Preserving State Government Digital Information
Minnesota Historical Society

Final Report on Tessella's Safety Deposit Box

As part of Minnesota's National Digital Information Infrastructure and Preservation Program (NDIIPP) project, "A Model Technological and Social Architecture for the Preservation of State Government Digital Information", the Minnesota Historical Society (MHS) worked with Tessella on a pilot project testing Tessella's digital preservation repository, the Safety Deposit Box (SDB). This report is written by the Minnesota Historical Society as a summary of the testing experiences of Minnesota, Illinois, Tennessee, and Vermont.

Any comments, corrections, or recommendations may be sent to the project team, care of:

Carol Kussmann
Collections Assistant, State Archives
Minnesota Historical Society
carol.kussmann@mnhs.org / 651.259.3262

Summary

As part of Minnesota's National Digital Information Infrastructure and Preservation Program (NDIIPP) project, "A Model Technological and Social Architecture for the Preservation of State Government Digital Information", the Minnesota Historical Society (MHS) worked with Tessella on a pilot project testing Tessella's digital preservation repository, the Safety Deposit Box (SDB).

The goal of this pilot was to provide MHS and other partner institutions with the opportunity to gain experience using a digital archiving solution that provided a desired set of functionality, business processes and policy for each participant in a shared services capacity. Participants included partners from Illinois, Minnesota, Tennessee and Vermont.[1]

Main repository functions tested included how files were ingested, stored, managed, preserved, and exported. The trustworthiness and ability to produce authentic records from within SDB was also tested. In addition to gaining practical hands on experience on how SDB approached basic repository functions, each state hoped to be able to address goals and objectives specific to their needs. Due to time, budget, and project scope, some but not all of these additional objectives were explored in the detail hoped. Cost considerations aside, the system met all general functional requirements of a digital preservation repository. These included overall system

[1] Partners included representatives from the Illinois State Library, Minnesota Historical Society, Tennessee State Library and Archive, the Vermont State Archives and Records Administration, and the Vermont Enterprise Project Management Office.

trustworthiness, as well as the basic repository functions of ingest, storage, preservation and management.

SDB General Architecture

SDB is a web based digital archiving solution that uses micro-services to increase the extensibility of the system. Micro-services address preservation, storage, access, reporting, and authenticity of digital materials as well as the integrity and scalability of the system. SDB uses the basic concepts of digital objects, information objects, and structural, descriptive and technical metadata to manage the archive.

Various deployments of SDB exist. An out-of-the-box solution includes core components with default policies and basic workflows. Customized solutions allow for individualized user interfaces, workflows, policies and integrations with other systems. A multi-tenancy instance supports multiple tenants who share a common infrastructure and core services but only have access to their own content, workflows, policy, storage, and branding.

This pilot project used a Tessella hosted, multi-tenancy instance of SDB based on a pre 4.1 release version with limited customizations.

Description of SDB Functional Entities

As a system, SDB uses portals to move between main system functions. These portals are based on the functional entities of the Open Archival Information System (OAIS) reference model[2].

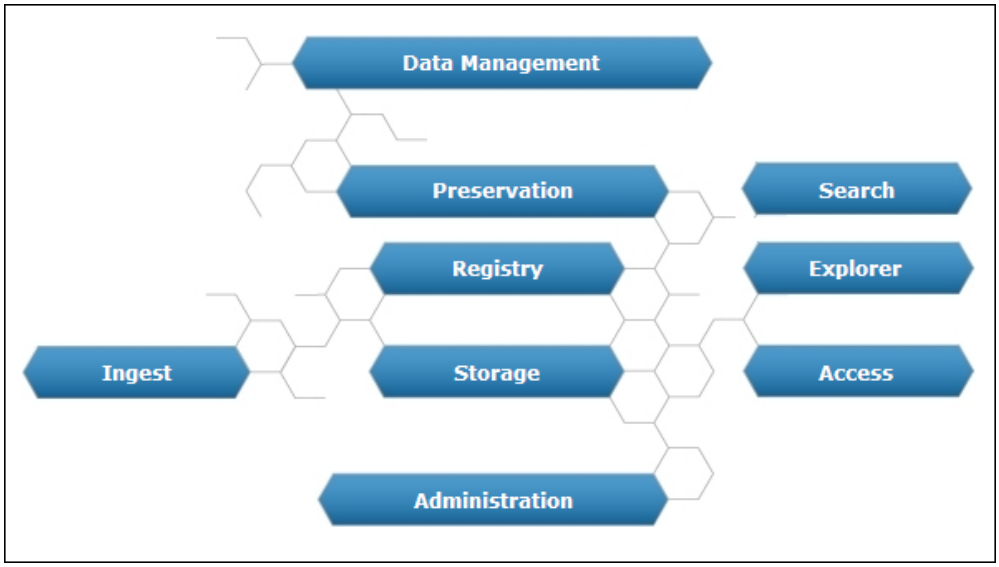


Figure 1: Portals on the Safety Deposit Box Home Page

Portals address the following functions as follows:

[2] Consultative Committee for Space Data Systems. *Reference Model for an Open Archival Information System (OAIS).* January 2002. http://public.ccsds.org/publications/archive/650x0b1.PDF

Ingest: Content is uploaded into SDB from this portal using ingest workflows.

Data Management: Includes tools to manage archive content, including deletion, automated reporting, and editing of metadata, as well as how SDB interfaces with other systems. *[Many of these functions parallel functions under Access. The Data Management portal was not used directly during this pilot.]*

Preservation: To keep content accessible, files can be migrated from one format to another using preservation workflows from this workflow.

Registry: The registry is a resource database of file format information (fmt/) and transformation pathways (pth/) that might be used within SDB. The registry provides the policy for preservation functions. *[In the pilot's multi-tenancy instance, this portal was not customized but shared across all instances.]*

Storage: This portal connects the system to storage adaptors, and shows information about the type of storage being used for system content. Storage adaptors may include local disks, network attached disks, and Amazon S3 cloud storage as well as other storage systems. This portal is also where fixity checking settings are chosen *[passive preservation]*.

Administration: System processes are administered from this portal. System configuration information is also found here. User security settings are controlled from here. It is also where you add/delete/modify tools that are available in the SDB instance. For example, before you can use workflows, reports, or metadata schemas, they must be uploaded into the system from this portal. This is the main entry point for all of the 'behind the scenes' actions or functions.

Search: A Google-like search portal that uses keyword searching to return a list of results in a web browser. *[Beta version added during pilot test.]*

Explorer: As the main access area into the archive, this portal is used to search, browse, view, modify, learn about and explore content within the archive. Content is displayed in a folder structure made up of collections, deliverable units, and individual files. Detailed information can be found on the collection, deliverable unit, or file level. Content can also be exported or downloaded from here.

Access: The access portal contains workflows that allow and control how content can be exported and downloaded from the archive. However, these actions are generally initiated from within the Explorer portal, not the Access portal.

Testing of Repository Functions

[This report includes experiences and comments from all participants, however the majority represents Minnesota's testing experience, as Minnesota had more resources available for testing.]

Ingest: Minnesota was interested in testing how files can be ingested into the system on an individual file basis, as well as in bulk. All participants wanted to ingest a variety of file formats to see how SDB handled diverse formats.

Before content could be ingested, a submission information packet (SIP) needed to be created. The information packet (SIP) includes desired content, a file of metadata describing the content, and a protocol file with general information about the SIP. The metadata is what SDB uses to create collections, deliverable units, and files within the system; the protocol file is a system trigger that contains basic information about the ingest itself.

Participants used the SIP Creator, a tool developed by Tessella, to create SIPs. The SIP Creator generated the SDB required metadata. At times the creation of SIPs was straightforward and easy, while in other instances it was time consuming. In a fully implemented instance of SDB, the creation of SIPs would most likely be part of a larger automated process, reducing the amount of time needed.

[View Appendix A: SIP Creator for more information on the SIP Creator. Page 20.]

Files contained within SIPs are ingested into SDB by using ingest workflows. Out-of-the-box workflow definitions created by Tessella were available for use and allowed ingests of individual folders or a collection of nested folders. Ingests of both large and small collections were tested using these workflows. Small submissions could be pulled directly from the user's computer, while larger submissions required the use of File Transfer Protocol (FTP) to move files to the appropriate location. Ingests of large files took longer than expected and required additional steps because the SDB server for the pilot was located in the United Kingdom. A fully implemented system would use its own server or storage system and the process would be more automated.

Two of the customized workflows that were created for the pilot are described below:

Web Harvest: A workflow was created to ingest web content of a selected web address. Participants could select a single web address, indicate how much of the site to crawl, and select a format (WARC, ARC, HTML) to ingest the content. Both Tennessee and Minnesota, who had previous experience with other web harvesting systems, tested the beta version of this workflow.

While content was captured and ingested into SDB, understanding the crawl configuration settings was more difficult than it should have been. The crawl itself was more limiting than other systems, as only one address could be used, and the only settings to choose were the number of links to follow and how deep

the crawls should go into a page hierarchy. Complete web archiving systems, whose main focus is web archiving, allow for further customizations and specifications.

It was found that small crawls were more successful. It was also learned that the mirrored crawl workflows failed when web addresses contained unsupported characters such as ':', '%', or ';'. Content could be ingested as .WARC, .ARC, or HTML files, however no matter which format was used, after content was captured and ingested it was difficult to know what was captured. This was not a surprise as SDB is designed as a preservation repository, not a file viewer or access mechanism. However, understanding the results of a crawl is often a visual experience. Some participants found that the lack of a WARC file viewer was psychologically difficult to overcome. These and other comments on user experience with this beta workflow are being used to improve the workflow for others to use in future releases.[3]

***Ingest with Preservation**:* Out-of-the-box, SDB could ingest content and then run preservation actions on said content. A workflow definition that combined these two processes was created for this pilot. Unlike the preservation action workflow run within the Preservation Portal, these new workflows created from the basic workflow definition could be re-used.

A separate workflow was created from the basic workflow definition for each transformation desired. For example, separate transformation workflows were set up for moving 1) .doc to .pdf, 2) .wav to .mp3, and 3) .jpg to .pdf. If desired, it was possible to create a workflow that addressed multiple transformations at once.

[View Appendix B: Ingest Workflows for examples of a basic ingest workflow, the web crawl, and ingest with preservation workflow. Page 21.]

Participants ingested a variety of file formats into SDB. Formats included iterations of: .pdf, .doc, .xml, .wav, .mp3, .jpg, .tiff, .html, .gif, .png, .css, .shape files, warc files, and more.

All formats tested were accepted into the system, but due to the limitations of available tools, not all formats could be uniquely identified by SDB. SDB uses DRIOD[4] and JHOVE[5] to try to confirm the identity of ingested files. Although a file may have an extension, SDB uses the above tools to examine the internal properties of the file to confirm the format identity.

---

[3] Release 4.1 now contains the ability to view a list of pages contained in a captured WARC file. This release also displays thumbnails of the files and or html pages if they have been characterized. It is expected that release 4.2 will include the ability to search and view crawled content. *[Conversation with Tessella November 1, 2011.]*
[4] Sourceforge. *DROID: Digital Record Object Identification.* http://droid.sourceforge.net/
[5] JSTORE and the President and Fellows of Harvard College. *JHOVE: JSTOR/Harvard Object Validation Environment*. 2009. http://hul.harvard.edu/jhove/ [JHOVE2 is currently being developed.]

A file whose unique identity could not be confirmed will still be stored, accessible, and preserved in its current format within SDB. However it will not be able to have any preservation actions performed on it (other than bit level integrity checking) until the file has been identified.

To characterize unidentified files, it is important to update DROID and JHOVE in SDB as updates are made to these tools. After updates are made, workflows can be written to find previously ingested unidentified files and re-characterize them. Overtime, this should reduce the number of unidentified files in the repository

[View Appendix C: DROID to see how DROID identifies file formats. Page 27.]

Storage System: The storage system used during the pilot was a Tessella hosted server housed in the United Kingdom. While this was not the ideal situation, participants were able to explore how the integrity of files was addressed and how access to the system was controlled.

SDB is based on the Open Archival Information System (OAIS) reference model (ISO standard 14721:2003)[6] and uses these standards as a framework. The storage portal manages the storage adaptors for the repository. Multiple adaptors can be utilized; these include but are not limited to local disks, network attached disks, or cloud services.

Storage integrity for each adaptor is also managed under the Storage System tab. Fixity checks can be run on all files, bringing attention to any files that may have been corrupted. Administrators can set the number of files to have fixity checks run on, each time the workflow is run. The integrity check workflow can also be scheduled on a yearly, monthly, weekly or daily schedule.

The Administration portal contains the role-based access controls. For example, an "Access User" is allowed to read metadata and read content, while an "Admin User" is allowed to read and update metadata, read and insert content, delete entities, and change certain parameters. These controls make sure content is protected from unauthorized access.

