Conceptualizing Policy-Driven Interoperability: The PODRI Project

Richard Marciano
Alexandra Chassanoff
David Pcolar
Bing Zhu
Chien-Yi Hu

March 24, 2010

PODRI's Key Research Questions

What is the feasibility of repository interoperability at the policy level?

- Can a preservation environment be assembled from two existing repositories?
- Can the policies of the federation be enforced across the repositories?
- What fundamental mechanisms are needed within a repository to implement new policies?

Requirements for architecture

- Persistent
- Interoperable
- Scalable
- Distributed
- Flexible
- Semantic

Rationale for using iRODS and Fedora

iRODS

- open source
- federated storage
- rule engine and microservices

Fedora

- open source
- file-centric
- front-end APIs
- richer metadata

Steps toward management architecture

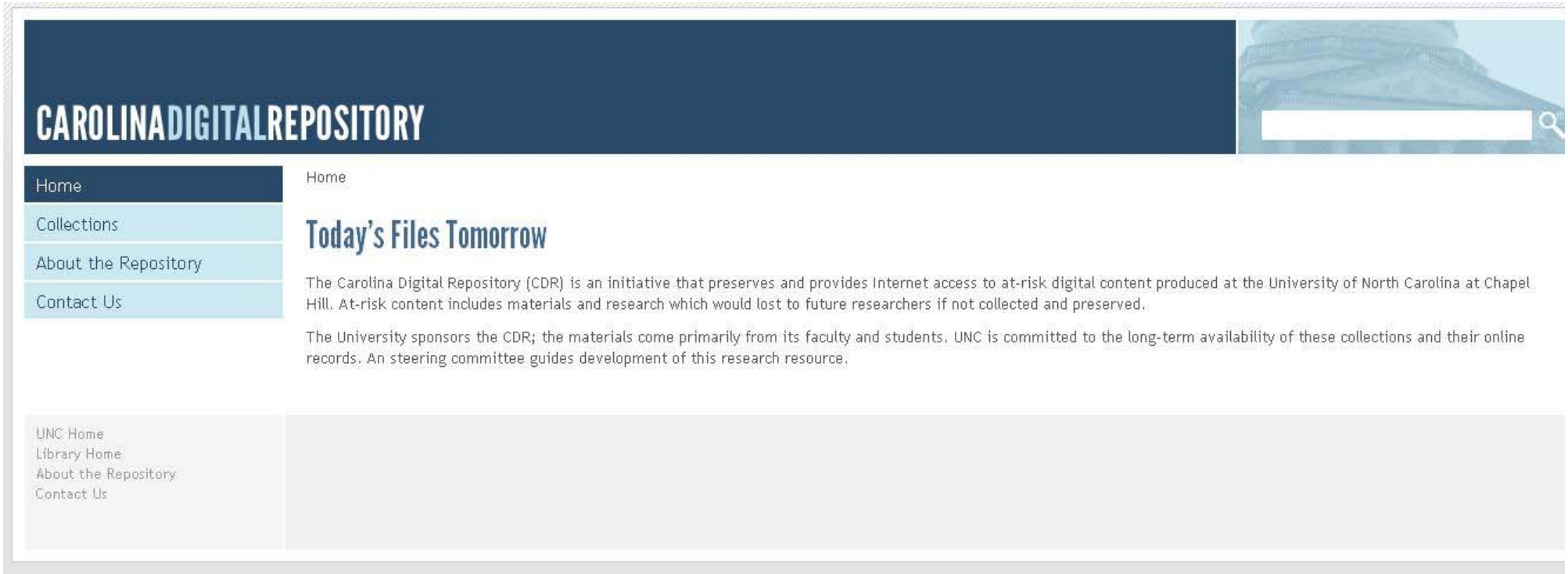
- Integrate views of content, original arrangement (hierarchy) and metadata

- Create an audit trail of policy execution events and related provenance information

- Demonstrate policy management through Fedora

- Demonstrate policy management from iRODS

Use Case:
Carolina Digital Repository



Carolina Digital Repository (CDR)

- CDR is designed as a repository for material in electronic formats produced by members of the University of North Carolina at Chapel Hill community. Its chief purpose is to provide for long-term preservation of digital assets. By preservation we mean the ability to ingest the material, index and search it, replicate it, and keep it safe from alteration.

