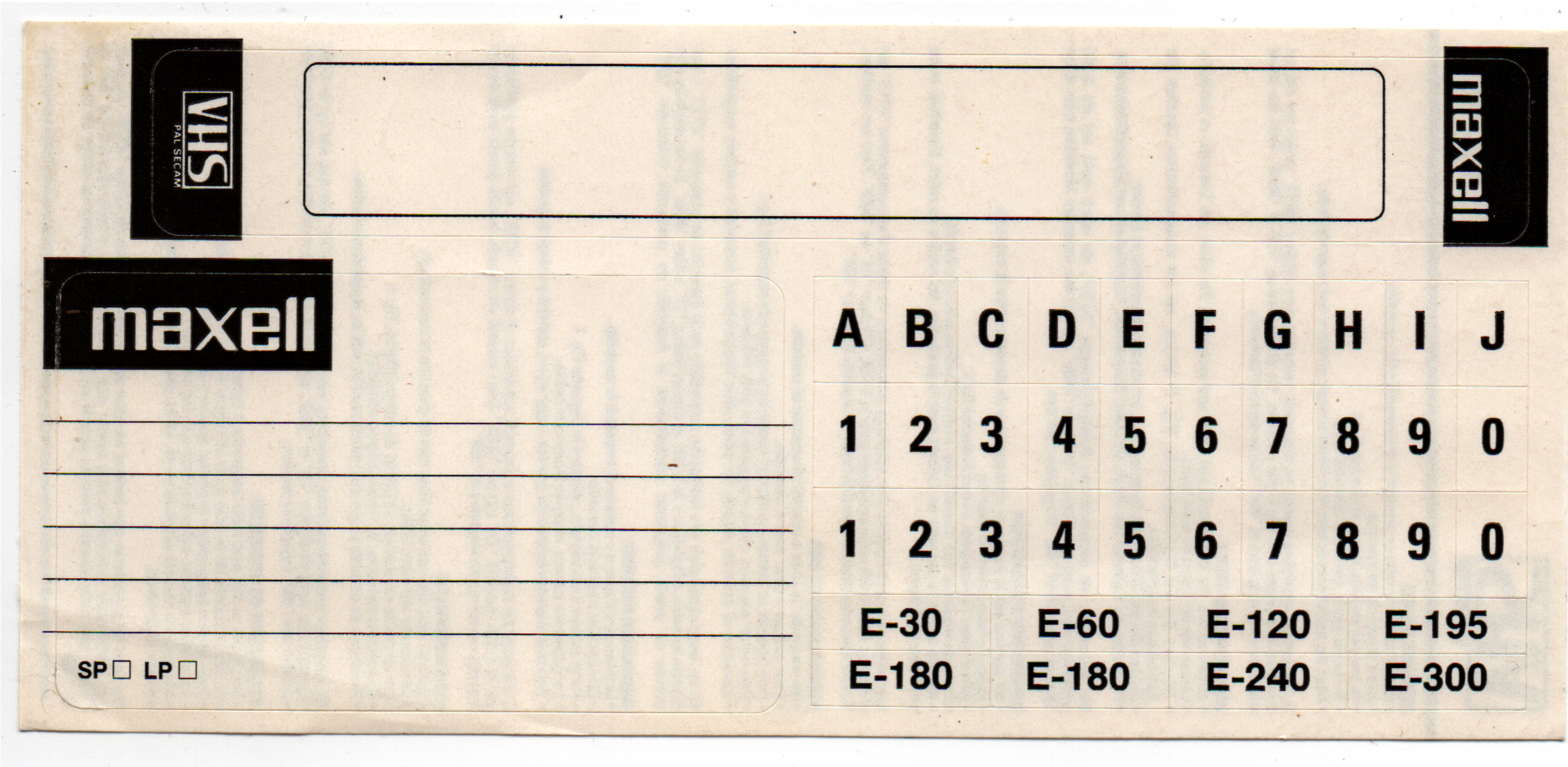# Topic 2 - Metadata

Examples? Ideas?

Retro Classic

Rockstars of

- Artist
- Title
- Album
- Date
- Cover art
- ...

