Moving repositories upstream in the content lifecycle

Richard Green and Simon Waddington

OR11, Austin TX, 11 June 2011
CLIF Project

CLIF - Content Lifecycle Integration Framework

Funded by JISC

01 July 2009 – 31\textsuperscript{st} March 2011

~$450,000

Project partners

University of Hull

King’s College London - Centre for e-Research (CeRch)
Background

• CLIF built on work within the JISC-funded RepoMMan and REMAP projects
  • In particular, REMAP explored how a repository could support records management and digital preservation as part of a lifecycle management approach for digital content

• Previous work had sought to push the repository upstream in the workflow
  • Dilemma was that the repository risked becoming another content silo alongside other content management systems on campus (in our case, Sakai and SharePoint)
  • How can the repository become more integrated in the institutional environment?
Lifecycle integration

Content flows between systems according to need in lifecycle
CLIF project objectives

• Understand how digital content can be managed across systems as part of the digital content lifecycle
  • Recognising that individual systems cannot always support the whole lifecycle from creation to preservation or deletion

• Specifically investigate the role of repositories in the digital content lifecycle
  • Where is the repository best positioned within the lifecycle?
  • What roles can digital repositories play?

• Understand how content will flow in and out of a repository as part of the lifecycle
  • CLIF has been agnostic about this
CLIF outputs

- Literature review on managing the digital content lifecycle across systems

- Technology integrations as exemplars of how a repository can support lifecycle management across systems
  - Fedora – Sakai integration
  - Fedora – SharePoint integration
  - Hydra-compliant digital objects
  - Software available on GitHub
  - Technical appendix to final report describing architecture and implementation
CLIF use cases

• Use cases cover research, teaching and administration

• Based on interviews with staff at partner institutions
  • Academic staff (Head of Department / Senior Lecturer)
  • Records Manager
  • Research active staff

• Interviews highlighted that staff were managing as best they could within single systems they were familiar with
  • Potential to exploit additional functionality in other systems welcomed
Sakai – Fedora integration

• Sakai 2.6.1 + Fedora v3.4

• Extends and enhances the JISC CTREP Fedora ContentHostingHandler plugin
  • CTREP was a previous JISC project (2007-2009) as part of which the University of the Highlands and Islands tried to integrate Sakai and Fedora 2.x
  • Not completely successful due to (then) limitations in Sakai code
  • Good starting point for CLIF
  • Content displayed in standard Sakai Resources Tool
    • Resources Tree view shows a ‘live view’ of a specific Fedora collection
    • ‘Show other sites’ allows files and/or nested folders to be copied/moved between MyWorkspace site and Fedora mounted site
Sakai – Fedora features

• The repository is embedded in Sakai as a set of resources that appear like any other set of resources
  • The majority of menu functions work in the same manner as with standard resources, e.g., upload, copy, paste, move, delete, create
  • This applies to folders as well as individual objects
    • Folders represent collection objects in the repository
  • Metadata can be captured in Sakai for use in Fedora (though Sakai is not able to re-use this when retrieving an object from Fedora)
  • User can browse Fedora collection (though not yet search)
• User does not (need to) know they are working with the repository
It looks easy, but…

... you don’t see what is going on underneath!
Copy/move to/from Repository

Copy or move files/folders/trees between Fedora and MyWorkspace is easy! Copy...
Copy/move to/from Repository

...paste!
Copy/move to/from Repository

...paste!
SharePoint – Fedora integration

- Microsoft Office SharePoint Server 2007
- “Swiss army knife” enterprise application
- Fedora Commons repository v3.4
- “Out-of-the box” – Hydra-centric content modelling
- Aimed to provide a “reference implementation”
- Produce a toolkit of features that can be deployed selectively to support different usage contexts
- Provide administrative and end user interfaces
- Documents can be assigned to “projects”
System requirements

• Deposit of documents to public repository through formal approval process (e.g. publications)
• Deposit to private repository “folders” for archival of internal documents, data, images
  • Archival of research projects and data
• Finding and retrieving documents
• Management of repository spaces
• Archiving of document collections (e.g. at end of a project or task)
Implementation

