Colorado Water Conservation Board's Data Harvesting Initiative: Design Recommendation & Cost Submission Dated 08/01/2008

Version : 3.0

Distribution : Susan Lesovsky

Prepared by : Solany™

1.0 Introduction

1.1 Purpose

This document contains our recommendation of the tools required to design, implement and support the Colorado Water Conservation Board (CWCB) Data Harvest Initiative (DHI). Included in our assessment are the following:

1. **Content Standardization**: Identification of high-level functionality specifications for the standardized Extensible Markup Language (XML) schema for information sharing, specifically the exchange of water content data;

2. **System Specific Specs**: Identification of the specific system specifications required to support free flowing content exchange, without compromising or altering the unique end user experience;

3. **Technical Architecture Specs**: Recommended development architecture for each of the three functional modules – Laserfiche, CONTENTdm, and future system integration.

In addition, we have provided an overview of the following:

1. **Risks**: Key project risks that must be managed throughout the overall development and implementation of the DHI;

2. **External Dependencies**: Identification of external dependencies which will impact the successful development and implementation of the DHI is dependent;

3. **Time & Resources Required**: Recommended project team and consulting resources required;

4. **Timeline**: Overall project timeline;

5. **Leverage Opportunities**: Leverage opportunities to benefit the public "customer base" by adopting data sharing standards and integrating with other entities.

1.2 Structure of the Document

The key findings are summarized in the Management Summary – Section 2.0

Section	Title	Note
2.0	Management Summary	A summary of the functional and architectural design specification for the DHI
3.0	Overall DHI Goals	High-level functionality requirements overview with specific exclusions identified
4.0	CWCB System Specifications	On overview of the CWCB functional requirements
5.0	CSU System Specifications	On overview of the CSU functional requirements
6.0	New Entities - System Specifications	An overview of new entities functional requirements
7.0	Architecture Recommendations	An overview of the technical architecture considerations – graphical depiction of information flows for each tool / phase
8.0	External Dependencies	An outline of internal and external dependencies that may affect the overall success and timeliness of tool delivery
9.0	Leveraging Points Future Growth	Identification of leverage opportunities for items developed for the initial CWCB → CSU Toolset that will be of benefit to the long term DHI
10.0	Time / Resources Required	Recommended team structure and skills required as well as an overview of consulting and development costs
11.0	Acronym Reference	Commonly used acronyms

2.0 Management Summary

3.0 Data HI Summary

Report Ref.	Issue	Description
3.1	Summary Functionality	There are three basic requirements for the DHI: 1.0 CWCB Data Collection Access 2.0 CSU Data Collection Access 3.0 Future Data Collection Access
3.1	Selection of Data Standard	We recommend adopting the Dublin Core Metadata Initiative (DCMI) to facilitate the search, retrieval and management of water information content.
3.2	Limitations on What 'Can' Be Developed	There is a lack of information regarding the long term system integration of future entities in relationship to utilized Content Management Systems.

4.0 CWCB System Specifications

Report Ref.	Issue	Description
4.1	Selection of Indexing Platform	We recommend employing a reconfigured version of the Laserfiche system – accomplishes a leveraging of existing platforms, infrastructure, tools, and skill sets.
4.2	High-level Functionality Requirements	The Laserfiche system would be reconfigured to handle: 1) Integrated search capabilities 2) Mapping to new DCMI metadata fields 3) Periodic adjustments to both internal Laserfiche Field Values as well as future DCMI changes 4) Ability to view external images from within the Laserfiche Weblink browser
4.3	Application Programming Interface (API) Utilization	An overview of technical requirements/enhancements that will be required to the Laserfiche API in order to meet DHI requirements
4.4	Design Specifications - Detail	An overview of detailed design specifications for the Laserfiche system.
4.5	Technical User Interface	User interface functionality overview.

5.0 CSU System Specifications

Report Ref.	Issue	Description
5.1	High-Level Functionality Requirements	The CONTENTdm system will be required to: 1) Receive DCMI formatted XML content 2) Export DCMI formatted XML file 3) Display Laserfiche content within native browser
5.2	API Utilization	An overview of technical requirements/enhancements that will be required for the CONTENTdm API in order to meet DHI requirements

6.0 New Entities – System Specifications

Report Ref.	Issue	Description
6.1	High Level Functionality	ALL Content Management systems joining the DHI will: 1) Provide API to the DHI participating parties 2) Provide and accept XML data in specific format 3) Provide a web based solution
6.2	Data Input Specifications	Pull automated DHI field values and translate to current system schema
6.3	API Utilization	Allow for image viewing within current application via URL (web) pointer.

7.0 Overall System Architecture and Process

Report Ref.	Issue	Description
7.1	Architecture Overview	Guidelines used to select DHI architecture.
7.2	Architecture Modifications	Laserfiche and CONTENTdm will require enhancements throughout the DHI development phase as outlined.
7.3	Data Sources	Import and export data feeds and their format(s) are identified.

