MAX PLANCK
digital library

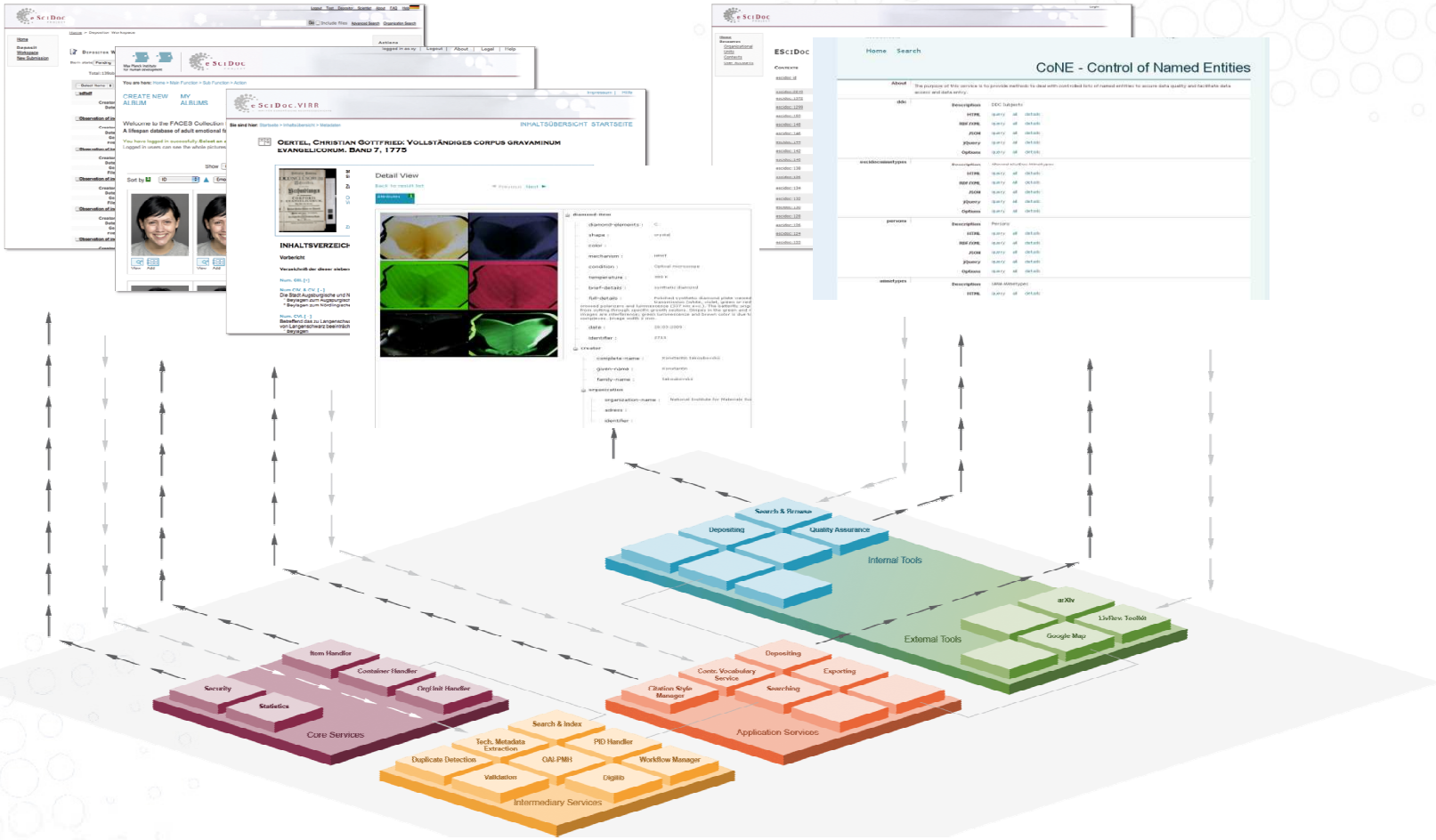
Persistent identification of resources in eSciDoc service-oriented infrastructure

