PREMIS PREservation Metadata Implementation Strategies

Proposed Data Model Changes for PREMIS 3.0

Angela Dappert
Digital Preservation Coalition

Changes in the PREMIS Data Model.

- Next major version of the PREMIS Data Dictionary
- Released by end of 2013
- Still in proposal phase in the Working Group

- Revised data model
  - Integrated Intellectual Entities
  - Better way to describe Environments

The PREMIS Data Model

Slide by S. Peyrard

Intellectual Entity

identifier

identifiers

Rights

identifiers

identifiers

Object

Agent

identifiers

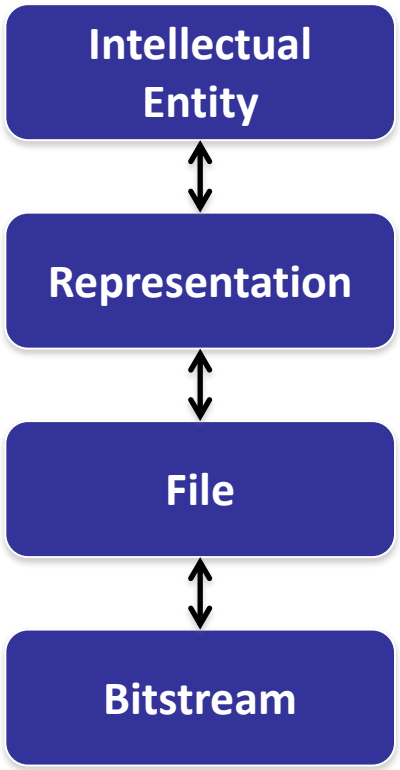
identifiers

Environment

Event

Intellectual Entities

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



Intellectual Entities Implementation

- Capture descriptive metadata.
- Assumed to be held in a container metadata schema.
- PREMIS Objects link to it.

- Can capture versioning information and metadata update events for intellectualEntities, such as articles and issues.
- Can represent a collection, FRBR work, FRBR expression, fonds, series, files (in the archival sense)

Represent a collection, FRBR work, FRBR expression, fonds, series, files … in order to

- capture descriptive metadata
- to have business requirements associated with them or to be referenced in business requirements (such as significant characteristics, risk definitions, guidelines for preservation actions, etc.)
- structural and derivative relationships
- rights and preservation rights information
- events and agents

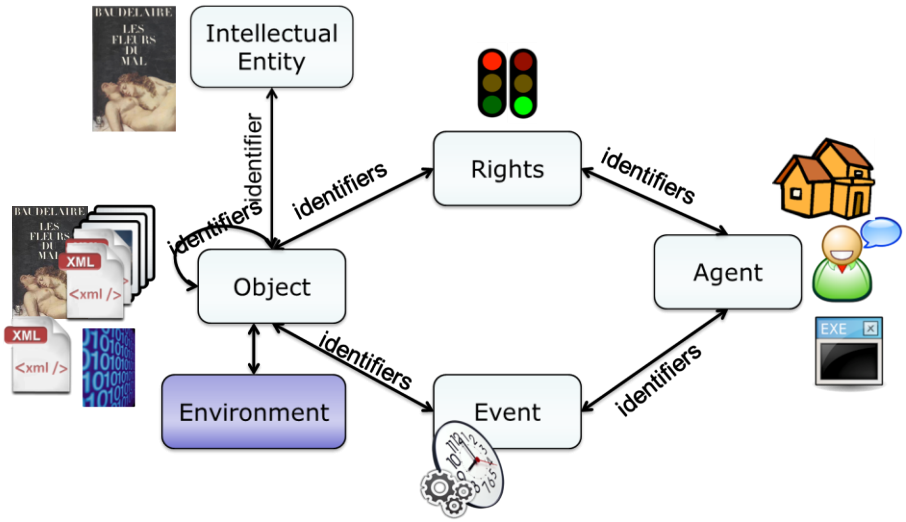
- This can only partially be accommodated by container metadata systems and their associated descriptive or administrative metadata.
- Core preservation metadata (provenance aspects)

Capture versioning information and metadata update events for intellectualEntities, such as articles and issues

- This should be accommodated on the container metadata level.
  METS does not record metadata of metadata
- This could be addressed by treating the metadata as a file that can have its own metadata
  which can record modification information.
  => awkward

