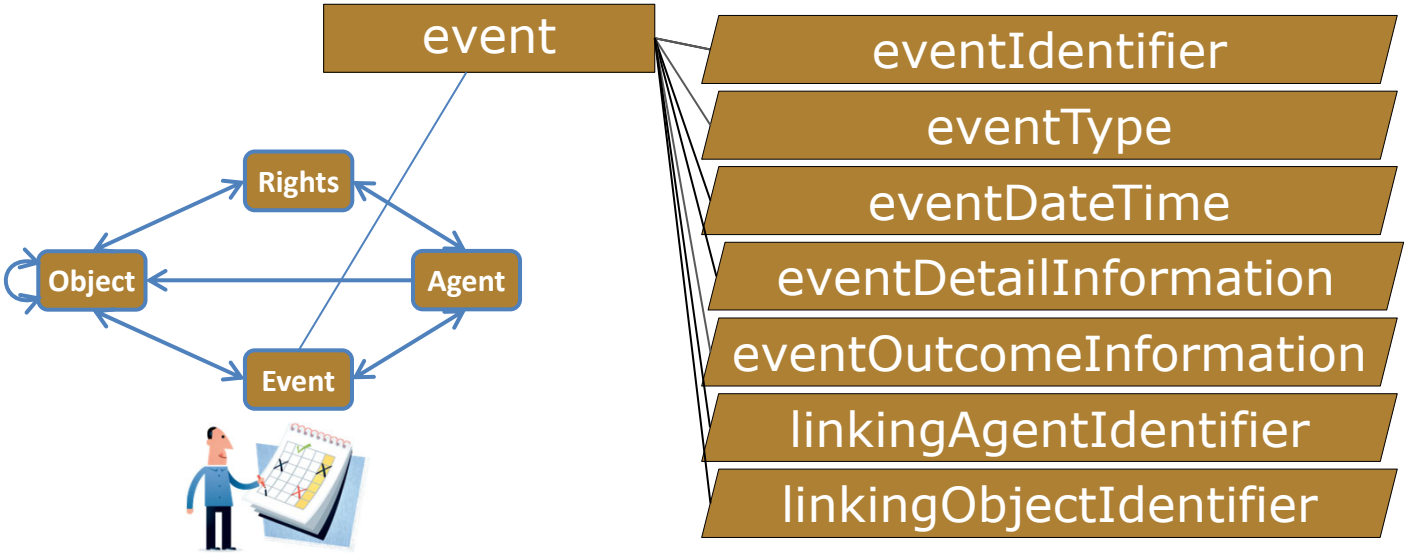
eventType

- Names the event

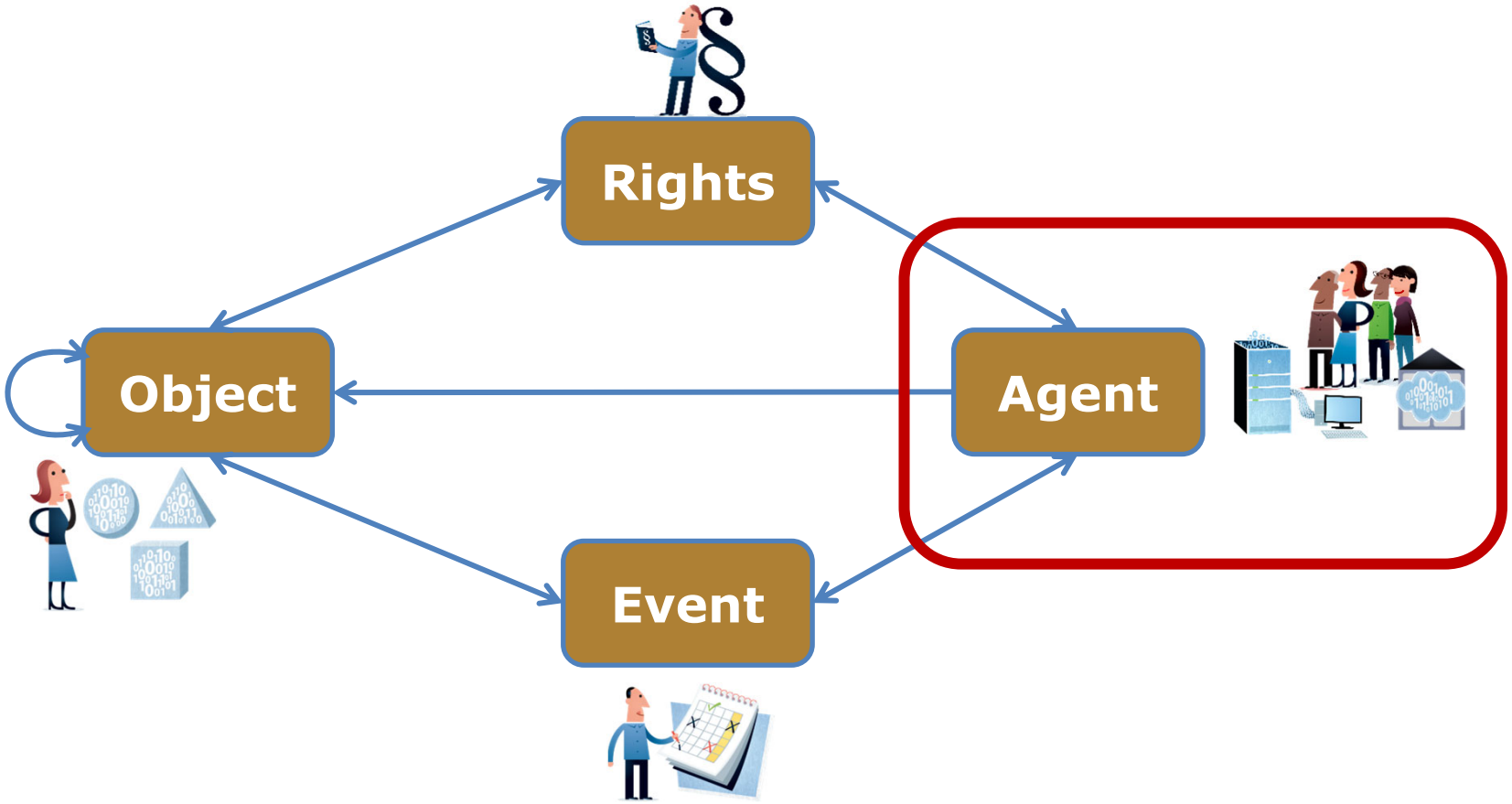
Ingestion **Virus check** **Validation** **Compression** **Message digest calculation** **Decompression** … **migration** **Fixity check**

- Recommended to use a controlled vocabulary, e.g. http://id.loc.gov/vocabulary/preservation/eventType.html
- Could use coded values
- Granularity is implementation-specific