Natasa Bulatovic
Research and Development
Max Planck Digital Library

Research and Development Activities

- Development, operation and maintenance
  - Solutions (PubMan, FACES, VIRR, WALS, Admin)
  - Services (Service Oriented Architecture - SOA)
- IT Infrastructure services
  - Development environment
  - Communication and collaboration platform
  - Wordpress MU platform
- eSciDoc project 2004-2009 (closed)
  - FIZ Karlsruhe and Max Planck Digital Library
  - Development is continued

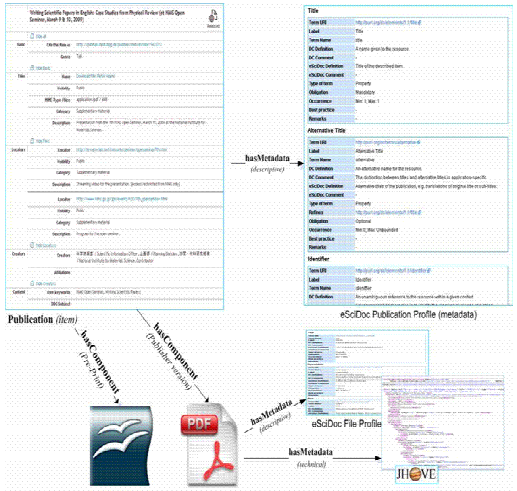
eSciDoc SOA Landscape



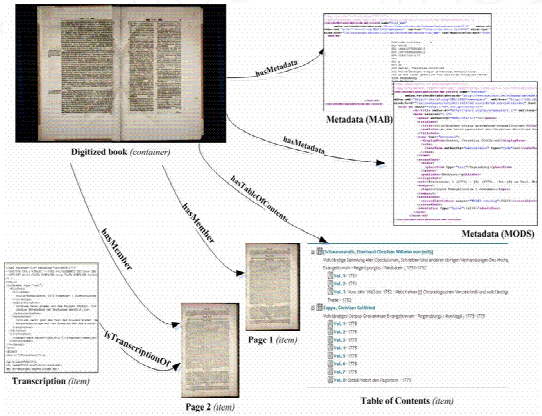
Diversity of MPG supported by a service infrastructure

- People
  - Librarians, researchers, developers, general public
- Data
  - Publications, research data, supplementary material etc.
- Processes
  - Various institutes apply various workflows supporting their data management

Which data are managed (content resources)?

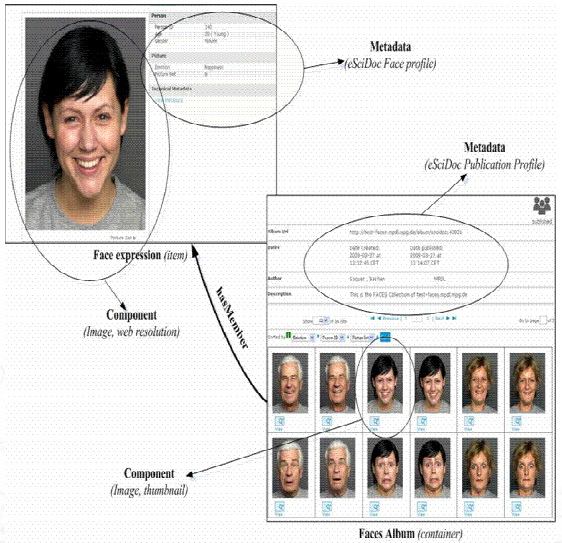


ContentModel	Publication Item
ObjectType	Item
Metadata	eSciDoc Publication Profile
Components	allowed
Mime	any mime type allowed
Content-category	any-fulltext, publisher-version, pre-print, post-print, supplementery-material, abstract, table-of-contents
Components metadata	File profile, technical metadata (JHove)
Storage	internally-managed, externally-references



