The Stanford Digital Repository, Version 2.0

Tom Cramer
Associate Director
Digital Library Systems & Services
October 2009
Agenda

• SDR to date
• Changes in the environment
• Version 2.0 design
• Three new, community developments:
  – Hydra Project
  – Blacklight
  – PASIG Repository Working Group
SDR is...

- a preservation system
- at version 1.7x
- in production since Dec 2006
- ready to jump to 2.0
Three Major Areas of Preservation Needs

- **Digital Library**
  - Legacy collections
  - Digitized collections
  - Licensed, locally loaded content
  - Born digital collections

- **Institutional Repository**
  - Research data,
  - Publications, dissertations,
  - Learning objects, university assets

- **External Depositors**
  - Publishers
  - Discipline-specific repositories
  - Reciprocal deposits with peer institutions

---

Google Books (’00s of TB)
Parker Manuscripts (75 TB)
MJF Media (50 TB)
NGDA (10 TB)
~30 other digi projects (15 TB)
Purchased collections (25 TB)
SDR 1.0: Core Repository vs. Repository Services

SDR Sample Use Case Diagram
SDR 1.0: Assumptions

Core Repository Functionality

• Preserving access to digital information over time
  ...through generations of technology obsolescence and change.

• Maintaining integrity of that information over time
  ...through generations of migration and reformatting.

Repository Services Functionality

• All (or almost all) user-facing services
• Enhanced access & delivery through applications
• Data mining, dry research, new indexing, e-science, etc.
• Federation
E.g., The Parker Project

- 559 Anglo-Saxon manuscripts, 200,000 pages
- For each page:
  - 22 MB JPEG2000 delivery surrogate
  - 22 MB JPEG2000 delivery surrogate
  - 110 MB submaster TIFF
  - 220 MB master TIFF

Parker.stanford.edu: Rich web application, tailored for general public, medievalists
SDR 1.0: Technical Architecture & Storage Platforms

- **Conversion Machine**: Creates the Transfer Manifest and the submission information package.
- **Ingest Gatekeeper**: Performs ingest validation and virus scanning.
- **Ingest**: Completes ingest and creates the archival information package.
- **Storage Preprocessing**
  - Online Storage: This software determines what goes online and what goes offline, according to the Transfer Manifest.
  - All metadata is stored separately from content, as well as with it.
  - Content is stored, along with metadata, in an archival information package.
- **Nearline and Offline Storage (Tape)**: All metadata and data goes on tape, and is stored in geographically distributed secure locations.
- **Access Layer**: "Air Gap": All access to the repository is through the access layer, which is read-only.
- **Federation**: Contains ADL, Aquifer, OAI, and SRB.
SDR today

- Version 1.7x stable & in production
- >80 TB of unique content
- >300 TB of managed data
- 15 different collections
- 5 content types: books, images, audio, manuscripts, GIS data
- Basic & proven ingest and admin capabilities
- Tens of successful retrievals (bulk and atomic)
- 3.5 storage migrations
SDR today – 1.x tasks still to be done

- More robust administration capabilities
  - Automation
  - Monitoring
  - Reporting
  - Preservation actions

- More robust access capabilities

- More collections to ingest
  - Content types
  - Size

- Increased activity on audits
What’s Changed: Staffing

Current staffing @ 4.5 FTE
- ‘07 reorg shifted 1 FTE to Web development
- ‘08 reorg shifted 3 FTE to Infrastructure & Metadata team
- ‘09 layoffs eliminated 1 direct and 1 indirect supporting position

Implications
- Reduction in dedicated staff
- Increase in complementary staff
- Challenges in continuity of knowledge
- Requirements to reduce & simplify SDR
What’s Changed: SDR in Stanford’s DL Ecosystem

Specialty applications provide context-specific, user-facing deposit, and access services tailored to content types and disciplines.

