# Transfer and Inventory Components of Developing Repository Services

LIBRARY OF CONGRESS

**Repository Development Group**
Office of Strategic Initiatives

**Leslie Johnston**
**Open Repositories 2009**

# STARTING DOWN A PATH TOWARDS BETTER CONTROL

- What are our most basic needs? What is the first step?
  - How do we know what we have, where it is, and who it belongs to?
  - How do we get files – new and legacy – from where they are to where they need to be?

# IDENTIFYING THE TRANSFER PROBLEM SPACE

- As part of its first phase repository development, the Library of Congress is working on solutions for a category of activities that we refer to as “Transfer.” At a high level, we define transfer as including the following human- and machine-performed tasks:
  - Adding digital content to the collections, whether from an external partner or created at LC;
  - Moving digital content between storage systems (external and internal);
  - Review of digital files for fixity, quality and/or authoritativeness; and
  - Inventorying and recording transfer life cycle events for digital files.

# RECENT TRANSFER EXPERIENCE

## During 2008 the Library of Congress received:

- 30 Tb from NDIIPP preservation partners, 20 Tb in Web Capture crawls to preserve identified web sites, 20 Tb from National Digital Newspaper Project (NDNP) partners, and 1 Tb from World Digital Library partners.
    - From 20 MB to over 2 Tb in a single transfer retrieved over the network.
- Dozens of hard drives with licensed, partner and vendor supplied content.
- All forms of content, some to be dark archived for preservation, some limited to Library use, and some to be made publicly available.
- There is also newly internally digitized content that has to be managed.