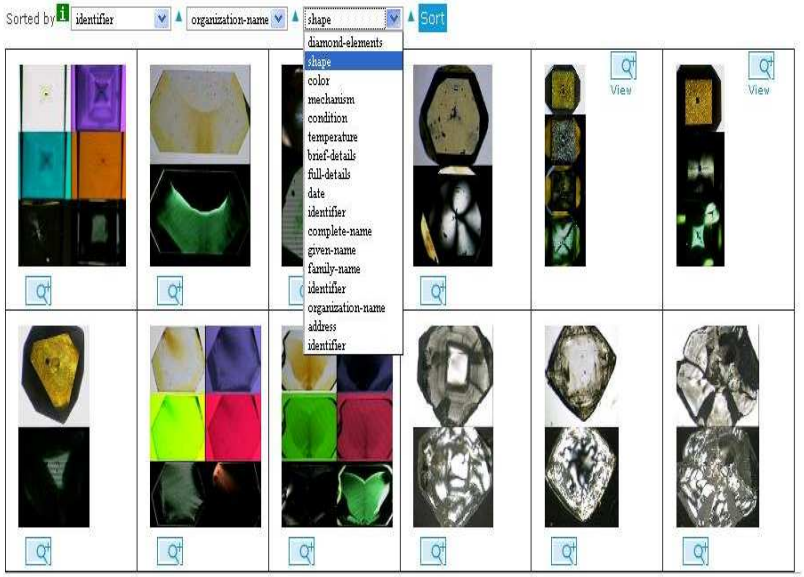
ContentModel	DigitizedBook
ObjectType	Container
Metadata	MODS profile
Components	Not allowed
Members	allowed
Member-types	Digitized pages, TOC

ContentModel	DigitizedPage
ObjectType	Item
Metadata	DC
Components	allowed
Mime	TIFF, GIF
Content-category	Original-size, web-resolution, thumbnail
Components metadata	File profile, technical metadata (JHove)
Storage	internally-managed