Reporting: Going hand in hand with a trustworthy system, it is important to be able to know what is in your system at any given time, who has accessed the system, and what actions have been taken on the content. Reporting tools are available in SDB.

Minnesota spent time evaluating SDB's out-of-the box reports. These reports covered some important issues, but were not always easy to read and understand. Many of the reports used internal codes generated by SDB that referenced collections and deliverable units. Reports covered topics on user activity (log in/out), file formats/details, list of ingested files, audit trails, submissions ingested, and more.

[6] Sharpe, Robert. *Safety Deposit Box: Technical Description*. Tessella. November 11, 2010. Issue: V1.R6.M0.

After evaluating the initial reports, Minnesota discussed with Tessella what modifications would make the reports more useful. New reports were also suggested. Tessella concentrated on making functional, rather than esthetic, modifications to reports as requested. Tessella also created new reports that addressed inventory and fixity concerns.

[View Appendix D: Reports for sample reports. Page 28.]

Tessella uses Jasper Reports[7] to create the SDB reports. Tessella explained the process of how these reports were created and modified to participants. To be most effective, report creators must have a deep understanding of the relationships between fields and the underlying data structure. Report workflows use this data structure to query fields to populate requested reports.

However, for security purposes, the database itself is not queried. A separate database 'report view' must be created to sit on top of the database for the report to query. This view is read-only and pulls necessary fields from the database. Because the pilot used a multi-tenancy system, tenant specific views were created on top of the report view. When a report is run, queries are made to the read-only tenant specific view of the database tables. This ensures that tenant specific content is reported on.

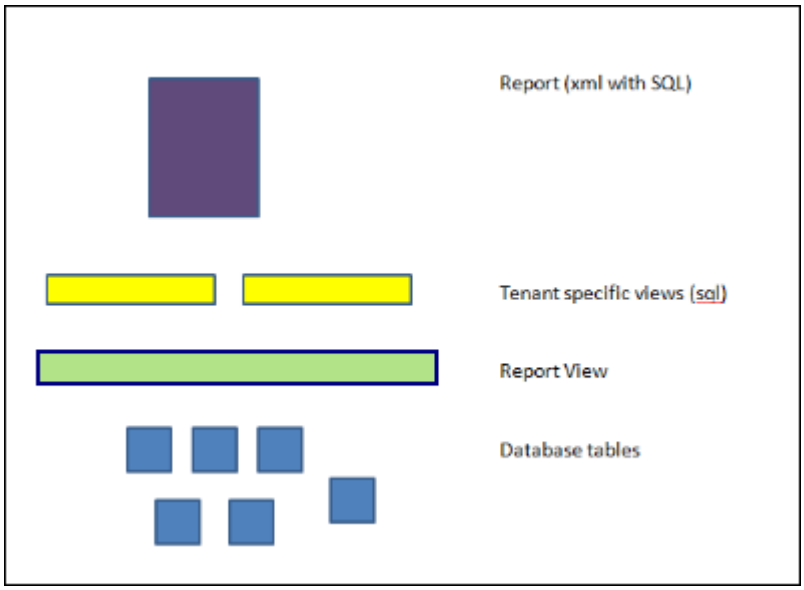


Figure 2: Data structure of how reports are built

Description: Descriptive metadata allows content to be found and provides information about an object. Many metadata 'standards' are in use by different organizations and each record type coming into an archive may use a different standard. These differences often create barriers to

[7] http://jasperforge.org/projects/jasperreports

digital preservation as many repository systems have their own metadata requirements for ingest. Minnesota wanted to learn more about SDB's requirements for metadata.

The SIP Creator populates its own XIP metadata schema that captures and manages descriptive, structure, technical, and administrative metadata about digital objects ingested into SDB.

In addition, descriptive metadata conforming to any schema can be provided by the users but is not required. The only requirement is that the provided metadata be well-formed XML; it must use both opening and closing tags. Descriptive metadata can be attached to content before or after ingest and in a variety of ways.

Before Ingest
If metadata describing content is contained in an XML file, the SIP creator allows users to point to this file when creating a SIP. Upon SIP creation, the SIP Creator then pulls this file out and associates it with the content by adding it to the metadata created by the SIP Creator.

Another method would be to create a SIP, but before completing the packaging step, to cut and paste the metadata into associated fields. Metadata can be associated at the collection, deliverable unit, or file level. This metadata is then placed with the metadata created by the SIP Creator.

After Ingest
After content is ingested, metadata can be associated to a collection or deliverable unit. This can be done by cutting and pasting the well formed XML metadata into a text box associated with the desired content.

Another method is to add metadata using a metadata template. Templates can conform to metadata standards such as Dublin Core or METS, but they could also be uniquely designed to meet your specific requirements or needs.

Any metadata provided using these methods will be indexed within the system, and therefore available for discovery.

In some of the testing that we performed, descriptive metadata was contained within the documents being ingested. In this case, the metadata was not pulled out and associated with the SDB metadata, i.e. it was treated as content. The information (metadata) however could still be used for discovery as XML files themselves are indexed and discoverable.

Metadata schemas can be registered with SDB. Once a metadata schema is registered, and metadata that is ingested and conforms to a registered schema, it can be validated accordingly.

In addition metadata transformations from one schema to another (using XSLT) can be registered in SDB. This provides the ability to create metadata transformations inside of the explorer interface. It is then possible to generate a workflow that will transform metadata form one scheme to another and make it available for download or export.

Preservation Actions: The main purpose of a digital preservation repository is to preserve electronic content and its context and provenance. Preservation can be a passive or active process, SDB utilizes both. Participants wanted to learn more about how SDB used passive preservation and test the active preservation methods.

Passive Preservation

As previously discussed, integrity checking can be set up to run on all files in the system. This is passive preservation. Performing fixity checks allows a system to monitor itself and verify that what is in the system has not changed.

Fixity can be set to be checked on a regular basis. Files are checked based on the length of time between checks; files with a longer date between checks are verified first. Users set the number of files to be checked at a time and the workflow can be turned on and off. The number of files to check at a time is up to the user, and the schedule can be daily (choose a time), weekly, monthly, or yearly. If any errors are discovered, an email will be sent to the email address listed in the fixity workflow definition.

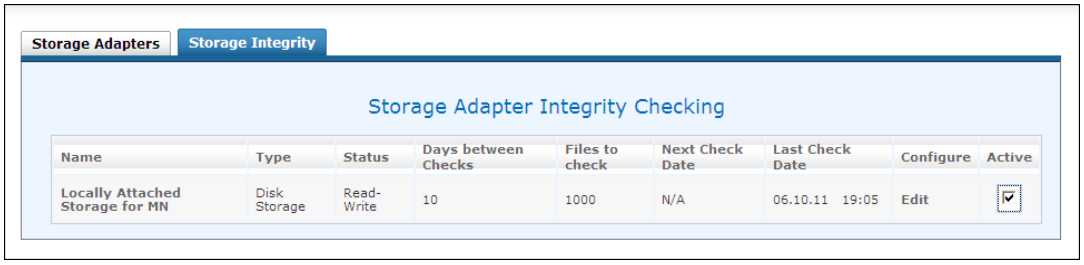


Figure 3: Example of a fixity checking settings

In addition to controlling the variables related to continuous integrity checking on files, it would be helpful to be able to select particular collections, deliverable units, or files to check on demand.

Active Preservation

Active preservation is addressed by SDB by using file format migrations that transforms 'at risk' or 'out of date' file formats to current or more desirable formats. This should keep files from becoming obsolete. It is however, up to the users and administrators of the system to determine what file types should be migrated. As a system, SDB is unique because of its ability to perform transformations on various file formats.

Preservation actions can be done upon ingest or on demand once records are in the system. If actions are to be done after items are ingested in the system, they are done from the Preservation portal.

Transformations can be used to assist with preservation (moving records from an outdated file format to a better supported format) or can also be used to create another 'copy' or 'version' of a file for presentation purposes (moving large files to smaller files that would be more appropriate as access or use copies). Internally, the only difference between a preservation action and a presentation action is how the original manifestation is treated after the transformation. With a presentation action, the original manifestation is kept in an active state, therefore available for additional preservation actions. With a preservation action, the original manifestation is moved to an inactive state and will not be available for future preservation actions.[8]

Unlike with the Ingest and Preservation workflows (discussed under Ingest), preservation actions done from the Preservation portal cannot currently be reused. This can and should be modified as most preservation actions will need to be run multiple times as new material is ingested. Unless addressed in acceptance policies, archives will continue to receive out of date files and the preservation repository will need to be monitored for files that are considered to be at risk formats.

The system, if set up to do so, could also allow users to search the archive for 'at risk' files based on a risk threshold. Administrators would need to evaluate each file format within the archive through the Registry to determine risk threshold numbers.

Participants in this pilot did not use the risk threshold method but tested individual transformations. Some of these included moving from .doc to .pdf, .jpg to .pdf, .wav to .mp3, and .wmv to .mov. Transformations involving Open Office documents were tested but failed because the tools, although listed as being in the system, were not deployed in the pilot instance. Time and efforts were spent making available and testing tools for audio migrations; specifically the FFMPEG tool was wrapped and deployed.

The process of creating and running a preservation workflow includes a lot of steps. It is difficult to go back and see what you did before as the file format that you performed the action on, is not recorded on an initial screen. It can be found but it is buried and not in a logical place.

[View Appendix E: Preservation Plan to follow the process of a transformation from within the Preservation portal. Page 32.]

[View Appendix B, Section 3 for information on the manual ingest with preservation workflow. Page 24.]

Export: It is just as important to be able to get things out of a repository as it is getting them in. It is your content and you should be able to do what you want with it, including downloading it or removing it. Participants wanted to see how this worked in SDB.

[8] It is important to understand these details as they may help or hinder your processes. Release 4.1 has restructured the 'labeling' of manifestations, but the details of these changes have not been discussed.

SDB allows content to be exported or downloaded as individual files or at the dissemination information package (DIP) level. Content could not be exported at the collection level during this pilot. However, it is understood that a workflow could be written to allow collections to be exported.

Knowing how files will be used will assist you in knowing how to organize your archive and determine at what level you will need content to be downloadable. Customizations can be made to fulfill your requirements. If, as in our situation with the pilot, you choose to upload duplicate content to satisfy different methods for download or another reason, SDB can be set up to store only one copy of the content which eliminates duplicate storage and saves overall space in the archive.

Search and Discovery: After content is ingested into a repository, it is important to be able to find it again. Participants explored how SDB accomplished this.

Some digital repositories, in addition to archival functions, focus their efforts on 'public' access. SDB is primarily focused on preservation needs, but provides the ability to build a comprehensive public access interface using exposed search APIs. *(See Implementations with Other Systems below)*. SDB's search and discovery methods were designed to be used by people familiar with the archive's structure and content. Participants understood this but were concerned that it would be difficult to demonstrate the system to people who were not familiar with the archive structure, i.e. people who may control funding. Tessella developed a beta version of the Search portal to show that a non-technical search tool could be designed to address specific needs.

Both methods use the same underlying principles of keyword searching. All metadata, both system and user generated, is indexed as well as the content of XML, PDF, Word, and other documents; indexed content is discoverable. Keywords found in any of these locations can be discovered. Wild cards can also be used in the searches. The system uses Solr Lucene for indexing and query functionality within SDB.

While underlying search methods are the same in these two search methods, results are returned differently. The 'public' search is done before you are in the 'archive' or 'explorer' view. Results are shown as a list and allow for immediate downloads or previews of some types of files, or link back into SDB explorer view for more details. The public search is currently strictly a keyword search while the 'archive' search also allows the use of filters based on Tessella's metadata (the XIP schema), multiple filters can be added. 'Archive' search results are shown within a window in the Explorer with some information about each result. Clicking on the result brings you to the result within the archive.

Both methods are being worked on as testing found inconsistencies and room for improvement.

[View Appendix F: Search for screenshots of the two methods of search. Page 38.]

Implementation with Other Systems: As indicated in the Search and Discovery section, SDB was designed so it could be used with other systems. This was a functionality that Illinois and Vermont wanted to test.

Illinois, understanding that it would be easy to copy files from one system and ingest them into SDB, Illinois wanted to see how SDB could be placed into existing workflows. Illinois explored a few different options of working with ContentDM and with Illinois' Digital Archive (IDA).

Experiment 1: Working with ContentDM, Illinois used OAI::PMH to retrieve a file with essential metadata about an object. This file was not a complete set of metadata and did not contain the digital object itself. Until OCLC properly implements the OAI::ORE protocol for retrieval of digital objects, OAI is not a viable option for building submission information packages (SIPS).

Experiment 2: ContentDM's native export facilities were also looked at. A collection of records can be exported into a tab delimited file or an XML file. Objects could be retrieved using Wget. This method was found to be useful for transport, but lacked reference to the described digital object.