MD Types for Preservation

- Descriptive
- Technical
- Environment
- Agent
- Rights
- Event
- Provenance
- Structure

Name some examples for each one of these types.

btw: Metadata is sometimes abbreviated as "MD".
Examples: techMD, rightsMD, sourceMD, digiprovMD etc

Examples:

- Descriptive: title, album
- Technical: resolution, bitrate
- Environment: tech required
- Agent: person, organization, tool
- Rights: copyright, ownership
- Event: an action that has a date
- Provenance: who owned it before
- Structure: file is part of a sequence

Important things

- Controlled vocabulary!
- Date/time: ISO 8601
  (2019-05-30T10:34:47+00:00)

- Standards: Interoperability

CoVoc Examples

- 35mm = 35 mm = 35 millimètre
- dup pos = duplicate positive
- de = deu = german = German = alemán
- YUV,4:2:2,10bpc = yuv422p10le
- Director = Directed by

Normalize date/time

- 4. May 2019
- May 4th, 2019
- 4.5.2019
- 5/4/2019
- …
- ISO 8601
  (2019-05-30T10:34:47+00:00)

Controlled Vocabulary

Your suggestions for:

- Descriptive?
- Technical?
- Other?

Standards

One To Rule Them All?

Exercise

Imagine some regular (preservation) workflows and identify:

- ## Events
  Which action is done?

- ## Agents
  Who is (involved in) doing it?

- ## Objects
  What does it apply to?

- ## Rights
  Who holds which rights to what?

- ## Relationships
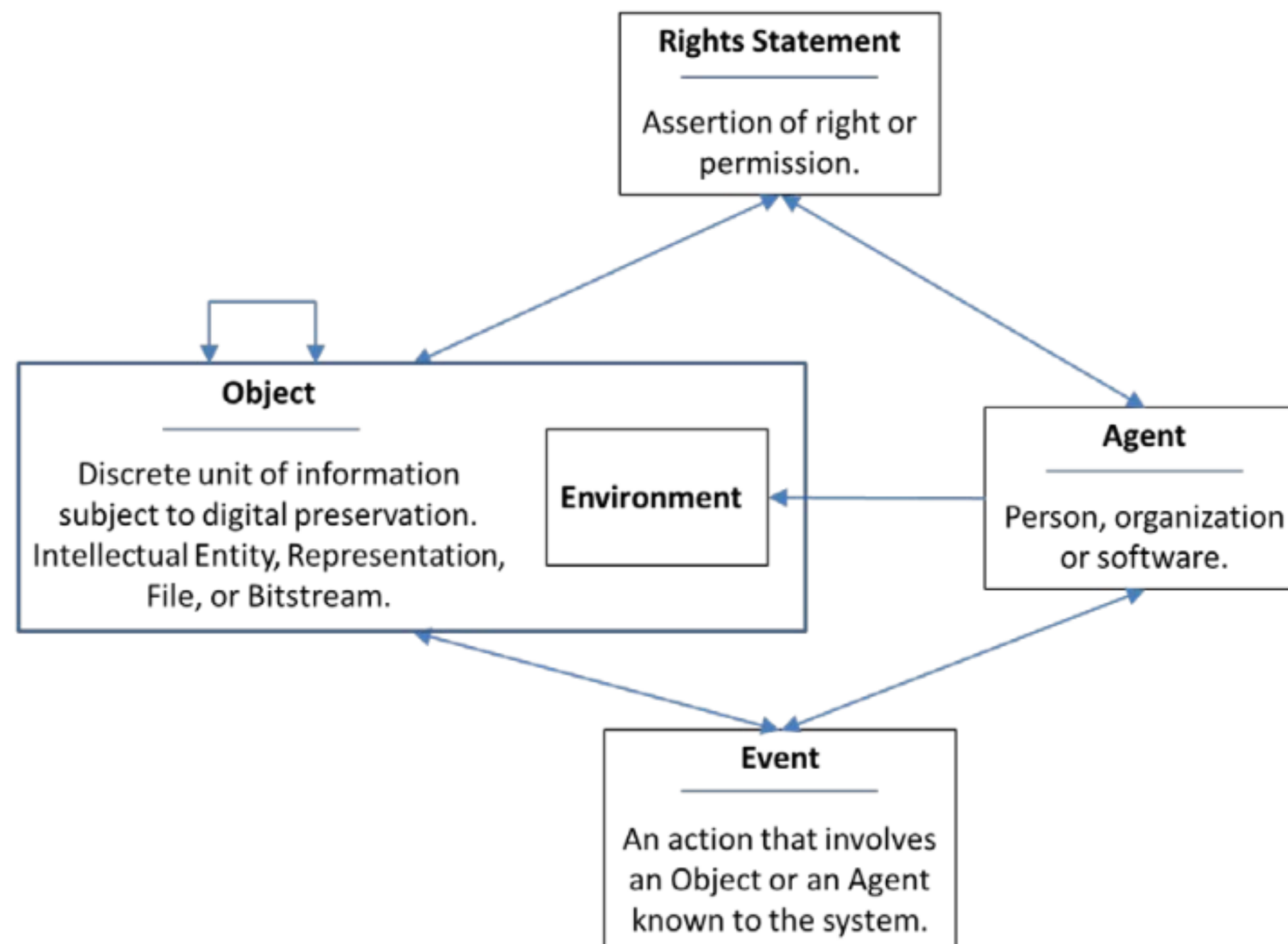 How do these entities relate to each other?