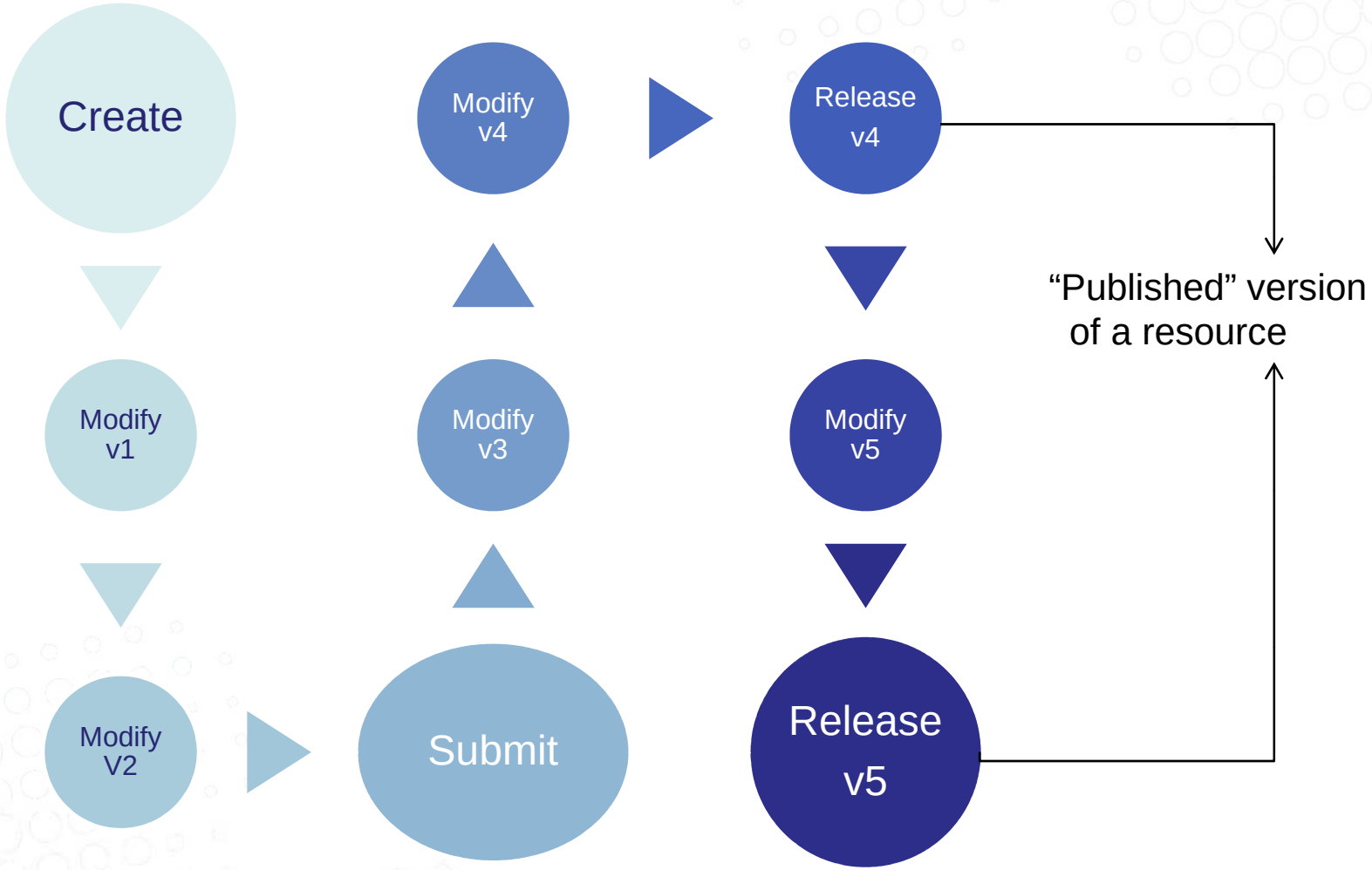


ContentModel	FaceImage
ObjectType	Item
Metadata	DC
Components	allowed
Mime	GIF
Content-category	Original-size, web-resolution, thumbnail
Components metadata	File profile, technical metadata (JHove)
Storage	internally-managed

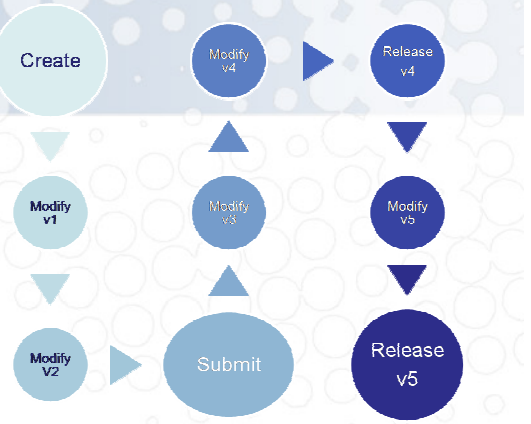
ContentModel	FaceAlbum
ObjectType	Container
Metadata	eSciDoc publication profile
Components	Not allowed
Members	allowed
Member-types	FaceImage



Which workflows are supported?

