Updates to SobekCM in first quarter 2015

There were numerous updates released in the first quarter of 2015, culminating in the release of the new installer for version 4.8.7. Development focused on general performance and design improvements, including embracing an international content delivery network for many of the static files used by the system and other changes to result in a large performance gain for real users. Customization support was added for MARC21 generation and OAI-PMH metadata formats, from new system-wide settings to ability to apply a XSLT to the MARC21 after generation.   Development focused on improving the ease of online collection management, editing the look and feel of the site, and other administrative options with many new features available to authenticated and permissioned users. And finally, the system underwent continuing architectural revolutions with a completely rewritten modular builder service and with an increasing acceptance of a REST-based microservices approach.

In total, we saw the release of eight versions. The code base saw 140 individual commits from users within four institutions.

The new version can be downloaded at: http://sobekrepository.org/software/download

Below is a look at the individual changes from the first quarter of this year:

General Performance and Design Improvements

1. Content Delivery Network
    a. Updated the way all the static version-specific files are linked throughout the system, affecting all standard images, javascript, stylesheets, and included libraries
    b. The static files to use are read from configuration files, allowing a user-specific configuration file to override just one or many of the file references
    c. By default, all these files are served from a content delivery network, increasing speed and performance for end users by 25-40%, and reducing the number of hits against the server by close to 40%
    d. Benefit of the content delivery network will greatly improve performance from international users, as the CDN has over 30 nodes internationally
2. Included Libraries (both javascript and stylesheets)
    a. Re-examined all included libraries and added some more logic to only include some libraries when actually needed
    b. Minimized all included libraries to improve performance
3. Design improvements
    a. Update the style of the messages for errors and successes across all admin screens to make the message and the severity more obvious
    b. Changed the design for all the system admin screen, removing the gray title box and adding the icon for that task to the screen, to allow users to better associate the icons with the tasks
    c. Updated most the internal and aggregation curator screens to use a similar look and feel as the main admin screens
4. *Contact Us* form

a. Completely recreated the contact us form
b. Can customize the form to have any number of elements, including hidden elements by using the configuration file
c. Can additionally customize the form for individual aggregations to have aggregation-specific forms
5. Email configuration
a. Can now choose to either use the Database Mail system, or to email directly via SMTP
b. Can now set the FROM email address and display name in the system-wide settings
c. Centralized all emailing to use the single *Email_Helper* class
d. A number of small improvements and bug fixes for emailing item and collection information
6. HTTPS support added
7. Changes to registration process
a. URL changed to /register, rather than duplicating /my/preferences to make cleaner URLs when sending them via email
b. If a user enters their name with all caps or all lower case, the name is automatically title-cased
8. Help now defaults from http://sobekrepository.org/sobekcm, rather than UFDC

Aggregation Viewer Updates

1. Updates to the public aggregation-level main menu
a. If no thematic headings exist, the variety of home pages (list, brief, tree) are suppressed
b. If a child static page is selected, the parent menu item is now highlighted correctly
2. Restored Tree View on the main instance home page to work correctly, and for the collapse and expand all features to work correctly without going back to the server
3. Aggregation Item Count - Corrected an exception thrown when there were no items in the aggregation

Item Viewer Updates

1. Added more JPEG2000 support, especially for separate IIP image server, by adding new system-wide setting – Image Server Root – to override the defaults
2. Updates to the PDF viewer
a. Allow PDF title to contain single and double quotes
b. Fixed a bug for PDF delete function
3. Updates to the HTML viewer, which allows a HTML file to be embedded as one of the item views
a. Restored the HTML item viewer
b. If a HTML/HTM file is referenced by a HTML viewer and in the METS, it no longer triggers the downloads tab.
4. Updated the way viewers are added back to the digital resource object from the database to be more clear and easier to update and support later changes
5. Citation Viewer updates
a. Switch to using the main thumbnail in the citation viewer for better performance
b. If the rights statement has a URL, it is correctly displayed as a link.

