

# A Technology Analysis of Repositories and Applications

## Interim Report

Submitted to the Mellon Foundation  
October 31, 2005

### Introduction

This document provides a status report for “A Technology Analysis of Repositories and Services.” To date, we have successfully collected a diverse group of scenarios, developed an initial set of use cases, identified what we are calling “key events” from scenarios that will inform the development of functional requirements, and identified content and installed repositories and applications for the analysis. Perhaps most importantly, we have built a community-based interest and effort that informs and guides this analysis.

The main project wiki page is available at:

<https://wiki.library.jhu.edu/display/RepoAnalysis/ProjectRepository>

As described in the proposal funded by the Mellon Foundation, the result of this analysis will be “a set of best practices and recommendations...that result in a typology of repositories and repository users.” These findings will inform the development of an interface layer that will facilitate the integration of modules from various applications with repositories. Perhaps equally importantly, this project is defining a taxonomy and methodology for conducting such analyses. Different audiences refer to different concepts when using the term “repository.” By adopting a clear, transparent, community-based process, this project advances a methodology for ongoing analyses and integration of repositories and applications in a manner that addresses both user needs and functional requirements.

In order to bridge the different perspectives, we have adopted a methodology that includes scenarios, use cases and functional requirements. For our analysis, we relied upon Kulak and Guiney<sup>1</sup> to distinguish among requirements, use cases and scenarios. Requirements, which are aspects that “a computer application must do for its users,” can be reduced in volume by removing conflicts, redundancy, and design assumptions. Use cases are a tool that should show the “what” of the interactions between the users and the computer system. About use cases, the authors state, “it is not only possible but also extremely wise to keep the number of use cases very small.” In producing a small number of use cases for functionality, the analysts and the users are forced to abstract the activities of the system until they truly represent what the system must accomplish. Scenarios are “individual instances of use cases that traverse a specific path using specific data.”

---

<sup>1</sup> Kulak, D. and Guiney, E. (2000). Use Cases: Requirements in Context. New York: ACM Press.

## Scenarios

With this approach in mind, we began with collecting scenarios that describe the diversity of content and functionality needs from the end user perspective. By accessing our network of contacts and partners at other institutions and within Johns Hopkins, we have collected seventy-eight scenarios from fifteen organizations (as of October 31, 2005). At least two other organizations have indicated that they might provide additional scenarios. We provided a basic set of guidelines for these scenarios in an effort to lower the barriers for submission. While this resulted in a diversity of scenarios from various institutions, it has required some additional evaluation from the project team. Every effort is being made to include all the information from the scenarios, without undue interpretation or modification. However, in some cases, the scenarios will require further evaluation by the project team, and possible modification by the individuals who created them.

The scenarios are available on the project wiki at:

[https://wiki.library.jhu.edu/display/RepoAnalysis/ProjectRepository\\_Scenarios](https://wiki.library.jhu.edu/display/RepoAnalysis/ProjectRepository_Scenarios)

As noted on this wiki page, we have classified the scenarios into three categories – “being used for analysis”, “under evaluation” and “revised.” The first category refers to scenarios that conform well to our guidelines, while the second category includes scenarios that might require further investigation or modification to fit well with our methodology. For scenarios for which the creator or submitter makes modifications based on our feedback, we will move them into the third category. The scenarios provide the basis for both traditional use cases and “key events” that will inform the development of the repository functional requirements.

## Use Cases

As mentioned previously, Kulak and Guiney describe use cases as the “what” of the interaction between the users and the computer system, emphasizing that a small number of use cases is not only possible, but also desirable. Given that scenarios represent specific instantiations of use cases, the relatively large number of scenarios can, and should, be mapped to a smaller, canonical set of use cases. In this regard, use cases represent an intermediate step between scenarios, which are essentially stories, and functional requirements. Additionally, many other projects or groups interested in repositories and applications recognize or relate to use cases. In this sense, they provide the useful feature of being a recognizable “benchmark.”