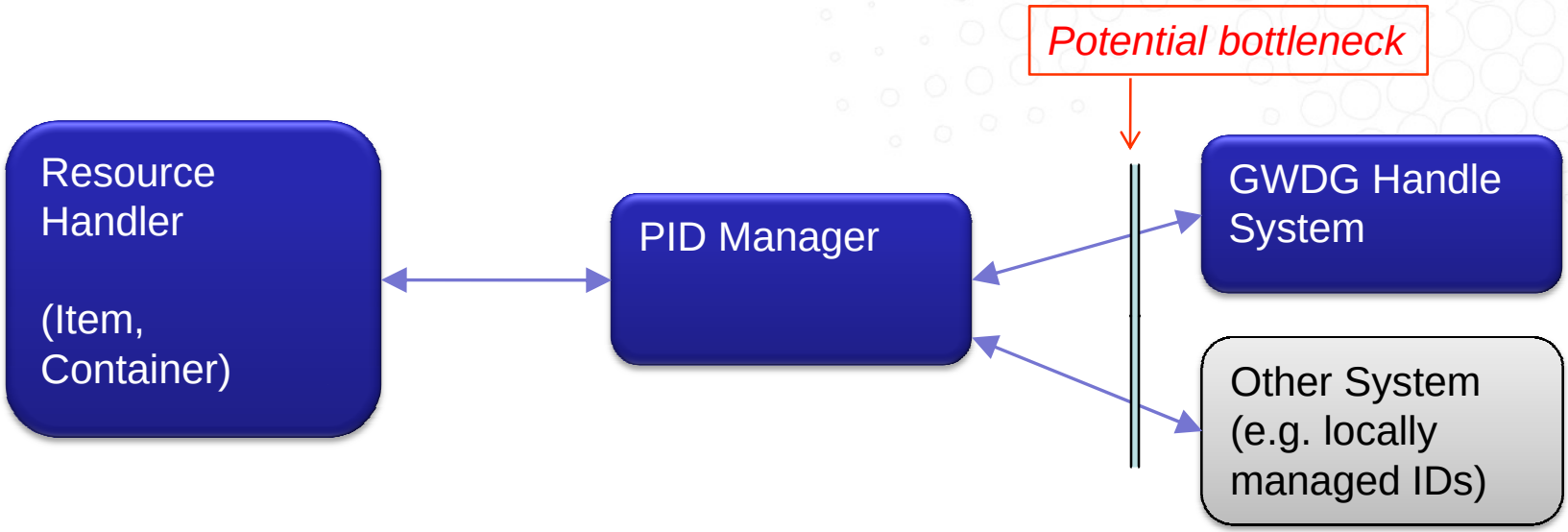


To what and when a PID is assigned?



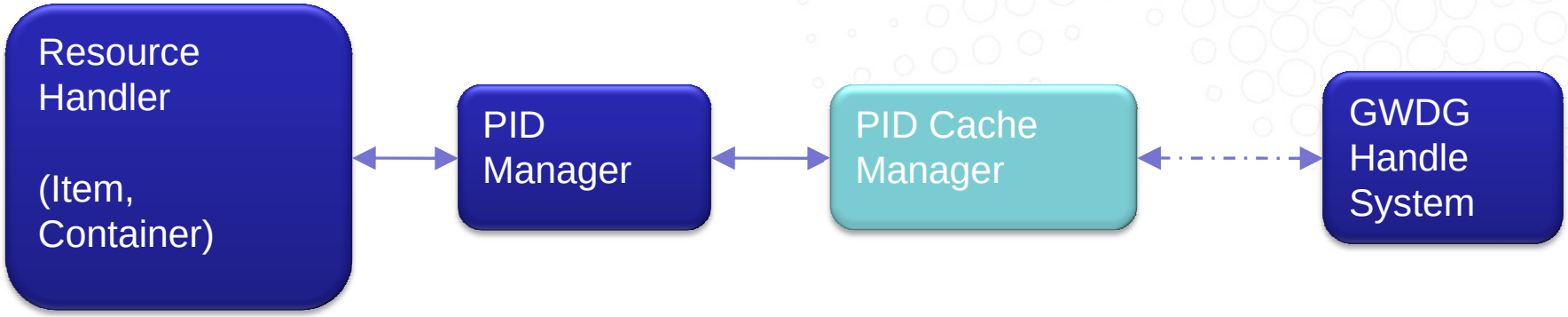
- A PID can be assigned to *each content resource*
- A PID can be assigned to *each version of the content resource*
- *Example: A publication* 1 full text file, 3 supplementary material files associated
  - has minimum 5 PIDs assigned (each version gets a new PID)
- A PID can be *assigned in each stage of the workflow* to the object as a whole, or to the object version