Metadata and OAI-PMH

1. MARC21 Generation
   a. Moved several tags into the standard MARC tags, to always be included.
      i. Collections are now listed in the 830 field
      ii. Thumbnail is included in the 992 field
   b. Added ability to customize certain MARC fields across an instance through system-wide settings
      i. Cataloging source code – for the 040 field, was defaulting to FUG
      ii. For reproductions only (not born digital)
         1. Agency responsible for reproduction, for the added 533 |a field
         2. Place of reproduction, for the added 533 |b field
      iii. Location code, for the 852 |a field
      iv. Set XSLT file to be applied to the MarcXML records after creation
2. Added extended Dublin Core (and Duke specific) tags to the Dublin Core reader class
3. If a TYPE does not map to a standardized type, it is mapped to Genre automatically
4. Accession number updates
   a. Accession number now saves as an item group external record number
   b. Accession number saves correctly in the metadata searching tables
5. Updates to metadata templates - Template XML files are now divided into two main folders – *default* and *user*
   a. Default includes the templates that come with the system and may be changed during an upgrade
   b. Templates put in the user folder override the template in the default and will not be changed during upgrades
6. Updates to metadata template elements
   a. Holding location and source institution fields now work correctly without specific aggregation rights
   b. Updates to a controlled rights statement – currently utilized by UF
7. OAI-PMH
   a. Added new configuration file that allows any metadata type to be served as OAI-PMH
   b. External libraries can easily be referenced to override the standard creation of metadata for OAI-PMH
   c. By default, MarcXML is included as an OAI-PMH metadata option
   d. Updated resumption token to work correctly – did not work in certain situations

Updates to Collection Management / Curation

1. Created *Add New Collection Wizard* to which steps an administrator or curator through the process of adding a new collection, with the following new capabilities:
   a. Aggregation type of the new aggregation is suggested based on the parent
   b. External link is hidden except for institutional type aggregations
   c. Short name is automatically populated from the name when leaving the name field
   d. Flag to include in the parent collection is hidden when the parent is the top-level home
   e. Can add a new thematic heading at the same time the new aggregation is added

f. Can upload the collection banner, or choose to let the system auto-generate the banner
g. Can upload the collection button, otherwise the system default is used
h. Better default values in general provided

2. Added a new aggregation admin/curator home page within the aggregation, available from the internal header, with access to all the aggregation-level management options.

3. Added new *Users Permissions* report which shows the list of all users that have special rights over the current collection. This report is accessible from the aggregation admin home menu and shows rights directly given to users, or assigned through user groups.

4. Added a new *Collection Change Log* which shows all the updates to the design files, settings, uploads, and home pages for each aggregation. This does not include changes to the digital resources, just changes to the collection itself.   This report is also accessible from the new aggregation admin home menu.

5. Major updates to the Single Aggregation administration screens
   a. Configuration files are now backed up when editing through the online interface
   b. Added a new *Uploads* tab which allows user to manage uploads
      i. Uploads can be deleted
      ii. Can upload images (several at time), for inclusion in the home page or child page text
      iii. Can upload documents to associate with the aggregation (such as tracking documents or the source files for the graphics used)
   c. Added a new SAVE button to save the changes and continue editing, and changed existing button to SAVE & EXIT.
   d. Some tab maintenance, such as hiding the *Highlights* tab if unused and moving the commonly used *Appearance* tab up to the second position.
   e. Hid the external link field if the aggregation is not an institution-type collection
   f. Added help to each of the help buttons on all the tabs of the aggregation admin screen
   g. Added an image of each search screen to view when selecting which searches to include in an aggregation
   h. Can now save multiple web skins to an aggregation
   i. Added ability to delete unused banner images
   j. Viewing the source files (most notably the CSS) now opens in a new window
   k. Top-level home page now supports multi-lingual versions, same as all the child aggregations
   l. Bug Fixes:
      i. Corrected issue where selecting an eight facet was causing the system to fail to show any results or facets
      ii. All changes from the admin screen take effect immediately now
      iii. Subcollections now show in the *SubCollection* tab, even if hidden or inactive.
      iv. Removed FULL from the possible search result views – deprecated
      v. Corrected the display of links for the source files (was white on white)

6. Aggregation (static) child pages
   a. Can now edit the home page html for all the child pages, including the browse-by version
   b. Putting child pages on the main menu under existing child pages now works, allowing hierarchical child page navigation directly from the publicly accessible aggregation menu

7. New Aggregations
   a. Automatically generated home page text updated to include a link to the parent collection and the description entered when the aggregation was added
   b. An email is now sent to the system administrators when a new aggregation is added to the system (can be controlled via new system-wide setting)

      c.   Aggregation permissions are copied from the parent collection to the child collection during creation
8. Exhibits now support subcollections, just like all other aggregations

