

<?xml version="1.0" encoding="UTF-8" ?>

<mets:mdWrap MDTYPE="MODS">

<mods:title>METS is the answer (What was the question?) </mods:title>

<mods:placeTerm type="text">DPC Awareness Day, London</mods:placeTerm>

<keyDate="yes">April, 2013</mods:dateIssued>

<mods:note>Dave Thompson</mods:note>

<mods:note>Wellcome Library</mods:note>

</mets:mets>

Jub Jub, he's everywhere you want to be!



The Simpsons, "A Fish Called Selma", 19th episode, 7th season, originally aired March 24, 1996

wellcometrust

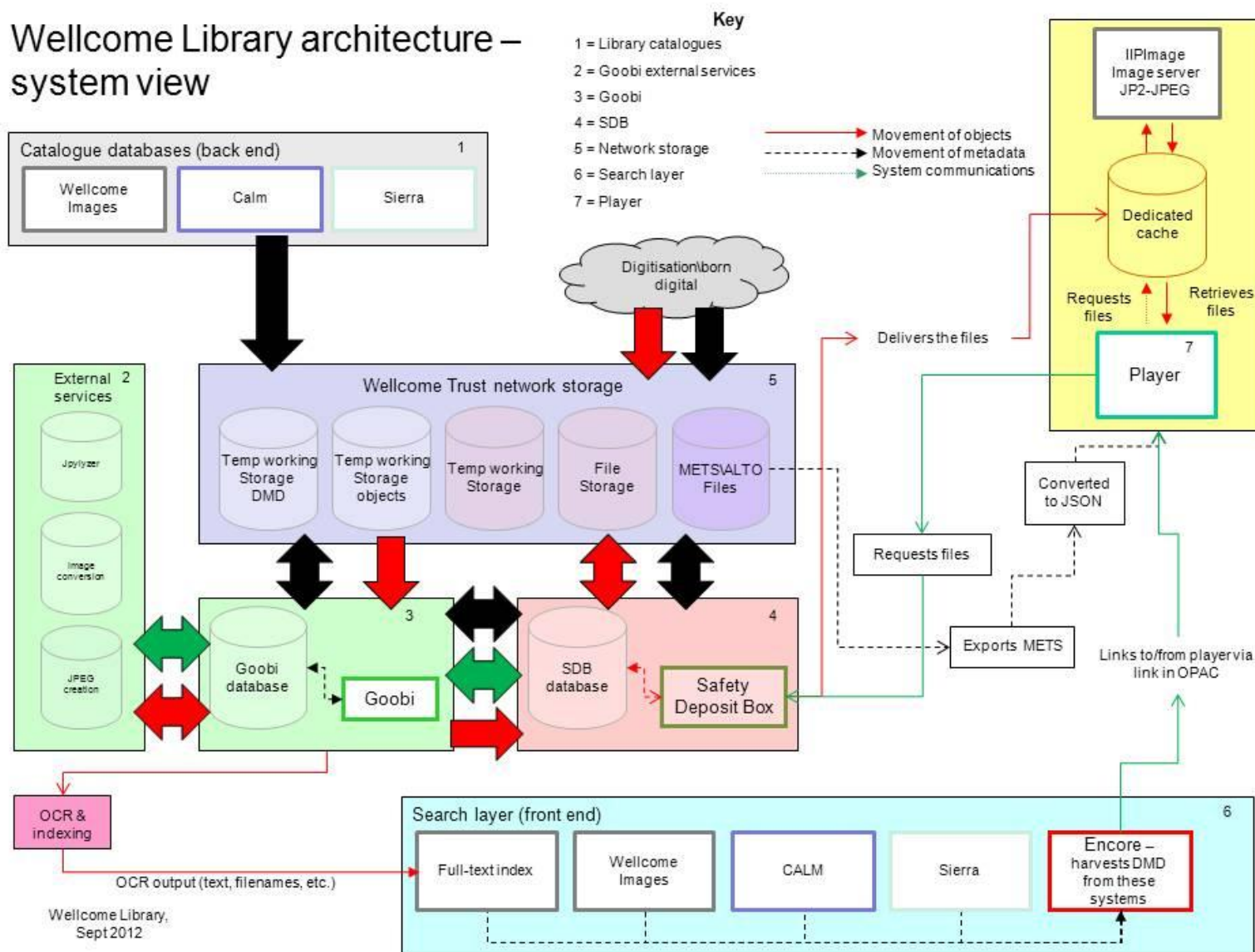
What's wrong with METS?