8.0 External Dependencies / Project Risks

Report Ref.	Issue	Description
8.1	Ongoing Clarification of Scope	.
8.2	Data Availability	Database Administrator (DBA) requirements.
8.3	Other Issues	Staff availability, scope creep, data architecture.

9.0 Leverage Points for Future Growth

Report Ref.	Issue	Description
9.1	Metadata Platform	Leveraging the DCMI XML standard toolset and design for DHI growth.
9.2	Communications Materials	Leveraging the communications materials and efforts throughout all phases of DHI to insure long-term participation and buy-in.
9.3	Team Management	Leveraging development skills and resources throughout all phases of DHI deployment.
9.4	Software Development Road Maps	Leveraging Software Development of both the Laserfiche and CONTENTdm systems for long-term DHI buy-in.

10.0 Development Effort / Fee Estimate

Report Ref.	Issue	Description
10.1	Personnel Requirements	Organizational chart outlining team size, roles, and responsibilities.
10.1	Phase I	Laserfiche API development and database configuration.
10.2	Phase II	CONTENTdm API development.
10.3	Phase III	Continued third party DHI connections.

11.0 List of Acronyms

Acronym	Description
API	Application Programming Interface
CWCB	Colorado Water Conservation Board
CSU	Colorado State University
DBA	Database Administrator
DCMI	Dublin Core Metadata Initiative
DHI	Data Harvest Initiative
DNR	Department of Natural Resources

Acronym	Description
ECM	Enterprise Content Management
OIT	Office of Information Technology
SQL	Standard Query Language
TB	Terabyte
URL	Uniform Resource Locator

DHI Overview

3.1 Summary Functionality

The DHI is required to perform three major functions:

CWCB Data Collection access (See Section 4.0 for Detailed Functionality Analysis)

- Provide integrated search and retrieval functionality between the existing CWCB Repository located within the Colorado Department of Natural Resources (DNR) Laserfiche Content Management System and other entities requiring access to water information;
- Translate all existing CWCB metadata to published DCMI standard as needed;
- Provide standardized data import and export functionality via XML;
- Systematically insure all transmitted data is up to date;

CSU Data Collection access (See Section 5.0 for Detailed Functionality Analysis)

- Provide integrated search and retrieval functionality between the existing CSU repository located within the university's CONTENTdm Content Management System and other entities requiring access to water information;
- Translate all existing CSU metadata to published DCMI standard as needed;
- Provide standardized data import and export functionality via XML;

Future Data Collections access (See Section 6.0 for Detailed Functionality Analysis)

- Provide integrated search and retrieval functionality between all member of the DHI;
- Accept metadata information and image pointer information seamlessly from an internal web based Content Management System;

Other Requirements

- Assess the accuracy and flow of data between systems and provide recommendations for the long term DHI using best practices and previous experience in data consolidation, database management, database optimization, and content management search and retrieval;

3.2 Selection of Data Standard

The DHI was developed to standardize the exchange of relevant water data/content. To accomplish this goal, the proposed architecture must be scalable and flexible. After evaluating leading technology standards and protocols, we are suggesting utilizing XML formatted to the DCMI to share data as its use allows systems utilizing different platforms

and technologies to share data easily and efficiently. We are recommending Dublin Core XML for the following reasons:

1) **Leverage Dublin Core:** Utilizing the DCMI standard, the framework for developing the database structure will make data sharing possible. The database structure will be developed utilizing the Dublin Core XML standard schema. Once an institution decides they want to share and search data with the DHI, they will utilize XML and the DCMI standard to easily transform their content making it accessible to anyone conforming to the DCMI standard;

2) **Proven Format**: The DCMI is an organization dedicated to promoting the widespread adoption of interoperable metadata standards and developing specialized metadata vocabularies for describing resources that enable more intelligent information discovery systems;

3) **Proven Methodology**: The DCMI organization was created with one major goal – *Standardize Content without bias.* The following characteristics prove highlight this mandate:

   - ***Independent***: DCMI is not controlled by specific commercial or other interests and is not biased towards specific domains nor does it mandate specific technical solutions.

   - ***International***: DCMI encourages participation from organizations anywhere in the world, respecting linguistic and cultural differences.

   - ***Influenceable***: DCMI is an open organization aiming at building consensus among the participants organizations; there are no prerequisites for participation.

4) **Maintainable:** The development and maintenance of a core set of metadata terms (the DCMI Metadata Terms) continues to be one of the main activities of DCMI. In addition, DCMI is developing guidelines and procedures to help implementers define and describe their usage of Dublin Core metadata in the form of application profiles. The work is done in a work structure that provides discussions and cooperation platforms for specific communities (e.g. education, government information, corporate knowledge management) or specific interests (e.g. technical architecture, accessibility);

5) **CSU Expertise:** Internal experience relating to the Dublin Core standard can be utilized and leveraged for the CWCB ←→ CSU pilot;

6) **Proven Standards:** DCMI provides simple standards to facilitate the finding, sharing and management of information. DCMI does this by:

   - Development and maintenance of international standards for describing resources;
   - Supporting a worldwide community of users and developers;
   - Promoting widespread use of Dublin Core solutions.