The initial set of use cases is available on our project wiki at:

[https://wiki.library.jhu.edu/display/RepoAnalysis/ProjectRepository\\_UseCases](https://wiki.library.jhu.edu/display/RepoAnalysis/ProjectRepository_UseCases)

While the development of these use cases is helpful for the reasons previously mentioned, the project team began to realize that the “traditional” use case may not be sufficient to define functional requirements from the perspective of repository services. Consequently, we have developed the concept of a “key event”, which is defined as “an interaction with the repository by an element of the system.”

## Key Events

By examining the scenarios with this focus on an interaction with the repository, we have defined a series of key events, which will form the basis for defining functional requirements at the repository or application level. We are currently working with a simple data model that includes objects, complex objects and containers as part of the underlying data structure, which will need to be further defined in order to support the functionality identified through the analysis. In many ways, this concept of key event is critical to understand service requirements at the repository level. It may mark the main “departure” from more traditional use cases.

The key events are available on our project wiki at:

<https://wiki.library.jhu.edu/display/RepoAnalysis/KeyEvents>

It should be noted that the key events are still a work in progress, especially given the importance of the concept for defining functional requirements, which will be mapped to the APIs of the various repositories and applications that we will analyze.

## Repositories and Applications

At the crux of this analysis lies the assumption that we must be “agnostic” about choices regarding repositories or applications. That is, different systems offer comparative advantage depending on the type of necessary content or functionality. For example, rather than assume one will use DSpace only (or Fedora only, or something else only), connected directly with a learning system (e.g., Sakai), our analysis emphasizes that, even within a single institution, one may require multiple repositories and applications (especially when one consider other service needs such as e-publishing or digital preservation). In order to test the potential for existing interface specifications such as JSR-170, OKI DR OSID, or IMS DRI to support widespread integration (rather than specific, pair-wise connections), it’s necessary for us to test multiple systems.

The systems we are evaluating (and accompanying notes) are described on our main project wiki page at:

<https://wiki.library.jhu.edu/display/RepoAnalysis/ProjectRepository>

We are working with the following systems:

- Digital Commons
- DiVA
- DPubS
- DSpace
- ePrints
- Fedora
- Open Journal System
- Sakai

We have communicated with DPubS about participating in this analysis, indicating our willingness to install DPubS locally. Cornell has indicated a desire to build this relationship, but we have not yet finalized the specific arrangement for collaboration. ProQuest will support an instance of Digital Commons for this analysis, providing us full access to their APIs. Regarding DiVA, we have installed the underlying Oracle database application, and we await final installation instructions from Uppsala University. In all other cases, we have installed test instances of each system. We hope to include MDID2 (<http://mdid.org>) in this analysis, especially given that the Visual Resources Collection (VRC) at Johns Hopkins is using this system to support teaching and learning. We have recently signed a service agreement with the VRC to move forward with installation of MDID2. Initially, we had hoped to include Luna Insight as well, but recent feedback from Luna leads us to believe that they are not (yet) prepared to offer access to APIs or source code.

It's worth noting that we are working with a number of e-publishing systems, thereby offering an opportunity to identify the comparative advantage of each system. We would welcome a chance to work with additional learning systems, but the proprietary nature of WebCT and Blackboard (now a merged entity) makes this difficult. The presence of ProQuest provides evidence that vendor-based offerings have a place in this analysis, but we need specific and explicit cooperation from vendors for this analysis. To date, we have not identified a means for engaging WebCT or Blackboard in this manner.

With the methodology for defining functional requirements, and an installed base of systems, the other remaining major "building block" of the analysis is content.

## **Content**

Through existing efforts at Johns Hopkins, and through our partnerships with other institutions, we have identified and collected a variety of content including text, images, audio, journal and proceeding papers, and associated metadata. While we do not own the 9/11 archive, which represented the content for the Archive Ingest Handing Test (AIHT), our experience with this Library of Congress-sponsored digital preservation initiative is included in our analysis. The 9/11 archive contained multiple file formats, and metadata in varying formats and quality.