Treat IntellectualEntity as Object Type

- Data model more compact
- Simplify the dictionary
  - drop linkingIntellectualIdentifier
- Data dictionary more self-contained
- Directly attach events, rights, indirectly attach agents to intellectual entities



Required Changes

- 1. Remove IntellectualEntities as stand-alone entity
- 2. Rename IntellectualEntities -> IntellectualEntity
- 3. Add IntellectualEntity as Object type
- 4. Define semantic units as for Representation

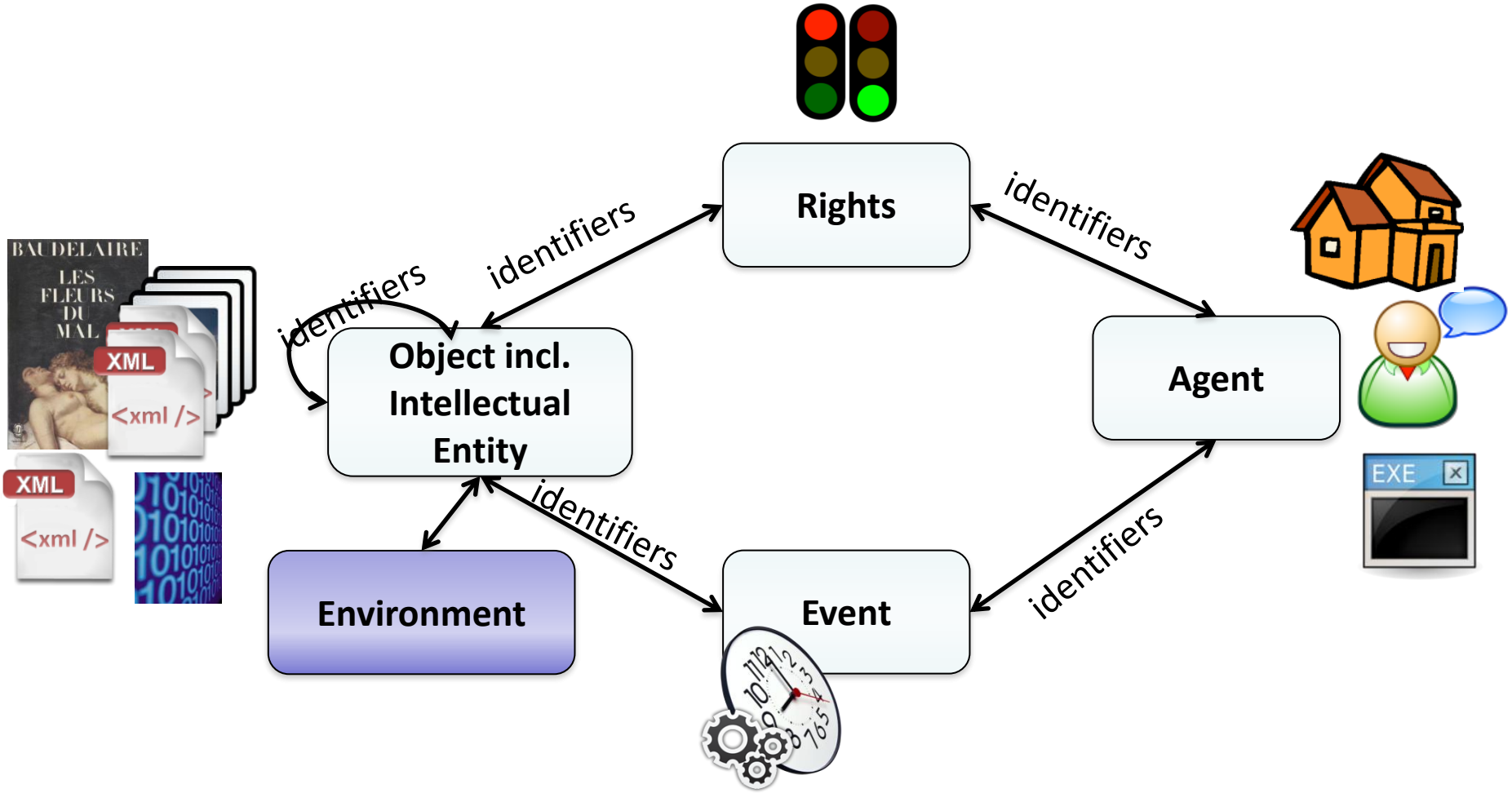
- [5. Decide whether "environment" semantic unit is considered applicable to IntellectualEntity.
  - No HW and SW (technical) environments.
  - Default technical environment for all representations
  - Policy environments (such as in which reading room it should be accessible, etc.)]