3.3 Limitations on What 'Can' Be Developed

Lack of Future System Identification: Our analysis of DHI requirements indicates that it is impossible to predict all of the potential system integration points of future members as there are infinite possibilities regarding future integrations – platforms, systems, database configurations and front-end applications. While the solution proposed in 100% platform and system independent, some development will be needed at the local level for system connectivity to DHI participants. A decision was made to recommend a phased approach within DNR and CWCB to 'smooth' the initial effects of development and system database configuration.

4.0 CWCB System Specifications

4.1 Selection of Metadata Platform

The CWCB has been involved with Enterprise Content Management (ECM) since 1999. Starting with document imaging, the CWCB has grown their data collection to over 75,000 documents and over 1,000,000 images. This collection is comprised of over 20+ electronic document types and is available free of charge to the public.

In an effort to reduce the duplication of content while at the same time increasing customer satisfaction, the CWCB designed the DHI to include other entities, mainly CSU, to join in the sharing of water information for public use. A tool must be developed to handle the new data sharing approach. In addition a standard must be developed and adopted to insure data integrity across entities.

As part of our analysis for determining the most appropriate tool to accomplish the harvesting of data, we considered two primary options:

1. **The 'Data Warehouse' Solution**: Using Standard Query Language (SQL), or similar database platform, create a tool that would import raw data from all participating entities, synchronize and maintain index tables, and server requested content to corresponding public client. In addition, this option would demand the creation of a 'new' system that would need to be built, housed and managed with two way conduits to each participating system – ultimately creating an additional layer of complexity;

2. **Create additional SQL Database tables within Existing Laserfiche System**: The existing Laserfiche system being used to manage over 2 terabytes (TB) of data for the DNR. By developing additional tables within Laserfiche, CWCB will obtain almost instant search access across all of the DNR repositories.

We have concluded that ***Option 2: Create additional SQL Database tables within Existing Laserfiche System*** is the recommended approach for the following reasons:

1) **Leverage existing standards / technology:** We can leverage the fact that an existing ECM system is in place, maintained, supported, and has been proven to process monthly transactions over the web to the public user committee with integrity and completeness. No further internal corporate audit would be required;

2) **Low Overhead Costs:** The system is a third party, proven tool, used by over 25,000 organizations – and is in a current stable state;

3) **Low Maintenance:** The resources required to maintain this system are fairly small given that it is currently up and running without difficulty;

4) **Minimal Training:** No further training would be required to operate the existing system;

5) **Internal Expertise:** Internal expertise can be leveraged for the reconfiguration effort;

6) **Leverage Laserfiche:** The parameters within Laserfiche can be updated to handle the unique requirements of the DHI – mostly leveraging internal DNR/Office of Information Technology (OIT) talent.

4.2 High-level Functionality Requirements

The DHI should provide the following functionality:

1) **Integrated Search Capabilities**: The ability to search available content of DHI participants through existing front-end applications from various entry points. It is important to be able to provide expansive search functionality across multiple data repositories while eliminating the need for end user training – users still utilize familiar systems and functions;

2) **Mapping to New DCMI metadata fields:** The CWCB has created over 100 field values to define its content within Laserfiche. The DHI will provide the ability to translate these field definitions to the Dublin Core standard;

3) **Periodic Adjustments to both internal Laserfiche Field Values as well as future DCMI changes**: Manage internal template changes based upon content, with the understanding that defined field changes (new and existing) will need to map to the relevant Dublin Core defined fields for enterprise search;

4) **View external images from within the Laserfiche Weblink:** Access search result location and render image within Laserfiche via URLs (web pointers).

4.3 Changes Required to Laserfiche - API Utilization

The CWCB will customize the Laserfiche application to allow for other entities to retrieve data, as well as render data from other participants within the existing Laserfiche framework. The following components will be utilized:

- LaserficheDeveloper Toolkit
- LaserficheServer Objects (LFSO)
- LaserficheImage Enable (LFAPI)
- LaserficheComponent Object Model (COM)

4.3.1 Actions Required

- Define new configurable interface between the Laserfiche and Dublin Core Defined Fields;
- Specify requirements for all data required from CWCB;
- Create new fields within Laserfiche if needed;
- Create new templates if necessary to organize existing data in line with standards;

- Provide revised requirements from above to DNR;
- Communicate with the Division of Water Resources (DWR) resources to include their internal water content, in Phase I of the initiative;
- Modify Laserfiche SQL database to receive data as defined from Dublin Core standardization;
- Implement database maintenance plan to update new index tables;
- Develop search capabilities to include internal and external repositories;
- Revise Weblink interface with respect to expanded search results to handle external image rendering;
- Develop process to ‘publish’ relevant water data, in Dublin Core XML standard, to DHI participants