Exercise

- **Event** eg: "transcode"

- **Agent** eg: "tool:ffmpeg"

- **Object** eg: "preservation file"

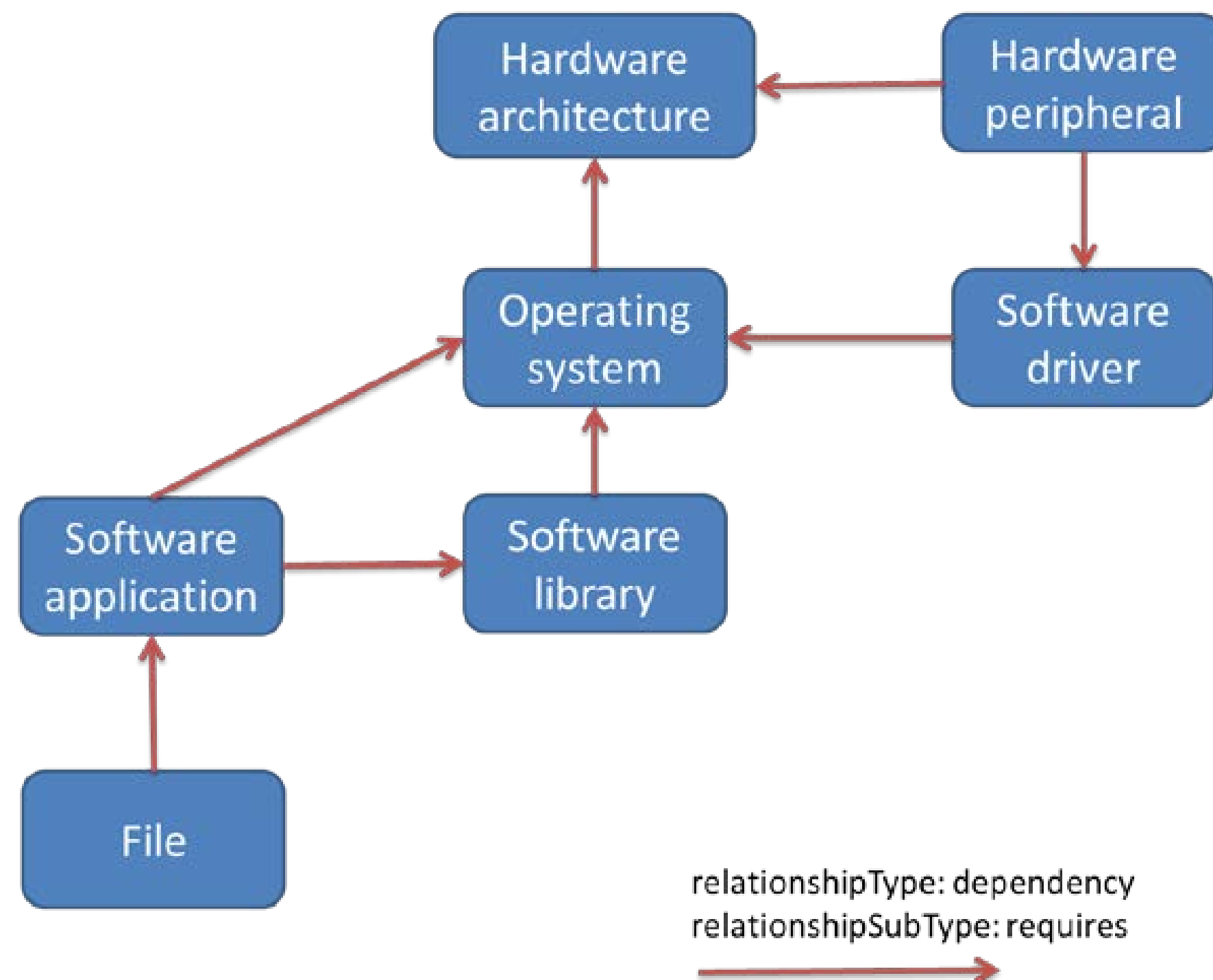- **Rights** eg: "item owner"

- **Relationship** eg: "converts", "is copy of"

Draw (and name) relationships between entities (one way/arrow). Write down what metadata you think should be captured in that entity.

