Implementing institutional Content Repositories with MyCoRe and MILESS

3rd OAI Workshop @ CERN, Feb 12, 2004



<MyC•Re>

http://miless.uni-essen.de/

http://www.mycore.de/

Dipl.-Wirt.Inform. Frank Lützenkirchen



University Library, Campus Essen Duisburg-Essen University, Germany

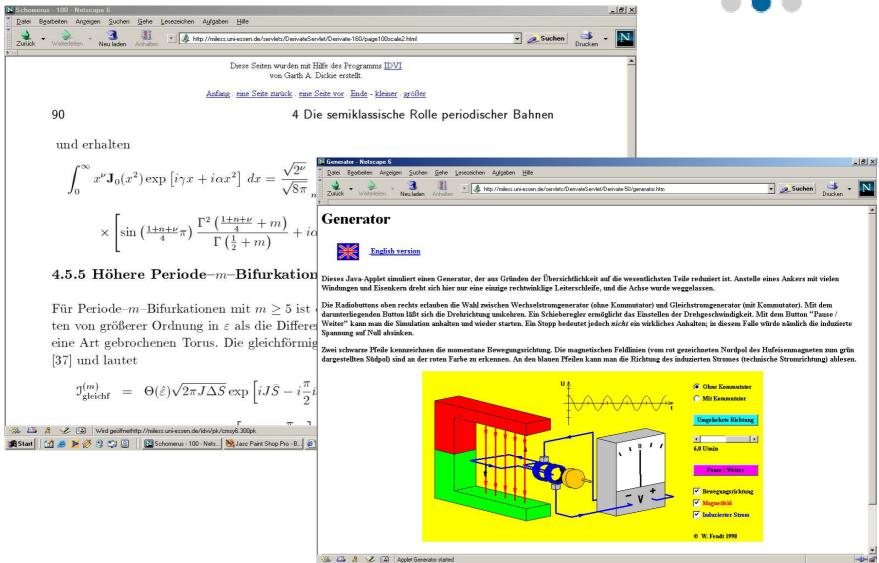
Context and Design Philosophy (I)



- MILESS: the origin of MyCoRe
 - = "Multimedialer Lehr- und Lernserver Essen"
 - = multi-media teaching and learning server Essen
 - = our local "digital library" or "document repository"
- Developed at Essen University since 1998, in production
- Java and XML/XSL application
- Open Source, reused currently at 12 german universities & Uppsala
- Hard-coded metadata model based on qualified **Dublin Core**
- Any file types, special support for audio/video streaming
- articles, animations, dissertations, audio/video, foils, lecture material
- Java applet allows staff to manage their content independently

Context and Design Philosophy (II)





Context and Design Philosophy (III)



- MyCoRe
 - = "MILESS Community Content Repository"
 - = the **core** to build **my** own local repository
- Grew out of the community of universities that reused MILESS
- Goal: build a successor for MILESS, which allows
 - flexible, customizable metadata models
 - customizable user interface
 - support for multiple languages in user interface and content
 - a **common "core"** of Java classes that could be used to build a more specialised, local content repository
 - support for multiple database backends, both commercial (IBM Content Manager) and Open Source
 - support for multiple audio/video streaming servers

Context and Design Philosophy (IV)



MyCoRe

- still under development, partially incomplete
- first "official" version will be launched in march, 2004
- main efforts currently in documentation and testing
- consists of "the core" (Java libraries, some XSL stylesheets)
- the official "sample application" that shows what you can do: an institutional document repository with Dublin Core datamodel
- currently one other application based on MyCoRe exists: "Papyri", a repository for digitized egyptian papyrus fragments

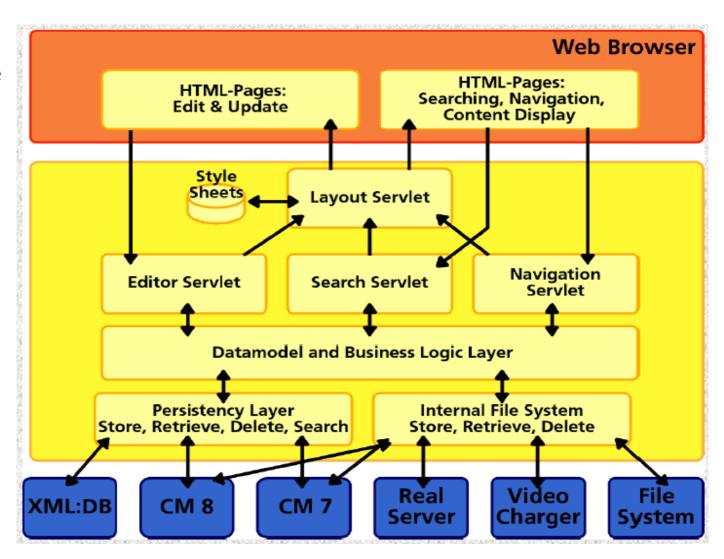
MILESS

- already uses large portions of the MyCoRe code
- will sometime completely substituted by a MyCoRe application

Context and Design Philosophy (V)



