A METS profile for kb.dk?

An attempt to an outline

Why such a beast?

We need a common framework, which should fulfill several functions

- Submission Information Package (**SIP**) for digital preservation
- Dissemination Information Package (**DIP**) for use on the web

DIP refer to what is retrieved from an archive, in the following I will use it in a more general sense, as a package of digital objects, data and metadata useful for services directed towards users.

A SIP is basically the same thing, but is directed towards future users.

METS overview

- Seven sections
- The green ones the most important ones for dissemination on the web
- The red is for preservation
- dmdSec contains ordinary metadata
- structMap and structLink describes the resource's internal relationships
- METS can be used to describe relationships between METS objects

```
<mets>
   <metsHdr/>
   <dmdSec/>
   <amdSec/>
   <fileSec/>
   <structMap/>
   <structLink/>
   <behaviorSec/>
</mets>
```

Modeling examples

Three examples (use cases) ordered from simple to complicated

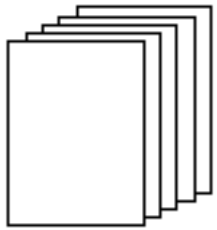
1. A letter
2. A sequence of aerial photographs
3. A photo album

Structure explained as maximum examples, to show the `<structMap>` idea

A sequence of aerial photographs

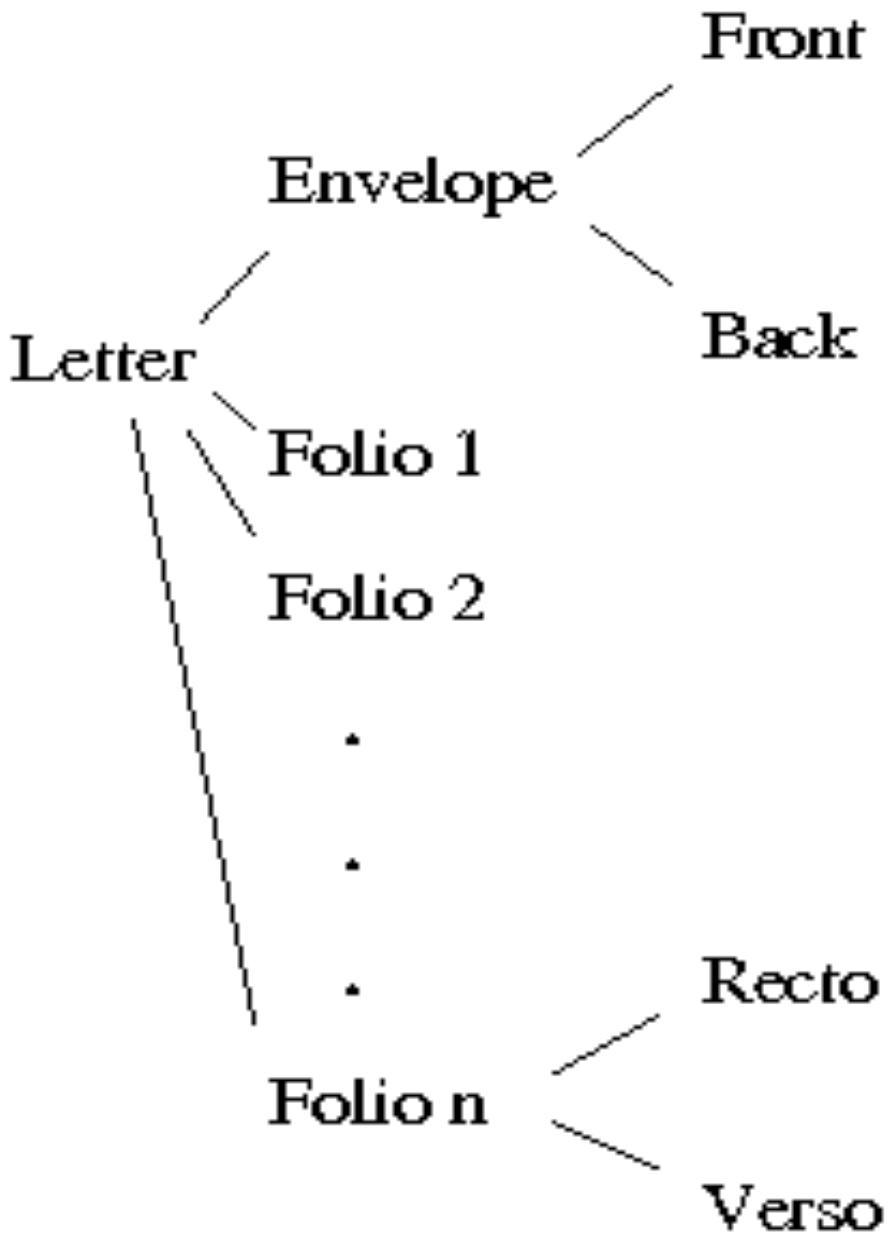
Positions (Dublin core <coverage> in <dmdSec>)