Speaker notes

- **Object** (or Digital Object): a discrete unit of information subject to digital preservation.

  - **Environment:** Technology (software or hardware) supporting a Digital Object in some way (e.g. rendering or execution). Environments can be described as Intellectual Entities and captured and preserved in the preservation repository as Representations, Files and/or Bitstreams.

  - **Event:** an action that involves or affects at least one Object or Agent associated with or known by the preservation repository.

  - **Agent:** person, organization, or software program/system associated with Events in the life of an Object, or with Rights attached to an Object. It can also be related to an environment Object that acts as an Agent.

  - **Rights Statement:** assertion of one or more Rights or permissions pertaining to an Object and/or Agent.

PREMIS

Speaker notes

Looks familiar? Congratulations! You've just derived the PREMIS data model on your own.

PREMIS - Environment Stack

Hardware architecture ← Hardware peripheral

Hardware peripheral → Software driver

Software driver → Operating system

Operating system → Hardware architecture

Software application → Operating system

Software application → Software library

Software library → Operating system

File → Software application

relationshipType: dependency
relationshipSubType: requires

Good to Know

- **Dublin Core**: Core fields (& their names)
- **METS**: (MD container) descriptive, administrative, structural
- **PREMIS**: A metadata framework
- **EBUCore**: descriptive & technical (broadcast use case focus)
- **CEN EN 15907**: Comprehensive description of cinematographic works
- **FRBR ("furbur")**: Comprehensive description of bibliographic works
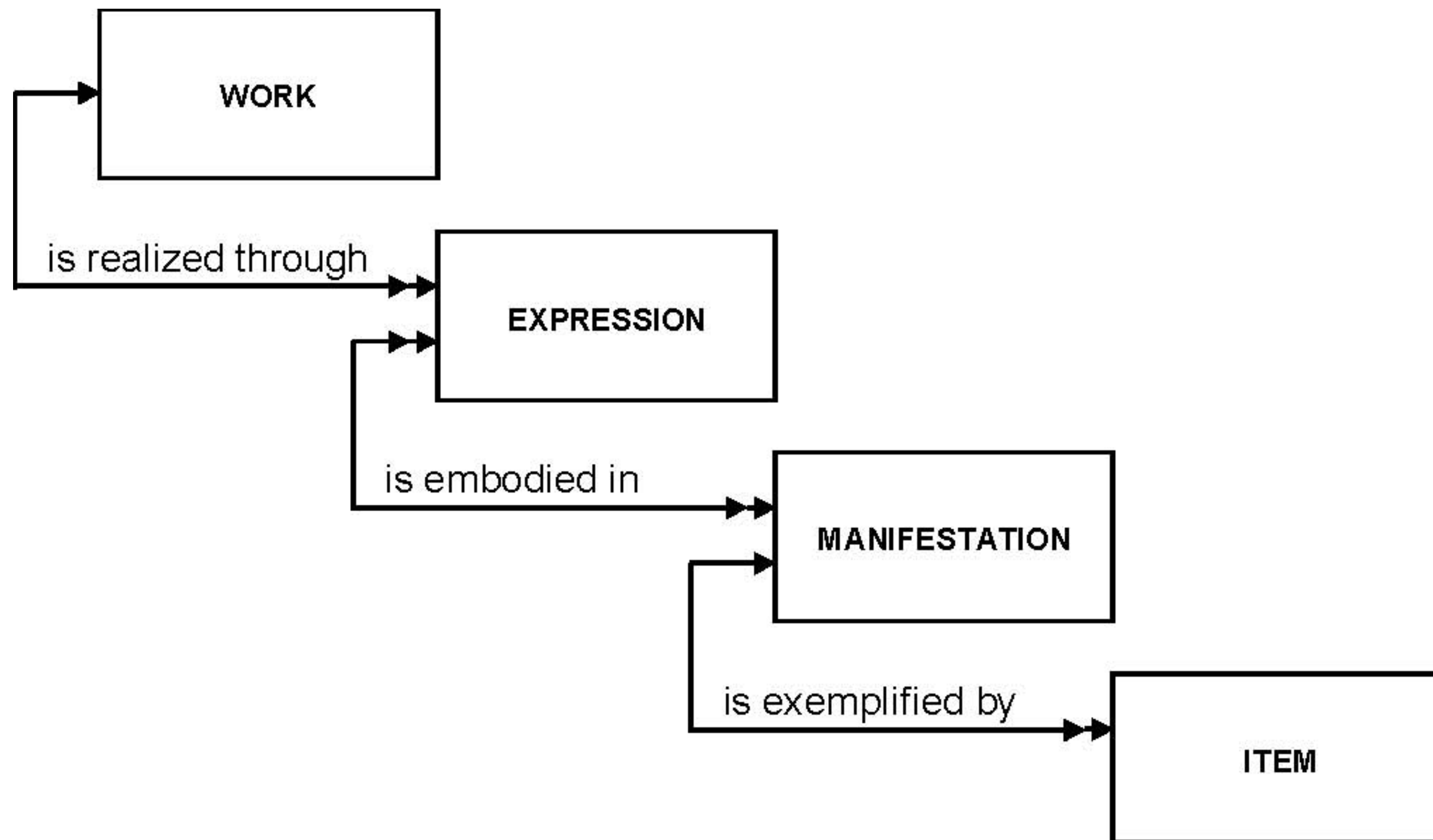- **Mediainfo XML**: Technical metadata (AV)