Experiment 3: OCLC's APIs were also used to access the collection. Retrieval can be presented as a JSON or an XML record. While powerful, the APIs do not return how many records a collection has or the starting and ending points.

From these experiments, Illinois created two solutions for automated retrieval of ContentDM records that could be used to build SIPs. For specifics, please review Illinois' preliminary report.[9]

To create a SIP from content already housed in the Electronic Documents of Illinois (EDI), Illinois' online electronic records repository, a Perl program was used to walk the directory tree. This Perl program created a SIP that consisted of the deposited digital object(s) as well as metadata derived from existing deposited metadata. The SIPs were created in such a way that they could be used for either deposit in dSpace or in Tessella's SDB. Further, the process would easily allow for the creation of a batch ingest file for upload into an existing ContentDM collection. Adding this SIP package creation method to EDI's existing workflow would be more straightforward than the roundabout method of walking and parsing a directory tree, as it could be added as an automated workflow step between Illinois State Library's cataloging of the record and its final disposition into the EDI system.

[9] Bullen, Andrew. *Illinois and SDB: A Preliminary Report*. September 14, 2011. Available on the NDIIPP project website: http://www.mnhs.org/preserve/records/legislativerecords/Tessella.htm

Vermont was interested in integrating SDB into enterprise wide practices, specifically with SharePoint. Due to time and budget constraints, Vermont was not able explore how SDB and SharePoint would work together. Tessella is aware of the desire of Vermont and other agencies to be able to integrate SDB with these larger systems. It is hoped that future implementations of SDB will be able to be integrated with such systems.

Documentation: To be able to use and understand a system, it is important to have findable and useable documentation. Tessella provided participants with documentation that covered various features on SDB, access to the SDB user group, and conducted a day and a half in-state training session. Minnesota also created user guides for specific processes used during the pilot.

Before working with SDB, Minnesota participants read the documentation. It provided a good overview of the system functions and was not overly technical. It was however out of date. Some of the examples were very specific and could not be followed easily. Minnesota provided Tessella with suggestions in places where documentation was unclear or outdated. Tessella is working on updating their documentation.

During the testing phase, Minnesota consulted Tessella's documentation in a limited manner. Minnesota staff had participated in multiple user trainings and was familiar with the basics of the system. When questions arose, Minnesota staff worked one-on-one with Tessella to get them answered. Troubleshooting was done via email, phone, and screen sharing.

Illinois staff reviewed and tried to make use of Tessella-supplied technical documentation in order to experiment with direct depositing of records into the SDB system using APIs. Other than as formal descriptions of API functionality, the documentation was found to be too technical and academic for practical use. Instead, Illinois staff used javadoc documentation to puzzle out the workings of the APIs. Illinois feels that there is a considerable gap between the language of Tessella-supplied documentation and documentation expected and useful to the primary consumers of the SDB product—librarians and archivists.

Compared to the other state participants, Vermont approached testing and evaluation in a different way. Their evaluation of SDB was performed according to the State of Vermont's standard information technology (IT) procurement process and focused on steps associated with user testing and user acceptance. This meant that Vermont opted to use the system documentation provided during the course of the project to resolve problems as opposed to relying on support from Minnesota and Tessella staff. Vermont used training slides, the SDB 4 Standard Workflows V1.R6.M1 document, materials posted on Tessella's website[10] [2], threads on the SDB LinkedIn User Group, and "Help" sections within SDB itself.

[10] http://www.digital-preservation.com

System documentation, such as the SDB 4 Standard Workflows V1.R6.M1, was the most detailed system documentation originally provided to the NDIIPP project team. It walks the user through different workflows with SDB and explains the different workflow steps. While errors are discussed, they describe why the errors occurred, from a technical standpoint, but not how to resolve the issue, from a user standpoint.

Likewise, when errors occurred within SDB, the displayed error message stated what went wrong, however there was no additional internal guidance for the user (i.e. what the user did wrong that caused the error) or information on how to resolve the error.

Some of the documentation provided did include information typically found in user or system administrator manuals; however, Vermont struggled to resolve problems using the existing documentation and hence adequately test and evaluate the system.

Figure 4: Internal SDB message. This message shows ID codes (that may or may not be understandable to users), and a description about the error. This description does not tell the user ***how*** to solve the problem, just want the error is related to.

Participants were not only interested in documentation that provided background information; they wanted to understand *how* to do things. "How To" guides were created during the project for specific processes/workflows, but not for trouble shooting or problem solving issues that arose during testing. It was felt that trouble-shooting guides would assist users greatly, as very few people will have a deep understanding of the ins and outs of a system like SDB.

Documentation could be written to address both audiences, however "How To" guides need to follow instance specific customizations. This may make it difficult for Tessella to produce "How To" guides. Tessella should consider creating a basic template for individual workflows that Tessella or users could modify to fit their customization.

Suggestions/General Comments by Function

In addition to gaining hands-on experience with a digital repository, participants provided Tessella with feedback on SDB. Feedback was given on overall user experience, general

usability, and specific processes. Tessella is using this feedback to improve future releases of SDB.

Ingest:
Many of the suggestions made to Tessella about the Ingest portal had to do with ease of use. Some of the field titles were not easy to understand – they could be interpreted by different audiences in different ways.

Consistency was also an issue – clicking on a workflow name should always do the same thing. In some cases, it described the workflow, while other times it continued the workflow running. These are two very separate tasks.

The system keeps track of all completed workflows. This list can be sorted by name, collection code, date, and other fields. These results can also be filtered by the workflow context (really the workflow definition) and date. It would be useful to also filter the results by user if a system expects to have a large number of users performing these functions.

Ingest Process:
The SIP Creator was used manually for creating submission information packets for this project. Problems arose when many people on several different computers, were creating SIPs. The SIP Creator, as currently designed, runs off a single workstation. This means that if one user creates a collection from their computer and would like someone else to add to the collection, the same computer must be used. The SIP Creator only remembers what collections were created from each workstation. All collections created from each particular workstation, even if no longer in SDB, would appear in the "existing collection" list. If a collection that no longer existed in SDB was selected, the submission would fail. It would be more useful if the SIP Creator would query the SDB to determine what collections were in the system. This would eliminate failed submissions for this reason and allow multiple users from multiple locations to add to the same collection. This may not be a problem with automated processes, but it was a stumbling block during the pilot.

Reporting:
As previously indicated, the reports were difficult to understand when they referenced internal codes. In general, reports should use more descriptive elements as demonstrated with the modifications made for the pilot.

Currently reports can be downloaded as a PDF, CSV, or XML document. (The pilot focused on the PDF versions.) When a report is produced on demand or emailed as part of a workflow, the file created is called "Report[1].pdf". It would be more useful to have the file name represent the report itself as well as when it was created. The file name could be generated from the report title and the date/time the report was run.

As designed, reports can be run on a schedule and emailed to an individual. It is suggested that being able to email a report could also be done on an on-demand basis. If working on demand, users should be able to select, each time, to whom a report should be emailed to. Often multiple users would find the reports useful. Currently only one individual can be emailed a scheduled report. If (like with the workflows) only one email address receives the reports, then there is a loss of use and value.

Description/Metadata:
Testing of the metadata schema transformations brought about more questions regarding system capabilities. More time and attention should be spent understanding and utilizing these features. These results should be documented.

Preservation Actions:
Preservation plans should be reusable. Currently a lot of time is spent recreating duplicate preservation plans.

Fixity checking should be able to run on demand. It would be useful to be able to check a collection or deliverable unit on-demand. Fixity is currently run on objects before they are exported as DIPs.

Export:
Export functions are very dependent on the structure of the archive. Users should be aware of this.

It would be useful to be able to export an entire collection, deliverable unit, or even the archive. In this pilot, only files and deliverable units can be exported. Being able to export an entire collection or entire archive would increase the flexibility of the system. Workflows could be written to achieve both of these.

Search and Discovery:
The search function within the Explorer portal allows users to search by keyword and limit results by SDB's metadata fields. The information returned is not labeled correctly. The Description field is not the description but the Collection Code for the collection. A more flexible search interface would be useful.

The newly created Search Portal was implemented to create a Google-like search experience. User testing found some inconstancies in terminology. For example, the Collection Code is being used as the Collection Title. *[As a comparison, the internal search function used the Collection Code as the description.]*

Within the search portal, the links to move from the search results into a file in the archive does not work. Users are always brought to the home page of the archive, not to the desired file. Users must be brought to the desired file directly for the search to be functional.

Additional testing should be done to make things more consistent between the two search functions if both will continued to be provided. When performing a search, the results should be the same. It is OK to be in a different structure, but some consistency is needed.

Implementation with Other System:
Potential users are very interested in how SDB can work with other systems. It would be useful to have case studies on how this has successfully worked for others. It would also be valuable to document how these processes were completed.

Documentation:
As previously indicated documentation needs to be updated and kept up to date. Documentation should be able to be understood by a variety of audiences. Templates of how to guides might be useful.

Requirements and Drivers for Digital Preservation

During the Pilot Project Wrap Up meeting, participants spent some time discussing general requirements of a preservation system as well as what type of environment is necessary for a successful digital preservation. The following requirements and drivers were selected from that discussion.

System

Information access: Content must be accessible over time and readily available. .

Enhance value of content: Over and above access, if the value of content can be increased the more valuable the system becomes.

Integrated RIM: A preservation system that can be integrated with other records management systems will be easier to adopt and fit into current workflows.

Ease of use: A system must have a low threshold for use and participation. If a system has too many barriers to entry, users will be frustrated and not use the system as intended.

Ability to build on existing systems: A preservation system that can be added on to existing systems lowers the technological barrier to entry and increases the flexibility of the system.

Environment

Governance: There must be a strategic plan for preservation. Information management needs to have a framework.

Cost allocation strategies: The first question is can we afford this followed by is this sustainable. However staff time is also a cost. If an institution has IT staff, managing this system may better fall under their expertise, rather than an archivist's.

Learn from past experiences: What has worked in the past? What hasn't? Why and why not? Low hanging fruit can be used as success stories or tests for pilot projects from which to learn from.

Cooperation and collaboration: Often this cannot be done alone. At what level can you cooperate and collaborate? As similar organizations? On a state level? With others that you share common goals and requirements?

IT consolidation: Many states are moving towards consolidating IT services across agencies or organizations. This trend allows for cost sharing savings and sharing of common goals.

Each state has their own drivers; currently Vermont is most concerned with integration while Minnesota is focused on authentication. Know and use these drivers to determine how to move forward. Identify key stakeholders. For long-term solutions, organizations need to move away from project based functions to program based solutions.

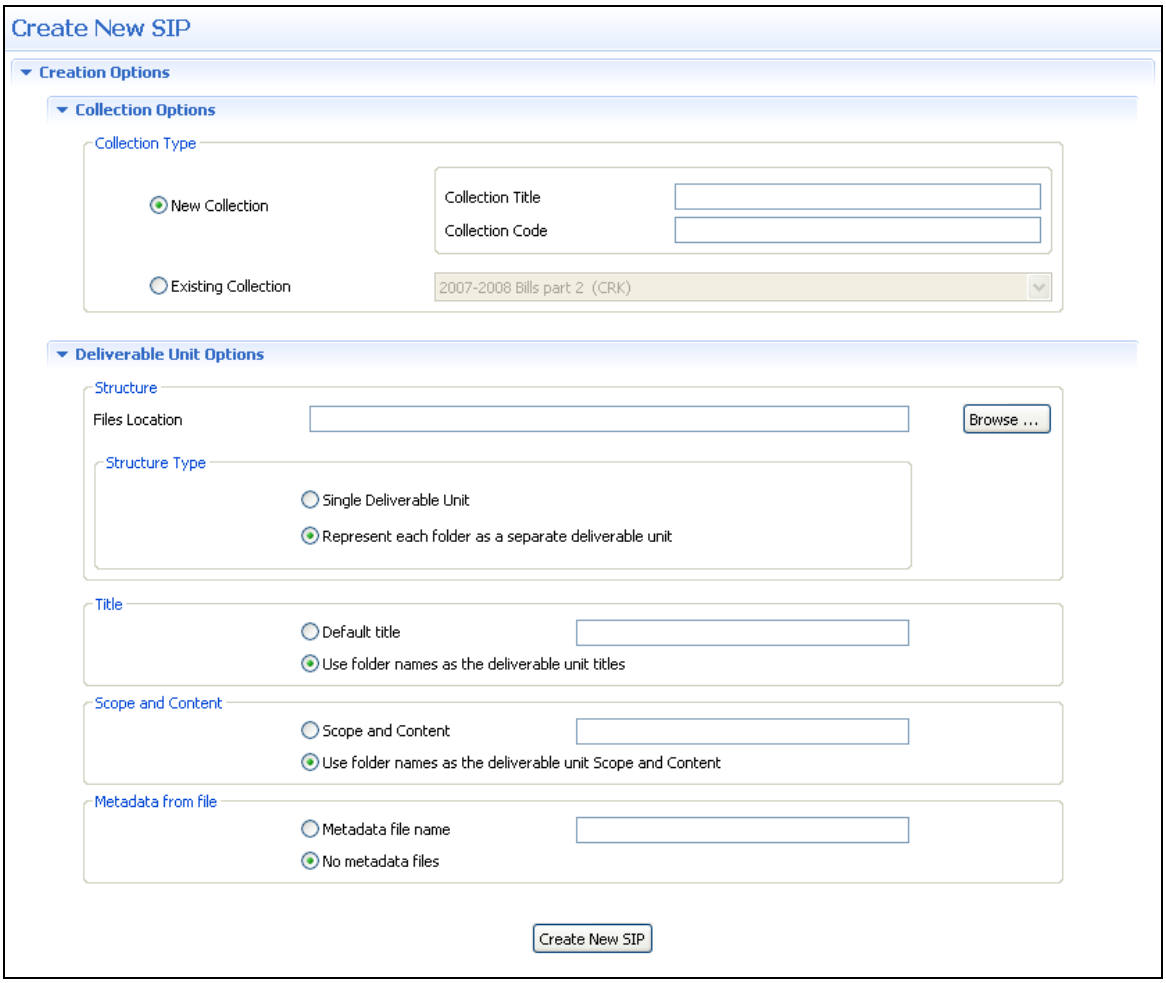
APPENDICES:

The following appendices describe testing processes and experiences in more detail.

