CMDRSB

CLARIN Metadata Repository/Service/Browser

http://clarin.aac.ac.at/MDService2/
**Basics - Interaction**

- **MDS**ervice accepts queries about metadata from **MetadataBrowser** (and external Applications)

- and passes them to the **Metadata Repository**s

- and/or to the **Virtual Collection Registry**

- optionally applying **Semantic Mapping** based on the information from **Component Registry**, **Data Category Registries** and **Relation Registry**

- receiving results and passing them (optionally formatted) back to the requesting node.
Functionality - Service

→ REST-interface (MDService2 WADL, HTML version)

- collections
  list the „natural“ hierarchical collections-structure of the repository

- model
  return xml-elems used in the repository (with usage statistics)

- terms
  return terms/indices/xml-elems used in the repository enriched with
  a) the usage statistics (count occurrences and distinct values)
  b) the corresponding CMD-components and data categories

- values
  list distinct values for given index (similar to facet functionality)

- recordset
  retrieve a list of MDrecords based on a query [CQL]

- record
  retrieve individual MDrecord based on the identifier
Functionality - Browser

- Dynamic Repositories
- Collections browsing
- Terms/Values browsing
- Query Input
  - Simple full-text query
  - Complex queries (CQL-searchclauses, boolean op)
  - Index auto-completion
- Queryset/Resultset
  - work with multiple results in parallel
  - Paging
  - Variable views (select columns, auto-columns)
- Workspace (storing queries, bookmarks)
- „Linkable“ Queries
- (Semantic Mapping)

http://clarin.aac.ac.at/MDService2/docs/htmlpage/info
Collections and Terms

- **collections**
  the „natural“ hierarchical collections structure of the repository (OLAC, IMDI, ... and subcollections)

- **terms**
  + information on the XML-structure of the records in the Repository = the profile usage statistics (occurrences and distinct values)
  + thanks to: CMD Profiles -> Schemas -> MD-Instances
    it is possible to track back from the instances to the profiles and from there to link to the data categories

```
<CMD><Header><MdProfile>{profileID}</MdProfile>
<Components><{profileName}>
 <{component}>
   <{element}>
</CMD_ComponentSpec>
<Header><ID>{profileID}...</Header></Header>
<CMD_Component name=“{profileName}”>
<CMD_Component name=“{component}”>
<CMD_Element name=“{element}”
  ConceptLink=“{datcat-uri}”>
</CMD_Element>
</CMD_Component>
</Components>
</CMD>
```

```
<dcif:dataCategorySelection>
<dcif:dataCategory
  pid=“{datcat-uri}” >
{detail-information}
</dcif:dataCategory>
</dcif:dataCategorySelection>
```
Query Language

- **simple full text queries:**
  - simple term **system**
  - any of multiple terms (OR) **child | acquisition**
  - all of multiple terms (AND - default) **child acquisition**
  - Phrase a sequence of terms "**longitudinal study**"

- **Bookmarks search/remember individual records (by handle/identifier):**
  - **clarin-at:aac-test-corpus:C4:158**

- **Complex search queries [SRU/CQL]:**
  - basic search clause (index relation term): **author any Adler**
  - boolean **title contains a and imprint.date between 1910 and 1920**

- **Combine simple query and search clauses**
  - **university and (title any system)**

- **Restriction by Collections:** **aac-test-corpus**

- **Search in a profile:** **LrtInventoryResource**

- **Search via DatCats ("Semantic Search"):** **isocat:creationDate contains 191**
Semantic Mapping

• Basic Idea

query: Actor.Name any Peter
+ relations: 
  (#DatCat)
  #sameAs (#Actor, #Person)
  #sameAs (#Name, #FullName)

= expanded query: 
  Actor.Name any Peter
  OR Actor.FullName any Peter
  OR Person.Name any Peter
  OR Person.FullName any Peter

• Levels

1. just mapping based on the ConceptLink resolvable via ComponentRegistry
2. use equivalence relation between DatCats from Relation Registry
3. use equivalence relation also between Component DatCats (yet to come)
4. use also other relations in Relation Registry (subClassOf, synonymy?, ...)
Public Space/Personal Workspace

- Shall allow for customization
  - save Perspectives/Views

- Currently supporting
  - Storing Queries
  - Bookmarks

- But User-management (join the federation) not solved yet
  Thus there is only one common shared space

- Public Space meant for News, as Dashboard, options...
  obviously function not clear yet
Standards Conformance

- Built on **CMD**
- Reading Data Category Registries
  - *isoCAT, dublincore, ...* (open for further DCRs)
- „Inspired by“ the standard-protocol **SRU/CQL**
  - Started opportunistically, but working towards conformance
  - Supported:
    - The query language CQL (parsing)
    - The format of the result `<searchRetrieveResponse>`
  - Main differences:
    - The current interface has to be mapped onto the protocol
    - mapping of collections - not solved (explain?)
    - Result-format: scan, explain
- Diagnostics

<table>
<thead>
<tr>
<th><strong>SRU/CQL</strong></th>
<th><strong>MDService</strong></th>
<th><strong>MDRepo</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>(explain, Zeerex)?</td>
<td>collections</td>
<td>getCollections</td>
</tr>
<tr>
<td>explain</td>
<td>model, terms</td>
<td>queryModel</td>
</tr>
<tr>
<td>scan</td>
<td>values</td>
<td>scanIndex</td>
</tr>
<tr>
<td>searchRetrieve (CQL)</td>
<td>recordset, record</td>
<td>searchRetrieve (but XPath!)</td>
</tr>
</tbody>
</table>
Planned/Phantasized Functionality

• Interaction with Virtual Collection Registry
  – Stored query → Intensional VC
  – Result of a (stored) query (Recordset) → Extensional VC
  – Collection of Bookmark → Extensional VC

• produce a SRU/CQL-protocol conformant REST-interface

• Custom Termsets

• Custom Views

• Queries Sorting

• Result Export (especially also only selected fields (not full records))

• translate UI, BUT mainly also the search-indices
  based on DCR language-sections

• Commenting – collaborative curation
  Allow to „annotate“ / comment / make notes on the MDRecords
  just get email. via POSTing trac.tickets?
Situation and Outlook

• The MDRepository currently contains around 109,000 records, mainly from the datasets: OLAC and IMDI (data statistics)

• Currently there are three instances of the MDRepository running providing similar but not identical datasets:
  – University of Gothenburg (main)
  – ICLTT, Vienna
  – MPI Psycholing, Nijmegen

• A first version of the MDService and Browser is online: clarin.aac.ac.at/MDService2 (this address may change)
  – Although the repository and interface already provide a lot of information and functionality, it is demo-quality and cannot yet be seen as reliable service.
  – Lot of work is still needed both on the data quality and user interface:
    • rework of the UI - based on feedback at CMDI-Workshop, Nijmegen20110117
    • continuous integration of new datasets (provided for harvest by the centres)

• Nevertheless we invite you to try it out and look forward to any critical remarks
Overview of functionality (not completely up-to-date but largely correct)

 MDBrowser

Metadata Service

simple search

complex search: AND/OR combined SRU/CQL search clauses: 'index relation term'

dynamic columns (auto-columns)

paging

working with queryset: multiple queries/results in parallel

view full record-detail

link to the resource (provide Resource Viewers)

Content Provider X

Content Provider Y

Provider Z

stored queries:
- Public Space
- Personal Workspace

browse collections hierarchy

explore and search by "terms" i.e. elements from used profiles, referenced data categories and relations between them