Dublin Core: Elements

1. Contributor
2. Coverage
3. **Creator**
4. **Date**
5. Description
6. Format
7. Identifier
8. Language
9. Publisher
10. Relation
11. Rights
12. Source
13. Subject
14. **Title**
15. Type

See: DC Reference Description
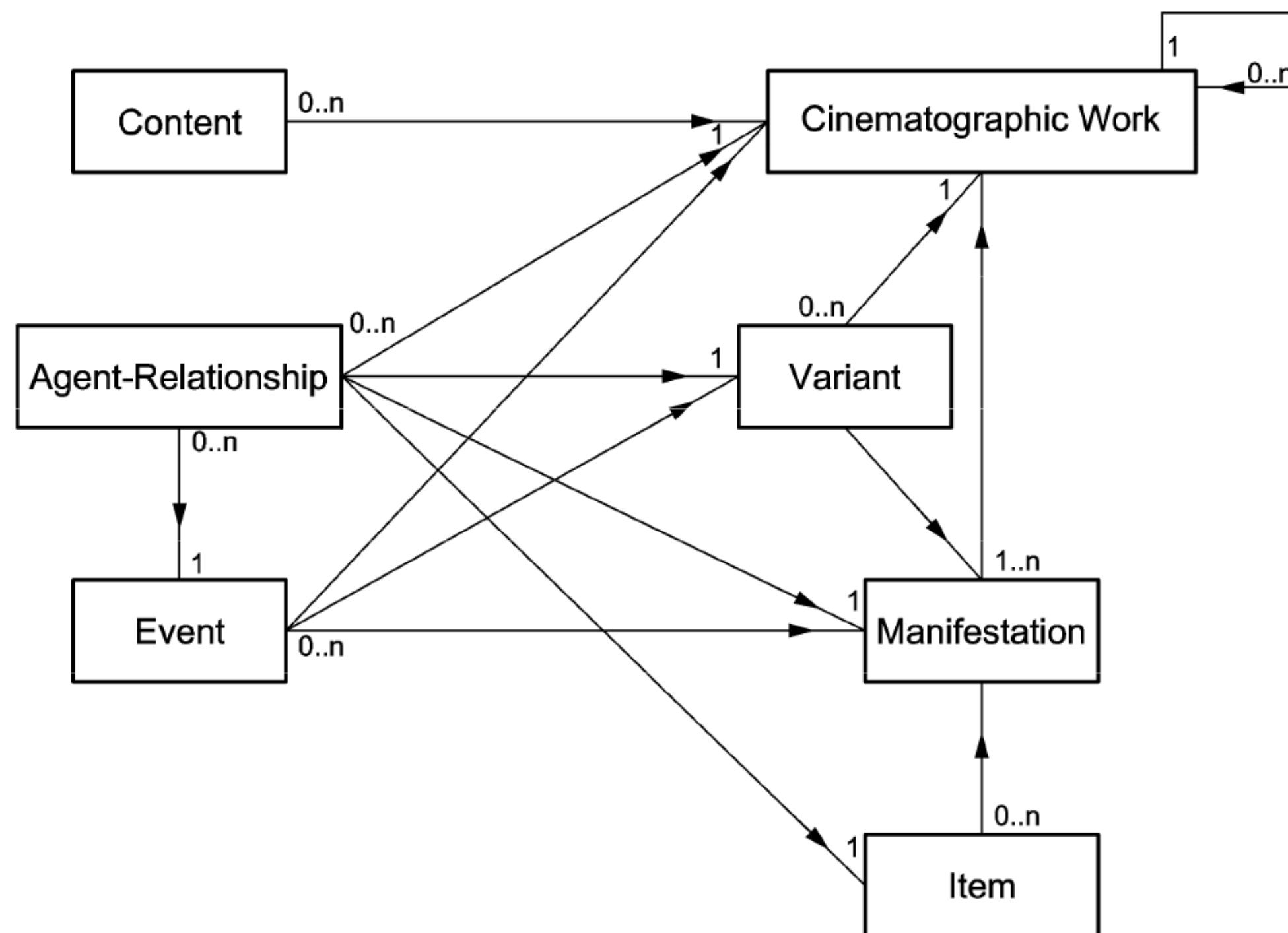
FRBR

EN 15907

Interoperability

- **Use a standard.**
  (at least as basis)
- **Use English field terms.**
  (at least in the background)

Metadata creation and capture

- When is which metadata created?
  - Ingest
  - Modifications (e.g. transcoding, rewrapping)
  - Restoration
  - Meta-Metadata:
    Who edited/updated which MD-field?
  - ...

Metadata creation and capture

- How can metadata be captured?
  - Automatic
  - Manual

Metadata: Usage & Storage

- Whatfor is metadata kept?
- How is metadata stored?

Plain Text

```
Sherlock
========================
Released in 2010 in United Kingdom
Actors: Benedict Cumberbatch, Martin Freeman
```

Plain text files are great! If you know the encoding, it's almost certain they can be opened and read by any computer/operating system/machine - even in the future.

But unstructured text makes it hard for a machine to index/find/structure things.

CSV

Comma Separated Value

```
"Title", "Release Date", "Country", "Actor 1", "Actor 2"
"Sherlock", 2010-07-25, "UK", "Benedict Cumberbatch", "Martin Freeman"
```

The CSV file

Technically, CSV is also plain text. But there's some structure defined on a meta level.

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<work version="0.1">
    <title> Sherlock </title>
    <date type="release"> 2010-07-25 </date>
    <country code="iso-3166" type="production"> gb </country>
    <actor>
        <first_name> Benedict </first_name>
        <last_name> Cumberbatch </last_name>
    </actor>
    <actor>
        <first_name> Martin </first_name>