eSciDoc PID Manager



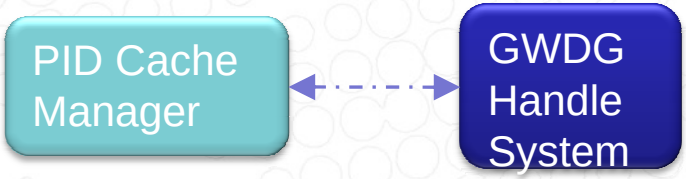
- Single Operation success (e.g. creation, modification, submit or release) depends on PID system (GWDG Handle system)
- Batch operations issue as well e.g. import or batch release requires several thousand PIDs (handles) to be created at once
- End users may be unable to finish their work

Extension with PID Cache Manager



- PID Cache Manager holds a user-defined threshold of pre-assigned handles (incoming buffer)
- Pre-assigned Handles have dummy resolution
- If GWDG Handle System is not available a handle is taken from the threshold and moved to an outgoing buffer
- An asynchronous process updates the resolution in the GWDG Handle system (in a user-defined frequency)

The PID Cache Manager



- Is independent service, can be used by anybody
- Technology: PID Cache Manager (Java), HSQLDB, JBoss
- Pre-assigned handles have dummy resolution
- If GWDG Handle System is not available a Handle is taken from the threshold and moved to an outgoing buffer
- An asynchronous process updates the resolution in the GWDG Handle system (in a user-defined frequency)

See also: http://colab.mpdl.mpg.de/mediawiki/Pid_cache

Requirements for the GWDG Handle system

- Interfaces
  - Stable
  - Clear specification of methods
  - The XML schema for create, update of handles is needed
  - REST interfaces: GET, POST, PUT, DELETE (Retrieve, Create, Update, Delete)
  - Backward compatible in case of interface changes
- Stable operation
- Support and documentation
- Community to discuss special services

Proposal: some extensions

- Allow to identify same content with multiple resolutions
  - Managed by single PID user
  - Example: Publication item with internal repository identifier escidoc:1234, has following representations: PubMan Item , Sengbush Wordpress blog post entry, Core-service style sheet

    http://pubman.mpdl.mpg.de/pubman/faces/viewItemFullPage.jsp?itemId=escidoc:39182:9

    http://sengbusch.blogs.mpdl.mpg.de/?s=escidoc%3A39182

    http://coreservice.mpdl.mpg.de/ir/item/escidoc:39182:9
- URL (resolution) validation
  - Active or passive
- LinkedData publishing

Extension example: Inter project: archiving LAMUS data into eSciDoc repository

- Resources from LAMUS are additionally archived in the eSciDoc repository
- PIDs managed by different PID users
  - Corpus -> Container (or Item, tbd)
  - Session -> Item
  - Session Resources->Item.components
- Do we re-assign a handles in eSciDoc repository or resources are getting a new handle
- What about resolutions (output from discussions with Daan Broeder, see next slide)?

Alternatives for multiple resolutions

- Alternative 1
  - PID is associated with two URLs: url_a and url_b,
  - We need to indicate which administrator/user should be able to modify each url record.
  - This could be done by adding an extra record specifying which HS_ADMIN can modify the URL information in record number "n“.
  - If no such extra record is present the owner of the PID is the only one that can administrate that URL. Of course this mechanism should be enforced by the PID service implementation
  - extra programming required
- Alternative 2
  - separate pid for each representation – each created by another owner
  - ”same-as” (linked data) relation provided by the owner of the PID to point to the other PID
  - Possible incosistencies on both copies? (are these indeed copies?)
  - extra programming required

Both alternatives have valid related use case at MPDL eSciDoc repository

Questions?

- bulatovic@mpdl.mpg.de