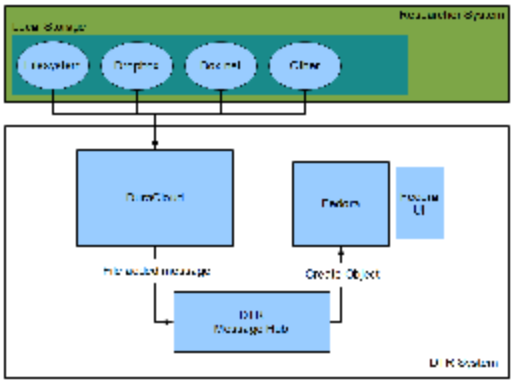
2011-11-10 - DTR Architecture



Pieces of DTR Architecture

Client Plug-Ins

- File system watcher (starting point)
- UI for selecting which files and directories need to be watched, can select from network shares as well
- Captures all metadata that is available from the file system
- Let users define metadata to be attached to files with particular characteristics (stored in a particular folder, certain size, certain type, etc)
- Could define a collection profile which include metadata
- Box.net data watcher
- Dropbox data watcher
- Omera, Electronic Lab Notebooks, other researcher tools which can be watched
- Each plug-in feeds content and metadata into DuraCloud

DuraCloud

- May need to add some additional information in the messaging to assist the message hub in being able to determine from where messages are originating

Message Hub w/ Fedora object creation plugin

- Message hub will allow for plugins to different systems where you may want to capture the events of an item being added to DuraCloud
- The only plug-in that we'll implement to start with the Fedora plug-in
- The message hub will listen for add/update/delete events from DuraCloud
- message hub includes a service bus, as well as the handlers that do work based on the messages encountered on the bus
- two plug-in points:
    1. content added to DuraCloud: plugins for systems (like Fedora) to have that content added to that system
    2. content added to other systems: plugins to retrieve/export that information (from a system like Fedora) and store it in DuraCloud

Fedora

- As Fedora objects are created, they are exported and stored back in DuraCloud
- This copy could be done by the message hub listening for Fedora ingest events and then exporting and storing the new object (with E datastreams) in DuraCloud
- Fedora would have its own local storage (on EBS)
- Would need to handle copying exported objects from Fedora to DuraCloud with the assumption that there may be a combination of Managed and External datastreams, since there may be objects/datastreams coming into the Fedora from another channel

UI/Fedora content display

Other UI components (e.g., plug-in configuration, etc.)

Client Plug-Ins (Retrieval side)

- Could poll DuraCloud for file changes and pull those files down locally

DTR meeting notes review

- citability, where does that fit? Seems like this would occur against the Fedora front end
- data management plans

- need to make a listing of the pieces of at DMP that DTR can help to fulfill
- provide a way to simplify the provisioning of DTR (for the lifetime that the DMP requires)
- there is need to specify locale in which data is stored (must be US, must not be US, etc)
- will need an encryption solution
- will need clear communication about what DTR does, how you can verify what it does, how the cloud works, and why DTR is a trusted solution
- will need to communicate: what happens if a storage provider folds, what happens if you want to stop using DuraCloud, what happens if a better solution comes along, what happens if DuraCloud fails, DuraCloud won't last forever
- would be very helpful if DTR could collect usage data to help determine how "valuable" or at least how well used some content is
- this may need to happen in DuraCloud, so that all access of the content (not just through Fedora) is captured
- would be useful to provide a document detailing where, when, and why cloud storage makes sense, but also indicating the situations where it's probably not the best solution. A simple pros-and-cons of the cloud listing would likely be useful
- search/discovery at the Fedora UI level is required
- How do we handle files which are updated? Do we create a new file in DuraCloud or overwrite the original? Do we create a new version of a datastream in Fedora, or a new datastream, or just overwrite the existing datastream?
- There may be no difference between the file system watcher and the dropbox monitor, as both would just watch a "local" path, then transfer anything changing there (along with any metadata that is collected) to DuraCloud
- If we were to use a single Fedora for an entire institution, is there existing capability to filter search results (via GSearch/Solr) to only show a researcher results for objects they have permission to access?
- Researchers would likely want to use their content in the cloud, rather than just downloading files. So a google docs kind of experience would be ideal.
- Does DTR want to handle versioning of files which come in?