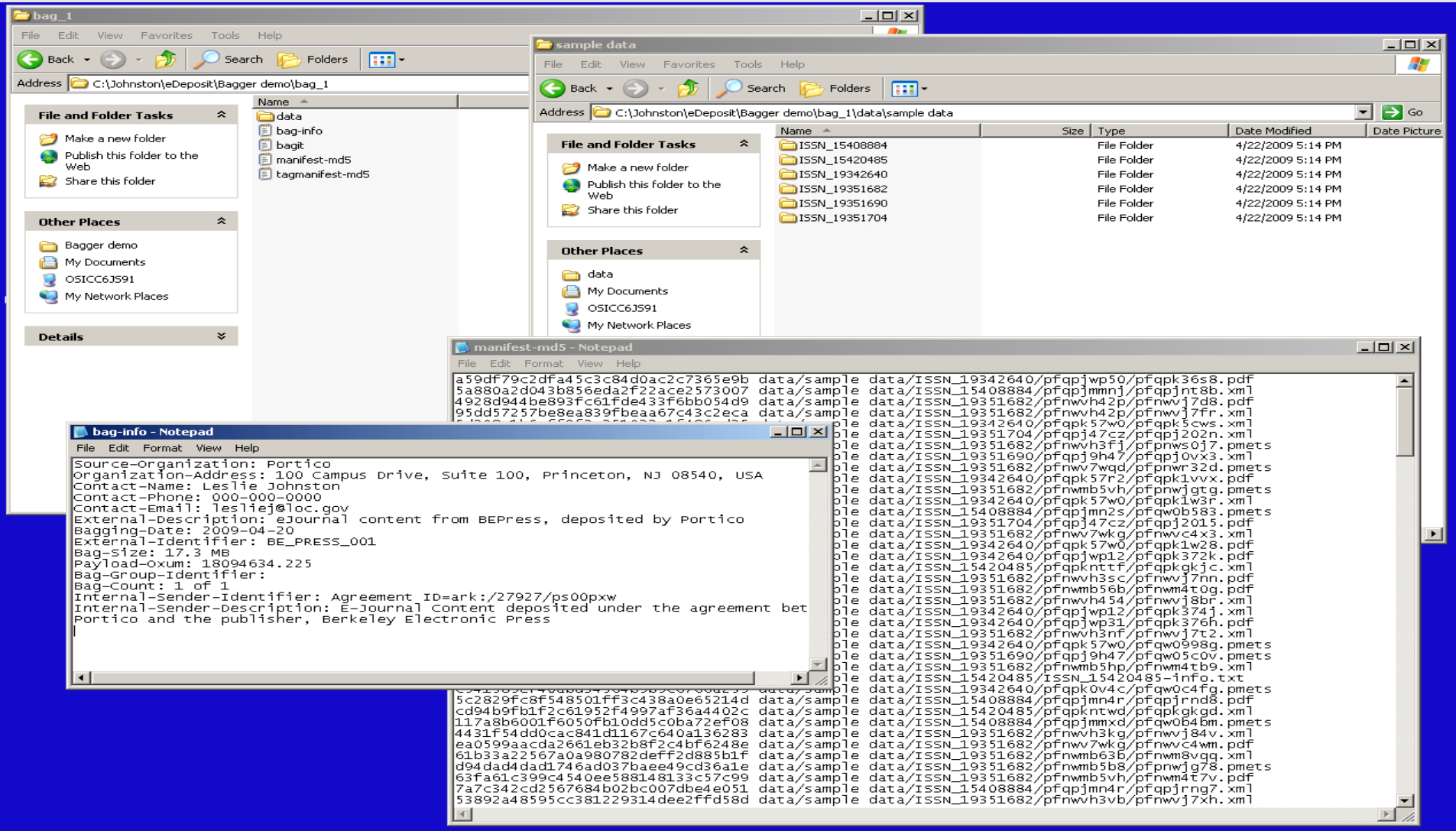
# DEVELOP A STANDARD TO OPTIMIZE TRANSFERS

**BagIt: A Packaging Specification for File Transfers.**

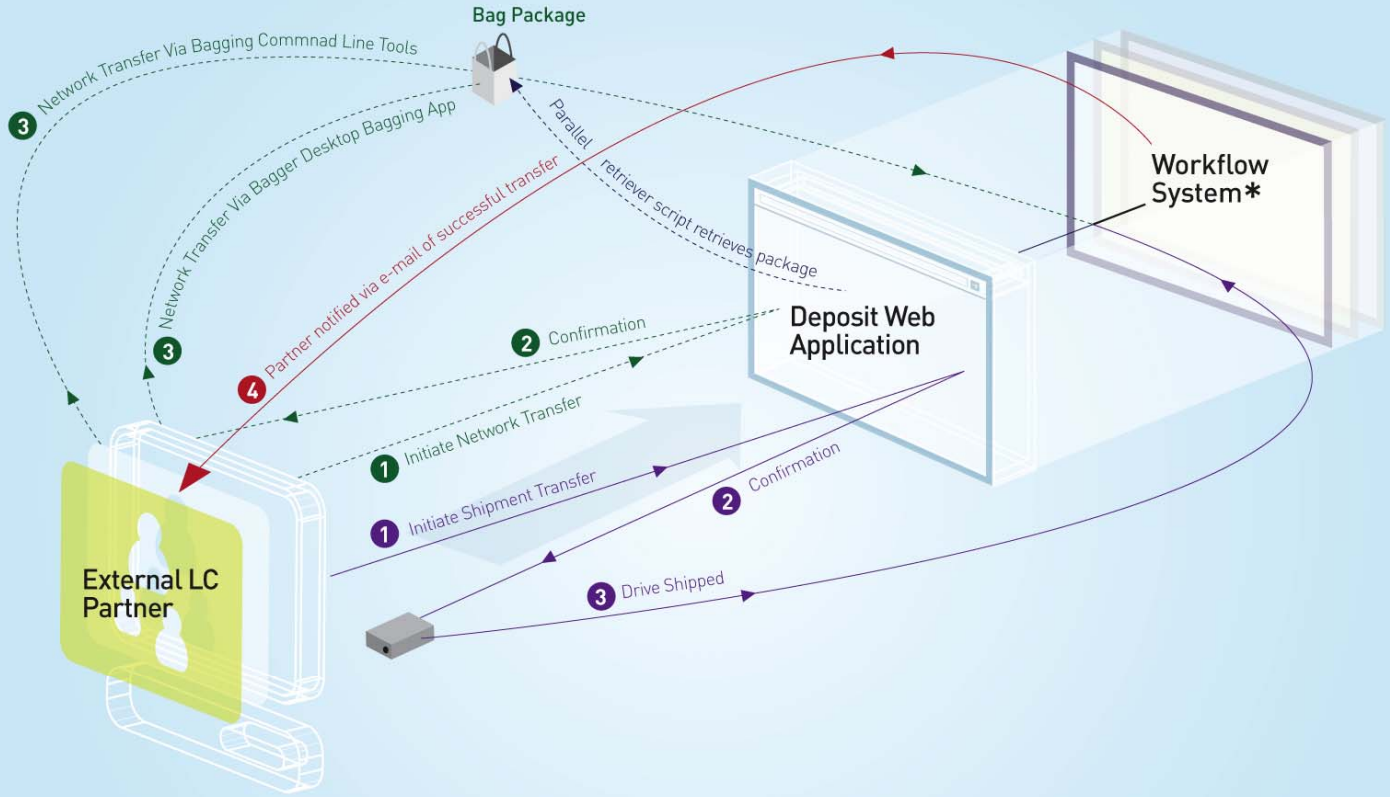
- Motivating use cases:
  - Transfer of content internally and between preservation partners.
  - Long-term storage of content.
- Needs:
  - Minimally self-identifying and self-describing packages.
  - Support for error detection and transfer optimization.
- Characteristics:
  - Low overhead
  - Content-type agnostic
  - Supported by off-the-shelf, easily supported tools.