Appendix A: SIP Creator…….….Page 20
Appendix B: Ingest workflows…….Page 21
Appendix C: Droid format identifier……...….Page 27
Appendix D: Reports……...….Page 28
Appendix E: Preservation plan…….Page 32
Appendix F: Search methods……..…….Page 38

Appendix A: SIP Creator

The SIP Creator allows users to create a new collection or add to an existing collection. If adding a new collection users needed to populate the Collection Title and Collection Code fields. If adding to an existing collection, a collection was selected from the drop down list.[11]



Users then needed to indicate the location of the folder they wanted to ingest and how they wanted to structure the content contained in the folder. This determined how the collections were organized within the archive. Titles could either be the folder names or a selected default title; scope and content could also be folder names or a selected words; and if a metadata was located in the folder to be uploaded it could be indicated here. This metadata file would then be pulled out and added to the SIP Creator's system metadata. Metadata could also be added in by cutting and pasting XML before the SIP was finalized.

After a SIP was finalized, it could be ingested into SDB using the desired workflow.

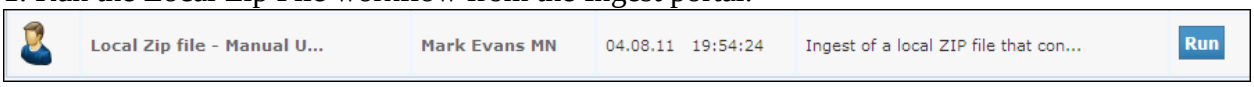
[11] This drop down list was the source of many problems. The list is not populated from the SDB itself, but from a local computer. So if another user created a collection you wanted to add to, you would have to use the same computer. In addition, some of the collections were deleted from SDB but still appeared in this list. If you used something that did not exist, the ingest workflow would fail. Modifications are being made to solve this.

Appendix B: Ingest Workflows

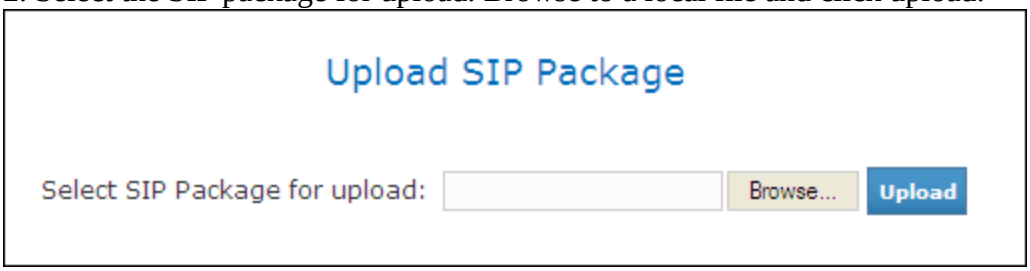
Below you will find examples of three different ingest workflows used during the pilot test. A basic ingest, a web harvest, and ingest combined with a preservation action.

Example 1: Basic Ingest – Local Zip File

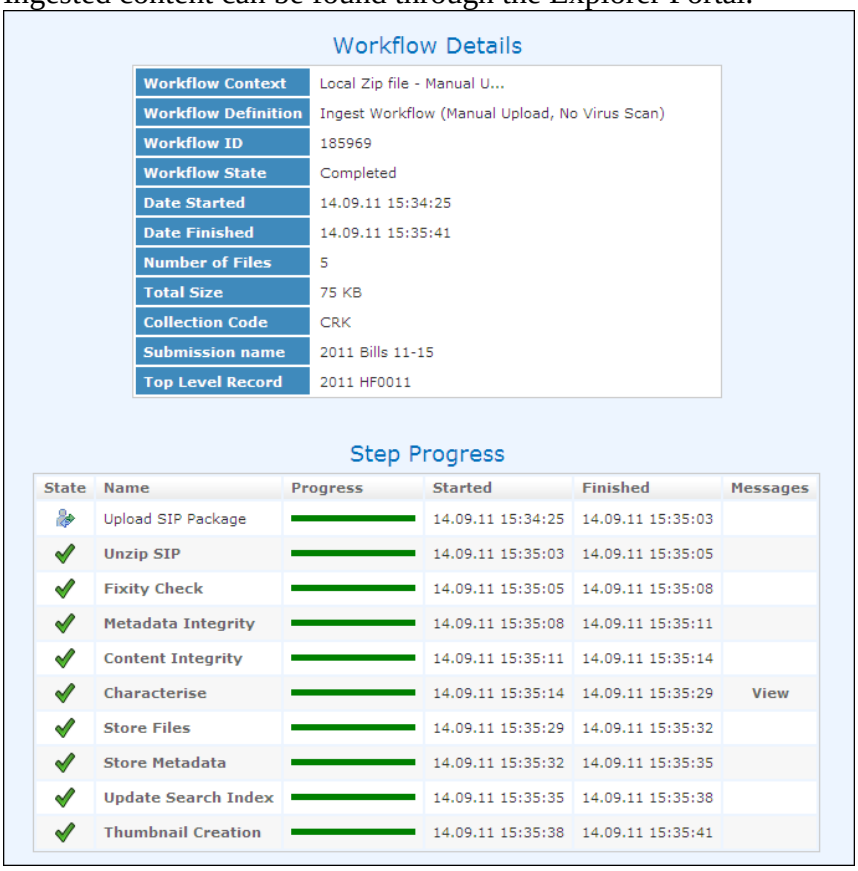
1. Run the Local Zip File workflow from the Ingest portal.



2. Select the SIP package for upload. Browse to a local file and click upload.



3. The workflow starts to run. A digest of a completed workflow is shown below. Ingested content can be found through the Explorer Portal.



Workflow Details

Workflow Context	Local Zip file - Manual U...
Workflow Definition	Ingest Workflow (Manual Upload, No Virus Scan)
Workflow ID	185969
Workflow State	Completed
Date Started	14.09.11 15:34:25
Date Finished	14.09.11 15:35:41
Number of Files	5
Total Size	75 KB
Collection Code	CRK
Submission name	2011 Bills 11-15
Top Level Record	2011 HF0011

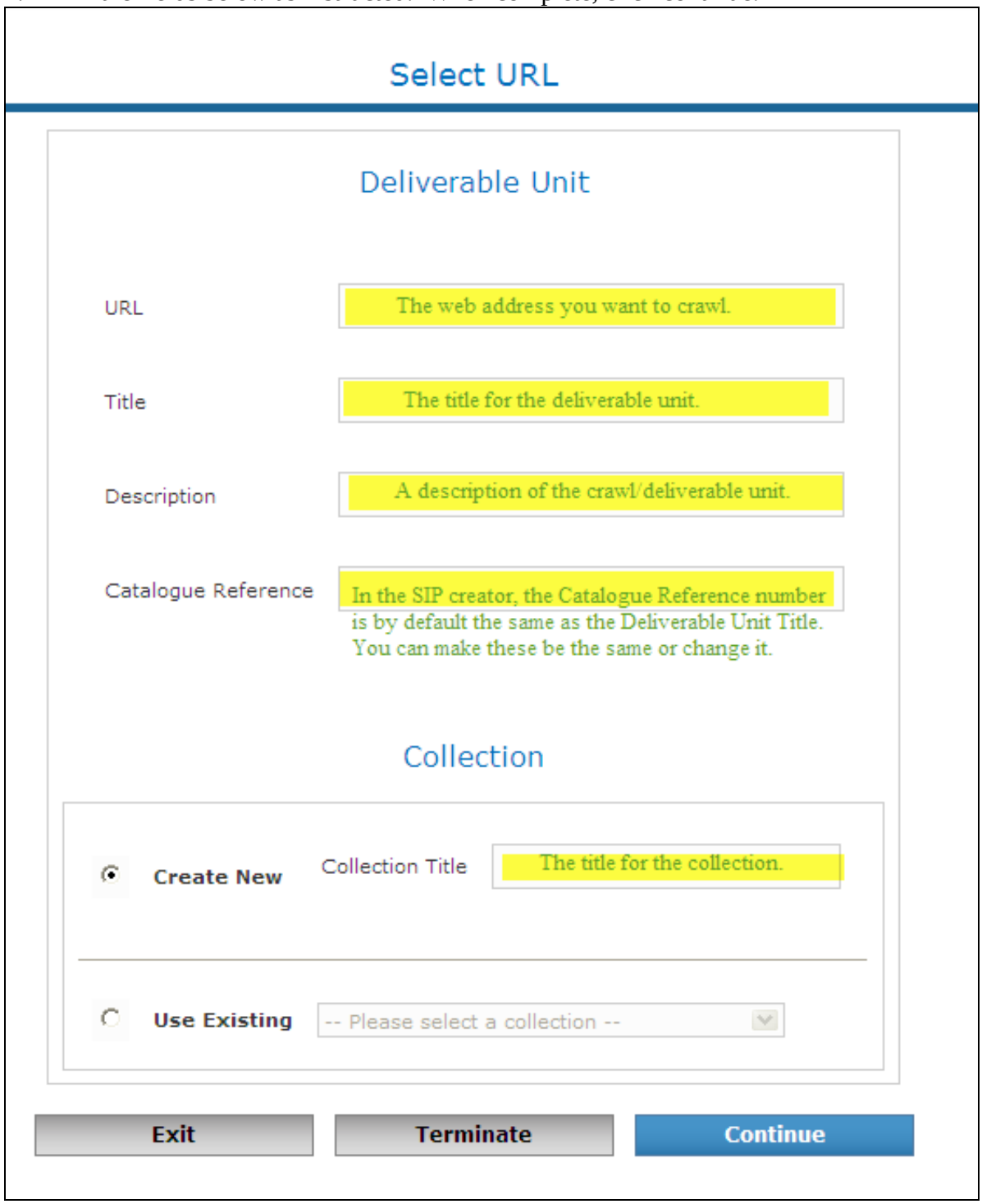
Step Progress

State	Name	Progress	Started	Finished	Messages
	Upload SIP Package		14.09.11 15:34:25	14.09.11 15:35:03	
✓	Unzip SIP		14.09.11 15:35:03	14.09.11 15:35:05	
✓	Fixity Check		14.09.11 15:35:05	14.09.11 15:35:08	
✓	Metadata Integrity		14.09.11 15:35:08	14.09.11 15:35:11	
✓	Content Integrity		14.09.11 15:35:11	14.09.11 15:35:14	
✓	Characterise		14.09.11 15:35:14	14.09.11 15:35:29	View
✓	Store Files		14.09.11 15:35:29	14.09.11 15:35:32	
✓	Store Metadata		14.09.11 15:35:32	14.09.11 15:35:35	
✓	Update Search Index		14.09.11 15:35:35	14.09.11 15:35:38	
✓	Thumbnail Creation		14.09.11 15:35:38	14.09.11 15:35:41	