PREMIS Event Entity



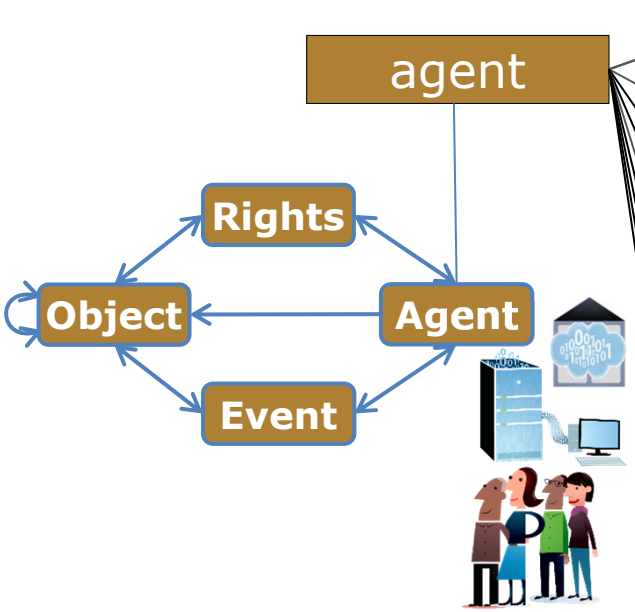
Properties of Entities - Semantic units



PREMIS Agent Entity

The only mandatory semantic unit is *agentIdentifier*

- May hold or grant one or more Rights.
- May carry out, authorize, or compel one or more Events.
- May create or act upon one or more Objects through an Event or with respect to a Rights statement.

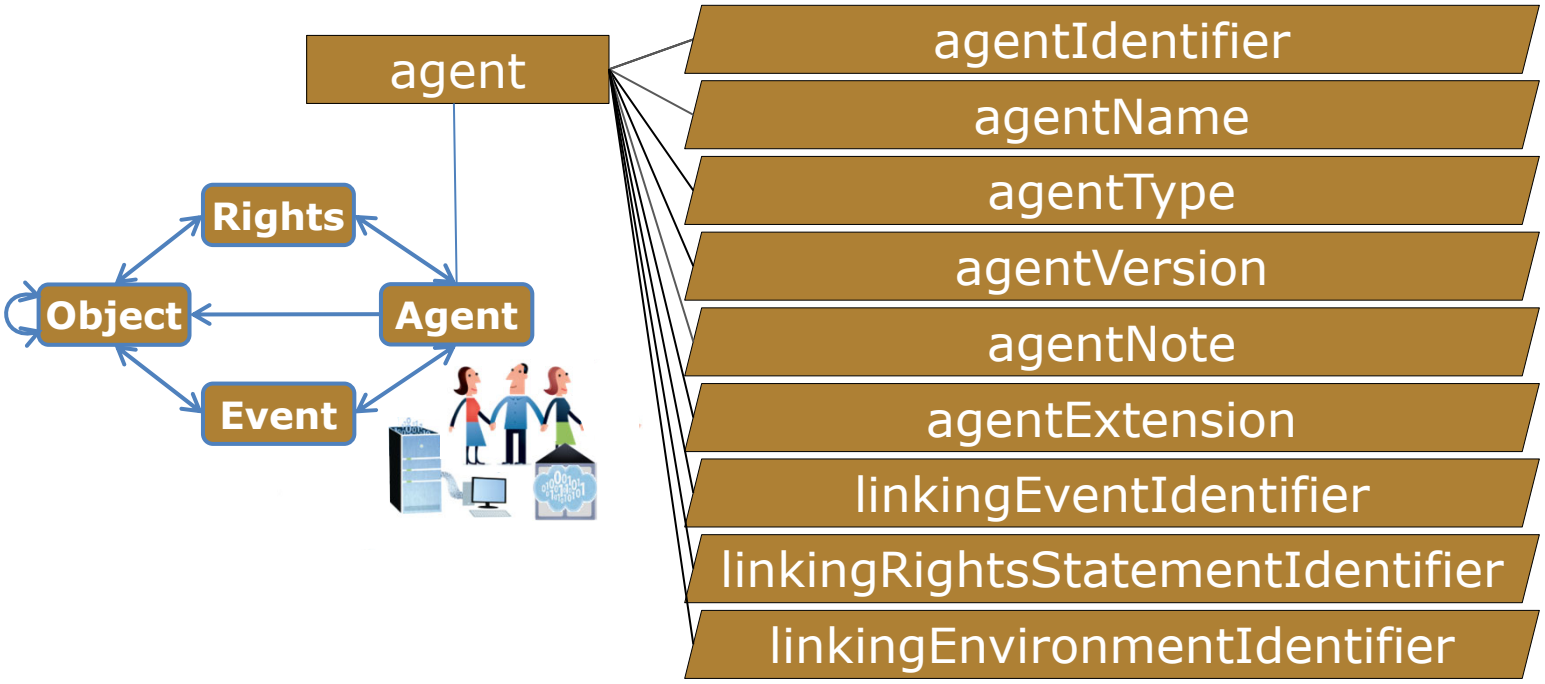


- agent
  - agentIdentifier
  - agentName
  - agentType
  - agentVersion
  - agentNote
  - agentExtension
  - linkingEventIdentifier
  - linkingRightsStatementIdentifier
  - linkingEnvironmentIdentifier