- Three institutions use METS, you'll get 5 different implementations.
- METS is a loose standard, difficult to identify best practice as implementations can vary.
- Requires clarity of thinking if METS to be used well & integrated with other systems.
- METS is so not designed for normal humans.

So...?

- METS is not an end in itself. Can't stand alone.
- Has to be part of a larger system, it is not the system.
- Tool is required to create METS, XML not eye friendly!
- Use of METS requires other systems to understand METS & be able to act on the metadata it contains.

Wellcome Library architecture – system view



So how does Wellcome use METS?

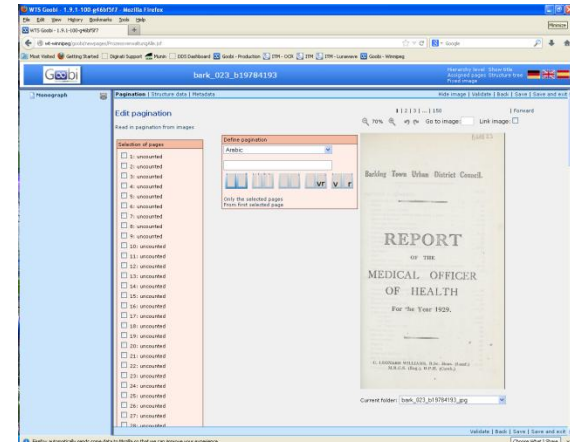
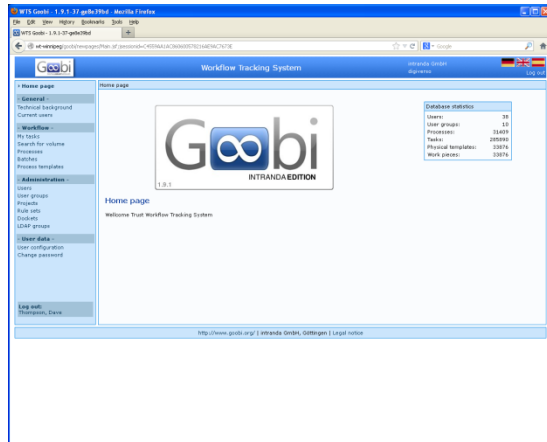
- METS is used to manage access to content.
- Amalgamates metadata from different systems.
- Carries that metadata between systems.
- Provides access to content via the front end player.

Remember digital preservation?

- Series of small scale, on-going interventions.
- No king hit. No one time deal.
- Life cycle management of every activity.
- Manageing access is part of digital preservation.
- METS is part of that process of on-going interventions.