- Following standards developed in the reference model for an Open Archival Information Systems, the CDR employs Fedora for data content models and uses iRODS as a data store.

- The University Library is a partner in the PoDRI, to build on the initial Fedora to iRODS connector and investigate interoperability issues in greater depth.

Carolina Digital Repository (CDR)

- UNC Libraries efforts are based on connecting technologies developed by Fedora, iRODS, PoDRI, and DCAPE and serves as a test bed for other developing technologies.

- In partnership with the Renaissance Computing Institute (RENCI), Duke University and North Carolina State University, the University Library and UNC Information Technology Services are participating in a project funded by the Triangle University Center for Advanced Studies, Inc. (TUCASI) to develop policies and infrastructures that will support the federation of independent repositories built on diverse hardware and software platforms. The CDR will form one of the core technologies for this project.

Carolina Digital Repository (CDR)

- Functionality focused on ‘Core Services’
- Ingestion
- Indexing & Discovery
- Preservation
- Dissemination
- Metadata Standards
- Dublin Core Based (FOXML)
- Minimal Elements
- Collection
- Folder
- Item
- Rights/Access/Use Metadata (XACML based)

Carolina Digital Repository: PoDRI Related Research Questions

- Where is PREMIS information stored (e.g., with an individual object or as an aggregate)?
- How will the collection structure be represented in the two products?
- How will Fedora be initialized for existing content in iRODS?
- How will Fedora be informed of content or metadata changes initiated directly in iRODS?
- How can content or metadata from Fedora be accessed by iRODS services?

Five Enabling Scenarios

1. New content ingest via Fedora
2. New content ingest via iRODS
3. Bulk registration from iRODS into Fedora
4. Update of content or metadata via Fedora
5. Update of content or metadata via iRODS