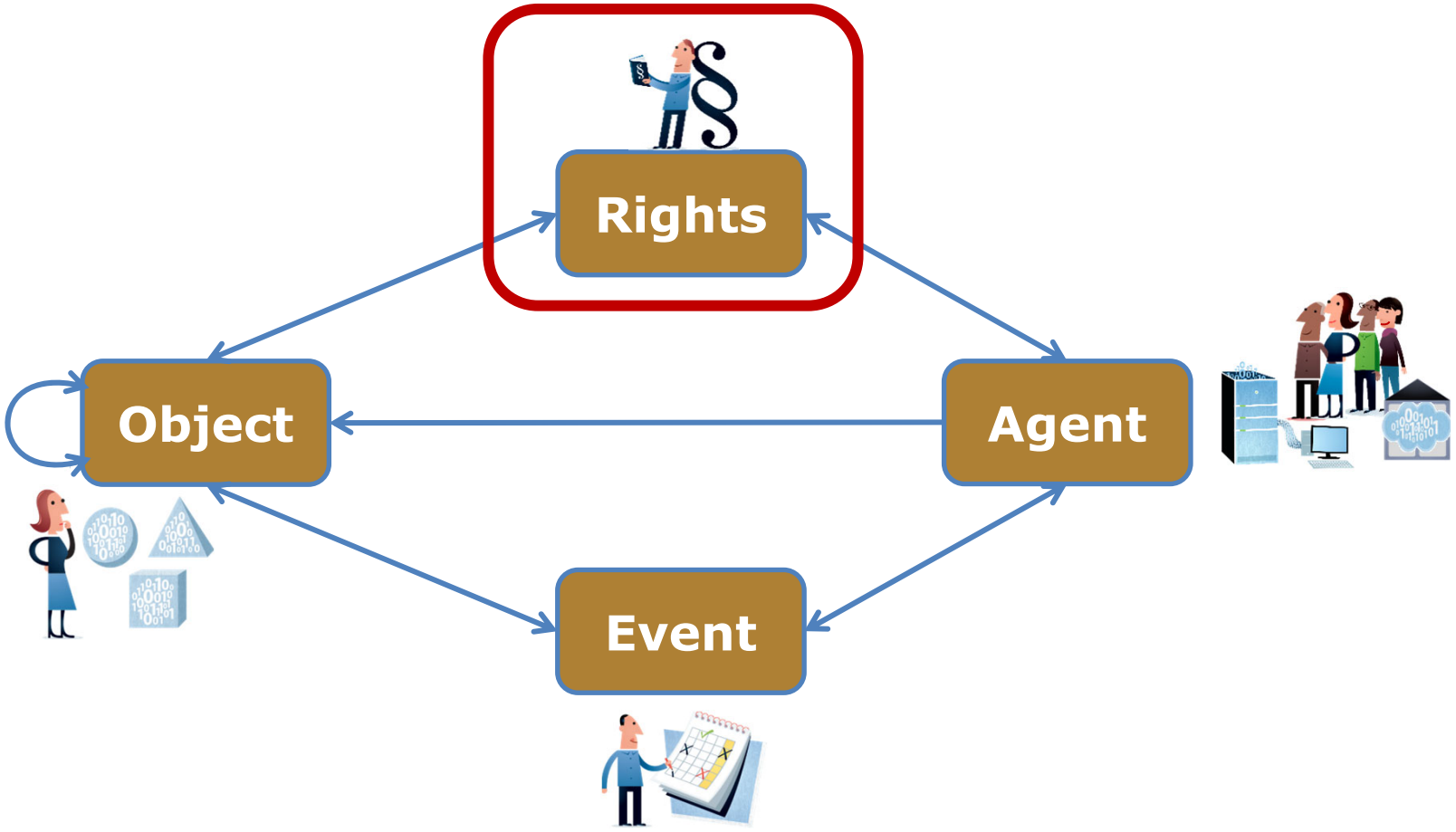
agentType

- Can use controlled vocabulary, e.g. http://id.loc.gov/vocabulary/preservation/agentType.html
  - hardware
  - organization
  - person
  - software

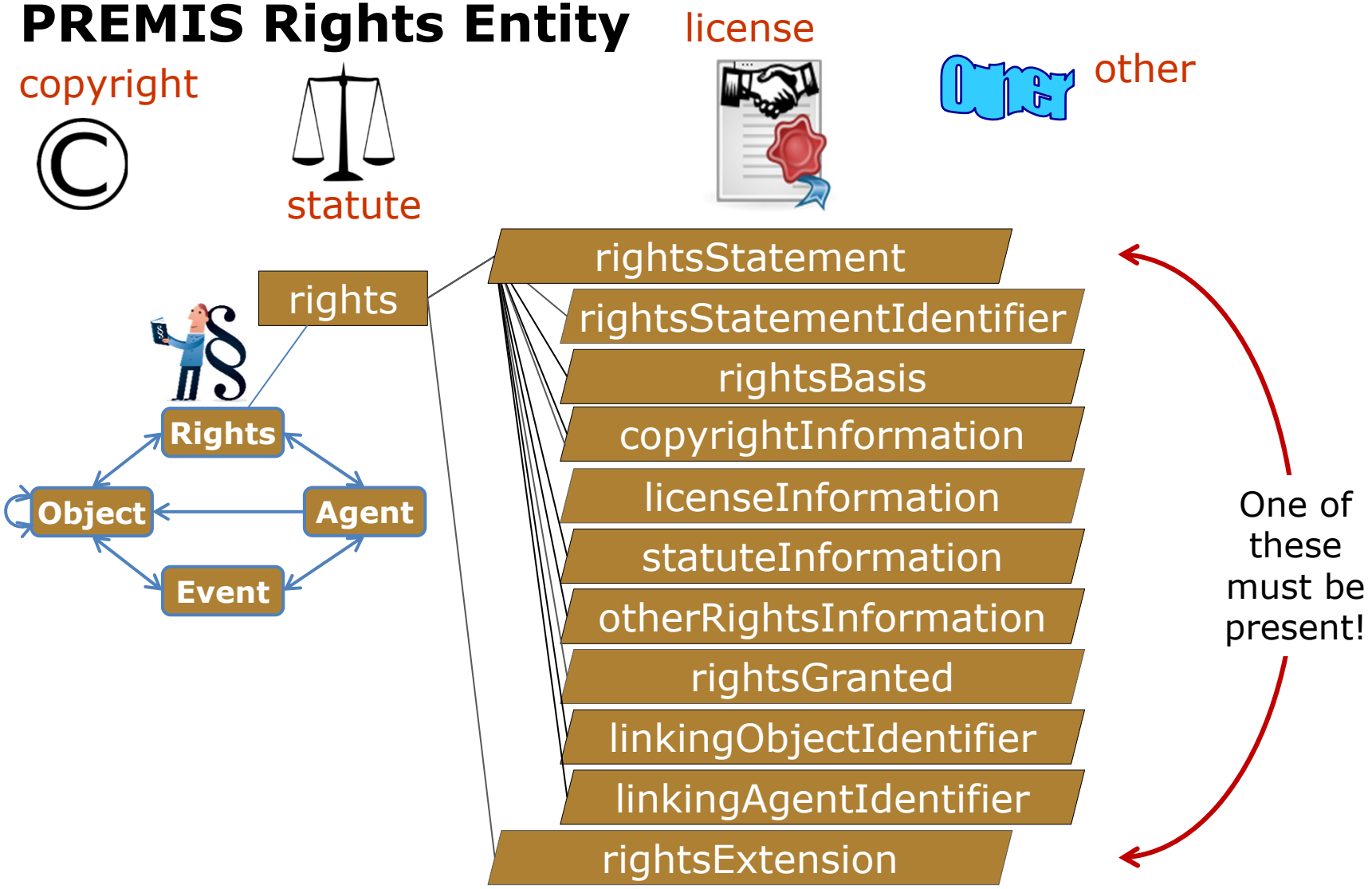
PREMIS Agent Entity



Properties of Entities - Semantic units



PREMIS Rights Entity



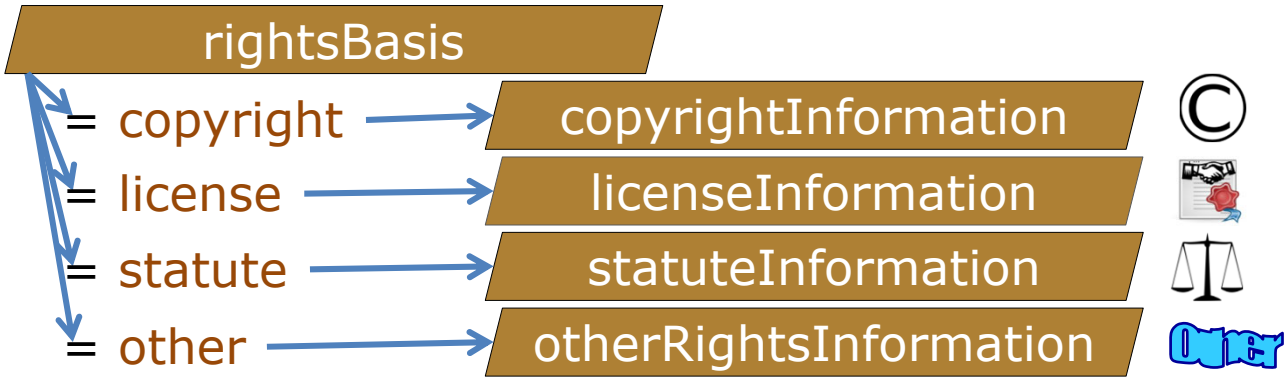
Dependent units about rights

Specifying different types of rights

- rightsBasis
- copyrightInformation
- licenseInformation
- statuteInformation
- otherRightsInformation