Startup Flow

- Institution signs up for DTR
- We provision a DuraCloud instance for them
- We provision a message hub for them (may or may not be on its own instance)
- we provision a Fedora w/ Islandora/Hydra for them
- Researcher chooses to use DTR for data management on their project
- Institution's IT provides them with credentials for the DTR system
- The researcher fills out some simple forms (likely through an Islandora/Hydra front end) saying what the project is about
- The DTR system provisions a space (or set of spaces) for the project
- The researcher is provided with a listing of client tools they can use to hook up their existing machines/tools/equipment to DTR
- The researcher downloads and installs the appropriate client tools
- DTR begins monitoring for their content and transferring data into DuraCloud, then into Fedora
- The researcher can now search/browse in the Islandora/Hydra UI to see the files they have available and find things
- The researcher can now download any of their content to their local system
- If the appropriate client tools were chosen, their content is transferred automatically to other systems as it is added to DTR

Characterizing the content

This section is a work in progress

A given discipline or collection's data may be characterized along a number of dimensions:

Sizes: **Portable** (can be sent to a phone), **Small** (fits in a tablet), **Medium** (fits in a server), **Large** (needs dedicated infrastructure)

Formats: **Commodity** (.doc,.xls, etc), **Open** (free software exists to manipulate it), **Closed** (software exists for common platforms), **Proprietary** (special hardware required)

Access Pattern: **Continuous**, **Occasional**, **Archival**,

We can then apply the characteristics to individual **data items**, to **data sets**, to a typical user's **working set** or **lab notebook**, and to the **corpus** available in the discipline. This gives some guidelines into the architectural constraints and boundaries of the system, and what services can be offered.

In thinking about potential integrations, some useful shorthands may be **BasicInternet** (smtp/ftp/http etc), **SocialNetworks**, **WebAPIs**, **Repository** (SWORD, etc), **OnlineBackup** (Dropbox, box.net, etc).

Potential services can include **Preservation**, **MetadataIndexing**, **Linking**, **ContentIndexing**, **FormatConversion**, **DataAnalysis**, **Visualization**, **Exhibit**, **Publishing**.

Examples:

Discipline	Size item/set/working /corpus	Formats	Potential Integrations	Potential Services	Useful Devices for Access
Example1	Portable/Portable/Small/Medium	Commodity	All	All	Phone, Tablet, workstation, server
Example2	Small/Small/Medium/Medium	Closed	All	Preservation, MetadataIndexing, Linking, Exhibit, Publishing	Tablet, workstation, server
Example3	Small/Medium/Large/Large	Open	All	All	Workstation, server
Example4	Any	Proprietary	OnlineBackup	Preservation, MetadataIndexing	Lab equipment

Questions to answer

1. What are the functional components of an idealized DTR system (The Pie that's in the Sky)?
2. What functions are in scope for the December, 2012 release
3. Which of the scoped functions will require a UI?
4. What is the comprehensive list of systems we might want to interface to to extract data from?

    a. What are the three first systems to build interfaces to?
5. What types of data will we support initially?
    a. What viewers / visualizers will we have?
6. Who are the collaborators we'll work with to develop the system?
    a. DG / Islandora
    b. Smithsonian
    c. researchers (grad students)
    d. ...
7. Who are the collaborators we will work with to understand researcher needs?
8. What is the first architecture (I.E. prototype to get started)
    a. What are the components to be integrated?
    b. How do we integrate those components?
9. What is the second architecture? (I.E. by end of the DTR initial funding)
    a. What dependencies do we have?
10. When would we like milestones to occur?
11. Where do we host the software?
12. What are the adjacent / competing environments, and how should we position against them?
    a. Islandora VRE
        i. Export to their visualizers?
        ii. Reuse components as appropriate?
    b. Smithsonian
        i. Use their data model?
    c. DataVerse
        i. Support cross-linking?
    d. Data Conservancy
        i. Support cross-linking?