5.0 CSU System Specifications

5.1 High-Level Functionality Requirements

CSU is currently leveraging the DCMI standard within their content management system. Having adopted this standard with the CONTENTdm product, CSU's implementation of the DHI is significantly reduced. In order to integrate with the DHI, the CONTENTdm system will need to be modified to utilize the Laserfiche API to retrieve data from the Laserfiche application and database. The following requirements must be met to integrate with the DHI:

1. Update existing database utilizing DCMI formatted XML data supplied by CWCB and integrate with CONTENTdm search capabilities;

2. Export DCMI formatted XML data to the DHI;

3. Customize CONTENTdm utilizing the Laserfiche API model to retrieve data from CWCB. This will also create a building block for sharing data with any other institution that uses the Laserfiche application model.

5.2 API Utilization

CSU must have the ability the render imagines within their current web environment. They will do this by utilizing the Laserfiche API model to customize CONTENTdm to display images that reside at CWCB within the Laserfiche application. It's extremely important for the developers at both CSU & CWCB to communicate their needs in order to successfully integrate the two applications. The development resources at each institution will be responsible for changes relating to their applications. The items below define the Laserfiche API.

- LaserficheDeveloper Toolkit
- LaserficheServer Objects (LFSO)
- LaserficheImage Enable (LFAPI)
- LaserficheComponent Object Model (COM)

5.2.1 Actions Required

- Import CWCB data from XML documents supplied by CWCB.

- Export water resource data that will be imported into CWCB's system in XML format.

- Decide whether or not to create a crossover environment to map CWCB data to be utilized in the CONTENTdm system.

- Customize CONTENTdm to retrieve images from the CWCB's Laserfiche application leveraging the Laserfiche API.

- Share the CONTENTdm API with CWCB so they can utilize to retrieve images from CSU and display them in the Laserfiche application.

- The API should include the following

- Component Object Model
- Developers Toolkit
- Server Objects
- And any other relevant information pertaining to the CONTENTdm API.

6.0 New Entities – System Specifications

6.1 High Level Functionality

As the DHI grows, new entities will come online. While each new participant will have a varying degree of internal system complexity, entry to the DHI will be standardized. All joining parties will:

1. Provide an open defined API to all of the existing DHI community;

2. Customize their existing system to be able to import and export XML data in the defined Dublin Core format. Database tables will also need to be built to provide translation between the existing data definition and the standard DHI definition;

3. Have the ability to receive URL (web) pointers to existing images from the DHI community. New entities must also grant access to their image and metadata information.

6.2 Data Input / Output Specifications

All DHI data delivered to DHI entities will follow the DCMI format. Interested parties will have to develop to this standard in order to join the initiative. Since the DCMI schema was designed to account for complete content sharing, all joining entities will also have to output all data to the DCMI XML standard.

6.3 API Utilization

All DHI members must have the ability to render images within their current web environment.

7.0 Overall System Architecture

7.1 Architecture Overview

The proposed architecture within this section outlines and defines how to expand search capabilities of the CWCB and CSU and allow users of either entity in either application to retrieve data from the other entity.

This document will make recommendations on how to integrate the two systems and develop a set of standards for which other institutions could adopt and utilize to also expand their search capabilities and share their data collections. In order to make a scalable solution we are suggesting utilizing XML to share data. The use of XML allows systems utilizing different platforms and technologies to share data easily and efficiently. Utilizing the DCMI standard as the framework for developing the database structure will make this possible.

The database structure will be developed utilizing XML schema files suggested under the Dublin Core standard. Once an institution decides they want to share their data and search other institutions, they will use XML and the DCMI standards to easily transform their data, making it accessible to anyone conforming to the DCMI standard. The two initial entities, CSU & the CWCB, utilize different platforms and systems, creating the ideal test environment for bringing on other institutions regardless of the technologies they utilize.

This document will include a series of Use Case Analysis documents and Functional Specification documents to better answer the questions of how, from a user perspective and developer perspective. The import of the data from each entity utilizing the DCMI standard will only bring in the Metadata elements necessary for searching, and the identifier record will point back to the data's physical location; the only foreign data that will be uploaded to each system is a searchable snapshot. The Laserfiche application will require modifications to search using the DCMI standard data format. A translation table will be established and will update on a predetermined schedule to allow for easy searching of data at CSU. This architecture was created using several guidelines:

- Leverage existing infrastructure and systems wherever possible;
- Minimize the data required to be transferred between systems;
- Provide access to all relevant data without leaving current system;
- Create a stable and robust environment that will scale as new entities join the initiative;
- Eliminate need to end user training.