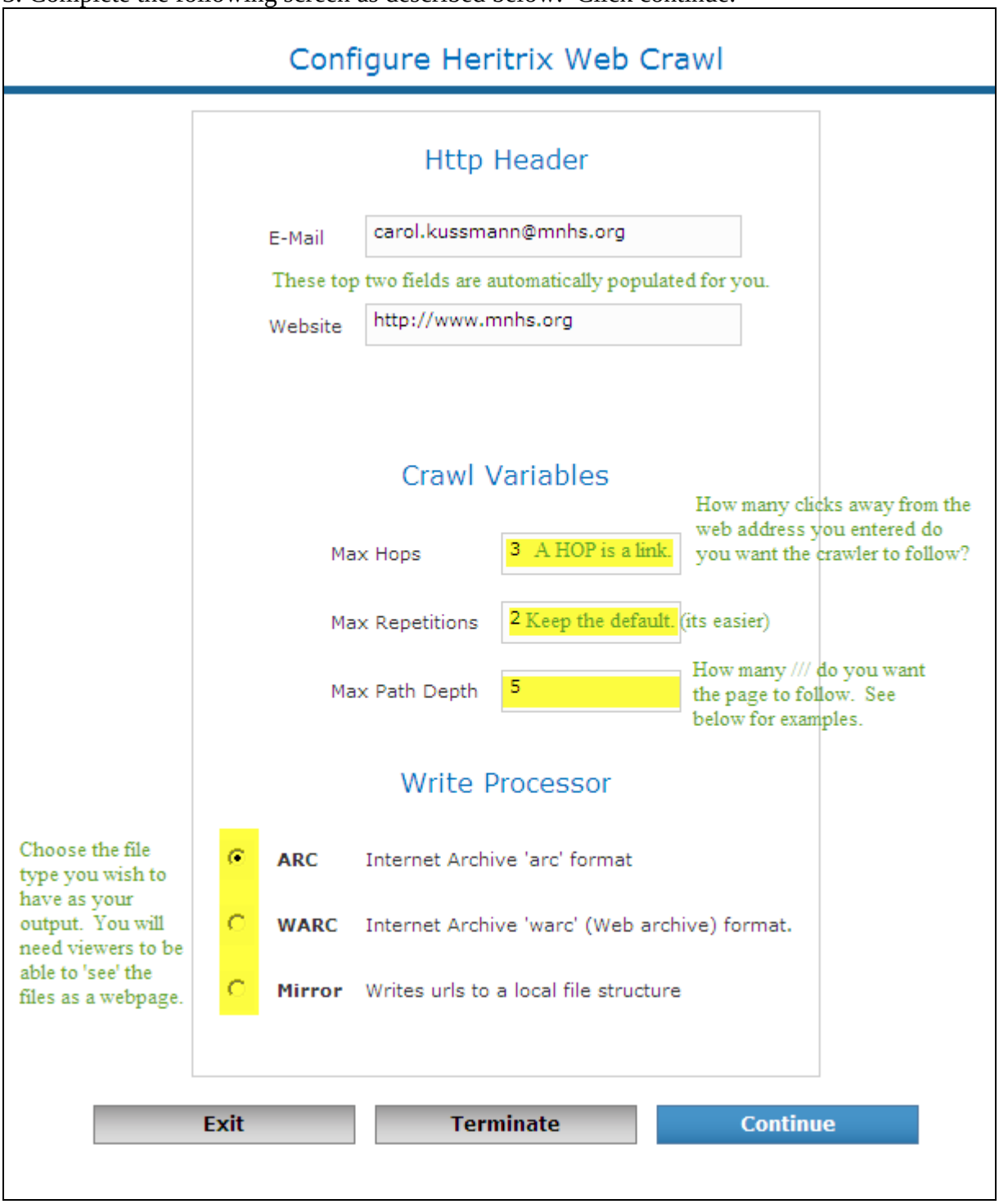
Example 2: Web Harvest

1. Run the Web Site URL Harvest workflow.

2. Fill in the fields below as instructed. When complete, click continue.



3. Complete the following screen as described below. Click continue.



This example is 3 deep...

http://www.uiuc.edu/classes/spring/2011

This example is 6 deep...

http://www.uiuc.edu/registration/student/new/information/housing/oncampus

Something to think about: if a HOP takes you do an address that is 10 deep in your webpage, and you have only selected to go 5 deep, the page will not be captured. You should have an idea about the page structure before setting your choices.

4. The crawl can be found through the Explorer portal. You can download the dip or file to view it.

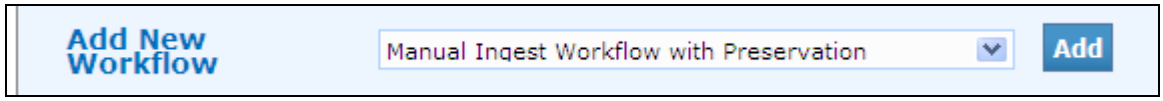
If you selected mirror, you can download individual pages or the entire capture as a DIP. You can then see the pages as 'screen shots'.

If you selected WARC files, you can download the file and view the file as XML in a web browser. Right clicking will show you an inventory of the WARC file.

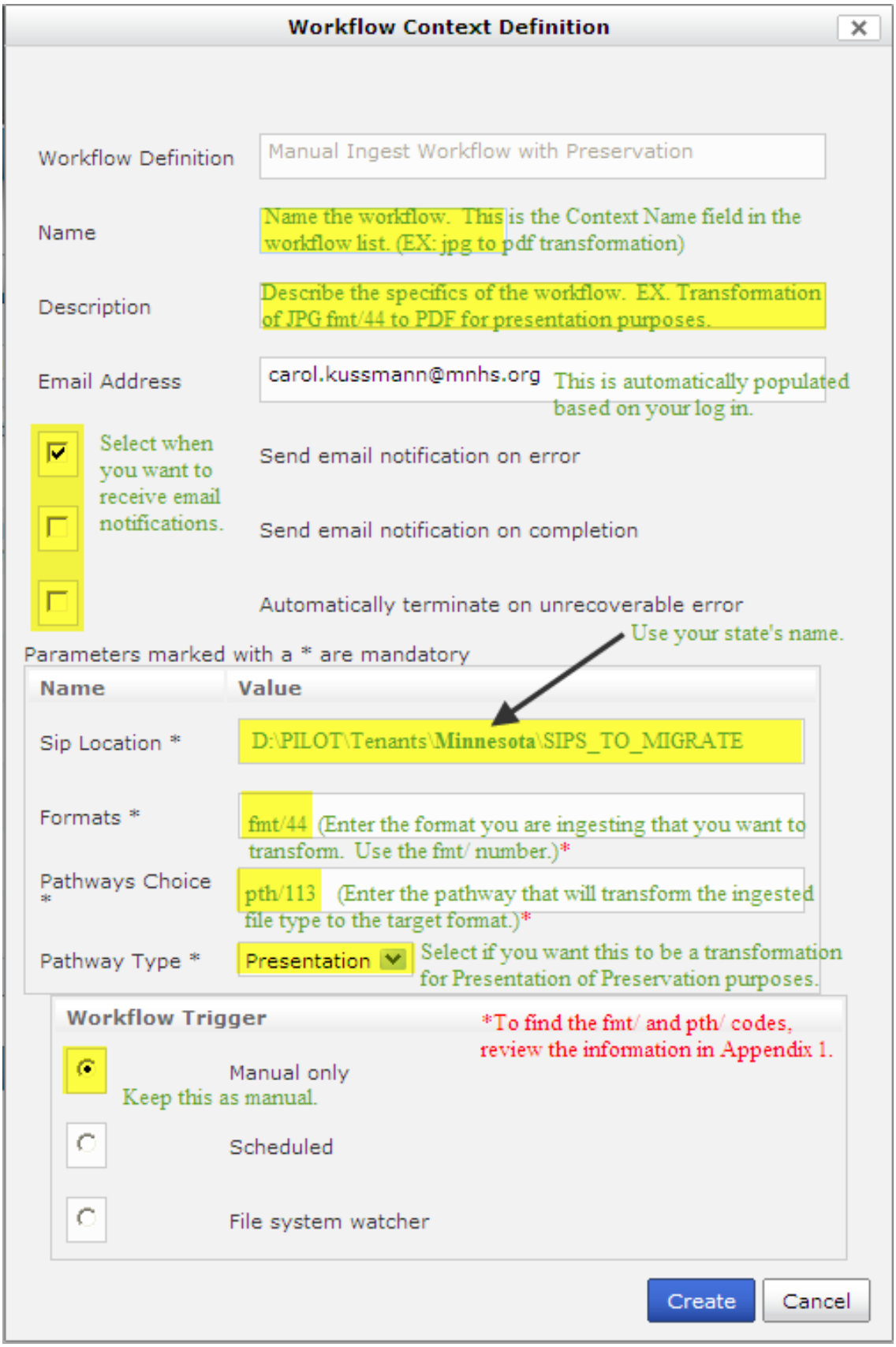
Example 3: Manual Ingest with Preservation Workflow

To perform a preservation action on items upon ingest you will need to 1) set up a preservation workflow and 2) start the workflow with a new SIP. A separate preservation workflow will need to be set up for each transformation you want to do. EX: To transform tiff → jpg you would set up one workflow and to transform doc → pdf you would set up another. Once a workflow is set up, you will be able to jump to starting the workflow.

1. From the manage tab in the Ingest portal, select the "Manual Ingest Workflow with Preservation"



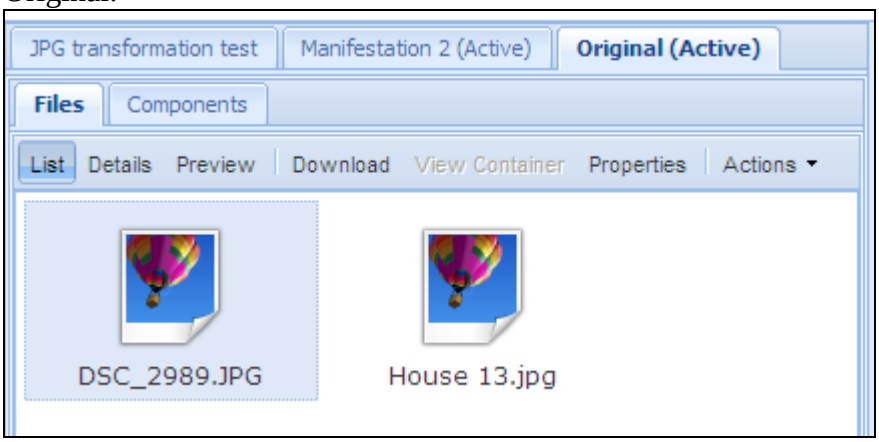
2. Create the workflow context definition by filling out the fields in the screen below. When complete, click create to add this new workflow context to the list of available workflows.


Workflow Context Definition
Workflow Definition
Manual Ingest Workflow with Preservation
Name
Name the workflow. This is the Context Name field in the workflow list. (EX: jpg to pdf transformation)
Description
Describe the specifics of the workflow. EX. Transformation of JPG fmt/44 to PDF for presentation purposes.
Email Address
carol.kussmann@mnhs.org
This is automatically populated based on your log in.
Select when you want to receive email notifications.
Send email notification on error
Send email notification on completion
Automatically terminate on unrecoverable error
Use your state's name.
Parameters marked with a * are mandatory
Name
Value
Sip Location *
D:\PILOT\Tenants\Minnesota\SIPS_TO_MIGRATE
Formats *
fmt/44 (Enter the format you are ingesting that you want to transform. Use the fmt/ number.)*
Pathways Choice *
pth/113 (Enter the pathway that will transform the ingested file type to the target format.)*
Pathway Type *
Presentation
Select if you want this to be a transformation for Presentation of Preservation purposes.
Workflow Trigger
*To find the fmt/ and pth/ codes, review the information in Appendix 1.
Manual only
Keep this as manual.
Scheduled
File system watcher
Create
Cancel

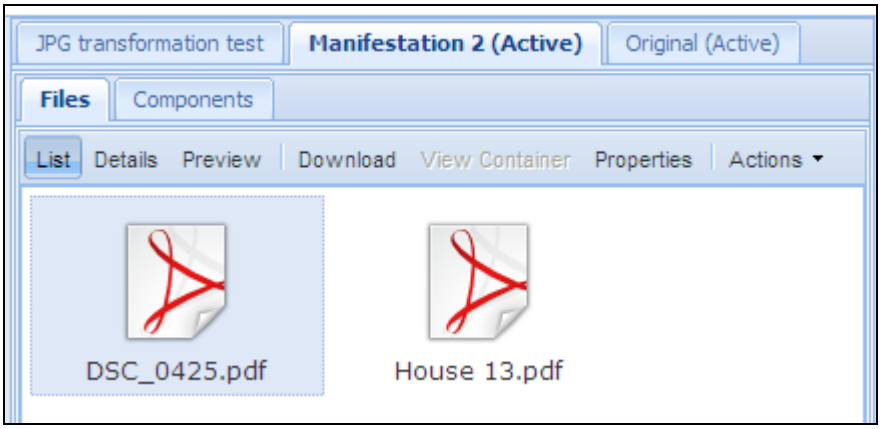
3. Now that a workflow has been set up for this particular transformation anyone with access to this SDB instance can use it. Like all other workflows, it can be used as often as necessary. To run the workflow, follow the same basic procedures as described in the basic workflow example.

