An OAI-PMH Endpoint for Fedora 6.x

CLARIN Centre Meeting 2022
Felix Helfer – helper@saw-leipzig.de
Fedora News

- In Leipzig: **Fedora 6.1.1** currently used in production.

- **Fedora 6.2.0** release soon.

- Arran Griffith replaced David Wilcox as program manager at Lyrasis.

- One of the project’s core contributors, Peter Winckles, left in April.
OAI-PMH

• Essential for interfacing with the CLARIN world.

• Still not officially supported in Fedora 6.

• Some verbal reception from Lyrasis in the past, but nothing concrete yet.
OAI-PMH Implementation for Fedora 6

- **Therefore**: custom implementation, by Nathanael Philipp.
- Based on his Django OAI-PMH app [1] (licensed under GNU GPL-3.0).
- Mostly extending the app’s import to interface with the Fedora 6 REST-API.

[1] https://github.com/jnphilipp/django_oai_pmh
Implementation Details

- **Django** application, with **Nginx** on top.

- **Dockerized**, quick setup and reset with `docker-compose`.

- Uses a **PostgreSQL** database in production mode per default (configurable).
Implementation Details

- Harvester starts at root, recursively checks all resources for relevant predicates:
  - `oai:identifier`
  - `oai:memberOf`
  - `example:oaiCMDIRecord`
  - `example:oaiDCMIRecord`
  - `oai:setName`
  - `oai:setSpec`

- Configurable predicates for the metadata records, e.g.
  - "cmdi": "http://example.org/vocabulary#oaiCMDIRecord"
Implementation Details II

- Supports **CMDI** and **DCMI** by default, but extensibility for additional formats should be very straightforward.

- **oai:identifier** in our case: relative path in the Fedora repo (includes PID variant).

- New implementation allows for designating parent resource containers as "sets" via set predicates (was less elegant in Fedora 3).

- Harvesting process not optimized yet (depth-first traversal of all resources).
  → Future work
Summary

• No concrete action on official OAI-PMH support by Lyrasis so far.

• Leipzig implemented its own solution.

• Based on Django, already in use.

• Will look into optimization in the future.

Thank you for listening!