• Based on SharePoint MySite
  • Personal site that can be shared with other users
• Implementation carried out as “SharePoint features”
  • Enables flexible configuration by administrator
• Implementation uses C# middleware
  • Wrapper for Fedora API-M and API-A web services via SOAP
  • Performs Hydra-compliant Fedora object creation in FOXML schema (native Fedora format)
• Fedora objects
  • Simple objects – “file” or “folder” linked by RDF relations
  • Capture DC metadata from SharePoint upload form
  • XAMCL Policy datastream created to handle access
Administration

• Root site for MySite administration
• Creation of MySite for new user automatically deploys CLIF solution – “feature stapling”
  • Includes creation of Fedora repository user account and private folders
• Configuration of public repository locations
• Assignment of approvers for each public repository
Deposit to Fedora

- Deposit options selected from menu on item in document library

- Copy to Repository
  - Copies content item and metadata to private repository folder
  - Bulk copy – copies multiple items

- Move to Repository
  - Moves content item and metadata to private repository folder
  - Replaces item in document library by a hyperlink

- Publish to Repository
  - Default is to run approval workflow
  - Option to provide MODS metadata entry form for entry of detailed preservation metadata
Deposit of single documents
Publish to repository

• Select location for publication for configured repositories
• Can be configured to run approval workflow
• Workflow creates task for approver – displayed on MySite homepage
• Publication initiated by approval
• Linkage to further appraisal systems
Copy and move to repository

- Selecting the location to deposit documents in the repository
- _private and _archive folders created as default
- User can add additional folders
Bulk copy and move

- Files for deposit can be selected using check boxes from a pop-up page
Retrieval of documents

- Retrieval of moved documents
  - Archive List in left side frame of Document Library page
  - Documents can be retrieved by selecting its hyperlink in Archive List
  - Metadata of moved documents is retained in SharePoint
  - Currently a “closed system” – only Hydra-compliant objects can be imported

- Search – based on SharePoint indexing
  - Search of document metadata in SharePoint
  - Full text search of documents in SharePoint
  - Search of document metadata in Fedora
  - Free text search box on MySite pages
Further developments

- Search
  - Integrate with Solr indexing to provide full text search of Fedora
- Repository browse functionality
  - It would be good to handle compound/complex Fedora objects
    - Currently focus on simple objects
  - Verification of Fedora objects for Hydra compatibility
  - Versioning of content items
- Security
  - Robust approach required for use in production systems
- SharePoint 2010 porting
  - All C# code can be reused
  - Modification required to feature definitions
  - There is interest in packaging the integration as a RIC plug-in
Evaluation

• There needs to be a clear understanding and view about where the boundaries are between the different systems being used, to avoid confusion

• There needs to be clarity over why different systems are being used, to overcome concerns about having to work with multiple systems

• There is a need for better preservation and a recognition that integrating the repository could support this, but also a need to be clear about what needs preserving

• There is benefit in being able to access other content stores from within your current working environment in order to see what is available more broadly
Conclusions

• Diverse content management systems can be effectively integrated to allow cross-system lifecycle management
  • Better adoption of interface standards would be helpful
  • Standardisation in the structure of the content being moved maximises how the content can be managed by the different systems

• Where the repository is one of the systems involved its current primary role appears to be as a recipient of content (for preservation)
  • Perception that content in the repository can be used there without moving it into the other integrated systems
Thank you

Chris Awre – c.awre@hull.ac.uk
Richard Green – r.green@hull.ac.uk
Andrew Thompson – andrew.thompson@hull.ac.uk
Simon Waddington – simon.waddington@kcl.ac.uk

Project website - http://www2.hull.ac.uk/discover/clif.aspx
Project GitHub - https://github.com/uohull/clif-sharepoint and https://github.com/uohull/clif-sakai

Project final report -
http://edocs.hull.ac.uk/splash.jsp?parentld=hull:1647%26pid=hull:4194