7.2 Phase I

Become Familiar with the Dublin Core Standard
The first phase will be a discovery phase for how to utilize the DCMI schemas to create a bulk import in to the database. A XML programmer will need to become intimate with the Dublin Core standard and create the XML necessary for data exchange with other institutions. Below is an explanation of the DCMI and why the organization created the

initiative. It is also possible to use “Crosswalks” to migrate from one standard to another allowing for data sharing with institutions that might adopt a different standard. Next, it is critical to become proficient in the following areas to ensure proper implementation of DCMI. The XML programmer will need a strong understanding of the following concepts and languages.

- **DCMI Metadata terms**
  - http://dublincore.org/documents/dcmi-terms/
  - http://dublincore.org/documents/2003/04/02/dc-XML-guidelines/
  - http://dublincore.org/documents/2008/01/14/dces/
- **Extensible Markup Language (XML)**
  - http://www.w3.org/XML/
- **XML Schema**
  - http://www.w3.org/XML/Schema
- **DCMI Metadata expressed in XML Schema Language**
  - http://dublincore.org/schemas/XMLs/
- **Namespaces in XML**
  - http://www.w3.org/TR/1999/REC-XML-names-19990114/
- **Namespace Policy for the Dublin Core Metadata Initiative (DCMI)**
  - http://dublincore.org/documents/dcmi-namespace/

7.3 Phase II

Phase II
Collaboration with CSU is necessary to utilize their experience with implementation and adoption of DCMI. Understanding what elements of the standard they utilize in their XML format will be important for data exchange between CSU & the CWCB. Below are the metadata elements used in the XML document supplied by CSU and an example of one record in a CSU XML document.

1) CSU XML format
Metadata Fields utilized by CSU

- Title
- Creator
- Description
- Language
- Subject
- Publisher
- Source
- Date
- Type
- Format
- Identifier

Example of one record in XML using Dublin Core Standard:

```
<?XML version="1.0" encoding="ISO-8859-1" ?>
- <OAI-PMH>
  <responseDate>2008-04-17T22:23:29Z</responseDate>
  <request verb="ListRecords"
metadataPrefix="qdc">http://digital.library.colostate.edu/cgi-bin/oai.exe</request>
- <ListRecords>
- <record>
```

- <header>
<identifier>**oai:digital.library.colostate.edu:cowaters/1951**</identifier>
<datestamp>**2005-09-29**</datestamp>
<setSpec>**cowaters**</setSpec>
</header>
- <metadata>
- <qdc:qualifieddc XMLns:qdc="http://epubs.cclrc.ac.uk/XMLns/qdc/" XMLns:dc="http://purl.org/dc/elements/1.1/" XMLns:dcterms="http://purl.org/dc/terms/" XMLns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="**http://epubs.cclrc.ac.uk/XMLns/qdc/ http://epubs.cclrc.ac.uk/xsd/qdc.xsd**">
<dc:title>**Cache la Poudre Basin Study Extension. Executive summary**</dc:title>
<dc:creator>**Colorado Water Resources and Power Development Authority;
Northern Colorado Water Conservancy District (Colo.);
EBASCO Environmental (Firm)**</dc:creator>
<dc:description>**Executive summary of a study to refine the environmental, engineering, and economic analyses prepared during a basin-wide water resources investigation completed by the Colorado Water Resources and Power Development Authority and the Northern Colorado Water Conservancy District (Colo.) in January 1987. The initial basin study identified and evaluated potential structural and non-structural measures to provide for the efficient and environmentally sound develpment of water and hydroelectric power resources in the Cache la Poudre Basin.**</dc:description>
<dc:language>**eng**</dc:language>
<dcterms:spatial>**Colorado;
Cache la Poudre River Watershed;
Larimer County (Colo.);
 Weld County (Colo.);
South Platte River Watershed (Colo. and Neb.)**</dcterms:spatial>
<dcterms:temporal>**1990**</dcterms:temporal>
<dc:subject>**Water resources development -- Colorado -- Cache la Poudre River Watershed;
 Reservoirs -- Environmental aspects -- Colorado -- Cache la Poudre River Watershed;
Hydroelectric power plants -- Environmental aspects -- Colorado -- Cache la Poudre River Watershed;
Environmental impact analysis -- Colorado -- Cache la Poudre River Watershed;
South Platte River Watershed (Colo. and Neb.)**</dc:subject>
<dc:publisher>**Colorado State University. Libraries**</dc:publisher>
<dc:source>**CWR1.2/C11/1990/v.3**</dc:source>
<dc:date>**1990-12**</dc:date>
<dc:type>**text;
map**</dc:type>
<dc:format>**image/jpeg**</dc:format>
<dc:date>**2005**</dc:date>
<dcterms:isPartOf>**Water Resources Archive;
Ival V. Goslin Water Resources Collection**</dcterms:isPartOf>

<dc:identifier>**http://digital.library.colostate.edu/cdm4/document.php?CISOROOT=/cowaters&CISOPTR=1951**</dc:identifier>
<dc:rights>**http://www.acns.colostate.edu/?page=copyright**</dc:rights>
<dc:identifier>**http://digital.library.colostate.edu/u?/cowaters,1951**</dc:identifier>
</qdc:qualifieddc>
</metadata>
</record>
- <record>