4. Viewing a completed workflow from the Explorer portal shows two manifestations, the original and the transformation.

Original:



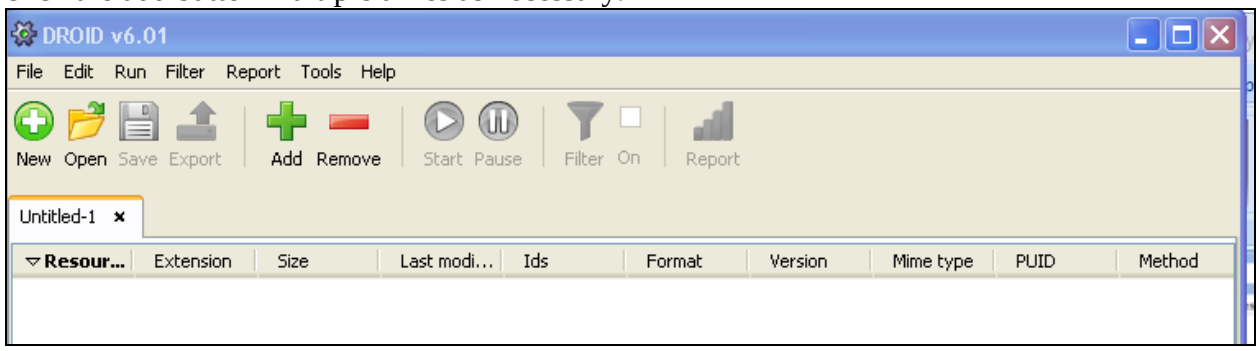
Transformation:



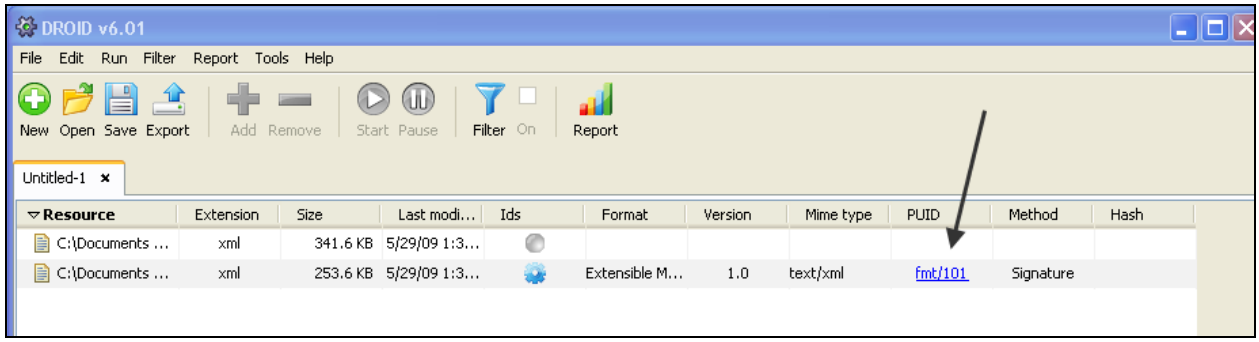
Appendix C: DROID Format Identifier

SDB uses DROID as one of the tools for identifying formats. Droid can also be installed locally and run to identify file formats prior to uploading if desired.

1. Add files to identify into DROID by clicking the green "Add" button. This will allow you to navigate to a file located on your computer; select the file/s in question. You can click the add button multiple times as necessary.



2. To start the identification process, press the blue "start" or play button. This will run DRIOD and the formats will be identified as shown below.



If the tool can figure out the format, the FMT/ is listed under PUID. (In this case, it found one but not the other – which was the expected result.)

Appendix D: Reports

Out-of-the-box reports were available in SDB. After reviewing these reports, modifications were made; before and after examples follow.

Below is the out-of-the-box "Ingested Files" report. Notice that the files are described by their "Ingested File Set ID".

SDB report: Ingested files

Tessella
Technology & Consulting

Start date: Not Specified Total files: 1000
End date: Not Specified Total size: 45.7 MB

Ingested File Set ID	File Name	Size (bytes)	Date Ingested
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2706-3-nopl.xml	12.2 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2716-1-nopl.xml	7.66 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2720-1-nopl.xml	596 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2720-2-nopl.xml	599 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2720-3-nopl.xml	592 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2724-1-nopl.xml	13.0 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2736-1-nopl.xml	23.4 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2747-1-nopl.xml	26.5 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2747-2-nopl.xml	26.6 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2749-1-nopl.xml	3.82 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2754-1-nopl.xml	3.83 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2755-1-nopl.xml	2.35 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2756-1-nopl.xml	2.21 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2756-2-nopl.xml	2.61 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2765-1-nopl.xml	4.63 KB	10 June 2011
3f77fe91-d354-49ed-a75e-6d07ef5925a5	S2766-1-nopl.xml	9.40 KB	10 June 2011

Out-of-the-box Ingested Files Report; uses Ingested File Set ID

This was not very informative so Tessella was asked to modify the report and replace the "Ingested File Set ID" with "Collection" and "Deliverable Unit" titles. These modifications are shown below.

SDB report: Ingested files

Start date: Not Specified Total files: 38

End date: Not Specified Total size: 290 MB

Collection	Deliverable Unit	File Name	Size (bytes)	Date Ingested
Transformations upon	Word to PDF	_Retrospective Digitization	175 KB	11 August 2011
Transformations upon	Word to PDF	ResourcesDigAudVid_CRK2009-04.doc	74.5 KB	11 August 2011
Transformations upon	Word to PDF	Legislative MetadataComparisonMNSunlightMarch2011.doc	323 KB	11 August 2011
Transformations upon	Word to PDF	DigitalAudioVideoWhitePaper_CRK_05-4-2009.doc	589 KB	11 August 2011
Transformations upon	Word to PDF	_Retrospective Digitization PaperFinal	424 KB	11 August 2011
Transformations upon	Word to PDF Transformation 2 -	Legislative MetadataComparisonMNSunlightMarch2011.doc	323 KB	11 August 2011
Transformations upon	Word to PDF Transformation 2 -	_Retrospective Digitization PaperFinal	424 KB	11 August 2011
Transformations upon	Word to PDF Transformation 2 -	ResourcesDigAudVid_CRK2009-04.doc	74.5 KB	11 August 2011
Transformations upon	Word to PDF Transformation 2 -	DigitalAudioVideoWhitePaper CRK 05-4-2009.doc	589 KB	11 August 2011
Transformations upon	Word to PDF Transformation 2 -	_Retrospective Digitization	175 KB	11 August 2011
Transformations upon	JPG transformation test	DSC_0425.JPG	98.0 KB	12 August 2011
Transformations upon	JPG transformation test	House 13.jpg	1.60 MB	12 August 2011
Transformations upon	JPG transformation test	DSC_2989.JPG	100 KB	12 August 2011
Transformations upon	JPG transformation test	DSC_6934.JPG	40.6 KB	12 August 2011
Transformations upon	JPG transformation test	DSC_0185.JPG	1.48 MB	12 August 2011
Transformations upon	JPG transformation test	DSC_0185.JPG	1.48 MB	12 August 2011
Transformations upon	JPG transformation test	DSC_0141.JPG	1.30 MB	12 August 2011
Transformations upon	JPG transformation test	DSC_0141.JPG	1.30 MB	12 August 2011

Modified Ingested Files report; uses the collection title and deliverable unit title rather than the Ingested File Set ID.

Tessella was also asked to create reports that specifically addressed administrative issues.

The Collections Details report was created to provide an overall view of the archive, the amount of space used, and organization of the collections. The report lists the name of each collection, the number of deliverable units and files within each collection. A total number of collections, deliverable units, and files as well as total file size is also recorded.

SDB report: Collection Details

Total # of Collections	23	Total Files	43745
Total Deliverable Units	116	Total File Size	9.50 GB

Collection	Deliverable Units	Files	Size (bytes)
2007-2008 Bills	1	6851	159 MB
2007-2008 Bills part 2	1	6851	355 MB
2011 Bills 1 to 10	14	44	11.6 MB
2011 Bills 11-15	1	5	75.1 KB
8-11-11	1	2	82.1 KB
Brainerd	1	1	310 KB
Brenda Ueland	2	3	351 KB
Docs	3	10	1.71 MB
EAD Documents	1	1	22.8 KB
Finding Aids	1	5	60.0 KB
Format Test	1	26	394 MB
Jpg	8	23	1.13 MB
Large File Test	1	1	1.30 GB
Metadata Test (Internal)	1	3	1.47 MB
metadata test sept 2 2011	2	2	141 KB
Minnesota Bills	40	9713	1.77 GB
Nested Folder Test 1 - Gov Ventura	17	431	137 MB
New Employee checklist	1	2	299 KB
Original Word Documents	1	4	1.30 MB
Transformations upon	10	73	323 MB
Video Conversion tests	2	11	3.61 GB
Website Crawls	5	19681	1.46 GB
XML Header Tests	1	2	376 KB

Collection Details Report

The Fixity Last Checked report records the fixity value for a file and shows when the fixity value was last verified. This report assists with the trustworthiness of the archive.

SDB report: Collection Fixity Last Checked

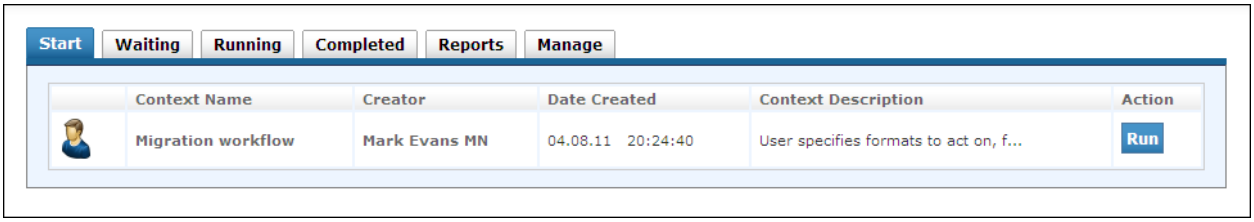
Collection	Deliverable Units	Files	Date Started	Date Finished	Fixity
2007-2008 Bills	2007-2008 Bills without Page Lines	07-0524-nopl.xml	2011-10-05	null	<File xmlns="http://www.tessella.com/XIP/v4" status="same"><FileRef>4b0c3e2a-011d-48af-b78d-236eef062f82</FileRef><IngestedFileSetRef>3f77fe91-d354-49ed-a75e-6d07ef5925a5</IngestedFileSetRef><FileName>07-0524-nopl.xml</FileName><Extant>true</Extant><Directory>false</Directory><FileSize>3781</FileSize><LastModifiedDate>2009-05-29T14:15:28.000+01:00</LastModifiedDate><FixityInfo><FixityAlgorithmRef>2</FixityAlgorithmRef><FixityValue>063e7cbd88cc773ba461f60717e644738838ca73</FixityValue></FixityInfo><IdentificationStatus>Notidentified</IdentificationStatus><IdentificationEvent><EventDate>2011-09-15T15:04:33.198+01:00</EventDate><EventAgent><Name>DROID</Name><Version>2.0</Version><Component>Droi
2007-2008 Bills	2007-2008 Bills without Page Lines	07-0524-nopl.xml	2011-10-04	null	<File xmlns="http://www.tessella.com/XIP/v4" status="same"><FileRef>4b0c3e2a-011d-48af-b78d-236eef062f82</FileRef><IngestedFileSetRef>3f77fe91-d354-49ed-a75e-6d07ef5925a5</IngestedFileSetRef><FileName>07-0524-nopl.xml</FileName><Extant>true</Extant><Directory>false</Directory><FileSize>3781</FileSize><LastModifiedDate>2009-05-29T14:15:28.000+01:00</LastModifiedDate><FixityInfo><FixityAlgorithmRef>2</FixityAlgorithmRef><FixityValue>063e7cbd88cc773ba461f60717e644738838ca73</FixityValue></FixityInfo><IdentificationStatus>Notidentified</IdentificationStatus><IdentificationEvent><EventDate>2011-09-15T15:04:33.198+01:00</EventDate><EventAgent><Name>DROID</Name><Version>2.0</Version><Component>Droi
2007-2008 Bills	2007-2008 Bills without Page Lines	07-0524-nopl.xml	2011-10-03	null	<File xmlns="http://www.tessella.com/XIP/v4" status="same"><FileRef>4b0c3e2a-011d-48af-b78d-236eef062f82</FileRef><IngestedFileSetRef>3f77fe91

