ARCHE Suite: A Flexible Approach to Repository Metadata Management

Mateusz Żółtak, Martina Trognitz and Matej Durco
mateusz.zoltak@oeaw.ac.at, martina.trognitz@oeaw.ac.at, matej.durco@oeaw.ac.at

CLARIN Annual Conference 2021
Why a new solution?

- Fedora 4/5
- Extensible (without Java)
- Invenio
- DSpace
- Dataverse
- Search API
- Reliable transactions
- Reliable backups
- Support for multiple identifiers
- Native RDF support
- Fedora 6
- Reliable transactions
- Native RDF support
- Reliable backups
Why do we need RDF?

• To migrate easily from Fedora 4
• Because we like LOD and Semantic Web
• To deal with named entities in a standards-compliant way

Diagram:

- res1 - author - John Doe
- res2 - author - Doe John
- res3 - author - https://orcid.org/123
- res4 - author - https://viaf.orf/789
- res5
  - author
  - res1
  - res2
  - res3
  - res4
  - name - John
  - surname - Doe
  - id
    - https://orcid.org/123
    - https://viaf.orf/789
How do we deal with the RDF? 1/2

• Repository is just an RDF graph
  (repository resource <=> node in RDF graph)
  • Each node can optionally store a binary content

• Automatic "same as" detection on metadata ingestion to avoid duplicated nodes
How do we deal with the RDF? 2/2

Easy way to fetch RDF subgraphs of interest, e.g.

- metadata of a resource and all its RDF neighbors
- metadata of a resource and all its RDF "relatives"
CLARIN - ready

- Support for
  - FCS 2 basic search
  - Switchboard
  - VCR

- Rich and fast OAI-PMH templating engine
  - Allows on-the-fly XML metadata generation from the repository's RDF metadata
  - Examples (basic CMDI, ARIADNE, DHA, Kulturpool)
Why RDBMS instead of a triplestore?

• Reduced resource usage and latency
  • 25 times lower RAM usage comparing to Fedora 4 + Blazegraph

• No synchronization problems
  • all features provided by a single component (including spatial and/or full text search)

• Proper transactions support

• We don't really need SPARQL
  • It can be easily plugged in if needed
Extensible architecture

• AMQP-based plugins
  • supported in 20+ programming languages
  • RDF on input and output

• Intensive use of microservices
  • can be written in any programming language
  • just have to stick either to the arche-core REST API or the RDBMS data model
Why it's written in PHP?

• There is no bad programming language but there are bad programmers

• We have good PHP programmers
  • Java programmers developing software which doesn't leak memory are beyond our reach

• It's written in a way you can extend it in any programming language you want