        <last_name> Freeman </last_name>
    </actor>
</work>
```

The XML file

DOC / PDF?

- Pros?
- Cons?

Embedded vs Sidecar

- Embedded =
  inside the media file

- Sidecar =
  additional files **next** to media file

Both have their advantages and disadvantages.

Embedded is more comfortable for files that are often moved around (less likely to get lost), but harder to change/augment/update/correct, because it requires applications to support read/write of these metadatas. This is not always the case.

For media objects, this often requires the whole file to be rewritten, regardless how trivial the metadata change was.

Furthermore: Any change to the media file changes their fixity information (hashcode, timestamp, etc), which must be dealt with accordingly. Consider runtimes of: read/write and hashcode generation!

Sidecar must be taken care of separately, but can more easily be changed/augmented/updated/corrected (because it's often text-based: CSV, XML, TXT, etc). May follow separate format specifications than the media object. This allows separate format migrations of the sidecar metadata.

Also: The sidecar data is usually significantly smaller than the media object (especially for AV): So if fixity information needs to be updated the large files may stay in place/untouched.

Again: Keep it as simple as possible and as complicated as necessary.

Good practice

- Use controlled vocabulary
- ...from public/common sources
- Prefer structured text (CSV or XML)
- ...over DOC(X) and PDF
- Embed at least basic info to ID the file
- ...and select which data may go better as sidecar
- Machine *and* human readable
- Field names ("backstage"): English, please.

Questions?

Comments?

Example XMLs

- Mediainfo
- METS
- PREMIS
- EBUCore

Links

- ISO 8601(Wikipedia)
- Date and time format - ISO 8601(ISO 2019)