7.4 Phase III

The next phase of the project will require the customization of the Laserfiche application to utilize the API supplied by CSU or any other entity to display foreign data in the Laserfiche application. It will require seeking not only its own data collection but also the metadata table/tables created utilizing the DCMI standard to locate and retrieve data from foreign systems. Below are the documents supplied by each institution, explaining the elements of each institution's API to be used for development. A programmer from each entity will be responsible for modifying the CONTENTdm and Laserfiche applications, respectively to locate and display data from each other. The documents below are the framework for that development. There is a diagram showing how the applications will retrieve the data and display it in each of the applications. A Use Case for the actors involved and how they would seek data is also included

1) Application Programming Interface (API) for CSU and CONTENTdm

This section outlines the necessary components for utilizing the CONTENTdm API. This will allow CWCB to seek the data from CONTENTdm and display them in the Laserfiche application. The process will be transparent to the end user.

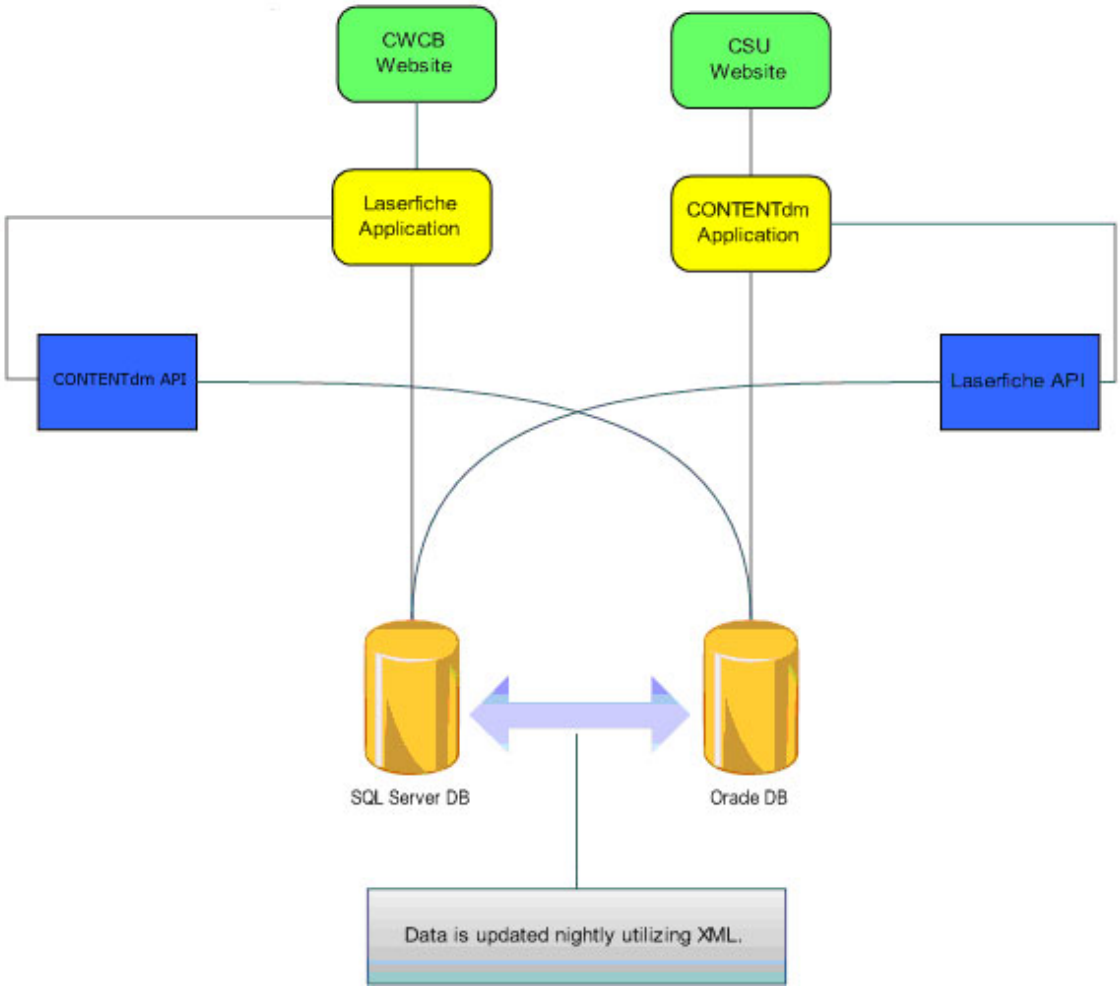
2) Application Programming Interface (API) for CWCB and Laserfiche

This section outlines all the relevant development information necessary to implement data retrieval from the CWCB's Laserfiche system. The Laserfiche API components are listed below. There is a supporting document for each, which CSU will utilize to extract data from the CWCB and display it in CONTENTdm.