Dependent units about rights

Specifying different types of rights



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

Example rightsBasis and copyrightInformation

rightsBasis = copyright

copyrightInformation

copyrightStatus = copyrighted

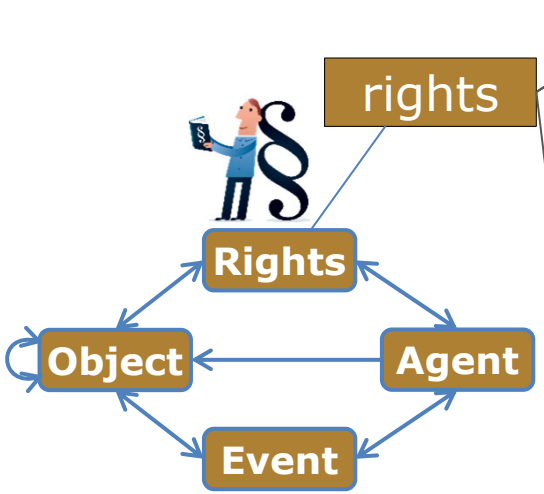
copyrightJurisdiction = us

copyrightStatusDeterminationDate = 2008-09-10

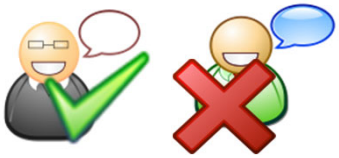
copyrightNote = Copyright expiration expected in 2022

copyrightDocumentationIdentifier = [link]

PREMIS Rights Entity



- rights
  - rightsStatement
  - rightsStatementIdentifier
  - rightsBasis
  - copyrightInformation
  - licenseInformation
  - statuteInformation
  - otherRightsInformation
  - rightsGranted
  - linkingObjectIdentifier
  - linkingAgentIdentifier
  - rightsExtension



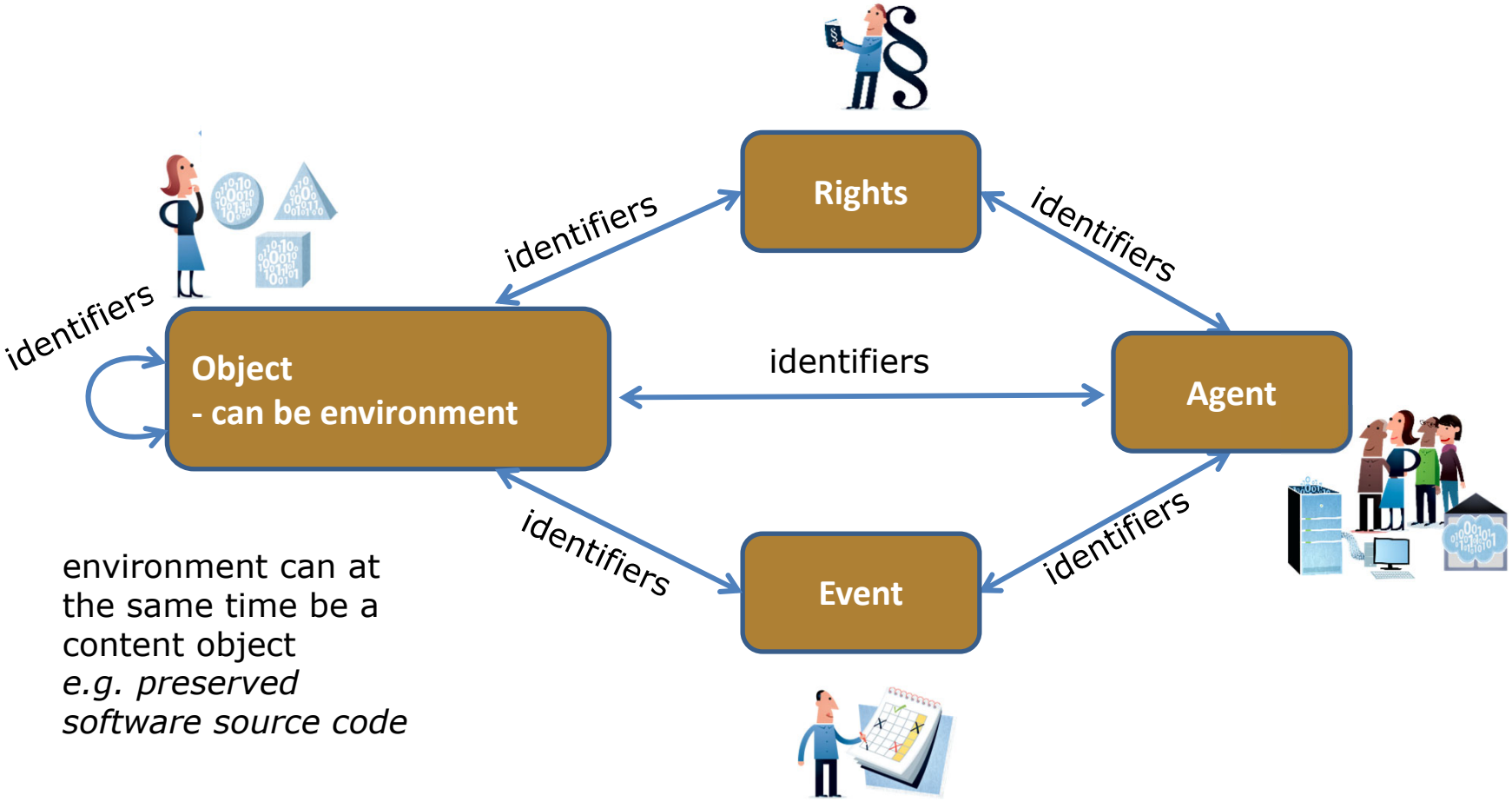
rightsGranted

- What action is allowed?
- Under what conditions?
- Are there time constraints?