Goobi is our METS tool

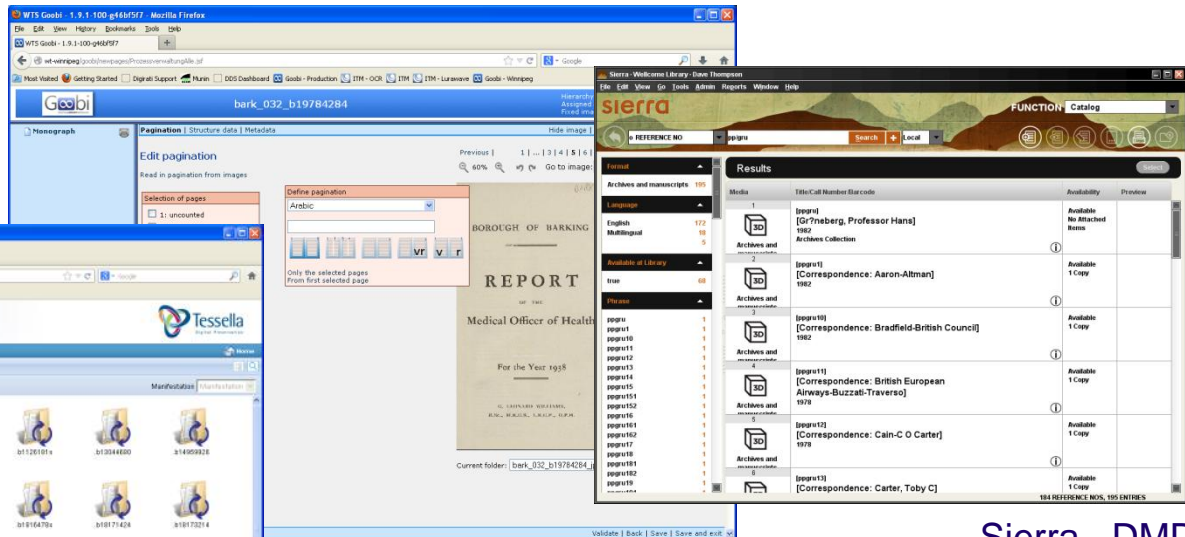
- Graphical interface to METS editor, no pointy bracket stuff!
- Easy to use, efficient interface.
- Mix of user input & automated metadata import.
- Automatic generation of METS files.



Data from existing Library systems

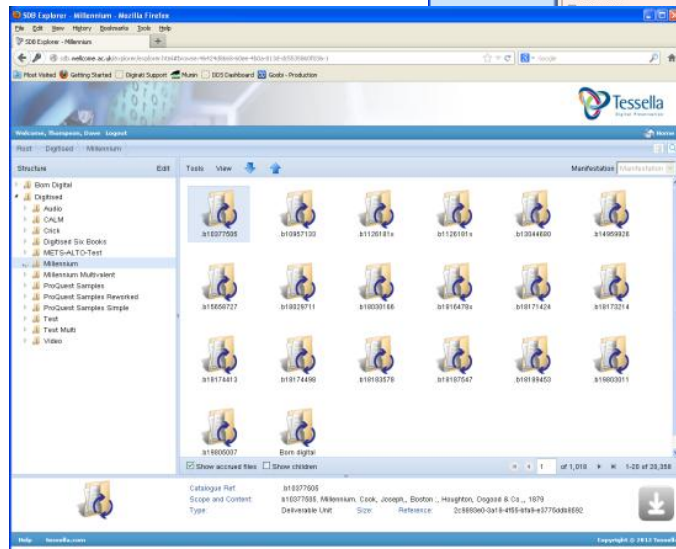
METS needs to contain/combine metadata from three Library systems.

Goobi – METS structure



Sierra - DMD

SDB – content & AMD



Administrative metadata

On ingest of content into our repository (SDB) AMD is exported to Goobi. This is the basis for access to that content. METS includes;