Required Changes

- 6. Update relationship:
  structural relationships may be used to record logical containment (e.g. between an article and an issue)
- 7. Remove linkingIntellectualEntityIdentifier:
  -> Use "relationship" instead.
- 8. Update definition of Object entity:
  need not be related to any digital object
- [Not:
  - 9. Add a semantic unit to store its type.
     TYPE attribute: article, monograph etc...
  - 10. Add a semantic unit to store the FRBR-level (works, expressions and manifestations) or archival categorizations.

    => use descriptive metadata instead]

Environments



Environment

- Software
- Hardware
- A format
- A document
  - A policy document
  - A manual
  - Documentation
- A cheat sheet
- A user behaviour study
- "Other representation information"
- Not: A process, workflow, preservation plan description

Modelling Choices

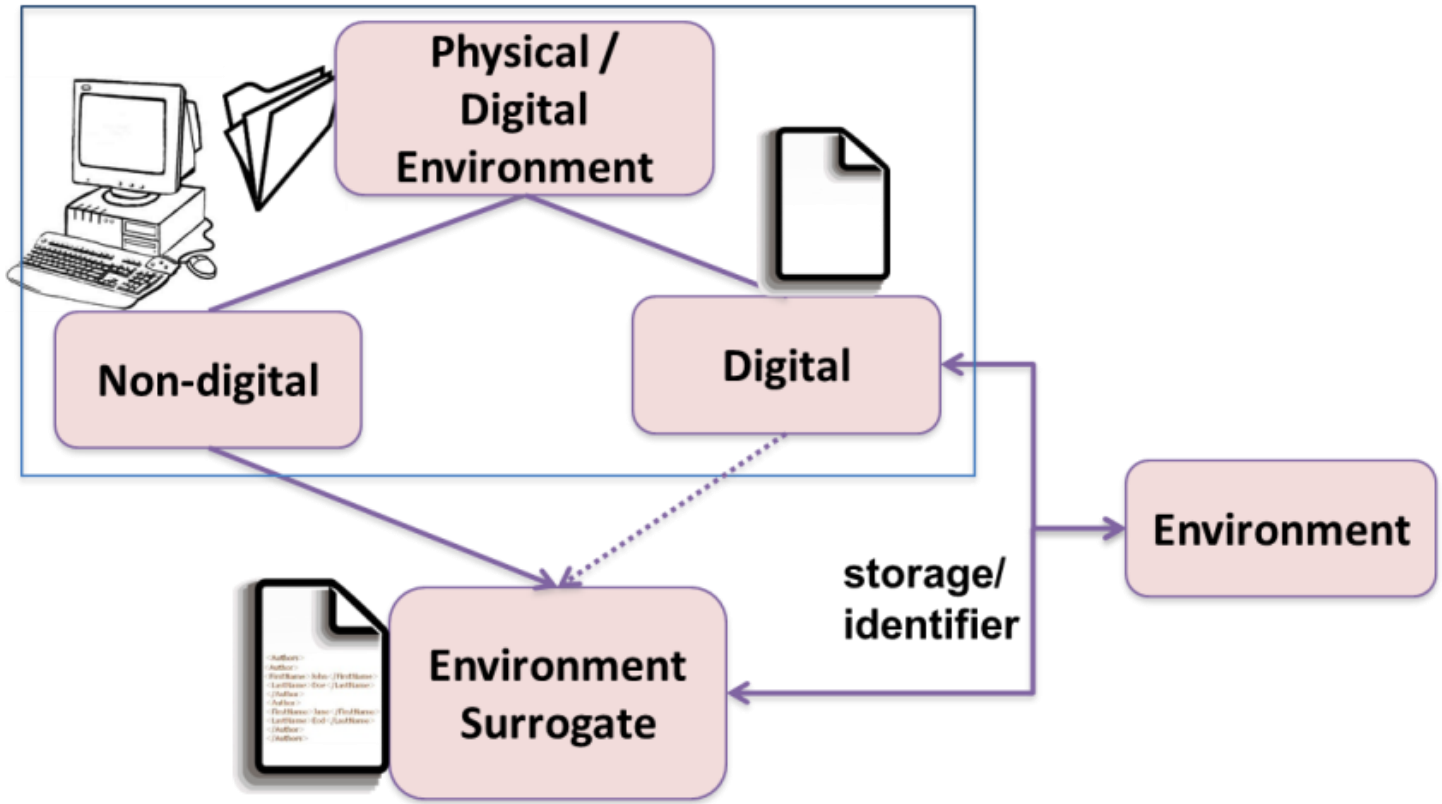
- Compliance with OAIS
- Straightforward Data Dictionary semantics
  - easy to implement
  - Implementable in the existing XML Schema and PREMIS ontology
- Backward compatibility
- Forward compatibility
  mapping PREMIS 2 -> PREMIS 3