Contains

- **Act** (e.g. migrate, modify, … could use *eventType values)*
- **Restriction** (description of condition or limitation on act)
- **termOfGrant** *(start and end dates of rights granted)*
  - **startDate** (e.g. 2005-01-01)
  - **endDate** (e.g. 2005-01-01)
- **termOfRestriction** *(start and end dates of restriction granted)*
  - **startDate** (e.g. 2005-01-01)
  - **endDate** (e.g. OPEN)
- **rightsGrantedNote** *(additional inf. about the rights granted)*

Data Model - environments

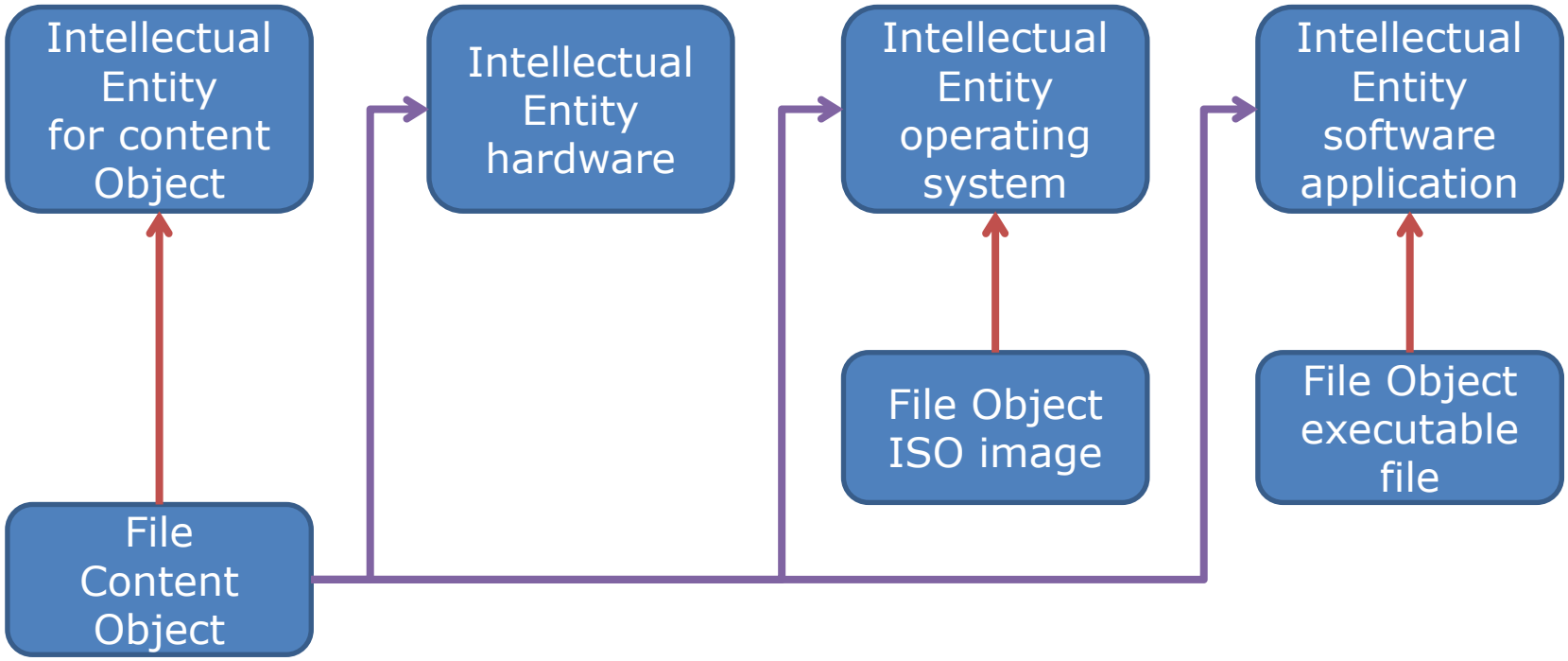


environment can at the same time be a content object
e.g. preserved software source code

Environments as independent objects

- What is needed to render or use a content object
  - Operating system
  - Application software
  - Hardware
  - Computing resources

Example:
An object and its rendering environment



represents =
relationshipType: structural
relationshipSubType: represents

requires =
relationshipType: dependency
relationshipSubType: requires

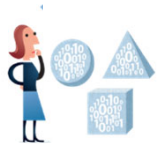
An agent is actually an environment acting as an agent

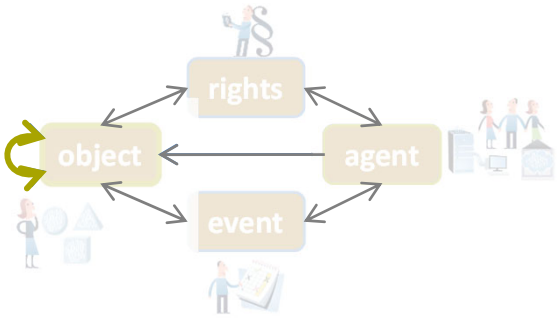
e.g. a format migration software agent involved in a preservation action

Relation points out the environment object acting as the agent

PREMIS PREservation Metadata Implementation Strategies

object





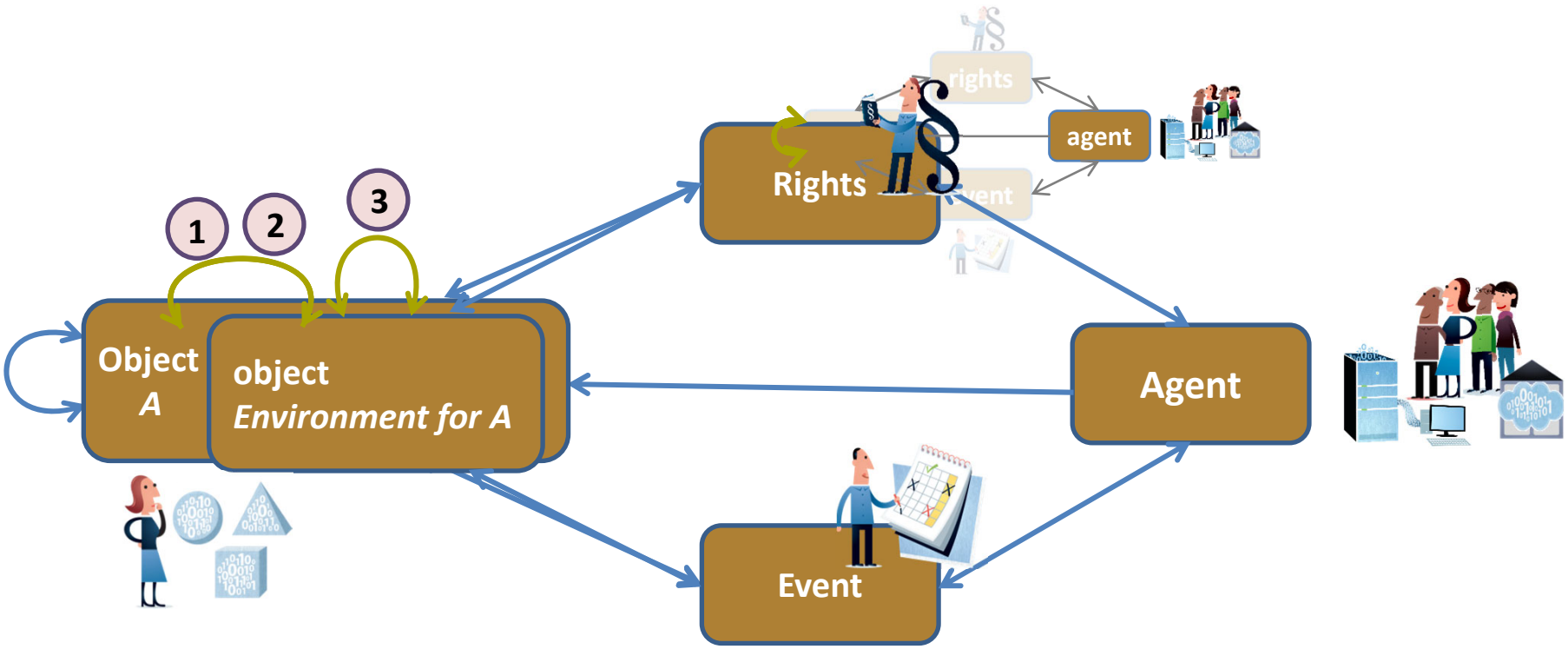
- objectIdentifier
- objectCategory
- objectCharacteristics
- significantProperties
- preservationLevel
- originalName
- storage
- signatureInformation
- environmentFunction
- environmentDesignation
- environmentRegistry
- environmentExtension
- relationship
- linkingEventIdentifier
- linkingRightsStatementIdentifier