Internal Views and System Administration

1. Improved the *Collection List* internal view – available to power/internal users
    a. Existing list view updated to allow filtering, sorting, and searching
    b. Added a new *Collection Hierarchy* view that shows ALL of the collection hierarchy (including inactive and hidden collections) in a collapsible tree view
2. Updated the system/portal admin main menu
    a. It is now a deeper categorized system, with more options
    b. Common tasks appear at the top, and include the most commonly used tasks
3. Updated the system/portal admin home screen
    a. Brief view includes a description of each option
    b. Classic view is similar to the old version display
    c. Categorized and Alphabetical display icons with just the title
    d. Recreated many of the icons used in the admin home screen
4. Updates to the *Users and User Groups* management
    a. Fixed issues reported when viewing user/user groups and individual permissions
    b. Updated the user group management screens to work more like the user management screens
    c. Added ability to delete user groups that are not attached to any users
    d. Validated all permissions changes in both the standard and detailed permissions modes
5. Added new *User Permissions Report* at the top-level, accessible to system and portal admins
    a. Displays users with permissions provided, either linked directly to the user or through user group membership
    b. Global list – shows all the users with top-level permissions (i.e., system admin, can delete all, can edit all, etc..) displayed as lists under each permission type
    c. Global table – shows users in a tabular format with filtering, sorting, and searching that have the same top-level permissions as above
    d. All Aggregations – lists every user that has specialized permissions on aggregations, along with a list of all the aggregations each user has permissions for. Permissions granted are displayed separately for those individually provisioned and those assigned through user group management
    e. Single Aggregations – lists the details of permissions granted to users and user groups for a single aggregation
    f. Can Submit Items – lists all users that have submittal rights and includes which templates and default metadata sets users are linked to.
6. *Aggregation Management* (top-level, under the portal/system admin options)
    a. Removed the DELETE link from the ALL collection, as it cannot be deleted
    b. Updated the list of aggregations
        i. All of the aggregations are now displayed simultaneously, but the list is now filterable, sortable, and searchable
        ii. Added columns for the parent collection codes and flags to indicate if the collection is active and if the collection is on the top-level home page

      c.   Collection codes for institutions now start with a lower case 'i', since the upper case was not always clear

      d.   Users are encouraged to use the *Add New Collection Wizard* here for new collections, but the old system to add (without uploading banners and buttons) remains, with some modifications to better support bulk adding of aggregations.

7. Deleting aggregations (from Aggregation Management and Single Aggregation Admin screens)
      a.   Corrected error message incorrectly reporting not all folders could be deleted, when they really were
      b.   Deleting an aggregation no longer fails if all children aggregations are also deleted
      c.   Adding an aggregation back that was previously deleted now works correctly

8. Updates to the System-Wide settings
      a.   Added and removed a number of settings (as noted in this documentation)
      b.   Settings viewer now handles deprecation of setting categories easily
      c.   Added new *isHosted* flag to the main configuration file which removes some system admin features and reserved them for a new class of admin, Host Administrator

9. Updates to the *Web Skin* management screen for system and portal admins
      a.   Checkbox to copy files from the current skin to the new web skin starts checked now
      b.   If the override banner flag is not set, the banner link input is hidden during input (and editing)
      c.   Removed strange O*verride Header/Footer* option for web skin – it now always overrides them
      d.   Can now delete an unused web skin
      e.   Changed the option to edit web skins to use the new single web skin management screen

10. Added new online web skin management screens to allow online editing of a single existing web skin
      a.   Can edit the basic information about the web skin (previously available in a pop-up from the main web skin management screen)
      b.   Can add support for new languages to the web skin and delete support
      c.   Can edit the headers and footers associated with the web skin for each language
      d.   Can edit the CSS stylesheet at the web skin level
      e.   When making changes through the new online interface, backups are kept for any files that are changed.
      f.   Can upload images (several at time), for inclusion in the headers, footers, or CSS
      g.   Can upload documents to associate with the web skin (such as tracking documents or the source files for the graphics used)

Installer

1. Created new MSI to install version 4.8.7, and upgrade database from previous versions
2. Simplified the process of installing a new instance – at least for non-production testing
3. Signed MSI with certificate for authenticity
4. Installs the SobekCM Builder scheduled task now

SobekCM Builder

1. Additional support for a single builder running over multiple instances

2. Builder modularization - Builder has been completely rewritten as a modularized workflow with complete ability to customize throughout the workflow. (Technical white paper to follow)
3. Can also reference external libraries to add your own custom steps into the builder
4. Builder as Windows Service - Currently testing the builder as a windows service, rather than a scheduled task

REST-based Web Architecture

1. Divided the former *SobekCM_Library* project into a number of projects to support better separation between the HTML generation routines and classes and the data access/update classes.
2. Added new SobekCM Engine Microservice handler and SobekCM Engine Client classes
    a. Engine is a loose collection of microservice endpoints, defined in XML
    b. Client class is also based on a different XML file, so external endpoints could be utilized if desired
    c. Performed performance testing and completed inclusion of two message types
        i. Serves JSON through the very quick JIL library
        ii. Serves protobuf messages through the even quicker protobuf-net library for inter-process communication
    d. Handles POST as well as GET REST request/endpoints
    e. Endpoints can be restricted by IP address
3. Exposed a large number of aggregation and web skin specific endpoints via JSON
    a. Optimized aggregation and web skin classes for better JSON and protobuf generation.
    b. Began using much of the final classes, without depending on the REST API code