High-level Requirements

- A high-level data model
- **No** detailed characteristics specific to an environment type
- A standardized way of treating environments
- Sharable and exchangeable

- Modularised environment descriptions (aggregates)
  (as a network)
- Re-usable environment description
  (across different Objects)
- Re-usable environment description
  (across repositories and registries)
- Distributed environment description
  (across repositories and registries)

Requirements



Requirements

- Capture designation information (name and version)
- Capture the function of an environment (type and use)
- Capture relationships with a variety of relationship types: whole/part, replacement, dependency (more later)
- Representations and Files -> Representations and Files
- Environment -> supporting documentation
- Object -> environment description in an external registry
- Environment -> complementary external description

- **Not**: business policies, preservation plans, business or research process descriptions and workflows referencing entities

The PREMIS Data Model

Slide by S. Peyrard

Intellectual Entity

identifier

identifiers

Rights

identifiers

identifiers

Object

Agent

Environment

identifiers

Event

identifiers

Example: Object Entity

Main types of information

- identifier
- technical object characteristics
- creation information
- software and hardware environment
- digital signatures
- relationships to other objects
- links to other types of entity

PREMIS – Environment Metadata

1.5.5 creatingApplication

1.5.5.1 creatingApplicationName
1.5.5.2 creatingApplicationVersion
1.5.5.3 dateCreatedByApplication
1.5.5.4 creatingApplicationExtension

Gap Analysis

- **OAIS focus on Object:**
  - **Creating Applications are Environments**
  - **Life-cycle view treating Environments uniformly**

Semantic Unit: Environment

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

PREMIS – Environment Metadata

1.8 **environment**
- 1.8.1 environmentCharacteristic
- 1.8.2 environmentPurpose
- 1.8.3 environmentNote

1.8.4 **dependency**
- 1.8.4.1 dependencyName
- 1.8.4.2 dependencyIdentifier
  - 1.8.4.2.1 dependencyIdentifierType
  - 1.8.4.2.2 dependencyIdentifierValue

1.8.5 **software**
- 1.8.5.1 swName
- 1.8.5.2 swVersion
- 1.8.5.3 swType
- 1.8.5.4 swOtherInformation
- 1.8.5.5 swDependency

1.8.6 **hardware**
- 1.8.6.1 hwName
- 1.8.6.2 hwType
- 1.8.6.3 hwOtherInformation