Fixity Last Checked Report

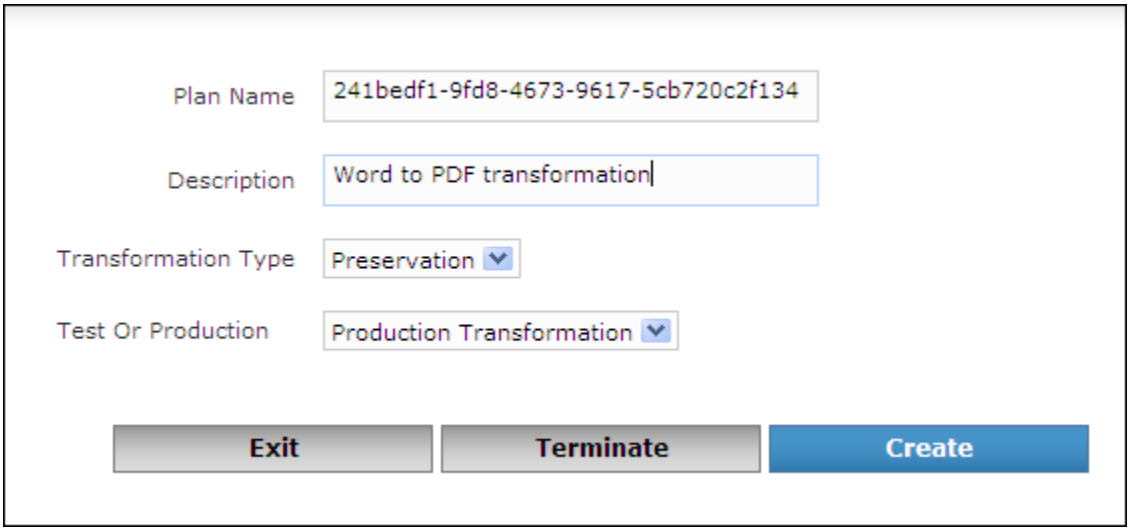
Appendix E: Preservation Plan

Creating a Preservation Plan

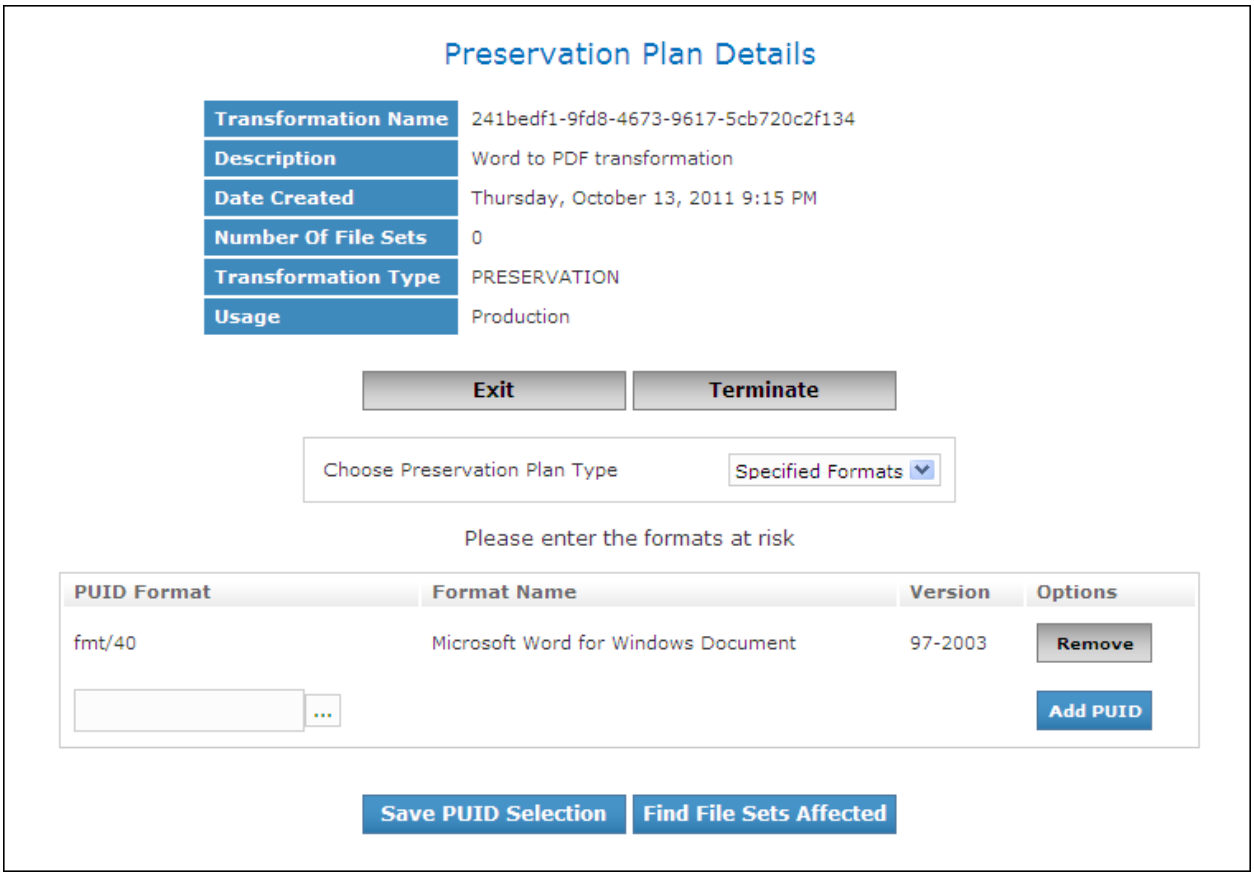
To perform a preservation action you need to create a 'preservation plan'. The preservation plan is where you choose what format type you want to act upon and decide if you want to run a preservation or presentation action. This plan is created in multiple steps as shown below.



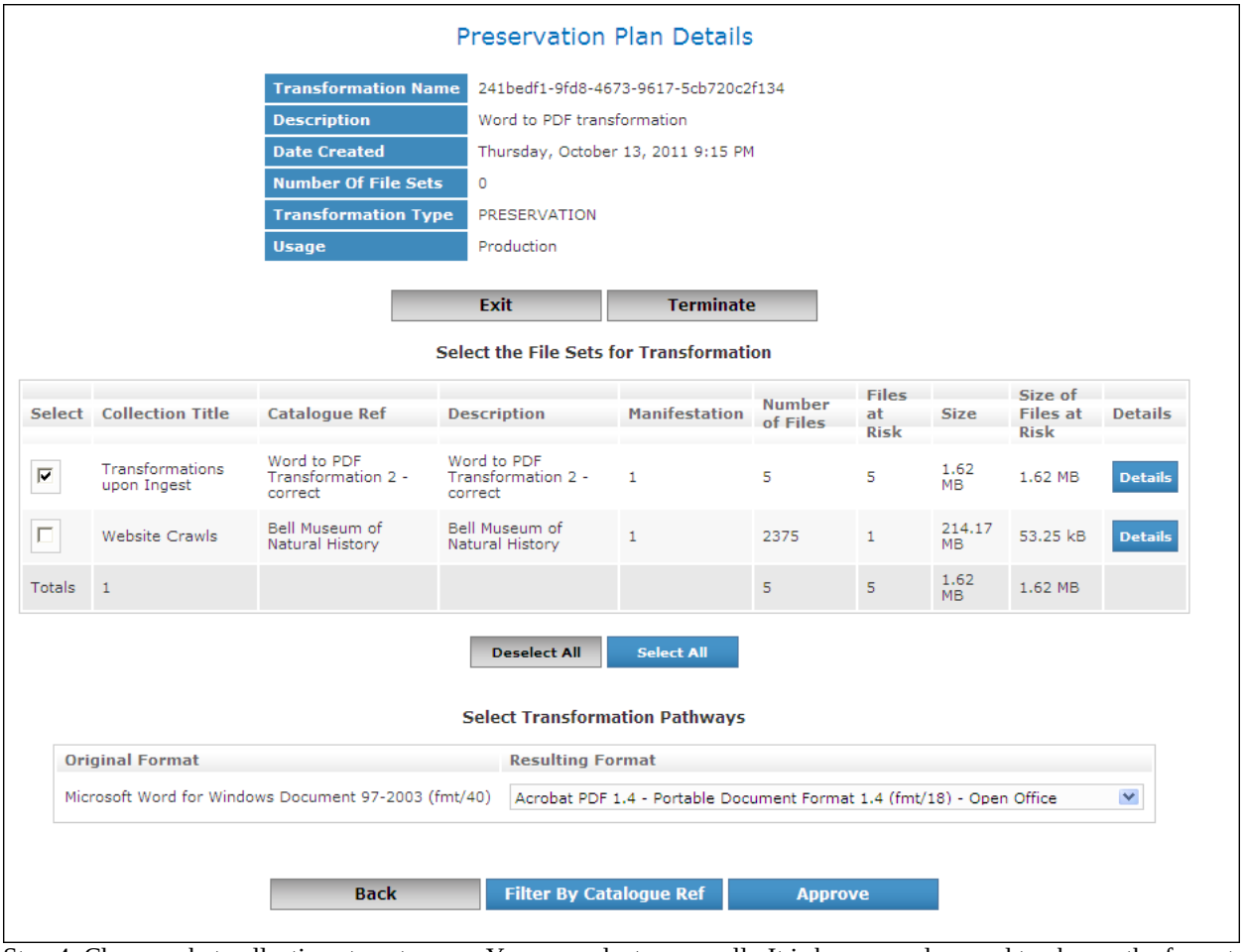
Step 1: Start a Migration workflow. Click Run.



Step 2: Create the 'preservation plan'. You can rename the Plan name, or keep it as is (the set of reference numbers). Describe what it is that you want to do. Select the transformation type: preservation or presentation. Choose test or production environment. A test transformation performs a transformation but does not ingest the transformed content. Transformed content can be reviewed by downloading it. Click create.



Step 3: Select the preservation plan type and formats to act upon. You can enter more than one format if desired. Click find file sets affected.



Preservation Plan Details

Transformation Name	241bedf1-9fd8-4673-9617-5cb720c2f134
Description	Word to PDF transformation
Date Created	Thursday, October 13, 2011 9:15 PM
Number Of File Sets	0
Transformation Type	PRESERVATION
Usage	Production

Exit | Terminate

Select the File Sets for Transformation

Select	Collection Title	Catalogue Ref	Description	Manifestation	Number of Files	Files at Risk	Size	Size of Files at Risk	Details
☑	Transformations upon Ingest	Word to PDF Transformation 2 - correct	Word to PDF Transformation 2 - correct	1	5	5	1.62 MB	1.62 MB	Details
☐	Website Crawls	Bell Museum of Natural History	Bell Museum of Natural History	1	2375	1	214.17 MB	53.25 kB	Details
Totals	1				5	5	1.62 MB	1.62 MB	

Deselect All | Select All

Select Transformation Pathways

Original Format	Resulting Format
Microsoft Word for Windows Document 97-2003 (fmt/40)	Acrobat PDF 1.4 - Portable Document Format 1.4 (fmt/18) - Open Office

Back | Filter By Catalogue Ref | Approve

Step 4: Choose what collections to act upon. You can select any or all. It is here you also need to choose the format you would like to move to. A dropdown list may provide various options for the transformation pathway. After you make your selection you can approve and run the workflow.

Preservation Plan Details

Transformation Name	241bedf1-9fd8-4673-9617-5cb720c2f134
Description	Word to PDF transformation
Date Created	Thursday, October 13, 2011 9:15 PM
Number Of File Sets	1
Transformation Type	PRESERVATION
Usage	Production

Exit **Terminate**

This transformation is ready for execution.