MyCoRe architecture



Interoperability (I)



Configurable OAI-PMH 2.0 implementation

- Sets are mapped to a selected MyCoRe classification
- Metadata formats are generated from XML using XSL stylesheet
- Allows different views (browsing sets) on the repository

• XML import and export command line tool

- Any metadata can be exported/imported as XML together with associated files

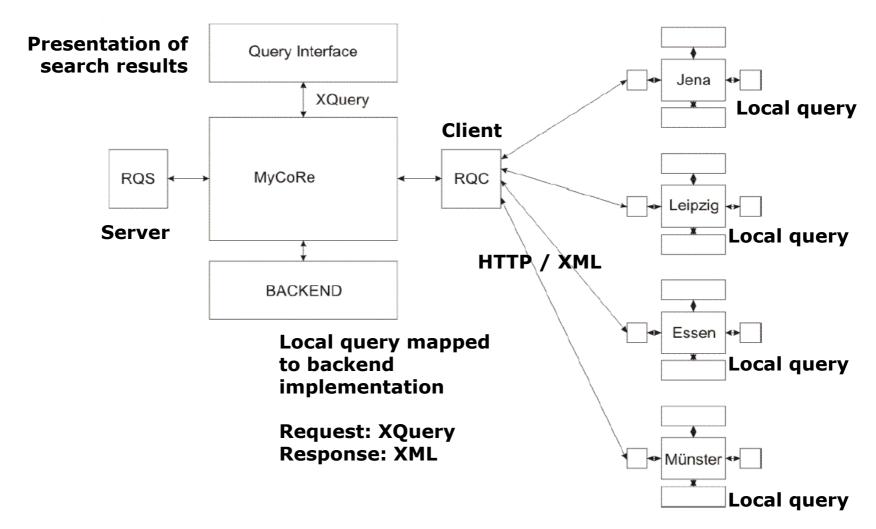
• Export of any metadata and query results as XML via HTTP

- Servlets generate XML output for metadata, query results etc.
- By default, this XML is rendered on server-side to HTML pages
- Optionally, XML can directly be delivered to HTTP client

Interoperability (II)



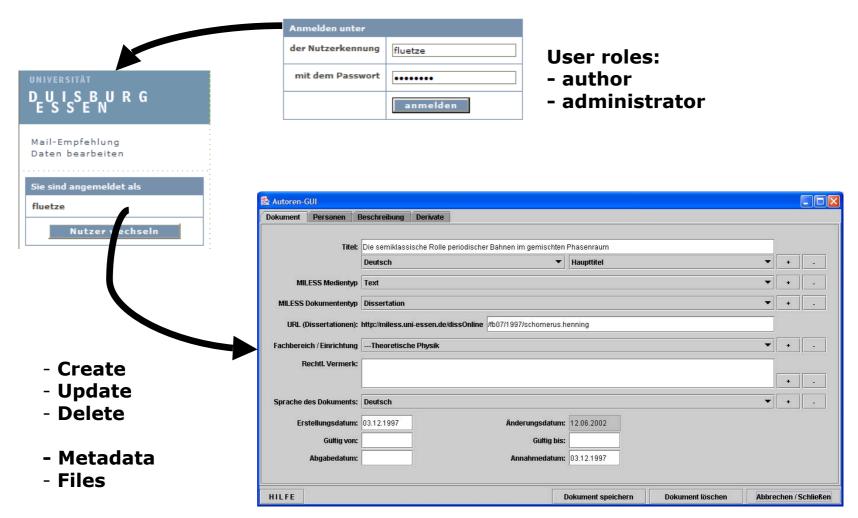
• Integrated distributed query across multiple MyCoRe instances



Content Submission (I)



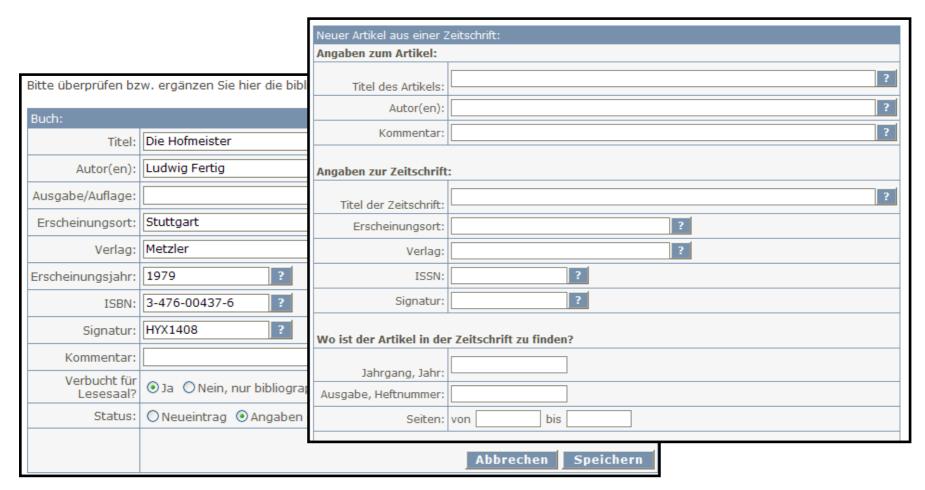
• MILESS: Java applet for metadata and content management