We will work with the following collections for this analysis:

- Papers from the 2003 ISMIR conference (when Johns Hopkins co-sponsored this conference)
- Images of brittle books from Johns Hopkins' Library collection
- Images from the Fowler architectural collection
- Images from the Visual Resources Collection (primarily Art History images)
- Articles from Project Muse (from Johns Hopkins University Press)
- Electronic theses and dissertations being submitted through a pilot project at Johns Hopkins
- Audio files from Indiana University (through a collaborative IMLS grant)
- (Lessons learned from 9/11 archive ingestion through AIHT)

We have successfully ingested the ISMIR papers, and images of the brittle books, Fowler collection, and Visual Resources Collection. Project Muse has already transferred full runs of a few journals from Hopkins Press (with access restricted to project team members only for the analysis), and we have already started receiving electronic theses and dissertations. As part of a collaborative IMLS grant focused on digital workflow for audio content, Indiana University will transfer audio files to Hopkins. While some of the DAAP content will eventually be publicly accessible, we can make no assumptions about how much of it and which performances that will be generally accessible.

The aforementioned components ranging from scenarios to key events, multiple repositories and applications, and a diversity of content represent the building blocks for this analysis. We have developed a framework to convey the results from our analysis, which was described during presentations at the Spring 2005 CNI Task Force Meeting and the 2005 Spring DLF Forum.

### Matrix Representation of Results

Given the interaction of the various aspects of this analysis, it is important to convey the eventual results in a concise, accessible format. In this regard, we have elected to use a matrix representation. The following images demonstrate a hypothetical set of cases. The first stage depicts the process of deriving functional requirements from use cases or key events. The image below provides an example:

	Use Case 1	Use Case 2	Use Case 3	Use Case 4
Functionality 1	*	*	*	*
Functionality 2		*	*	
Functionality 3	*			
Functionality 4		*	*	*
Functionality 5	*			
Functionality 6		*		
Functionality 7			*	
Functionality 8				*
Functionality 9	*			*
Functionality 10		*	*	*

In this hypothetical case, use case 1 requires functionality 1, 5, and 9. The second stage highlights the repository support for each functionality, as depicted below:

	Fedora 2.0	Dspace 1.2.1	Other Repo	...
Functionality 1	*	*		*
Functionality 2	*		*	
Functionality 3	*			
Functionality 4		*	*	*
Functionality 5	*	*		
Functionality 6	*	*	*	
Functionality 7		*	*	
Functionality 8				
Functionality 9	*	*		
Functionality 10	*	*	*	

Continuing with the hypothetical examples, this diagram expresses that DSpace 1.2.1 supports Functionalities 1, 4-7, and 9-10; Fedora 2.0 supports Functionalities 1-3, 5-6, and 9-10; and some other repositories have support for some of the functionalities, as well.

The next stage outlines the mapping of repository interface specifications to each functionality, as shown below. We are interested in these interfaces to explore repositories working together (e.g., DSpace and Fedora).

	JSR-170 0.16.3	DR OSID 2.0	IMS DRI 1.0	Other API
Functionality 1	*	*		
Functionality 2	*	*		
Functionality 3	*	*		
Functionality 4		*	*	*
Functionality 5	*	*		
Functionality 6	*		*	*
Functionality 7	*		*	
Functionality 8	*	*		
Functionality 9	*	*	*	
Functionality 10	*	*	*	

The above diagram depicts (again, hypothetically) that JSR-170 version 0.16.3 supports 1-3 and 5-10; OKI DR OSID 2.0 supports 1-5 and 8-10.

The final stage brings each of the components together, highlighting potential gaps, as depicted below:

	Use Cases				Repositories			API Specifications			
	UC 1	UC 2	UC 3	UC 4	Fedora	Dspace	Other	JSR-170	DR OSID	IMS DRI	Other
Functionality 1	*	*	*	*	*	*	-	*	*	-	-
Functionality 2		*	*		*	-	*	*	*	-	-
Functionality 3	*				*	-	-	*	*	-	-
Functionality 4		*	*	*	-	*	*	-	*	*	*
Functionality 5	*				*	*	-	*	*	-	-
Functionality 6		*			*	*	*	*	-	*	*
Functionality 7			*		-	*	*	*	-	*	-
Functionality 8				*	-	-	-	*	*	-	-
Functionality 9	*			*	*	*	-	*	*	*	-
Functionality 10		*	*	*	*	*	*	*	*	*	-

The elements identified in red (and the minus sign) express gaps in support for particular functionality through a combination of systems and interface specifications. For example, in this hypothetical depiction, the combination of DSpace and IMS DRI do not support functionality 2, which is required by use cases 2 and 3.