  - http://www.digitalpreservation.gov/library/resources/tools/docs/bagitspec.pdf

# WHAT'S IN A BAG?



# TRANSFER ARCHITECTURE OVERVIEW




LEGEND

- Network Transfer
- Shipping Transfer

*** Workflow System**

Appropriate project-based **Workflow UI** (NDNP, NDIIPP, Internet Archive Capture,eDeposit, etc) launched.

Tasks vary by project, but includes **Bag validation** using Bag Validator script, file fixity checking using **Verifyit** script, format validation using JHOVE or DVV, transport of files to a production server, and transport of files to Sun29 for archival storage

# TRANSFER TOOL DEVELOPMENT

- Parallel Retriever script
  - Python script that operates multi-threaded  transfers using rsync, Wget, or cURL over multiple protocols (rsync, FTP, HTTP, HTTPS)
- Validation script
  - Python script that validates Bags against the BagIt specification
- VerifyIt script
  - Shell script used to verify that files are uncorrupted

# TRANSFER TOOL DEVELOPMENT

- BIL Java Library
  - Used for application and command line tool development
- Bagger Desktop application
  - Java Webstart graphical desktop tool to create/update/validate Bags
- LocDrop “Deposit” Web application
  - Django/Python/mySQL web app for use by partners to register and initiate transfers, whether shipping a hard drive or sending files over the network.
  - A prototype SWORD service has been created to support push transfers through LocDrop or the Bagger application. The LoC SWORD extension (BOB, or Bag-of-Bits) supports the deposit of compound, un-serialized resources (bob:acceptUnserialized element, accompanied by the bob:oxum element). Completion of the PUT is through a bob:completed element.