1.8.7 environmentExtension

Environment Example: PDF File

environmentCharacteristic = known to work

environmentPurpose = render

hardware/hwName = Intel Pentium II

hardware/hwType = processor

dependency/dependencyName= Mathematica 5.2
True Type math fonts

software/swName = Adobe Acrobat Reader

software/swVersion = 6.1

software/swType = renderer

software/swDependency = Windows NT

software/swName = Windows NT

software/swVersion = 5.0

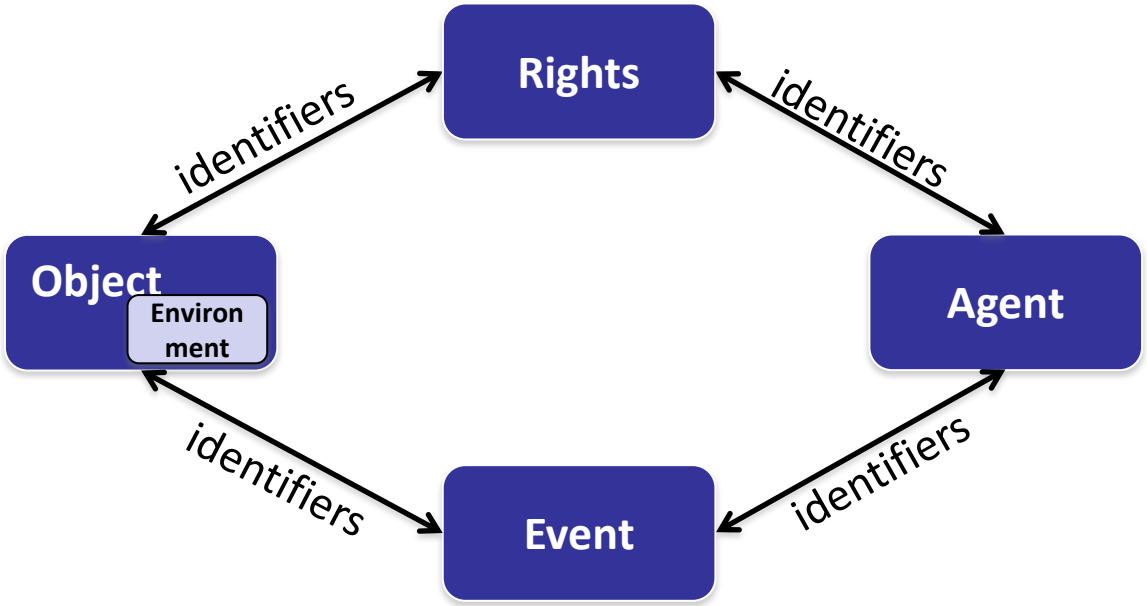
software/swType = operatingSystem

Gap Analysis I - Subordinate to Object

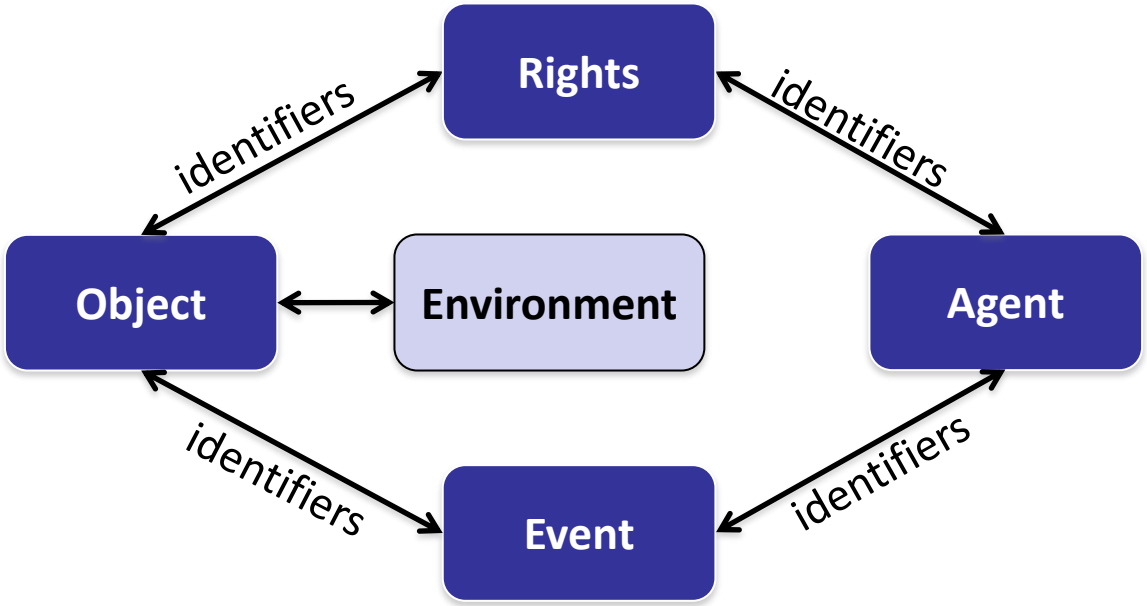
- **Environments can be too complex** to handle in an Object repository.
- **Solution too redundant** (verbose, cumbersome to manage evolution)
  Rarely specific to a single Object.
- **Unable to describe stand-alone Environments** independent of Objects
  Repositories and registries need to speak the same language
- **Unable to decouple Object or Agent descriptions from environment related information**
  (to version and maintain environments separately)