As we move forward and complete our analysis, we will develop such matrices with actual functional specifications, use cases (or key events) and gaps using repositories and applications. The Hopkins project team will focus on a specific set of repositories and applications described earlier, but we hope that other institutions in the digital library community will join us in evaluating other combinations of functionality, repositories, applications and interface specifications.

## **Community Outreach and Collaboration**

The original proposal highlighted the importance of involving MIT and UVA as participants in this analysis, given their prominent roles with DSpace and Fedora, respectively. The Hopkins project team has apprised both institutions regarding this analysis, and directed them to the project wiki. UVA elected to submit scenarios. As indicated in the proposal, we will invite individuals from MIT and UVA to a meeting at Hopkins to discuss the analysis and results. As an example of this working relationship between these institutions, it's worth noting that the Hopkins team has already communicated our findings from the AIHT project with both MIT and UVA.

In addition to this specific outreach and collaboration effort, the Hopkins project team has engaged the broader community in substantive and significant ways. By doing so, this analysis has moved beyond a Hopkins-centric examination to a broader, community-based effort. There has been a great deal of interest from other institutions within the US and UK, which reflects the desire for a rigorous, objective, transparent understanding of repositories and applications. The response from the community regarding scenarios provides ample evidence of this interest.

Through the aforementioned CNI and DLF presentations, the wiki, and extensive communication via email, the Hopkins project team has built a network of collaborators and interested parties. Complementing the US institutions, the JISC Digital Repositories Programme has expressed a serious interest and willingness to collaborate with Hopkins. Specifically, Neil Jacobs from JISC invited Project Director Choudhury to the recent, joint Digital Repositories and Digital Preservation meeting in Glasgow. Hopkins is formally connected to two projects in the JISC Digital Repositories Programme, one of which has adopted our methodology. During the JISC meeting, Choudhury provided a presentation outlining the methodology and definitions from this analysis.

Bill Olivier, Chief Systems Architect from JISC, suggested that the JISC projects consider using our approach to determine how each of their projects fit into the UK E-framework. During a follow-up discussion after the JISC meeting, Choudhury met with the JISC Digital Repositories Programme support team. As a result of this dialogue, JISC has agreed to examine our methodology and taxonomy to determine areas of collaboration and coordination. It's possible

that JISC may choose to work within our framework (e.g., submit their scenarios to our wiki), adopt aspects of our analysis or, at the least, work in parallel with our analysis understanding each others' methodologies. This validation of our approach is most promising and encouraging.

## **Next Steps**

The next steps include a comparison of a use case reference identified by the JISC Digital Repositories Programme support team. Specifically, they are using Cockburn<sup>2</sup> so our usability expert will compare this treatment of use cases to the Kulak and Guiney reference.

We will focus on a deeper consideration of key events so that we might define carefully the functional requirements for repositories. We will map those requirements to the APIs of the repositories and applications. Finally, we can evaluate the repository interface specifications (e.g., JSR-170) for their ability to support moving content between repositories and applications in a manner that supports the needs identified through scenarios. These efforts will illuminate the gaps in the aforementioned matrix framework.

We are eager to move from the building blocks and foundation that has been developed over the first part of this analysis of repositories and applications. It will be very useful to examine specific content moving between repositories and applications in ways that support needs identified through community-identified scenarios. We thank the Mellon Foundation for the opportunity to conduct this important analysis.

---

<sup>2</sup> Cockburn, A. (2000). *Writing Effective Use Cases*. Boston: Addison-Wesley.