DIGITARQ
Creating and Managing a Digital Archive

Miguel Ferreira
mferreira@dsi.uminho.pt

José Carlos Ramalho
jcr@di.uminho.pt

ELPUB ’04
26 June 2004

Porto District Archive [ADP]
The proposal

- To build an unique repository according to international standards
- To have an unique interface to access and query the repository
- To have the system available through the Web

Contents

- Introduction to the project
- Archive’s Metadata
- XML throughout the project
- Storing the metadata
- Model evaluation
- Conclusions and future work
DigitArq Project [main goals]

- Information centralisation
- International Standards
  - *International Standard Archival Description* - ISAD(g)
  - *Encoded Archival Description* - EAD/XML
- Paper use reduction/elimination
- Publish the archive’s catalogue on the Web

Paper finding aids [digitalisation]
Different databases [migration]

- Access
- Filemaker
- Anqbase
- Word
- XML
- Excel
- Transformers
- <EAD>

Results of migration

- Hundreds of EAD/XML files
- The need to manage all this information
- Publish adequately on the Web

How do we store this information?
Archival metadata (EAD)

- **Hierarchical structure**
- A top-down, **general to specific** description
- Organised in description **levels**
- Unit ID, title, unit dates, physical description, scope and content, etc.

XML Pros and Cons

**Pros**
- The **hierarchical** model is assured
- **Portability** – easy to exchange information with other archives

**Cons**
- Too **Low-level** for user consumption
- Difficult to store and **maintain**
Metadata storage architecture

- Download()
- Upload()
- Children()
- CreateNode()
- AppendChild(child)
- RemoveChild(child)
- HasChildren()
- Clone()
- Parent()

Model implementations

- EAD/XML Files
  - DOM
- Relational Databases
  - A column for each description property
  - Id, ParentId, HasChildren
  - Circular relationship
  - In optional properties allow NULL values
  - Additional tables for list-based data
Model evaluation

- Pros
  - Simple to use and implement
  - Accounts for the use of Catamorphisms
  - Simplified migration of data between different storage-level types (e.g. relational vs XML)

- Cons
  - Doesn’t take full advantage of RDB
  - Finding a particular records usually means traversing the whole tree of description

DigitArq [final product]
Finding aids management software [innovative features]

- Uses relative references
- Automatic description revision
- Inference mechanisms

Conclusions

- Abstract model for hierarchical metadata representation
- Simple and transparent
- Independence of storage-level
  - Different types of databases can be used
- Easy to migrate data between different storage-levels
Future work

- Model optimisation for Relation DB
  - Stored procedures
- Cache
  - Minimises traffic between application and DB
- Pre-loading
  - Predict what information is going to be requested next

Questions?

DIGITARQ
Creating and Managing a Digital Archive

Miguel Ferreira
mferreira@dsi.uminho.pt

José Carlos Ramalho
jcr@di.uminho.pt