- **Cause:** Environment Subordinate to Object
  - **Solution**: Environment as first class entity

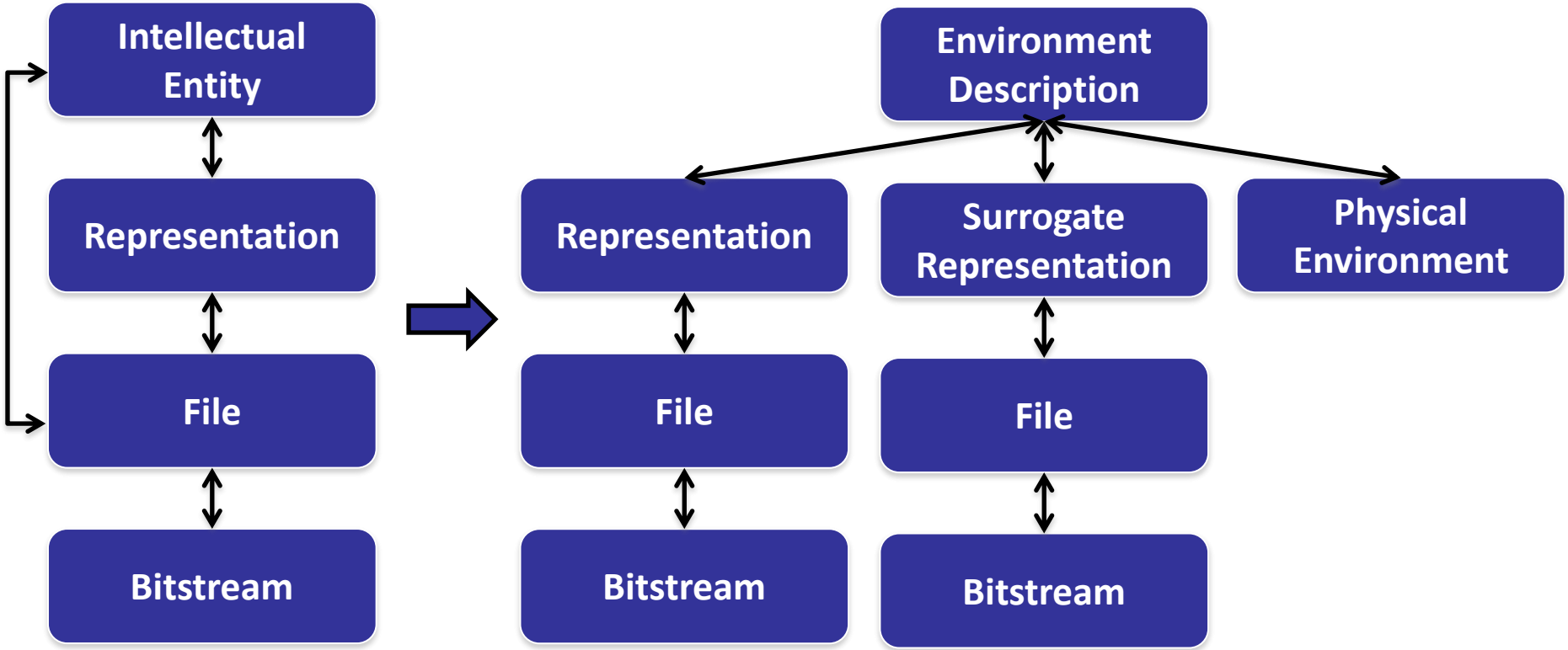
Gap Analysis I



Gap Analysis I



New or Re-Used Entity?