**Library Management Applications**
- EEMS (acquiring born digital content), digitization workflow, etc.

**Institutional Repository**
- ETDs, open access articles, faculty “papers”, research data, web sites, etc.

**National Geospatial Digital Archive (NGDA)**
- Geospatial data

**SULAIR Digital Stacks**
- Delivery for text, images, mss, media, data, & curated collections

Stanford Digital Repository (SDR): content agnostic, preservation repository

and SDR provides “back-office” preservation services: replication, auditing, migration, and retrieval in a secure, sustainable, scalable stewardship environment.
An Ecosystem of Activities to the Digital Library

Management Components
- Deposit
- Process
- Access Control
- ILS (Integrated Library System)
- DOR (Digital Object Registry)
- Monitor
- Work flow
- Report
- Preserve (SDR)

Discovery Environments
- Search Works
- Fed Search
- Alerts
- OPAC
- Music
- GIS
- Gov Docs
- EAD
- Art
- Soc Sci

Delivery Tools
- Text
- MSS
- Web
- Data
- Image
- Media

“Portals” and Platforms
- Sakai
- Collab
- Drupal
DOR Enables Deposit & Management of Digitalia
Deposit, Management & Access Happen “Above” SDR

Deposit & Manage

Discovery & Delivery

Preservation
Digital Object Registry (DOR) Details

• *Registry*, not repository

• Registers, tracks and relates versions and instances of content across the DL and their lifecycle
  • deposit, processing, preservation, indexing for discovery, mining, delivery

• Enables management of & reporting on objects through DOR- (and other) web services
  • E.g., *fetch metadata, transform object, create derivative, send to SDR*, etc.
  • Supports workflow & choreography of atomic services

• Based on Fedora
Google: End-to-End Full Text Workflow

Stage 1: Download Prep
- Richard DB
  1.1 request to GRIN for all books available
  1.2 analysis & selection of titles to download
  1.3 create list of barcodes
- GRIN

Stage 2: Download & DOR Management
- SURI
- Page Map
- PDF Maker
- Make METS

Stage 3: SDR
- DOR
  2.1 register object
  2.2 conversion requests
  2.3 create descriptive metadata
  2.4 download
  2.5 create page objects
  2.6 create PDF
  2.7 request deposit

Stage 4: Discovery/Delivery
- Symphony
  MARC to SOLR
- process
  update index
- SearchWorks Bib Index
- SearchWorks App
  4.1 get queue
  4.2 get contentMap
  4.3 create index
  4.4 detect text
  4.5 search text
  4.6 find books
  4.7 read book online
- SBooks App
- processes
  3.1 get deposit
  3.2 ingest
  3.3 confirm

User

Update index
What’s Changed: Surrounding Ecosystem

• Scoped repository: differentiation between *preservation* (provided by SDR) and
  ...
  *content middleware* (provided by DOR)
  ...
  *access* (provided by the Digital Stacks apps)

• Implications:
  – Reduces pressure on SDR to be all things to all depositors, for all content
  – Reinforces need to provide managed & secure storage at scale
  – Reinforces requirement to focus on content integrity services
  – Emphasizes need to integrate SDR to management & access services through stable API’s
SDR 1.0: In Production & Working, BUT...

- Custom code, maintained by evolving & smaller team
  
- Bottlenecks
  - Need to be quicker to add new content types
  - Need to be quicker to add new collections
  - Need to increase the throughput to the storage layer
  
- Need to refine & hone the SDR service model
  - Need to deemphasize *curation* approach, shift away from “just in case” to a “just in time” mentality
  - Need to emphasize non-technical solutions & strategies
  
- Need to leverage other parts of the DL ecosystem
  - API’s for management, ingest, retrieval, reporting
  - Reuse of code
SDR 2.0

- Redefine (reduce & hone) service model
- Revise & streamline the data model
- Modify the technical architecture
- Reengineer the storage layer
- Integrate with other parts of the DL ecosystem
SDR 2.0: Refine the SDR Service Model

The Stanford Digital Repository (SDR) provides services to make scholarly resources available over the long term by helping ensure their integrity, authenticity, and reusability.