e.g. function: SW, level: 1
function: OS, level: 2

name, version, origin, …
e.g. for SW Windows XP Professional

name, key, role
-alternative: Link to an external registry

relationshipType: dependency
relationshipSubType: requires



1. object to environment - specify computational context
2. environment to object - documentation, specifications, surrogates
3. environment to environment - inclusion, dependency, derivation, other

Additional environment information

- relationship
  Different environments can support different uses/*purposes* of objects
  create, **edit**, **modify**, **render**…

- relationship
  Characteristics describing how the environment supports its purpose
  unspecified, **minimum**, **known to work**, **recommended** …

PREMIS Object Entity – Semantic Units

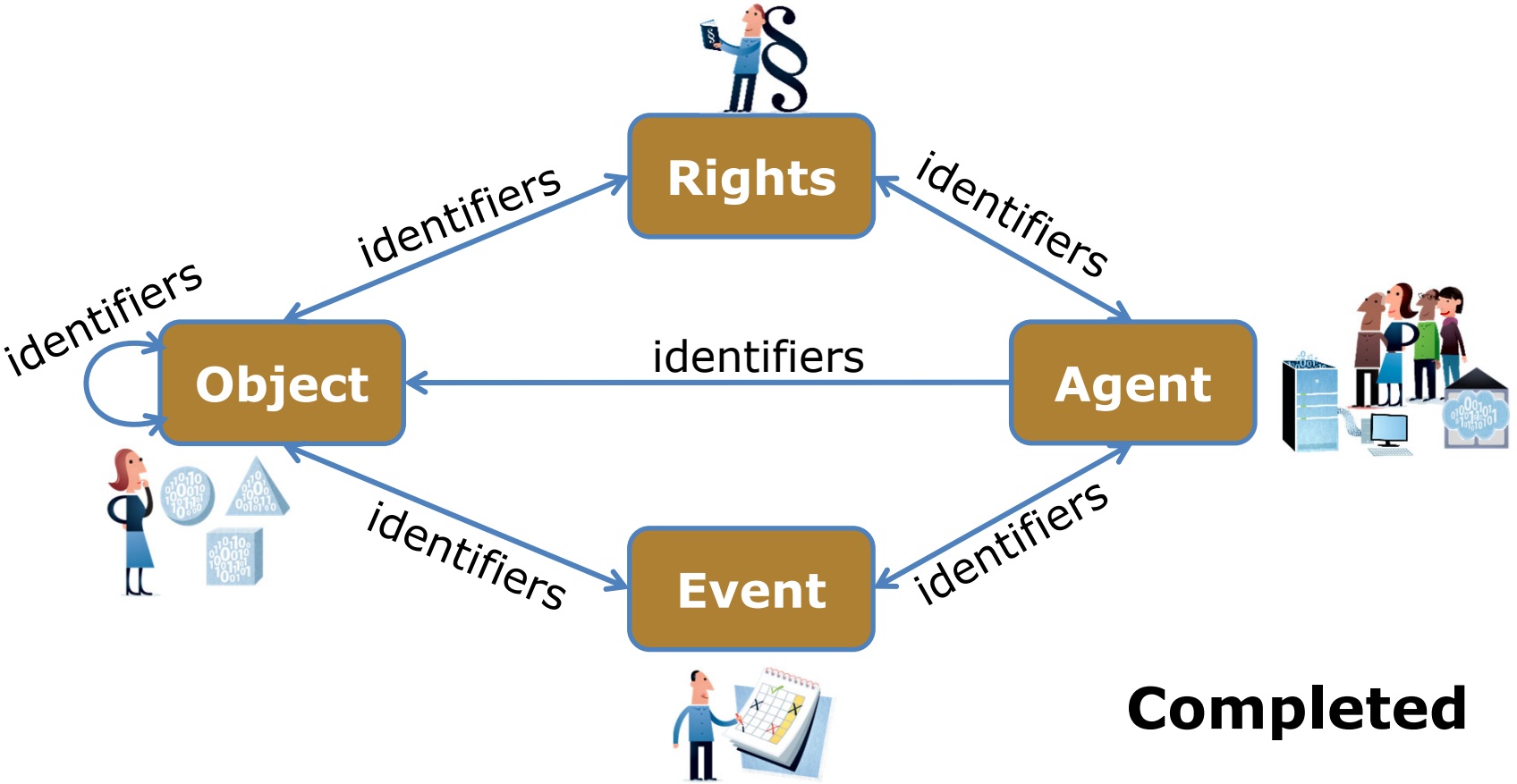
- relationship
  - relationshipType
  - relationshipSubType
  - relatedObjectIdentifier
    - (type, value, sequence)
  - relatedEventIdentifier
    - (type, value, sequence)
  - relatedEnvironmentPurpose
  - relatedEnvironmentCharacteristic

A controlled vocabulary is available at: http://id.loc.gov/vocabulary/preservation/environmentPurpose

e.g. create, edit, modify, render

e.g. unspecified, minimum, recommended, known to work

PREMIS 3 - Entities



Completed

PREMIS PREservation Metadata Implementation Strategies

PREMIS Conformance & interoperability

Micky Lindlar
TIB - German National Library of Science and Technology

HOW TO USE PREMIS?

