Integration: Knowing the Limits of a Repository

MacKenzie Smith
Associate Director for Technology
MIT Libraries
What is a Repository?

Main Entry: 1re·pos·i·to·ry

1 : a place, room, or container where something is deposited or stored : depository

2 : a side altar in a Roman Catholic church where the consecrated Host is reserved from Maundy Thursday until Good Friday

3 : one that contains or stores something nonmaterial <considered the book a repository of knowledge>

4 : a place or region richly supplied with a natural resource

5 : a person to whom something is confided or entrusted

Merriam-Webster dictionary
Institutional Repositories

- Institution-based
- Scholarly material in [all] digital formats
- Cumulative and perpetual
- Open and interoperable

SPARC, 2002
Digital Libraries

- *Digitized* library material
  - Special collections
  - Archival collections
  - Non-print collections (images, multimedia, etc.)

- Resembles publishing
- Often requires specialized viewing tools or applications
Learning Object Repositories

- Discovery/reuse of discrete “Learning Objects” of teaching material

- Must interoperate with
  - Course Management Systems
  - Virtual Learning Environments
  - Collaborative Learning Environments

- Standards from IEEE, IMS, ADL, OKI...
Data Repositories

- Typically specialized for one type of data, e.g.
  - Statistical or geospatial datasets
  - Biological pathways or micro-array data
  - Genomics or proteomics data
  - Radio telescope or ocean sensor data

- Specialized requirements for deposit, search, viewing, manipulating data
Non-academic Repositories

- Back end for commercial Content Management Systems, other applications
- Accessed by JSR 170, WebDAV, proprietary APIs
- Next generation file systems which support versioning, rich metadata, security, etc.
All of These are called “ Repositories”

- DSpace
- Eprints
- Fedora
- SRB
- Jackrabbit
- Digital Commons
- Documentum
- Harvest Road
- Digitool
- i-Tor
- ARNO
- CDSWare

- Function at different levels of the technical architecture
- Have different features, functions
- Support different (or many) service models
DSpace Example: Core Functionality

- **Asset Capture**
  Provide a workflow system that allows people (or their proxies) to deposit material directly into the archive

- **Description**
  Provide tools for searching, browsing metadata and/or full-text of archived material

- **Redistribution**
  Provide web-based access to archived material once found

- **Preservation**
  Provide long-term access to archived materials by preserving the contents over archival time frames (i.e. by format migration, emulations tools, etc.)
Not Core (for DSpace)

Specialized functionality e.g.

- Traditional archives functionality
  provenance, integrity, finding aids, etc.
- Records Management
- Publishing (e-journals or other)
- Classroom delivery
- Tools for content manipulation
  e.g. VDC for statistical datasets, ESRI for GIS, M:Media
  for annotated multimedia collections
Interoperability

- Solution for non-core functionality

Make DSpace interoperate with other systems and tools that extend its functionality to support non-core services
Interoperability Dimensions

Systems and Tools interoperate at the protocol level

- OAI-PMH
- Web Services
- Z39.50
- CORBA
- OKI
Interoperability Dimensions

Applications interoperate at the functional level

- Institutional Repository
- Course Management (e.g. SAKAI, Blackboard, WebCT)
- Electronic Records Management
- E-Publishing
- E-Research (e.g. Grid)
Interoperability Dimensions

*Data* interoperates at the **semantic** level

- **Descriptive**
  - MODS, DC, MARC, VRA, LOM, FDGC, DDI...
- **Administrative**
  - ODRL, <indecs>
- **Technical**
  - PREMIS, MPEG7
- **Structural**
  - METS, IMS-CP, MPEG21
Interoperability is the Intersection of

Semantics

Functions

Protocols
Defining Limits

How do you define the limits of

- an IR?
- a library or an archive?
- “academic computing”?
- “content management” or “asset management”?
Defining Limits

Focus on Mission

- Collection/asset management?
  - Business layer
  - Policies, description, long-term preservation, reporting, legal oversight, etc.

- Search?
  - Business layer
  - But what about Google Scholar? Scirus? Thomson Web Citation Index? Worldcat?
Defining Limits

• Publishing?
  • User Interface or application layer
  • Manikin, other Cocoon-based UIs
  • Specialized content delivery tools
    e.g. classroom delivery, dataset manipulation

• Storage?
  • Storage layer
  • SRB, data grid
  • Requires serious technical expertise
Conclusion

- Start with a *well-defined service model* (business process, e.g. an IR)
- Focus application development on *that model*
- Accomplish other functionality through *interoperation*
- *Technical* interoperation between component systems is easy...
- Defining how those systems will work together *functionally* is harder... need a *service framework*
- Achieving useful *semantic* interoperability is very hard... need fewer “standards” and better *data models*