Selected File Sets

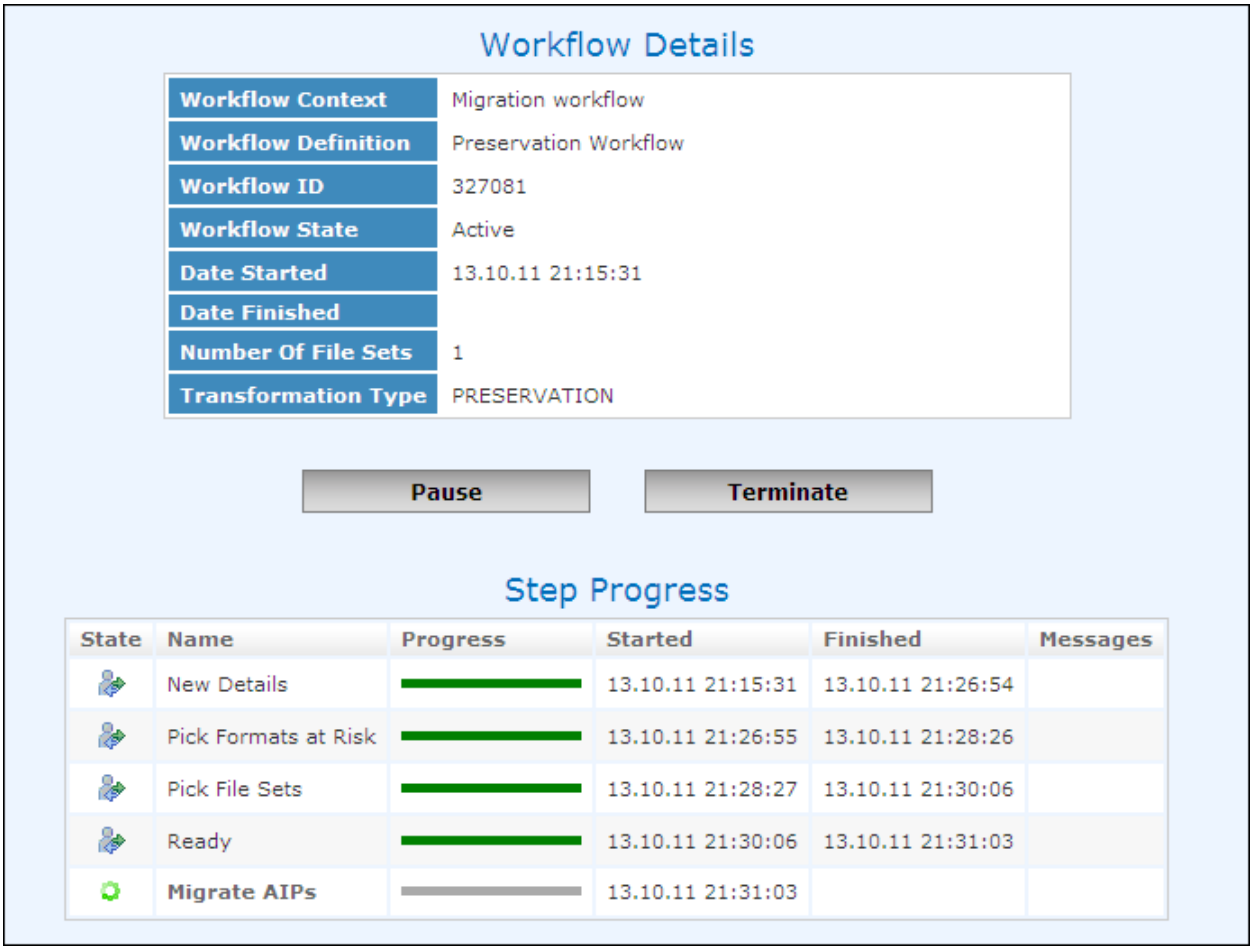
	Collection Title	Catalogue Ref	Description	Manifestation	Number of Files	Files at Risk	Size	Size of Files at Risk	Details
	Transformations upon Ingest	Word to PDF Transformation 2 - correct	Word to PDF Transformation 2 - correct	1	5	5	1.62 MB	1.62 MB	Details
Totals	1				5	5	1.62 MB	1.62 MB	

Selected Migration Pathways

Source Format	Version	PUID	Target Format	Version	PUID	Tool
Microsoft Word for Windows Document	97-2003	fmt/40	Acrobat PDF 1.4 - Portable Document Format	1.4	fmt/18	Open Office

Back **Confirm**

Step 5: Review the plan and click Confirm.



Workflow Details

Workflow Context	Migration workflow
Workflow Definition	Preservation Workflow
Workflow ID	327081
Workflow State	Active
Date Started	13.10.11 21:15:31
Date Finished	
Number Of File Sets	1
Transformation Type	PRESERVATION

Pause Terminate

Step Progress

State	Name	Progress	Started	Finished	Messages
	New Details		13.10.11 21:15:31	13.10.11 21:26:54	
	Pick Formats at Risk		13.10.11 21:26:55	13.10.11 21:28:26	
	Pick File Sets		13.10.11 21:28:27	13.10.11 21:30:06	
	Ready		13.10.11 21:30:06	13.10.11 21:31:03	
	Migrate AIPs		13.10.11 21:31:03		

Step 6: The workflow starts.

Workflow Details

Workflow Context	Migration workflow
Workflow Definition	Preservation Workflow
Workflow ID	327081
Workflow State	Completed
Date Started	13.10.11 21:15:31
Date Finished	13.10.11 21:31:36
Number Of File Sets	1
Transformation Type	PRESERVATION

Step Progress

State	Name	Progress	Started	Finished	Messages
	New Details		13.10.11 21:15:31	13.10.11 21:26:54	
	Pick Formats at Risk		13.10.11 21:26:55	13.10.11 21:28:26	
	Pick File Sets		13.10.11 21:28:27	13.10.11 21:30:06	
	Ready		13.10.11 21:30:06	13.10.11 21:31:03	
✓	**Migrate AIPs**		13.10.11 21:31:03	13.10.11 21:31:24	**View**
✓	Launch Ingest Workflows		13.10.11 21:31:24	13.10.11 21:31:24	

Preservation Plan Details

Transformation Name	241bedf1-9fd8-4673-9617-5cb720c2f134
Description	Word to PDF transformation
Date Created	Thursday, October 13, 2011 9:15 PM
Number Of File Sets	1
Transformation Type	PRESERVATION
Usage	Production

The transformation has completed.

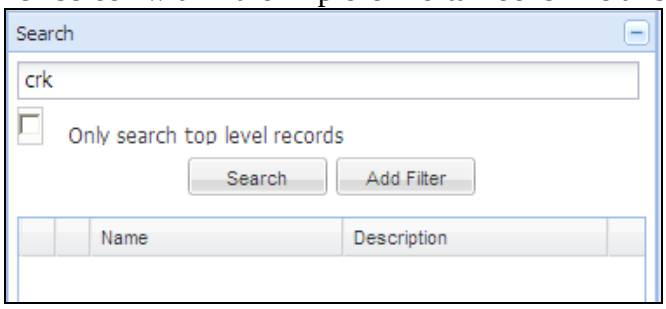
Top Level Record	Collection Title	Outcome
Word to PDF Transformation 2 - correct	Transformations upon Ingest	Transformation Complete

Completed workflow.

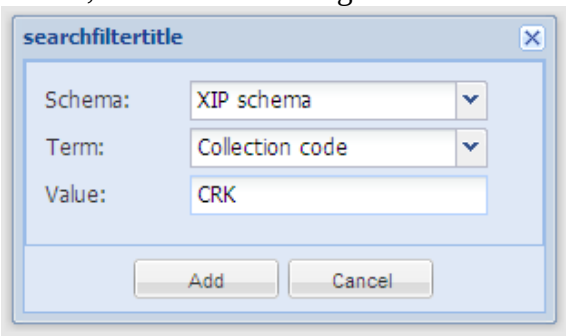
Appendix F: Search Methods

Explorer Portal Search

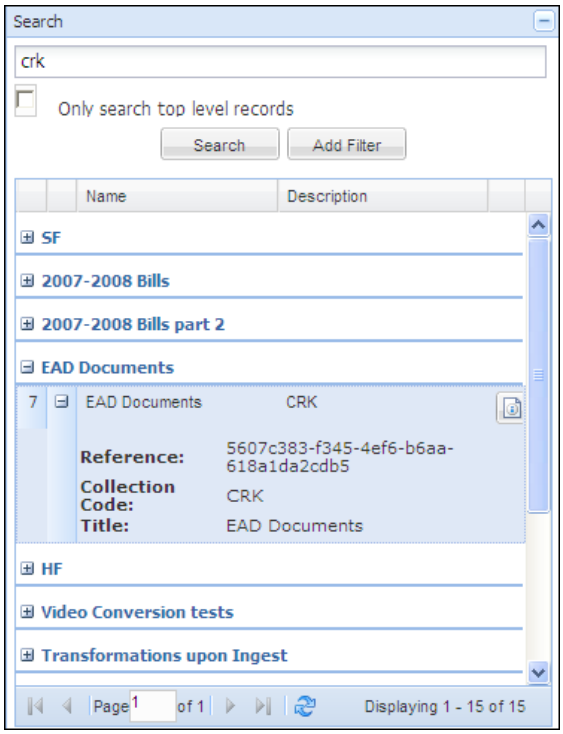
The search screen within the Explorer Portal looks like this:



To add a filter, fill in the following screen:

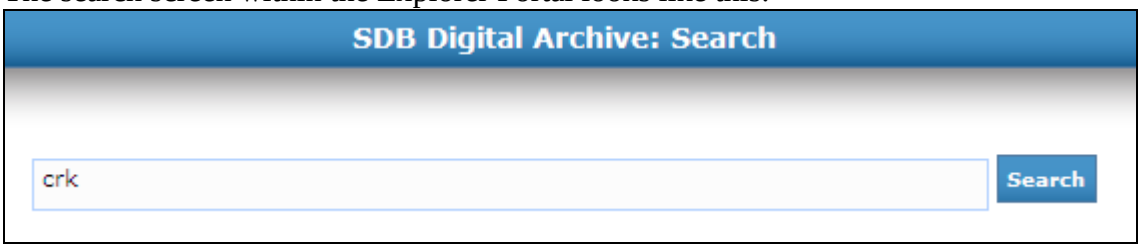


The results look like this:

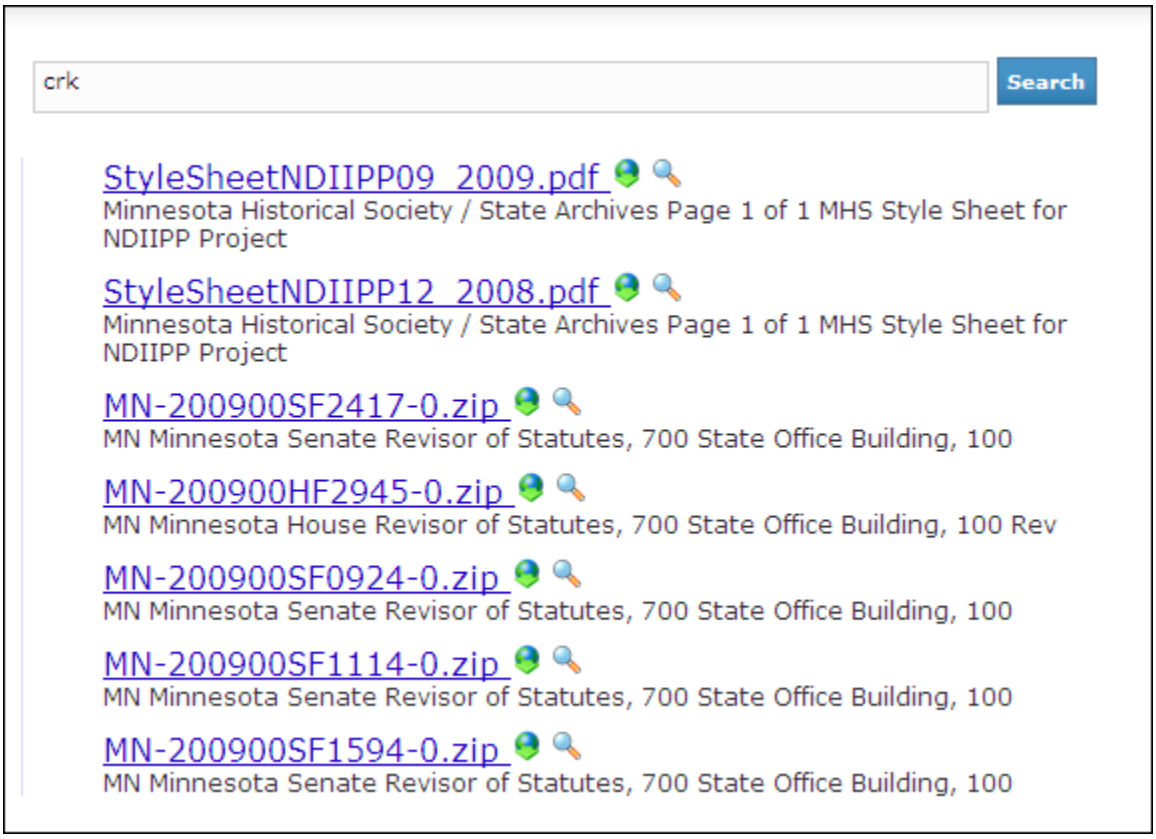
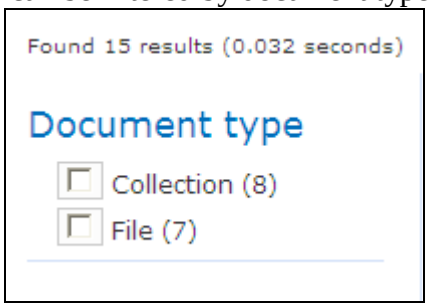


Search Portal Search

The search screen within the Explorer Portal looks like this:



The results look like this:



Results can be filtered by document type after the results are returned.