Intellectual Entity
Software operating system
Ubuntu 32-bit, version 12.10

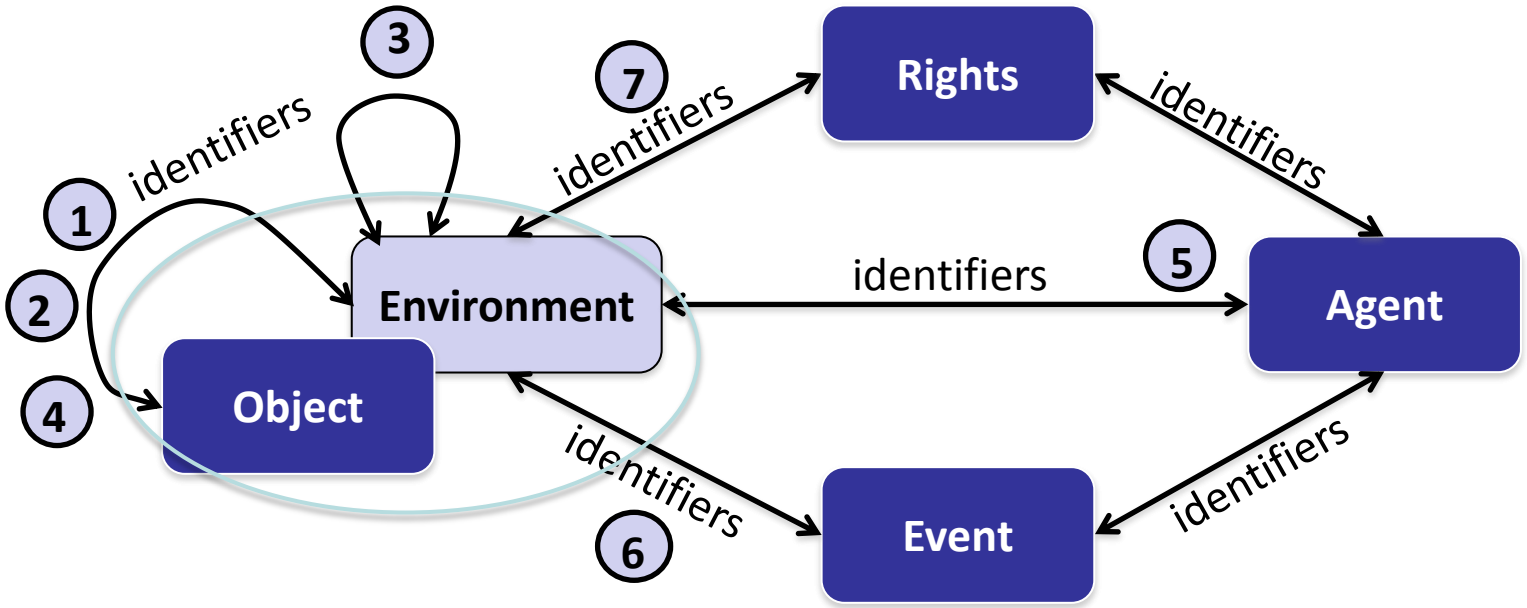
structural relationship
represents

File
size: 726970368
format name: application/x-iso9660-image

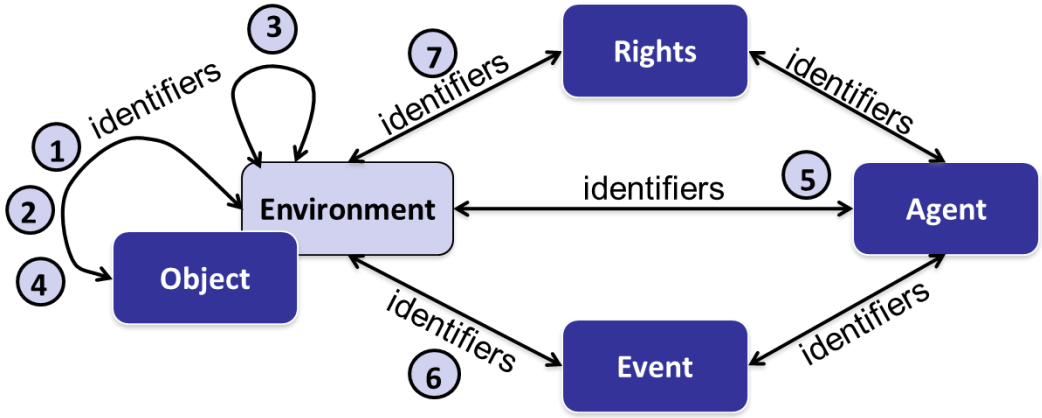
The ISO image is described as a file with technical characteristics. I also want to record that this file captures Ubuntu version 12.10