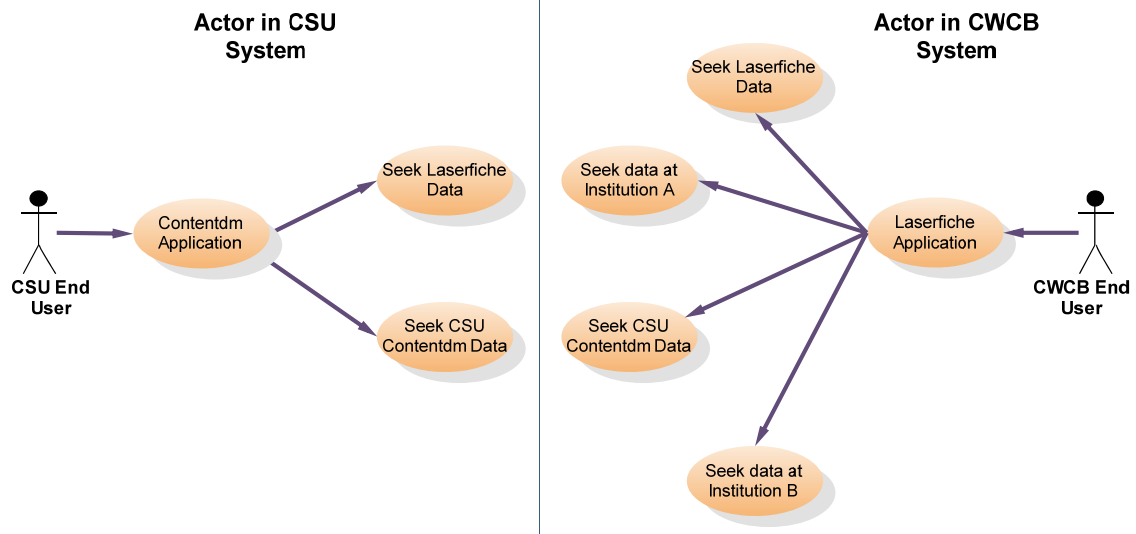
- Lasterfiche Developer Toolkit
- Laserfiche Server Objects (LFSO)
- Laserfiche Image Enable (LFAPI)
- Laserfiche Component Object Model (COM)

a. **Application Design Diagram**

The diagram below shows how the application for each institution would first find data in their own database from the metadata harvest. Then go out and get the images and relevant data from the other institution database if that's where the data is located. If a user seeks data that is contained in the harvest from CSU then the retrieval would utilize the Contentdm API to display the images in the Laserfiche application.



b. **Use Case Diagram**
The Use Case Diagram below shows the users at CSU and the CWCB as the actors and where they would be retrieving data.



7.5 Phase IV

The last phase includes the data harvest and creating the schedule for keeping the databases up to date on both sides. The use of XML files that utilize the DCMI standard are the vehicle for carrying out the data population. Below there is a description of the three components of harvesting the data with a diagram to illustrate the process in simplistic terms.

1) Data sharing management

a. **Initial Data Population**

i. **Bulk upload of Metadata from CSU to the CWCB:** Utilize existing data import procedures to pull XML data into DB utilizing Dublin Core standard.

ii. **Bulk upload of Metadata from the CWCB to CSU:** Create data import procedures to pull XML data into DB utilizing Dublin Core standard.

b. **Update Schedule**

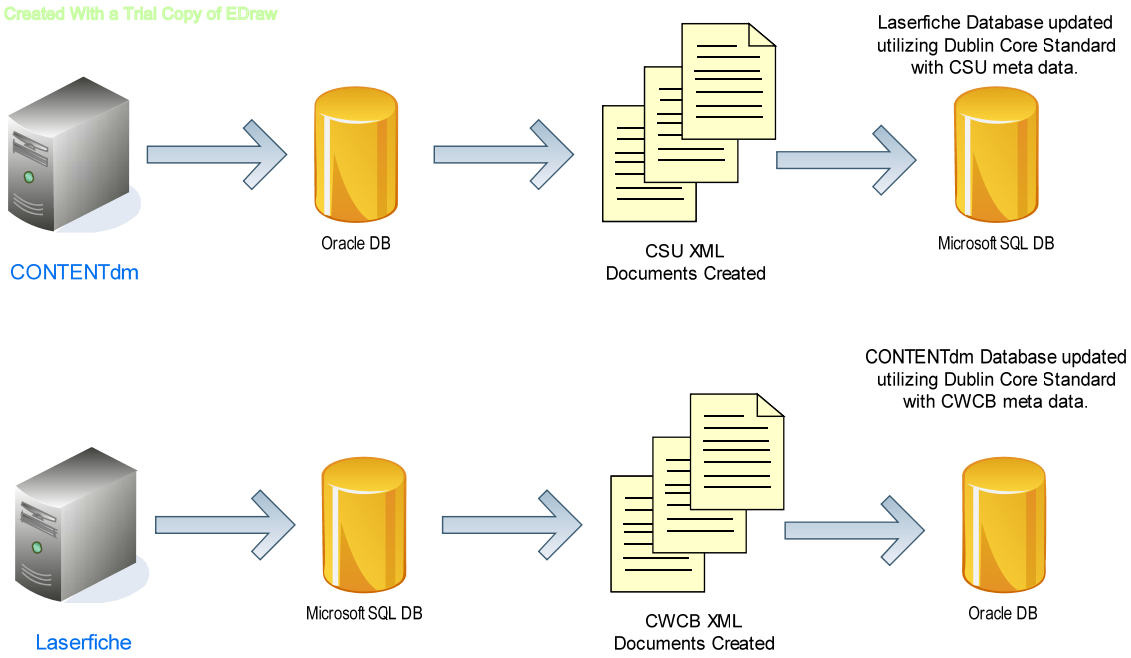
Create automated schedule to check for updates to the metadata collection for harvesting. (Daily, Weekly, Monthly, to meet the needs of each institution.