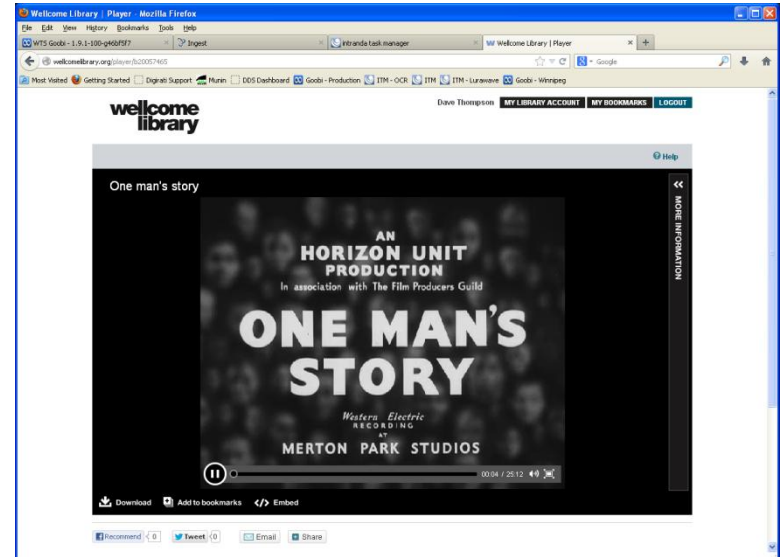
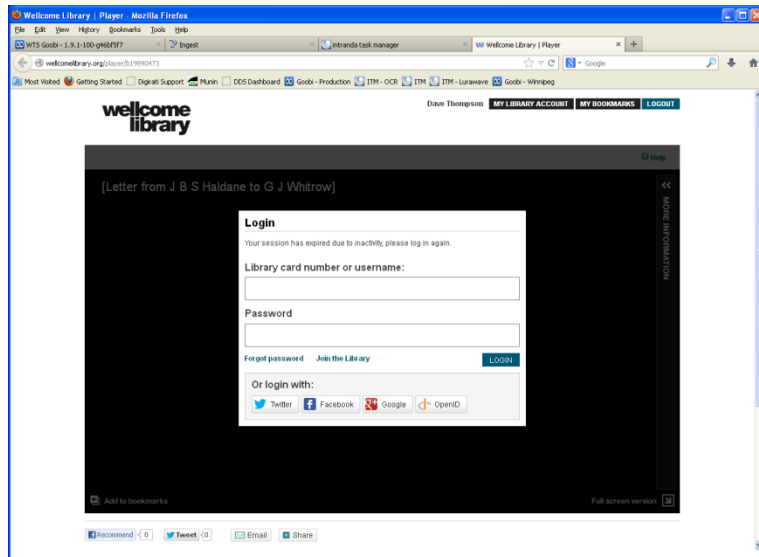
- Unique repository identifier for both object, e.g. book, & for individual files. Used by player to retrieve content.
- Image dimensions (HxW) for individual images. Used by player to display images.
- Checksum, may use for born digital.

Working with DMD

- Convert MARC records from Sierra to MARC XML for upload into Goobi.
- DMD in METS not delivered to end user, DMD displayed in player is from live catalogue.
- No issues about synchronising DMD between catalogue & METS.
- DMD in Goobi/SDB allows humans to identify material for snagging.

Neat tricks with METS

- Set material type & default values for access based on DMD.
- Goobi METS editor allows defaults to be applied but to be easily overridden.
- Player presents content in appropriate context.



Access – defining material type access

We broadly define the material ‘type’ using the MARC leader 6 field, text, archive, etc. This allows us to set default access conditions for a ‘project’ in Goobi.

```
<mods:note type="wellcomeidentifier">PP/CRI/J/1/5/18/1:Box 140</mods:note>  
<mods:note type="leader6">t</mods:note>
```

- Efficient & consistent process.
- Saves time & mouse clicks.
- Can be manually over ridden on an ad hoc basis.

Access – defining terms access

We define access conditions in the METS; 'open', 'closed', 'requires registration', etc. This is used by the front end player to manage access to content – or not – for instance by making a login box pop up when 'requires registration' material is requested.

```
<mods:accessCondition type="dz">J</mods:accessCondition>  
<mods:accessCondition type="status">Requires  
registration</mods:accessCondition>
```

- Efficient & consistent process as set at the 'project' level.
- Saves time & mouse clicks.
- Enactment of the policy is done by the front end player.

However....

- Our METS potentially contains details of how to retrieve content that is 'closed' or 'restricted'.
- METS could be reverse engineered to retrieve that content. In theory.
- Front end player designed to prevent this happening.
- Don't make our METS publicly available.

So, what was the question...?