Scenario 1:
New content ingest via Fedora

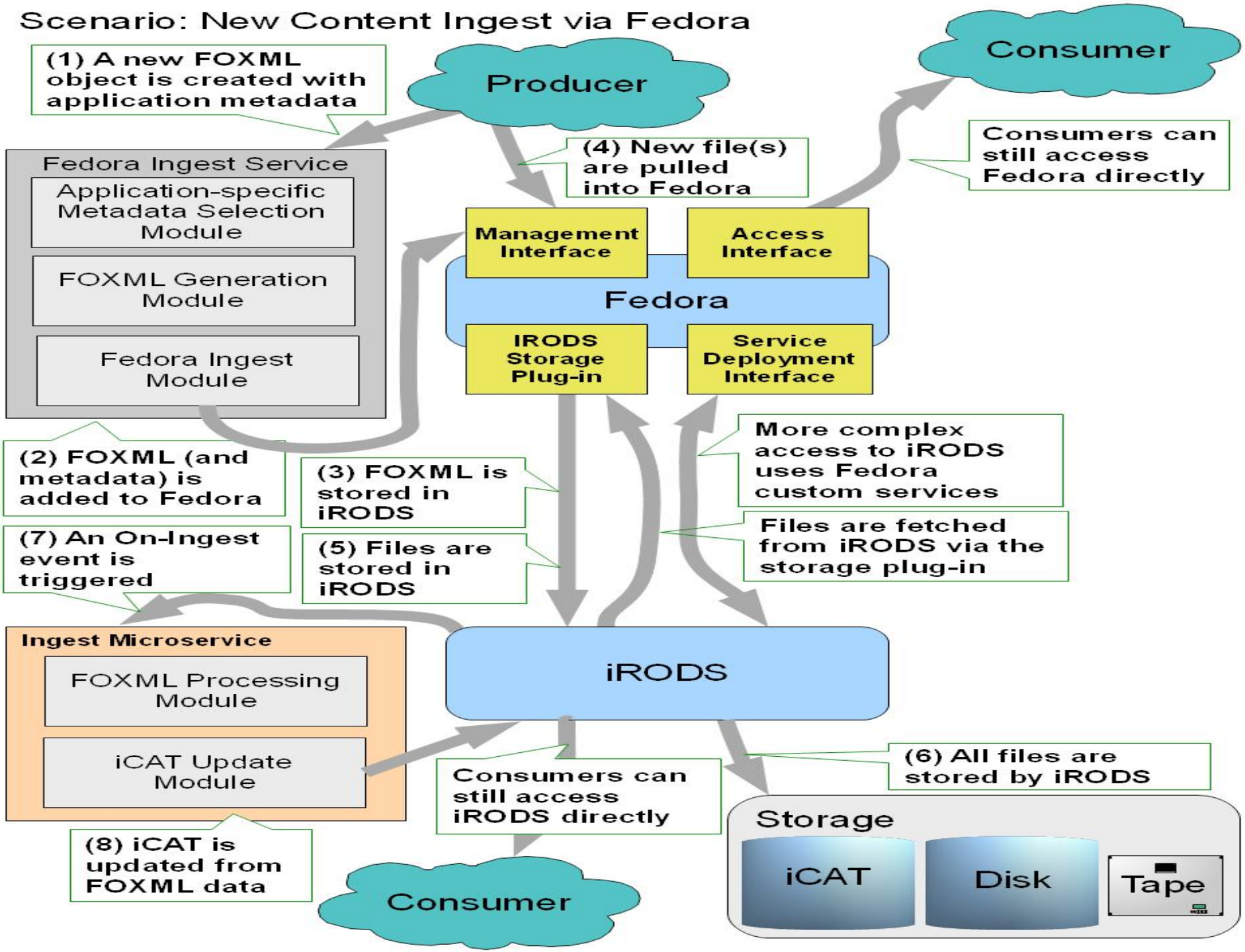
- Use Fedora features to add and relate rich metadata including policy, provenance and authenticity information.

- Fedora Digital Object (FDO) and Content Stores in iRODS.

- Index selected metadata for speedy access, services bindings, etc.

- Model tree structure through RDF relation and navigate through a triplestore.