Images (links in <fileSec>)

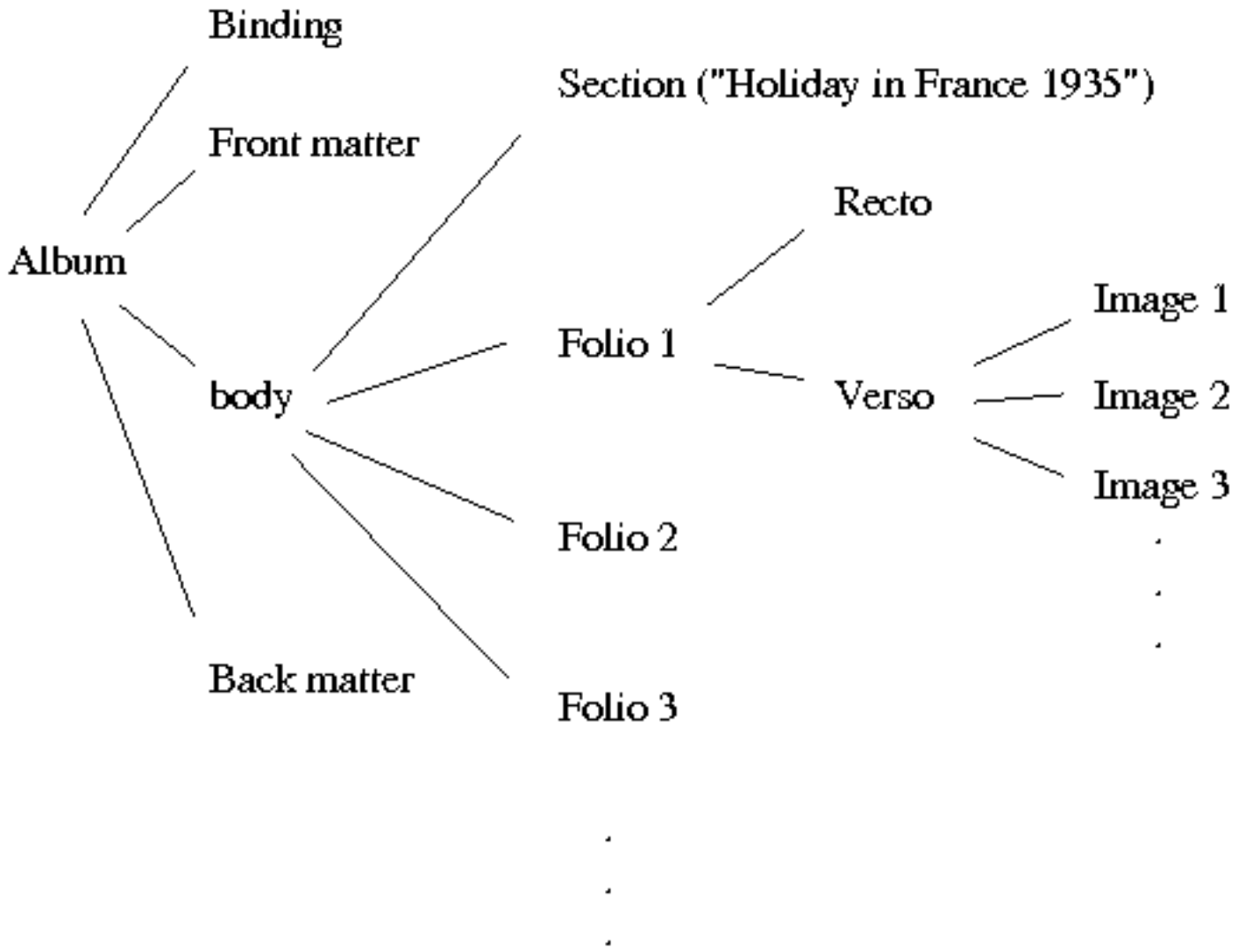


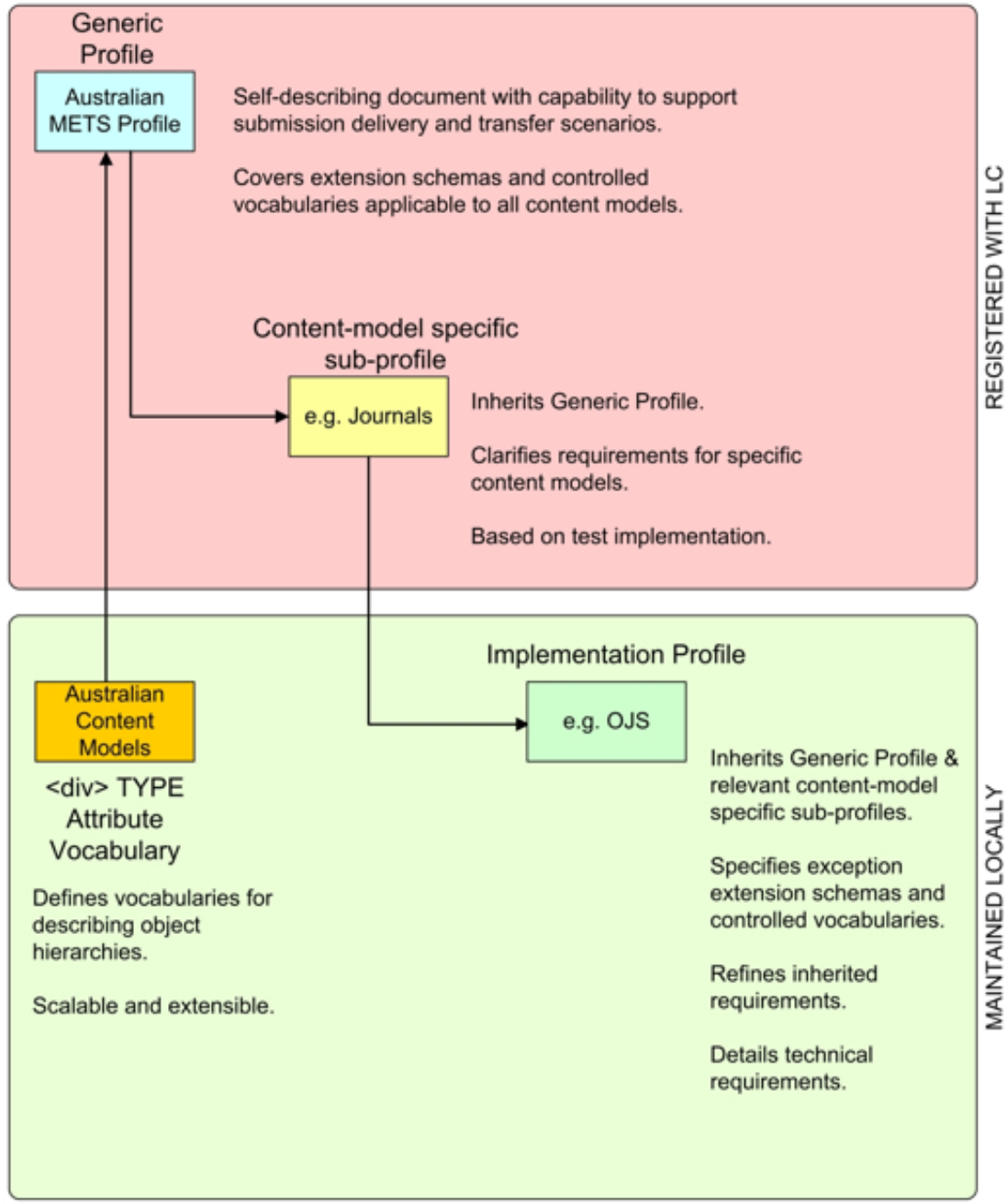
```
<structMap>
  <div>
    <div>
    <div>
    <div>
  </div>
</structMap>
```

A letter


Front
Envelope
Back
Letter
Folio 1
Folio 2
Recto
Folio n
Verso

A photo album





http://www.dlib.org/dlib/march08/pearce/03pearce.html

Guidelines

- If distributing the PREMIS metadata among METS sections, use <techMD> for the PREMIS Object entity, <digiProvMD> for Event, <rightsMD> for Rights, and either <digiProvMD> or <rightsMD> for the Agent entity, depending upon which is applicable. If distributed, the PREMIS container is not used.

- PREMIS metadata may be kept together, in which case it is encoded under <digiProvMD> with the PREMIS container.

Guidelines (cont'd)

- An application may decide whether it is easier to include the information redundantly, based on how the data will be used and/or supplied. Implementers should consider the use of the data (e.g., display or preservation) and whether the PREMIS or the METS metadata is primary when deciding which to use and whether to record it redundantly. If preservation is the goal, PREMIS elements should be used regardless of redundancy.

Guidelines (cont'd)

- The METS <structMap> is more expressive for structural relationships and should be used for structural metadata. If the scope of exchanging the object is preservation, the information should also be included in PREMIS relationship elements.
- Linking should use METS ID/IDRef constructs, particularly if the PREMIS metadata is wrapped in the METS document, but the PREMIS linking elements should also be used in case the PREMIS metadata is separated.
- Implementers should document decisions, especially those concerning redundancy of elements, in a profile.

http://www.dlib.org/dlib/july08/guenther/07guenther.html