



# Adoption Story: DFT Model in Fedora Commons

Thomas Zastrow, Peter Wittenburg

research data sharing without barriers  
[rd-alliance.org](http://rd-alliance.org)

- Goal & DFT Model
- Overview: Fedora Commons
- Containers & Binaries
- PID Assignment
- Example: Dataset of Maps
- Resolving

- The Data Foundation and Terminology WG task is to describe a **basic, abstract data organization model** which can be used to derive a reference data terminology that can be used across communities and stakeholders to better synchronize conceptualization, to enable better understanding within and between communities and finally to stimulate tool building, such as for data services, supportive of the basic model's use.

<https://rd-alliance.org/groups/data-foundation-and-terminology-wg.html>

## Definition:

A digital object (DO) is represented by a **bitstream**, is referenced and identified by a **persistent identifier** and has properties that are described by **metadata**.

RDA Data Foundation and Terminology - DFT: Results RFC

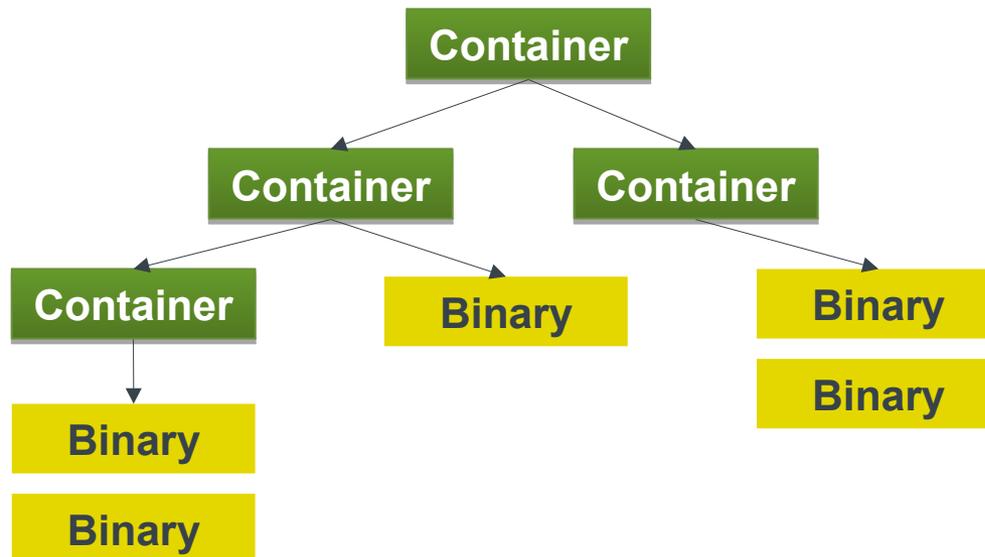
[dx.doi.org/10.15497/06825049-8CA4-40BD-BCAF-DE9F0EA2FADF](https://dx.doi.org/10.15497/06825049-8CA4-40BD-BCAF-DE9F0EA2FADF)

- Is DFT model just on paper?
- Can we implement it?
- Can it be used out of the box?
- Can we help people who need to setup a proper repository?

**So let's try to implement it in Fedora Commons.**

- Fedora Commons (FC) is a generic data repository software
  - Useful for middlesized datasets, but probably not for big data
- FC comes along with basic functionality common for modern data repositories:
  - User management
  - Metadata handling
  - APIs for automatic access
  - ...
- FC can be used as backend for individual and discipline specific data repository implementations

- A FC based data repository can consist of two elements:
  - Containers
  - Binaries
- Containers are used to group binary elements or other containers. With nested containers, a folder-like hierarchy can be build up
- Binaries are representing uploaded files / **bitstreams**
- Both containers and binaries can be assigned with **metadata**