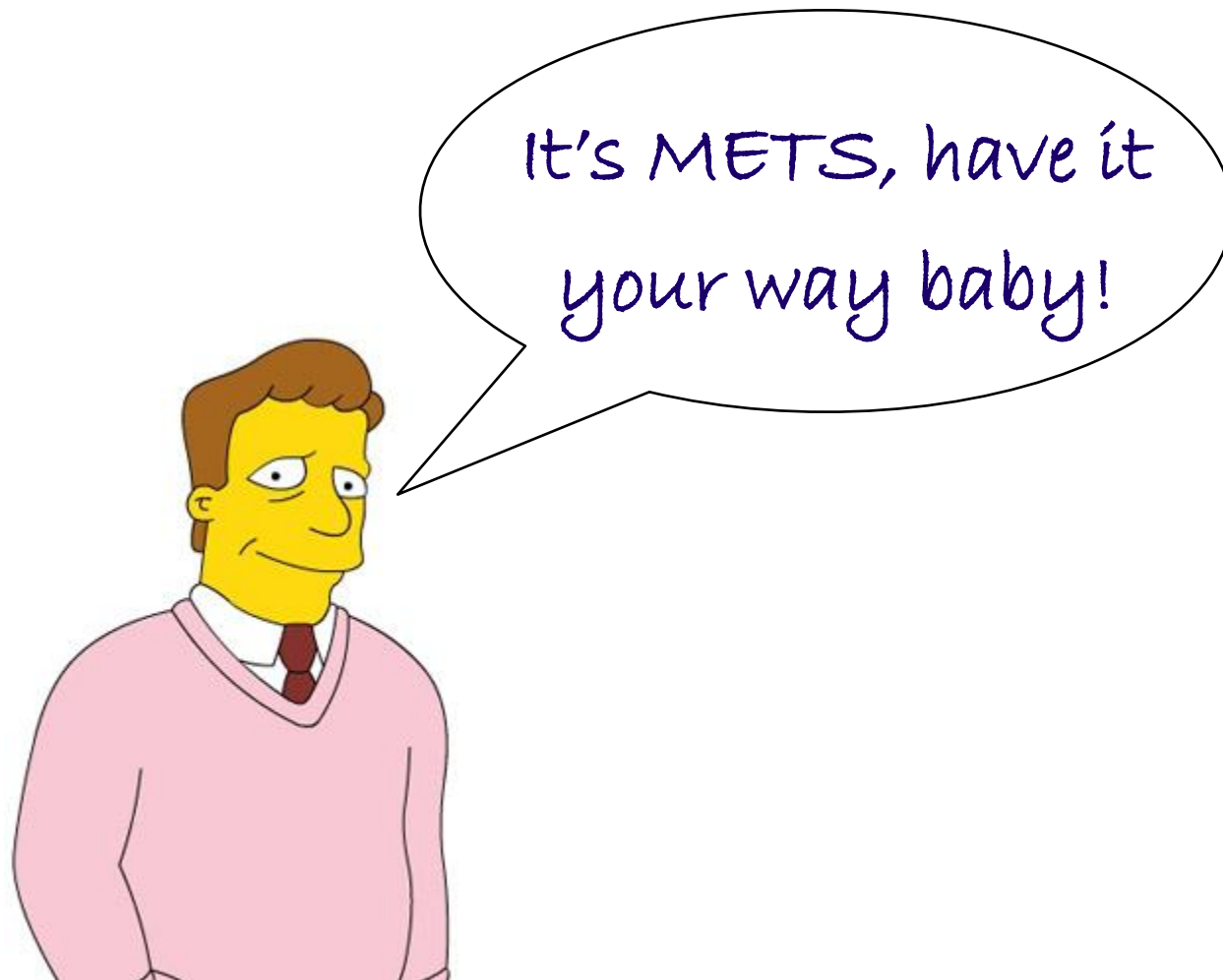
- Diverse sources (Different systems) of metadata.
- Different types of metadata, AMD, DMD.
- Different purposes for metadata, retrieval, access.
- Different systems at work, Sierra, Goobi, SDB.
- Different formats, though mostly XML.



From chaos comes order, beautiful order

- This is exactly what METS is designed to handle.
- A single portable framework able to combine different types metadata from disparate sources.
- Without METS individual systems would have to communicate individual pieces of metadata separately.
- METS allows us to fully manage access to content by consolidating metadata relating to access.

Of course, you
may think
differently...



<?xml version="1.0" encoding="UTF-8" ?>

Thank you

Questions now, questions later...?

Dave Thompson, Digital Curator
Wellcome Library

d.thompson@wellcome.ac.uk

<http://wellcomelibrary.org/>

</mets:mets>