Content Submission (II)



• MyCoRe: Configurable data model, therefore also configurable HTML-based editor for metadata and file upload



Content Submission (III)



- Philosophy: users with "author" privileges are allowed to create, edit, delete their documents any time, immediately
- Exception: dissertations etc, "manual" workflow
- Currently **no automated workflow** mechanisms (planned)
- But:

MyCoRe allows to **send email** from within the application Notification of a person for quality control etc. possible

- Some additional metadata is generated during submission, e. g.
 - dates of creation and modification
 - file content types, mime types, size
 - audio/video technical metadata like bitrate, framerate, ...

End User Interface (I)



Search masks to query all or parts of the repository



- Multiple search masks for different departments, content types etc.
- Fields shown in the mask can be configured
- Search mask can be limited to query only selected parts of repository

End User Interface (II)



- Browsing content through custom "classifications"
- Classification := a hierarchical tree or list of categories
- Examples: DDC, organizational structure of your institution, ...
- Custom classifications can be loaded from XML



End User Interface (III)



• Any content is directly accessible through http URL: document metadata, files, videos, ...



- Persistent Identifier: Implementation of "National bibliographic names" URNs, urn:nbn:de:... as defined by "Deutsche Bibliothek, Frankfurt"
- End user interface is completely HTML based
- MyCoRe: multiple language user interface possible
- HTML pages are generated from XML using **XSL stylesheets** on server side
- Allows easy customization and localisation

Long term preservation issues (I)



- Any metadata is stored in XML format
- Any content (files) is stored together with metadata internally
- NBN URNs as persistent identifiers to access content
- Support for multiple database backends includes
 IBM Content Manager, which has
 hierarchical storage management functions (Tivoli Storage Mgr)
 to allow very large repositories and archiving of content
- But: currently **no formal implementation** of long term preservation standards, e. g. OAIS standard

Strengths (I)



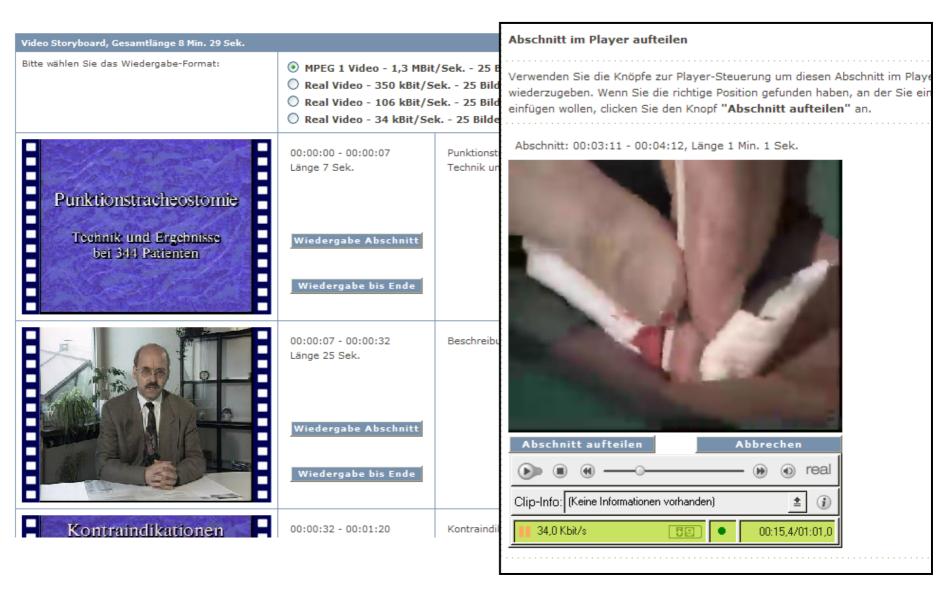
- Open Source
- Custom metadata models in MyCoRe
- Custom classification trees with any categories
- Any files as content, also multimedia, audio, video, ...
- Special support for audio/video files:
 - **Streaming** of mpeg, real, mp3, avi, wav, ... with integrated Real Server, Helix Server, IBM VideoCharger Server
 - Technical metadata for audio/video files
 - Online **storyboard editor** to segment and annotate video

Multiple database backends possible

- Commercial: IBM Content Manager, Tamino, Oracle, DB2
- Open Source: MySQL, Apache Lucene, XML:DB (Xindice etc.)

Strengths (II)





Weaknesses (I)



- Currently two systems:
 - MILESS: production, but hard-coded DC metadata model
 - MyCoRe: more flexible, but partially still incomplete
- Automated submission workflow to be implemented
- No peer reviewing support etc. currently
- Formal long term preservation support yet to be implemented: conformance to OAIS standard etc.

Thank you for your patience. Any questions? luetzenkirchen@bibl.uni-essen.de





http://www.mycore.de/



http://miless.uni-essen.de/