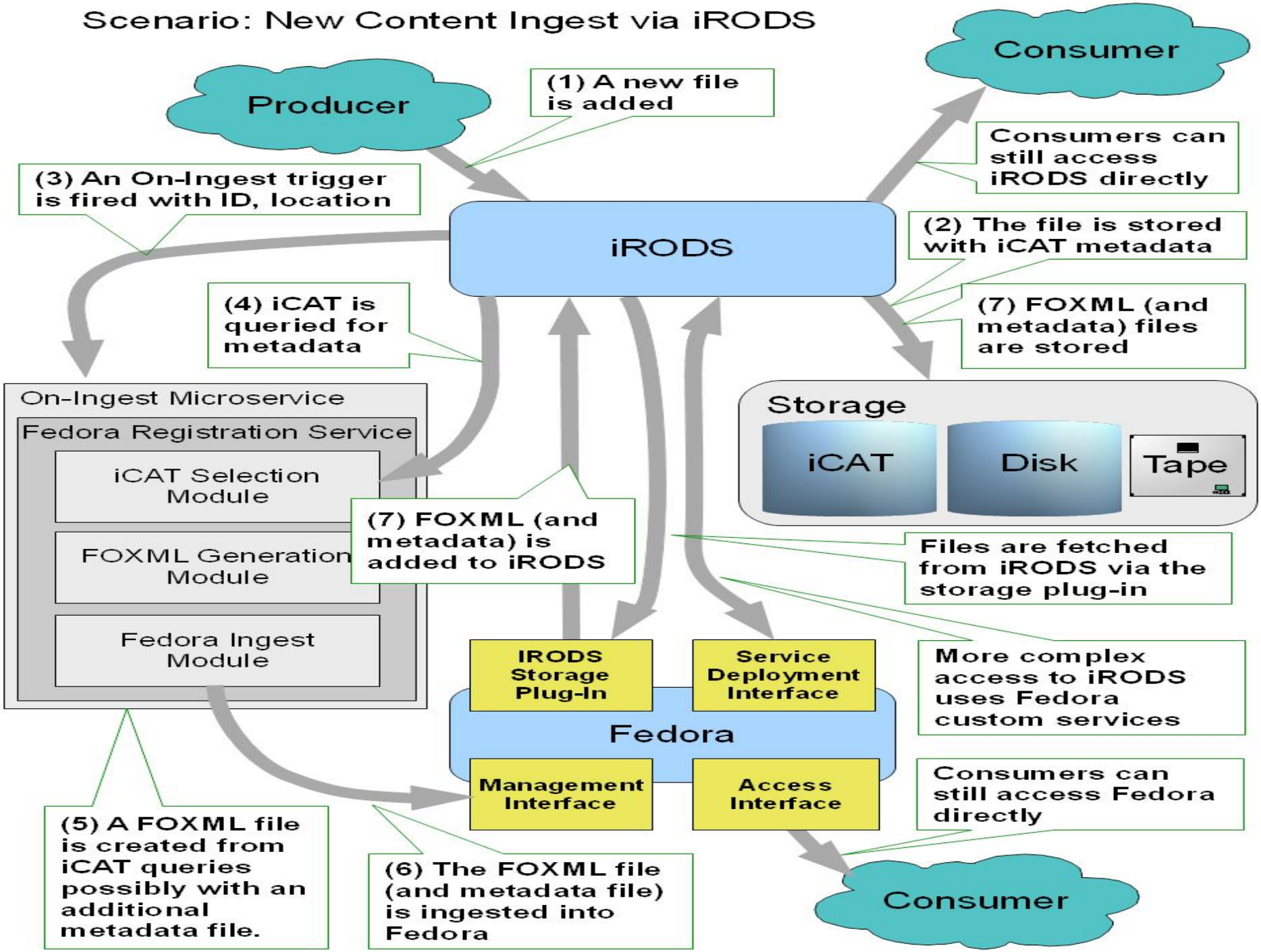
- Use iRODS rules to monitor new content for indexing and metadata extraction.


Scenario: New Content Ingest via Fedora
(1) A new FOXML object is created with application metadata
Producer
Consumer
Fedora Ingest Service
Application-specific Metadata Selection Module
FOXML Generation Module
Fedora Ingest Module
(4) New file(s) are pulled into Fedora
Consumers can still access Fedora directly
Management Interface
Access Interface
Fedora
IRODS Storage Plug-in
Service Deployment Interface
(2) FOXML (and metadata) is added to Fedora
(3) FOXML is stored in iRODS
More complex access to iRODS uses Fedora custom services
(7) An On-Ingest event is triggered
(5) Files are stored in iRODS
Files are fetched from iRODS via the storage plug-in
Ingest Microservice
FOXML Processing Module
iCAT Update Module
iRODS
Consumers can still access iRODS directly
(6) All files are stored by iRODS
Storage
iCAT
Disk
Tape
(8) iCAT is updated from FOXML data
Consumer

Scenario 2:
New content ingest via iRODS

- Microservice is triggered after each new iRODS data object is created.
- iRODS files are registered in Fedora through micro-service trigger.
- Fedora Data Object (FDO) and FOXML are created.

Scenario: New Content Ingest via iRODS



Other utilities

- iRODS Storage module
- iRODS Data Harvester for Fedora

Future Work

- Continuing working on use cases
  - Bulk registration from iRODS into Fedora
  - Update of content or metadata via Fedora
  - Update of content or metadata via iRODS

- Applying preservation policies on iRODS
  - integrity check
  - replication
  - and more...