c. **Archive Schedule**

Remove outdated or irrelevant data from the metadata search collection on a pre-defined schedule.

2) Data Harvest Diagram

a. The data harvest diagram below shows how the initial population and the scheduled updates would pull data. The data would be extracted from the database in XML format utilizing the Dublin Core Standard and its schema definitions and import that data into the Meta data collection at each institution. The scheduled updates would utilize the same


Created With a Trial Copy of EDraw
Laserfiche Database updated utilizing Dublin Core Standard with CSU meta data.
Oracle DB
CSU XML Documents Created
Microsoft SQL DB
CONTENTdm
CONTENTdm Database updated utilizing Dublin Core Standard with CWCB meta data.
Microsoft SQL DB
CWCB XML Documents Created
Oracle DB
Laserfiche

8.0 External Dependencies / Project Risks

8.1 Ongoing Clarification of Scope

Define – High level clarification of deliverable, as entities join DHI:

8.2 Data Availability

There will be a need for the DBA of CSU and the CWCB to collaborate the data population. Each will have to create the XML documents defined utilizing the DCMI schema. In order to keep the project timeline in check management of this data exchange will need to be closely managed.

8.3 Other Issues

Staff Availability: Programmers will need to be available from the CWCB and CSU in order to manage the information needed for the system integration between CONTENTdm and Laserfiche. Selected project management personnel will need to make sure there is clear assignment of tasks for each institution.

Scope Creep: To address this challenge, we will ensure that significant efforts will be made throughout the early phase of the project to 'show' users, not just 'tell' them about what they will eventually be receiving. From that point, strong version and date controls will be used to keep the early deliverables on track – while allowing for future enhancements to be logged as users become more familiar with the material.

Data Architecture: The database architecture will be built to suit the application it's servicing. The data population will utilize XML and the standard as defined by DCMI. The collection of cross system information will contain just the metadata elements.

9.0 Leverage Points for Future Growth

9.1 Metadata Platform

The Dublin Core Standard selected for the standardization of content is the ideal schema to be integrated with future participants of the DHI. This avoids the need to have a 'throw away' integration tool or solution. Adapting and developing to a standard of this magnitude establishes one to n functionality.

9.2 Communications Materials

The DHI will require a substantial amount of communication and rollout during the three phases of development. A consistent message and timeline should be presented in all communication and rollout materials – a message that will help prepare the user community for the DHI rollout. In addition, if some users are somewhat disappointed with the level of functionality provided by the DHI, the communication of the vision may help to address their concerns.

9.3 Team Management

Clearly, there is a benefit of the three-phase implementation approach being coordinated for a number of important reasons:

1) Many of the CWCB Content Management expertise required to support the initial DHI phases will become valuable assets as other entities join the DHI;
2) Even though some CWCB personnel will only be involved in the Design and Implementation of the DHI, their expertise may still be required to support the ongoing development efforts in the future;
3) Consultants used in the design and implementation of the DHI will acquire a substantial knowledge base on the processes and information requirements for both the CWCB and CSU.

In order to achieve the maximum leverage potential within the teams themselves, a cross utilization and mapping of staffing should be maintained with strong coordination through the execution of both initiatives.

9.4 Software Development Road Maps

In the process of developing the overall functionality of the DHI, detailed documentation of the overall design need be prepared – a foundation which will be beneficial to all of the future DHI participants and will go a long way in reducing buy in costs.

10.0 Development Effort / Fee Estimate

Estimated Development Costs	Price
Phase I	
API Development 500 Hours @ $100/hour	$50,000
Database Configuration 250 Hours @ $100/hour	$25,000
Phase II	
API Development 200 Hours @ $100/hour	$20,000
Database Configuration 200 Hours @ $100/hour	$20,000
Phase III	
Estimated cost to join DHI	$10,000
Total DHI Project Costs:	$125,000
	TOTAL AMOUNT
TOTAL PROJECT AMOUNT	$125,000