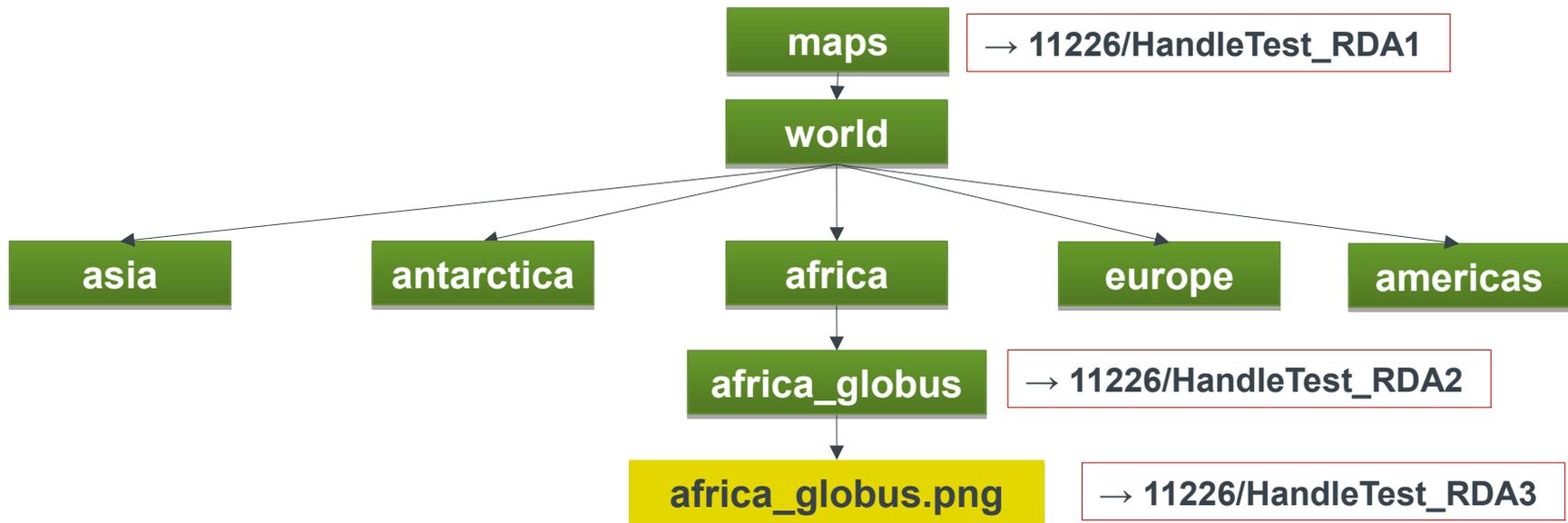


- In FC version 4.x, **PIDs** are part of the metadata of an FC object
- Assigning PIDs to FC objects has to be done by the user
- PIDs can be assigned to containers and / or binaries

- Assigning a PID to a **binary**
- Pros:
  - Checksums can be calculated and can be used for validity checks etc.
- Cons:
  - As a representation of a single file, the binary itself is often not a semantic meaningful unit – it has to be seen in its context, „standalone“ usage makes no sense
  - Scalability: large datasets can contain a lot of binaries which can lead to performance issues when assigning, maintaining and resolving PIDs

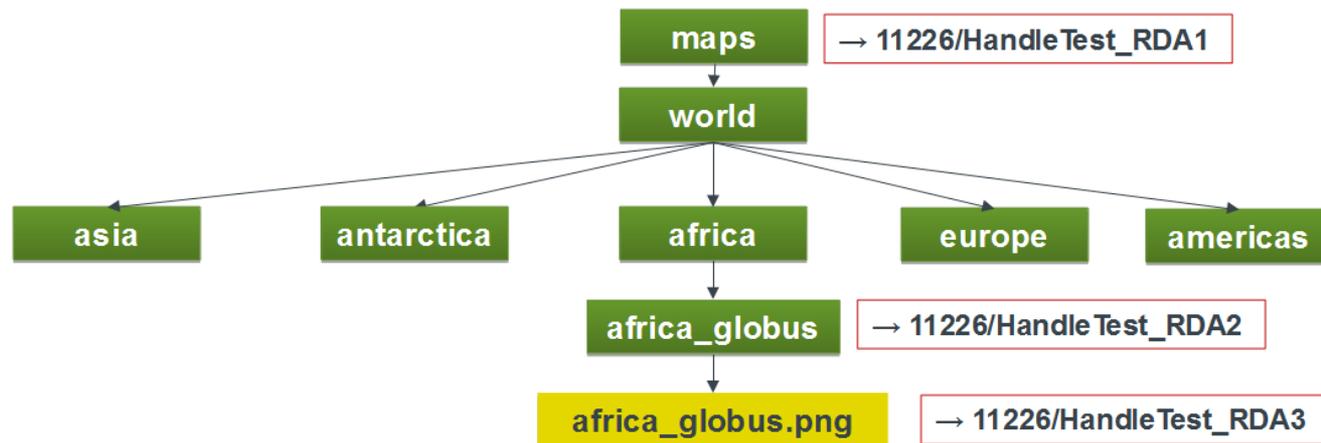
- Assigning a PID to a **container**
- Pros:
  - Discipline specific, semantic meaningful unit
- Cons:
  - Changes inside a container can not be tracked by PIDs

# Example: Dataset of Maps



## Example: Dataset of Maps

- In the example, 3 PIDs were assigned (two to containers, one to a binary)
- It would be possible to assign PIDs also to any other FC object: for example, the „continent“ containers could be also adressable via PIDs



- The assigned PIDs can be resolved from any Handle resolver:

**Handle.Net®**

**Resolve a Handle and View the Values**

The web form below will enable you to resolve individual handles and view their associated values. It uses a proxy server which understands both the Handle protocol and HTTP protocol.

If you type a handle into the text box, and that handle has a URL associated with it as one of its values, the proxy server will instruct your browser to display the location of that URL. If you select "Don't Redirect to URL's", the proxy will display the handle record.

The Handle proxy server uses caching to speed handle resolution. If you check "Authenticate Query", the proxy will bypass the cache, go directly to the responsible handle server, and then return the cache with the data for that handle.

Simply appending a handle to the URL <http://hdl.handle.net> and giving the string as a browser as a location will also resolve that handle. The proxy also supports a REST API, returning handle records in JSON format. Further documentation is available [here](#).

Handle:

Authenticate Query  
 Don't Redirect to URL's  
 Don't Follow Aliases

Handle Proxy Server Documentation  
[HandleNet.Pdx.Us](#)

[hdl.handle.net](http://hdl.handle.net)

The screenshot shows the DOI System website with a navigation bar and a search section titled "Resolve a DOI Name". The search form includes a text input field and a "GO" button. Below the form, there is a "Clicking a DOI link" section with a link to "http://dx.doi.org/10.1002/anie.201500000" and a "DOI.ORG® in the News" section with a link to "2015 DOI Staff Changes Announced".

[www.doi.org](http://www.doi.org)

1. The DFT model can be implemented easily in Fedora Commons
2. Fedora Commons can be downloaded in several forms from the Fedora Commons webpage or activated as Docker container:

```
docker run -p 1111:8080 -t -i thomazastrow/fedora-commons /bin/bash
```

It should download the image automatically from docker hub and run the image - after a few seconds, point your local browser to:

**<http://localhost:1111>**

# Thank you.

Thomas Zastrow  
thomas.zastrow@mpcdf.mpg.de

Peter Wittenburg  
Peter.Wittenburg@mpi.nl