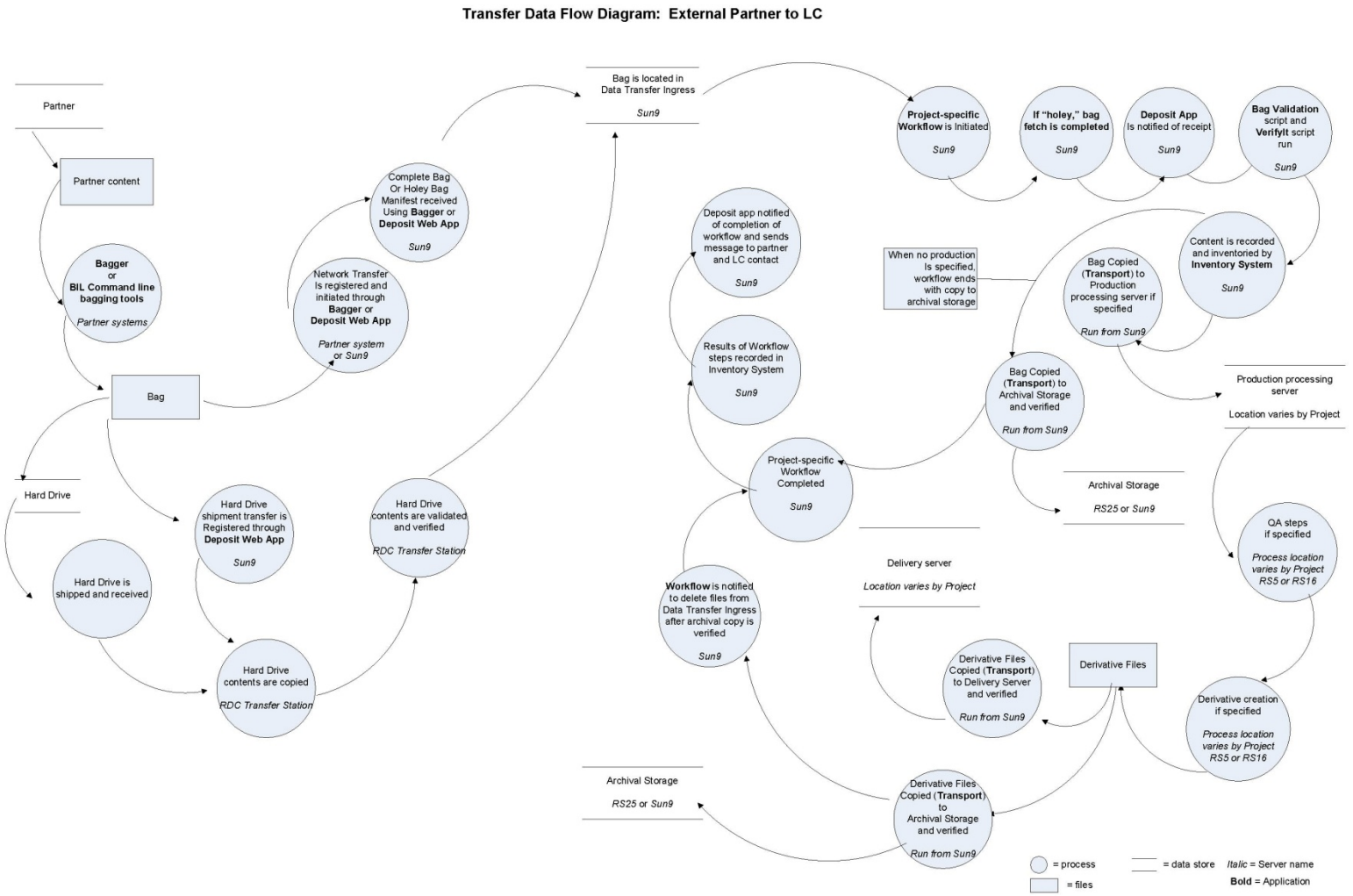
# INVENTORY TOOL DEVELOPMENT

- Inventory Tool
  - Record Package Events
    - Examples of Package Events include “Package Received Events,” which are recorded when a project receives a package; and “Package Accepted Events,” which are recorded when a project accepts curatorial responsibility for a package.
  - Record File Events.
    - Examples of File Events include “File Copy Events,” which are recorded when a package is copied from one File Location to another; and “Quality Review Events,” which are recorded when quality review is performed.
  - For legacy collections the Inventory Tool can be pointed at existing directories to package, checksum, and record life cycle events to bring the files under initial control.
  - The Inventory Tool is implemented on top of the BIL Java Library, with its Java objects mapped to a mySQL database using Hibernate for object-relational mapping.