Gap Analysis II

- Unable to express all but the first type of relationship



Relationships



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

"Object": here a traditional repository Object

Gap Analysis III

- **Refer to external registries**: only for file formats
- **Versions**: only for software
- **Software dependencies**: maximally 2 levels
- Only applicable to **technical computing** environments – not: documentation, events, licenses and policies
- Cannot document the **nature of dependencies**
- **Software or hardware** used as an **Agent** cannot be **identified**

Requirements

- Describe tangible and intangible items
- Capture designation information (name and version)
- Capture the function of an environment (type and use)
- Capture relationships with a variety of relationship types: whole/part, replacement, dependency
- Representations and Files -> Representations and Files
- Environment -> supporting documentation
- Object -> environment description in an external registry
- Environment -> complementary external description

- **Not**: business policies, preservation plans, business or research process descriptions and workflows referencing entities

Example

Description of the Ubuntu Version 12.10 operating system and reference to its installation manual

objectIdentifier
    objectIdentifierType: ARK
    objectIdentifierValue: ark:/9999/b1
objectCategory: intellectual entity
objectFunction
    objectType: software
    objectSubType: operating system
objectDesignation
    objectName: Ubuntu
    objectVersion: Version: 12.10
    objectOtherInformation: 32-bit version
    objectNote: maintenance deadline: 2014-04
objectDesignation
    objectName: Ubuntu
    objectVersion: Quantal Quetzal

relationshipType: documentation
relationshipSubType: is documented in
relationshipPurpose: install
relationshipCharacteristic: known to work
relatedObjectIdentification
    relatedObjectIdentifierType: URL
relatedObjectIdentifierValue:
https://wiki.ubuntu.com/QuantalQuetzal/TechnicalOverview

Object:
<Documentation>

Example

objectIdentifier
objectIdentifierType: ARK
objectIdentifierValue: ark:/9999/b1
objectCategory: intellectual entity
objectFunction
objectType: software
objectSubType: operating system
Ubuntu 32-bit, version 12.10

relationshipType:
structural,
relationshipSubType:
represents

I am preserving the Ubuntu operating system as an ISO image. I can identify two levels of description (the File and the abstract Intellectual Entity) and link them with a PREMIS relationship.

objectIdentifier
objectIdentifierType: ARK
objectIdentifierValue: ark:/9999/c1
objectCategory: file
objectCharacteristics
compositionLevel: 0
size: 726970368
format
formatDesignation
format name: application/x-iso9660-image

Example

objectCategory: intellectual entity
objectFunction
    objectType: software
    objectSubType: operating system
objectDesignation
    objectName: Windows XP Professional
    objectVersion: Service Pack 3
objectRegistry
    objectRegistryName: PRONOM
    objectRegistryKey: x-sfw/8
    objectRegistryRole: generalization
objectRegistry
    objectRegistryName: IIPC Database
    objectRegistryKey:
http://gator1355.hostgator.com/~iipc/pwg/software.php?id=1006
    objectRegistryRole: identity
objectRegistry
    objectRegistryName: UDFR
    objectRegistryKey: http://udfr.org/udfr/u1r2415
    objectRegistryRole: generalization

This ISO image contains Windows XP Pro, SP3. I have 3 descriptions of Windows XP outside the registry.

3 different registry entries about Windows XP Professional

I am preserving a Word file, thus want to link to available information in external registries **without duplicating it**.
I use a simple link to an external entry

relationshipType: dependency
relationshipSubType: requires
relationshipPurpose: render
relationshipCharacteristic: recommended
relatedObjectIdentification
    relatedObjectIdentifierType: PUID
    relatedObjectIdentifierValue: x-sfw/1

objectCategory: file
size: 12348
format
    formatDesignation
        formatName: application/msword
        formatVersion: 97-2003

x-sfw/1
Description of Word 97-2003 in PRONOM

Thank you!