The Data Dictionary in action: PREMIS Conformance and repository interoperability

PREMIS Conformance statement

- http://www.loc.gov/standards/premis/premis-conformance-20150429.pdf

Baseline requirements:

- For every implemented Entity (Objects, Events, Rights, Agents) mandatory semantic units must be captured
  - For those levels of Object that the repository supports (IE, representation, file, bitstream)
- Requirements for
- Shared name = Shared definition!
- Shared definition without shared name -> needs documentation

PREMIS Conformance Levels

	A – Object Entity Only	B – Object, Event & Agent
Level 1 – Mapping	Internal metadata is mapped to PREMIS & documented	
Level 2 – Export	Internal metadata can be exported (via a tool-/process-supported routine) to PREMIS	
Level 3 – Internal Implementation	PREMIS is implemented as internal metadata schema	

Example: What's conformant and what isn't?

1.1 objectIdentifier (M, R)
1.1.1 objectIdentifierType (M, NR)
1.1.2 objectIdentifierValue (M, NR)

✗ None

✓ DOI: 10.5281/zenodo.5569542

✓ Eindeutiger Bezeichner:10.5281/zenodo.5569542

✓
```
<dc:identifier>https://zenodo.org/record/5569578</dc:identifier>
<dc:identifier>10.5281/zenodo.5569578</dc:identifier>
<dc:identifier>oai:zenodo.org:5569578</dc:identifier>
```

✗
```
<objectIdentifier>fmt/18</objectIdentifer>
```

✓
```
premis:objectIdentifier
        premis:objectIdentiferType=„doi“
        premis:objectIdentifierValtue=„10.5281/zenodo.5569542“
```

Which Entities to implement?

- Object is the core Entity (level A);
- Event and Agent are closely related (level B); implementing Agents has strong implications: it means the repository is able to manage and follow the use of its Agents in the Object lifecycle.
- The Rights Entity (excluded from the conformance statement) helps a repository tracking the intellectual property rights governing the Object, or some institutional policy.

Use case: For building other standards

- As a basis extended with locally-defined elements: Preservation Metadata Dictionary (Netherlands Institute for Sound and Vision). https://publications.beeldengeluid.nl/pub/389/BIJLAGE-C_Metadatadictionary-English.pdf
- As a free source of inspiration: DEPIP (Data Exchange Protocol for Interoperability and Preservation), ISO 20614. https://www.iso.org/standard/68562.html

- No conformance, inspiration!

Use case: As a self-assessment tool

- Am I able to provide information about my digital assets following the Data Dictionary structure and requirements?
  - I.e., documenting the mapping between my metadata structure and PREMIS semantic units.

- Conformance level 1 « through mapping » (see the Conformance Statement document, p. 5: http://www.loc.gov/standards/premis/premis-conformance-20150429.pdf#page=5)

Use Case: As an export format

- Preferably in a PREMIS-endorsed expression (XML or RDF)

- Conformance level 2 « through export » (see the Conformance Statement document, p. 5: http://www.loc.gov/standards/premis/premis-conformance-20150429.pdf#page=5)

Use case: As the native format of the repository Data Management module

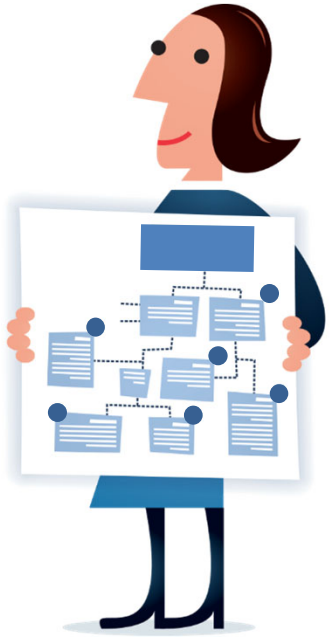
Where to store PREMIS data?

- Any technology, using a PREMIS-endorsed expression or not, can be used
  - XML database
  - RDF triple store
  - relational database
  - etc.

- Conformance level 3 « through internal implementation » (see the Conformance Statement document, p. 6: http://www.loc.gov/standards/premis/premis-conformance-20150429.pdf#page=6)

PREMIS PREservation Metadata Implementation Strategies

Karin Bredenberg
Kommunalförbundet Sydarkivera



WRAP UP

Sum up – Data Dictionary

Lots of other information

Where? and What??

- Where
  - Resources: http://www.loc.gov/standards/premis/
  - Zenodo: https://zenodo.org/communities/premis
  - PREMIS Implementors Group Forum: PIG@listserv.loc.gov
- What PREMIS is for today have given you a good explanation to and you now need to explore it further

Images in this style is taken from digitalbevaring.dk

Resources

- Understanding PREMIS
- PREMIS-in-METS guidelines
- Conformance statement
- Examples of implementation

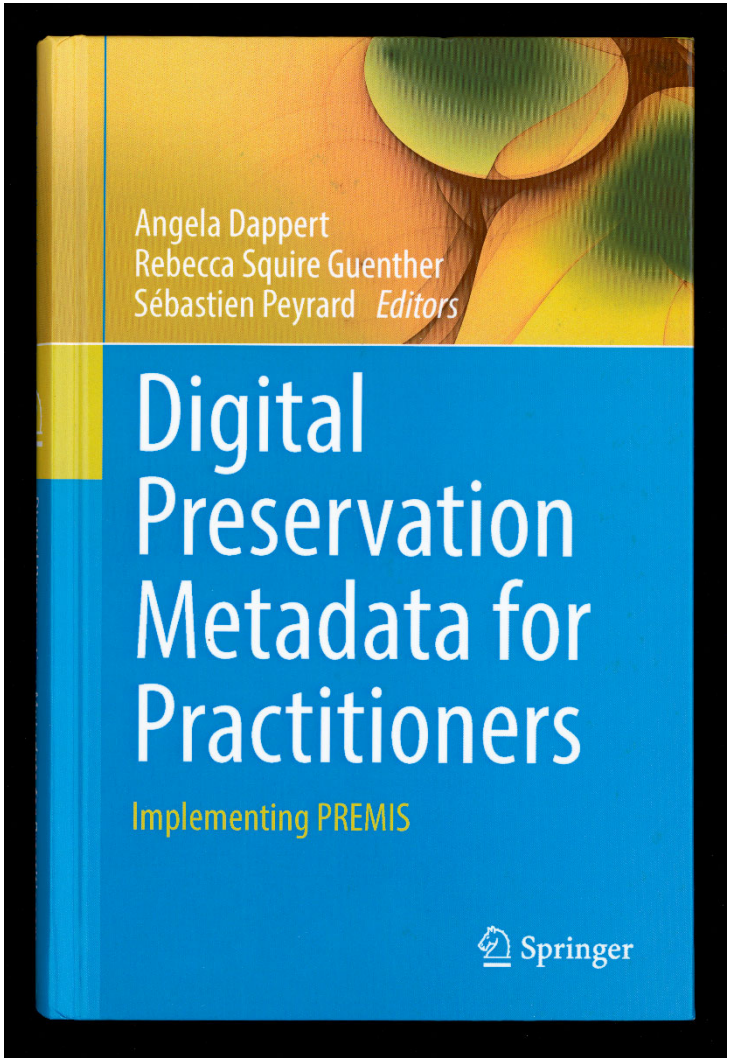
http://www.loc.gov/standards/premis/

Current activity

- Move the DD to a TEI-format to simplify maintenance and transformations to publications
- DD updates following the ontology work
- Set up of Zenodo
- Setup of new wiki
- Rights overhaul