To fulfill its mission, the SDR must be secure, sustainable and trusted.
SDR 2.0: Refine the SDR Service Model

What does this mean?

SDR (the system) provides “back-office” preservation services: replication, auditing, migration, and retrieval in a secure, sustainable, and scalable environment.

SDR (staff) provide consulting, common-sense & commitment to long-term access for the institution.

Other (systems & people beyond SDR) will provide the domain expertise, curation functions, access support and overarching content management services.
SDR 2.0: Revise the Data Model

Currently use a METS-based “transfer manifest” as SIP, AIP and DIP, with many issues:

- Each Transfer Manifest is content & collection specific → Doesn’t scale
- Transfer manifests require too much interpretation and analysis to change, augment
- Too complex: Stanford METS structure breaks apart related data across the object
- Wraps (somewhat dynamic) metadata with (mostly static) data files in same envelope
- Recursive nature of transfer manifest makes versioning self-referential, complex
- No one speaks METS natively: depositors, SDR & clients all forced to perform translation at handshakes
SDR 2.0: Revise the Data Model

• Revised data model currently in design
• Support “Zip ‘n’ SIP” for deposits
• Deprecate generation & storage of so much technical metadata
• Break object metadata into discrete files, track changes in change logs
  – Descriptive MD
  – Structural MD (aka the “content map”)
  – Technical MD
  – Rights MD
  – Provenance MD
• Store content files in bags
SDR 2.0: New Technical Architecture

• Adopt Fedora as a metadata management system
  – Clean mapping of new data model to Fedora content models
  – Reuse same design pattern, core technology as in DOR

• Support for highly parallelized operations
  – Multiple ingest streams to increase throughput
  – Decompose one process (e.g., “ingest”) into discrete, loosely coupled operations (“checksum”)

• Adopt a RESTful architecture
  – Shared services with DOR, access layer
SDR 2.0: Reengineer storage layer

- Current storage layer comprises:
  - Honeycomb for all MD and some data
  - TSM and tape robot for three LTO4 tape copies
  - Home-grown storage management system
- Throughput is acceptable for large objects, less than acceptable for many small objects
- 2.0 will adopt a full disk layer
  - Pool of SATA drives
- Currently evaluating storage management and abstraction options
SDR 2.0: Integrate with Stanford’s Ecosystem

- Fully automated download -> process -> ingest pipeline for Google-scanned books
  - Google -> DOR -> SDR chain
  - RESTful architecture of atomic web services
  - DOR manages object registration, validation, processing, indexing & packaging for deposit
  - Automatic handoff to SDR for ingest & call back to confirm successful deposit
  - Indexing & access to delivery surrogates managed through “Digital Stacks” environment

- Bulk retrievals from SDR for replenishing “digital stacks”, applying new operations
SDR 2.0: Integrate with Stanford’s Ecosystem

• Reuse of the Google Books pattern for preservation / management / access of other Stanford content

• “Hydra Project”
  – Reusable application framework that sits on top of Fedora, uses Blacklight for search & browse
  – Basis for Stanford’s user-facing (institutional) repository services (electronic theses, open access articles, data deposit, etc.)
  – Support for deposit, edit, annotate, set permissions, search, browse, view objects in a Fedora environment
  – Also the basis for an SDR administrative user interface
Conclusion

• SDR 1.x has largely achieved its mission
• SDR 2.0 will address necessary points to jump to the next curve of maturity
  – Focus on scoped set of services
  – Simplified data model
  – Reuse of technology (Fedora, Hydra, Blacklight)
  – Scale up on storage

• Mindset
  – Shift from “just in case” to “just in time”
  – Rely on integration with other services to provide holistic framework for management & access
PASIG Repository Working Group

- Identify principles and overarching design of repositories in larger systems context
- Identify successful technology patterns, independent of any specific software or hardware products
- Share code, strategies and methods