CUSTOMIZED OAI-ORE AND OAI-PMH EXPORTS OF COMPOUND OBJECTS FOR FEDORA REPOSITORIES

Alessia Bardi, Sandro La Bruzzo, Paolo Manghi
name.surname@isti.cnr.it
http://nemis.isti.cnr.it/groups/infrascience
Digital Library Systems (DLSs) need to export content

- Export of compound objects: packages of information objects with an identity

- OAI protocols (typical solution)

DL management systems (DLMSs) need to support export protocols

DLMS issues:

- Absence of OAI-exports: some DLMS do not provide support for OAI protocols (Relational Databases)

- Pre-defined OAI exports of compound objects: shape of objects as exported by OAI protocols is pre-defined by the DLMS (Fedora)
A Fedora instance manages graphs of Data Objects.

Data Objects conform to Content Models.

Content Models define:

- Type and name of mandatory datastreams (files)
- Allowed relationships to other Data Objects
FEDORA OBJECTS AND COMPOUND OBJECTS

CM_article

ds: ART, PDF
ds: DC, XML

USES
USED_BY

GENERATES
GENERATED_BY

CM_data

ds: DATA, binary
ds: DC, XML
ds: DDI, XML

Document model: implementation with Fedora Content Models

hasModel

Art1

ds:DC

d:ART

GENERATES

Compound Object 2

GENERATED_BY

Art2

ds:DC

d:ART

USES
USED_BY

Compound Object 1

Data 1

ds:DATA
ds:DC
ds:DDI

USED_BY

USES

GENERATES

USES

GENERATED_BY

Fourteenth International Conference on Grey Literature - Rome, 2012
## EXISTING OAI-PMH SOLUTIONS

<table>
<thead>
<tr>
<th></th>
<th>Basic OAI-PMH Provider</th>
<th>OAI Provider</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMH-Set</td>
<td>Each Fedora Data Object is annotated with relationships to the Sets it belongs to</td>
<td></td>
</tr>
<tr>
<td>PMH-Item</td>
<td>Fedora Data Object</td>
<td>Fedora Data Object</td>
</tr>
<tr>
<td>PMH Metadata Records</td>
<td>Datastream</td>
<td>Datastream</td>
</tr>
<tr>
<td>PMH Metadata Format</td>
<td>Dublin Core</td>
<td>Any format in existing datastreams</td>
</tr>
<tr>
<td>Compound Object Boundaries</td>
<td>Fedora Data Object</td>
<td>Fedora Data Object</td>
</tr>
</tbody>
</table>
# EXISTING OAI-ORE SOLUTIONS

<table>
<thead>
<tr>
<th></th>
<th>OREProvider</th>
<th>Fedora2ORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORE Aggregation</td>
<td>Defined by annotation</td>
<td>Sub graph visited starting from a given Fedora Data Object</td>
</tr>
<tr>
<td>ORE Aggregated Resources</td>
<td>Datastreams with a given name (e.g., DC)</td>
<td>Fedora Data Objects in the sub graph</td>
</tr>
<tr>
<td>ORE Proxy</td>
<td>not supported</td>
<td></td>
</tr>
<tr>
<td>Relationships between Aggregated Resources</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>Compound Object Boundaries</td>
<td>Fedora Data Objects annotated with the same tag</td>
<td>Navigation depth</td>
</tr>
</tbody>
</table>
EXPORTING COMPOUND OBJECTS: 
THE OAIZER SOLUTION

- OAizer is a generic software for the customization of compound objects in Digital Libraries
  - Software layer over the Digital Library Management System
  - ORE exports include relationships and their semantics
  - PMH metadata formats can be generated on request

- DB-OAizer:
  - works on RDBMS;
  - prototype in the OpenAIRE project, to be integrated with OpenAIRE+ (Open Access Infrastructure for Research in Europe)

- Fedora-OAizer:
  - for Digital Library Systems based on Fedora frameworks with Content Models
FEDORA-OAIZER: HIGH-LEVEL ARCHITECTURE

OAlzer

Entity Graph

OAI view

View Interpreter

OAI-ORE exporter

OAI-PMH publisher

DLS document model implementation

Graphs of Content Models

Compliant to

FEDORA

Graphs of Data Objects

Fourteenth International Conference on Grey Literature - Rome, 2012
PHASE 1: ENTITY GRAPH

Entity Graph: Graph representation of the document model

- Created exploiting information about Content Models
- Graph nodes represent Content Models
- Properties of a node represent the datastreams defined by the Content Model
- Edges between nodes represent the relationships defined by the Content Models
**PHASE 2: DEFINITION OF THE OAI VIEW**

**OAI View**: Tree representation of the structure of the compound objects to export

- Sub graph of the entity graph with entities and relationships to include in the compound object
  - Ex. The DATA datastream of CM_data is excluded
- The root node is the entry point of the OAI view
PHASE 3.1: INTERPRETATION OF THE OAI VIEW FOR OAI-PMH

Instance View: created by navigating the graph of Data Objects according to the paths in the view

Data Transformation Rule (XSLT): OaizerXML -> MF1

OAI-PMH View Interpreter

Set View

- Item Instance_View_1
  - DC
  - OAIIZER-XML
  - MF1

- Item Instance_View_2
  - DC
  - OAIIZER-XML
  - MF1

-...

OAIIZER-XML Xml Serialization of the Instance View
PHASE 3.2: INTERPRETATION OF THE OAI VIEW FOR OAI-ORE

Legend:
- Aggregation
- Aggregated resource
- Proxy
- ore:aggregates relation

Art2

uses
Data 1

uses
Art1

uses

GENERATED_BY

http://<server>/oaizer/Art1/OAIZER-XML

http://<server>/oaizer/Data1/OAIZER-XML

http://<server>/oaizer/Art2/OAIZER-XML

OAI-ORE View Interpreter

Aggr

Proxy

proxyFor

proxyIn

proxyFor

proxyIn

proxyFor

proxyIn

GENERATED_BY

Legenda:
- Aggregation
- Aggregated resource
- Proxy
- ore:aggregates relation
CONCLUSIONS

Don’t adapt your document model for the exports, adapt the exports to your document model!

• Customized, domain-dependent exports of compound objects

• Future work:
  • Better graphical support for the definition of views in case of entity graphs with cycles
  • Implementations for different back-ends
  • Integration of DB-OAIzer with OpenAIRE