Book

- ISBN E-book: 978-3-319-43763-7
- ISBN Hardcover: 978-3-319-43761-3
- http://www.springer.com/gp/book/9783319437613

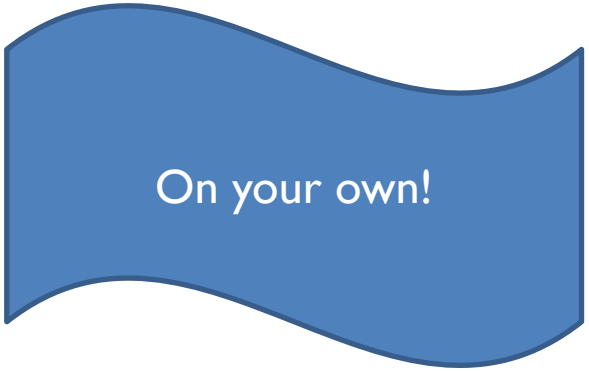


Exercises

- Today have been really filled!
- Three exercises to start working with PREMIS metadata at home
  - Print them out!
- Solutions is also published!

Exercises

- Today have been really filled!
- Three exercises to start working with PREMIS metadata
  - Print them out!
- Solutions is also published!
- The aid is seen on next slide



Sample Data Dictionary table of contents

- Version 3 Hierarchical listing of semantics units.pdf https://doi.org/10.5281/zenodo.5569578

Entity semantic units

NB: Semantic units are applicable for Intellectual Entities, Representations, Files and Bitstreams unless otherwise indicated.

1.1 objectIdentifier (M, R)
- 1.1.1 objectIdentifierType (M, NR)
- 1.1.2 objectIdentifierValue (M, NR)

1.2 objectCategory (M, NR)

1.3 preservationLevel (O, R) [Intellectual Entity, Representation, File]
- 1.3.1 preservationLevelType (O, NR) [Intellectual Entity, Representation, File]
- 1.3.2 preservationLevelValue (M, NR) [Intellectual Entity, Representation, File]
- 1.3.3 preservationLevelRole (O, NR) [Intellectual Entity, Representation, File]
- 1.3.4 preservationLevelRationale (O, R) [Intellectual Entity, Representation, File]
- 1.3.5 preservationLevelDateAssigned (O, NR) [Intellectual Entity, Representation, File]

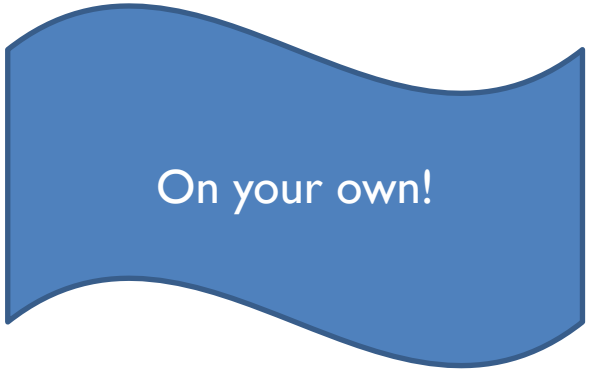
1.4 significantProperties (O, R)
- 1.4.1 significantPropertiesType (O, NR)
- 1.4.2 significantPropertiesValue (O, NR)
- 1.4.3 significantPropertiesExtension (O, R)

1.5 objectCharacteristics (M, R) [File, Bitstream]
- 1.5.1 compositionLevel (O, NR) [File, Bitstream]
- 1.5.2 fixity (O, R) [File, Bitstream]
  - 1.5.2.1 messageDigestAlgorithm (M, NR) [File, Bitstream]
  - 1.5.2.2 messageDigest (M, NR) [File, Bitstream]

On your own!

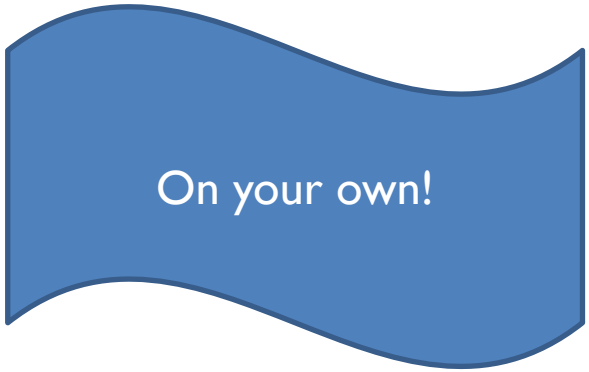
PREMIS Object Entity – Exercise

- Exercise to get a feeling for the object!
  - Objectexercise.pdf
  - https://doi.org/10.5281/zenodo.5569614
- Page 1
  - Find the different object types!
- The rest of the pages
  - With the data stated, fill your PREMIS semantic units.
  - Take help from the hand-out with all the semantic units!



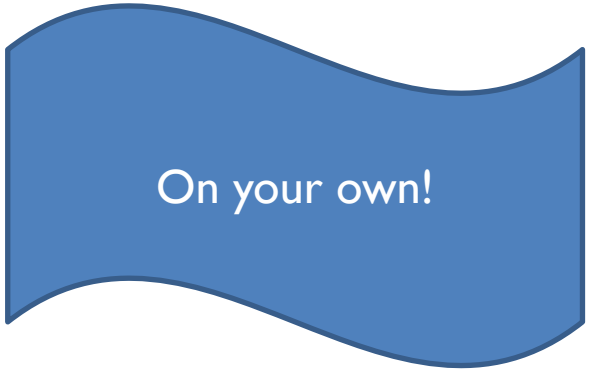
PREMIS Events, Agents and Rights Entity – Exercise

- Exercise to get a feeling for the events, agents and rights!
  - Eventsagentsrightsexercise.pdf
  - https://doi.org/10.5281/zenodo.5569644
- For the pages
  - With the data stated, fill your PREMIS semantic units.
  - Take help from the hand-out with all the semantic units!



PREMIS Environments – Exercise

- Exercise to get a feeling for the environments!
  - Environmentsexercise.pdf
  - https://doi.org/10.5281/zenodo.5569651
- For the pages
  - With the data stated, fill your PREMIS semantic units.
  - Take help from the hand-out with all the semantic units!



Today

- Have been comprehensive!
- A longer Tutorial recording avilable!
  - Part 1: https://youtu.be/GiQCNgw-HOE
  - Part 2: https://youtu.be/LhRMF6vP_PU

- Thank you! BIBLIOTEKA NARODOWA

Finally…

PREMIS is a community standard.

- Send examples
- Ask questions
- Send suggestions
- Take part!

Thank you!

Eld, Micky and Karin