# WORKFLOW DEVELOPMENT

- The Transfer components and Inventory Tool are tied together through multiple project-based Workflow systems.
  - Through case study development with stakeholders we identify the data flow and tasks to be performed.
  - Workflow tasks formalized through the system include transfer, validation by an format validation application, manual quality review inspection, and file copying to archival storage and production storage.
  - A workflow UI allows users to initiate, monitor and administer processes; and notify the workflow engine of the outcome of manual tasks, including task completion.
- The BIL Java Library supports the workflows. The underlying workflow engine is jBPM. jBPM Process Definition Language (jPDL) -- the native process definition language of jBPM -- is used to encode the workflow process steps as XML. The UI was developed using Spring MVC.

# DATA FLOW & WORK FLOWS



# WHY IS TRANSFER SO IMPORTANT?

- While our initial interest in this problem space came from the need to better manage transfers from external partners to the Library, the transfer and transport of files within the organization for the purpose of archiving, transformation, and delivery is an increasingly large part of daily operations.
- The digitization of an item can create one or hundreds of files, each of which might have many derivative versions, and which might reside in multiple locations simultaneously to serve different purposes.
- Developing tools to manage such transfer tasks reduce the number of tasks performed and tracked by humans, and automatically provides for the validation and verification of files with each transfer event.

# WHY IS INVENTORY SO IMPORTANT?

- Why are we looking at close integration between transfer and inventory functions?
  - Inventory services can bring several benefits, including collection risk assessment and storage infrastructure audits.
  - Realizing any benefits for effective data management relies on knowledge of data holdings.
  - Knowledge of file-level holdings and recording of life cycle events related to those files from the moment that they enter the collection and in every future action reduces future risk by storing information that can be used in discovery, assessment, and recovery if and when a failure occurs.

# WHY IS MODULAR SO IMPORTANT?

- Identifying needed services as modular rather than monolithic has allowed the Library of Congress to research and implement each of these functions in a more nimble way, all the while planning to fit those services into a larger scheme of repository services.
- The integration of modular transfer and inventory services as well as workflows allows for separation of tasks based on project or collection or format needs while supporting backend data integration where required.
- Modules can be independently re-implemented in the future when the need arises. This also allows for extensions to services and functionality that we have not yet even considered, let alone planned for.

# Is This a Repository?

- Not yet. These modular services do not yet equate to everything needed to call a system a repository.
  - There are only detached end-user discovery and delivery applications.
  - Descriptive metadata is not yet tracked with the media files.
  - There are currently no granular rights and access policies nor means to enforce them.
  - Preservation monitoring is not yet in place.
- But there is a set of services that equate to many aspects of "ingest" and "archiving" – the registry of a deposit activity, the controlled transfer and transport of files, and an inventory system that can be used to track files, record events in those files' life cycles, and provide basic file-level discovery and auditing.
- Through the Inventory tools we expect to be able to provide persistent access at a file level. In other words, it may not be a full-blown repository yet, but is the first stage in the development of a suite of tools to help the Library ensure long-term stewardship of its digital assets.

# OUTCOMES FOR THE LIBRARY

- The Library's first Open Source software release via SourceForge.
  - Transfer scripts and BIL Java library released.
  - http://sourceforge.net/projects/loc-xferutils/
- BagIt is in use with multiple NDIIPP partners, in the eDeposit pilot project, and for the packaging and transport of file packages internally.
- Gradual development of graphical workflow tools for all active projects
- The transfer of partner content has informed the Library's own preservation efforts, building our understanding about what we need to know about files and what events in their life cycle we need to record and track.
- The Inventory Tool will support the Library's initial efforts in a file-level preservation audit.
- Put all tools and services into full production during 2009

# THANK YOU!

Leslie Johnston
Library of Congress Repository Development Group
Office of Strategic Initiatives
lesliej@loc.gov