Combining Content, Semantic Relationships, and Web Services – Fedora

CS 431 – April 11, 2007

Carl Lagoze – Cornell University

Acknowledgements:

Sandy Payette (Cornell)

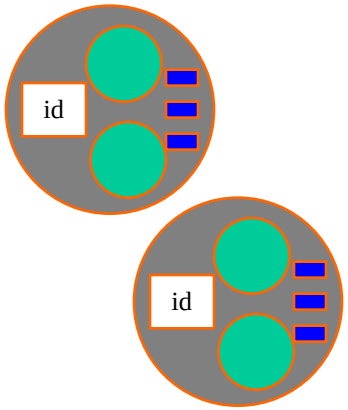
Herbert Van de Sompel (LANL)

Sang Shin (Sun)

Compound Information Objects

- Aggregations of distinct information units that when combined form a logical whole.
- Examples
  - digitized book that is an aggregation of chapters, where each chapter is an aggregation of scanned pages;
  - a CD that is the aggregation of several audio tracks;
  - an image object that is the aggregation of a high quality master, a medium quality derivative and a low quality thumbnail;
  - a scholarly publication that is aggregation of text and supporting materials such as datasets, software tools, and video recordings of an experiment;
  - a multi-page web document with an HTML table of contents that points to multiple interlinked HTML individual pages.

Compound Information Objects

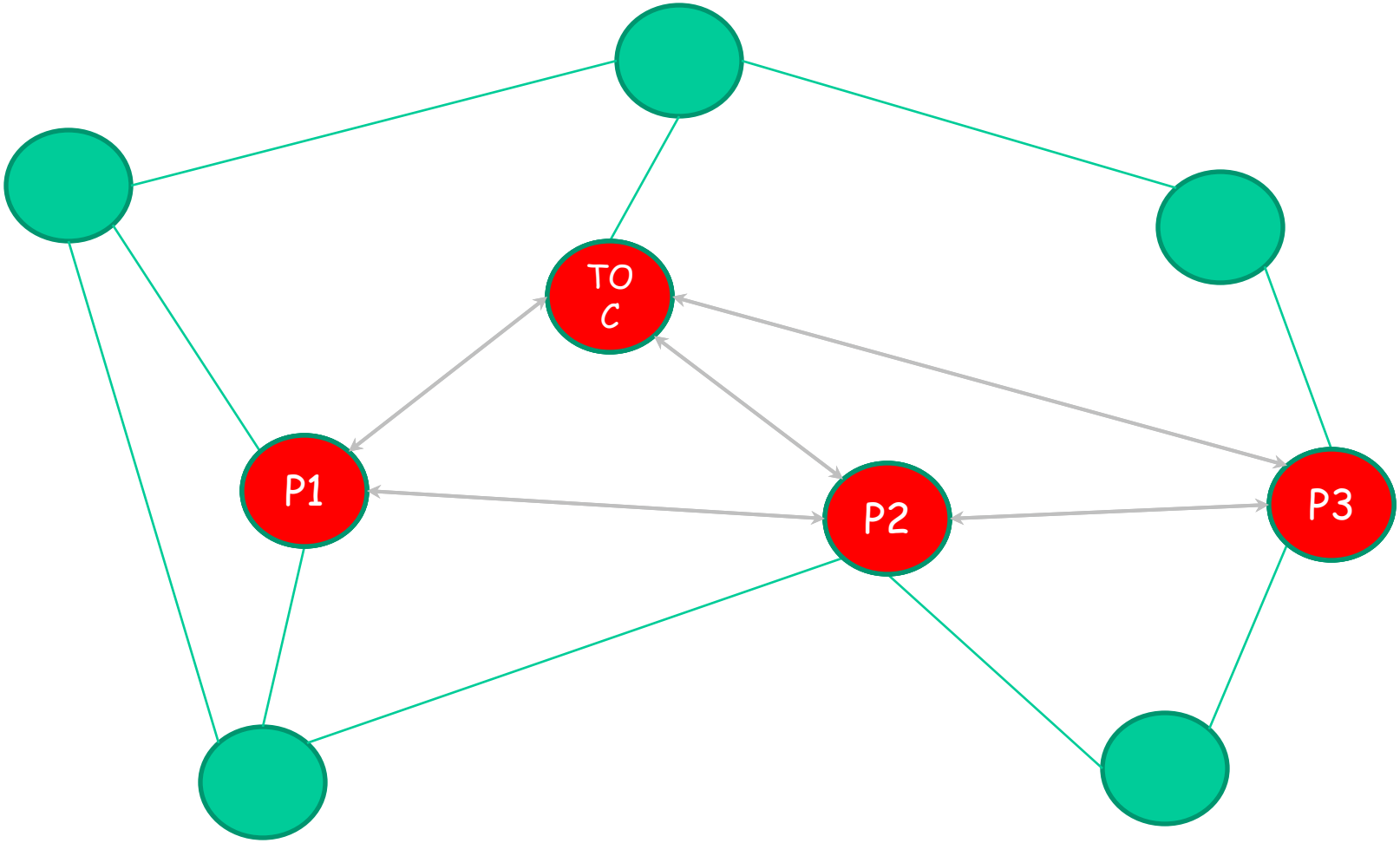


Digital Objects

Digital content with multiple components varying on:

- **Content (semantic) types** including:
  - Text
  - Datasets
  - Simulations
  - Software
  - Dynamic knowledge representations
  - Machine readable chemical structures
  - Bibliographic and other types of metadata
- **Media types** including
  - IANA registered MIME types
  - Other type registries such as GDFR
- **Network locations** including content from:
  - Institutional repositories
  - Scientific data repositories
  - Social networking sites
  - General web
- **Relationships** including:
  - Lineage
  - Versions
  - Derivations

Compound objects in the web graph



Compound Information Objects in Common Use

Back to the Flickr photo page

Uploaded on April 10, 2007
by Jai-to-Z

Available sizes: Square (75 x 75) | Thumbnail (67 x 100) | **Small** (161 x 240) | Medium (334 x 500) | Large (685 x 1024) | Original (2592 x 3872)

Download the Small size

Compound Information Objects in Common Use

Pathways: Augmenting interoperability across scholarly repositories

Simeon Warner, Jeroen Bekaert, Carl Lagoze, Xiaoming Liu, Sandy Payette, Herbert Van de Sompel

(Submitted on 5 Oct 2006)

In the emerging eScience environment, repositories of papers, datasets, software, etc., should be the foundation of a global and natively-digital scholarly communications system. The current infrastructure falls far short of this goal. Cross-repository interoperability must be augmented to support the many workflows and value-chains involved in scholarly communication. This will not be achieved through the promotion of single repository architecture or content representation, but instead requires an interoperability framework to connect the many heterogeneous systems that will exist.
We present a simple data model and service architecture that augments repository interoperability to enable scholarly value-chains to be implemented. We describe an experiment that demonstrates how the proposed infrastructure can be deployed to implement the workflow involved in the creation of an overlay journal over several different repository systems (Fedora, aDORe, DSpace and arXiv).

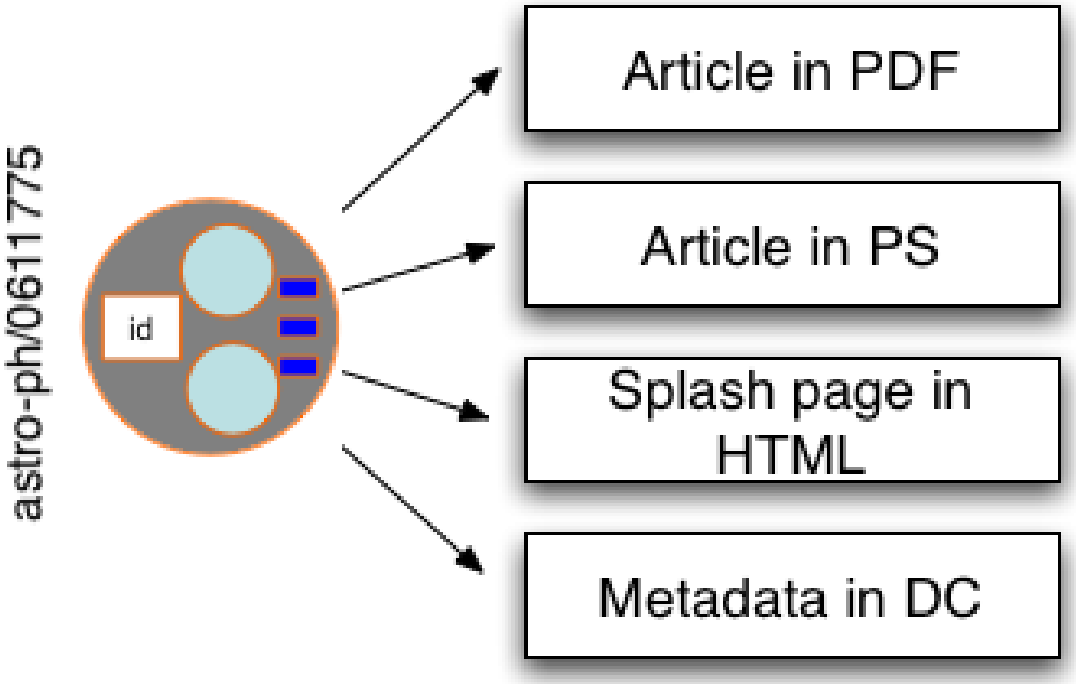
Comments:
18 pages. Accepted for International Journal on Digital Libraries special issue on Digital Libraries and eScience
Subjects:
Digital Libraries (cs.DL)
ACM classes:
H.3.7
Cite as:
arXiv:cs/0610031v1 [cs.DL]

Submission history

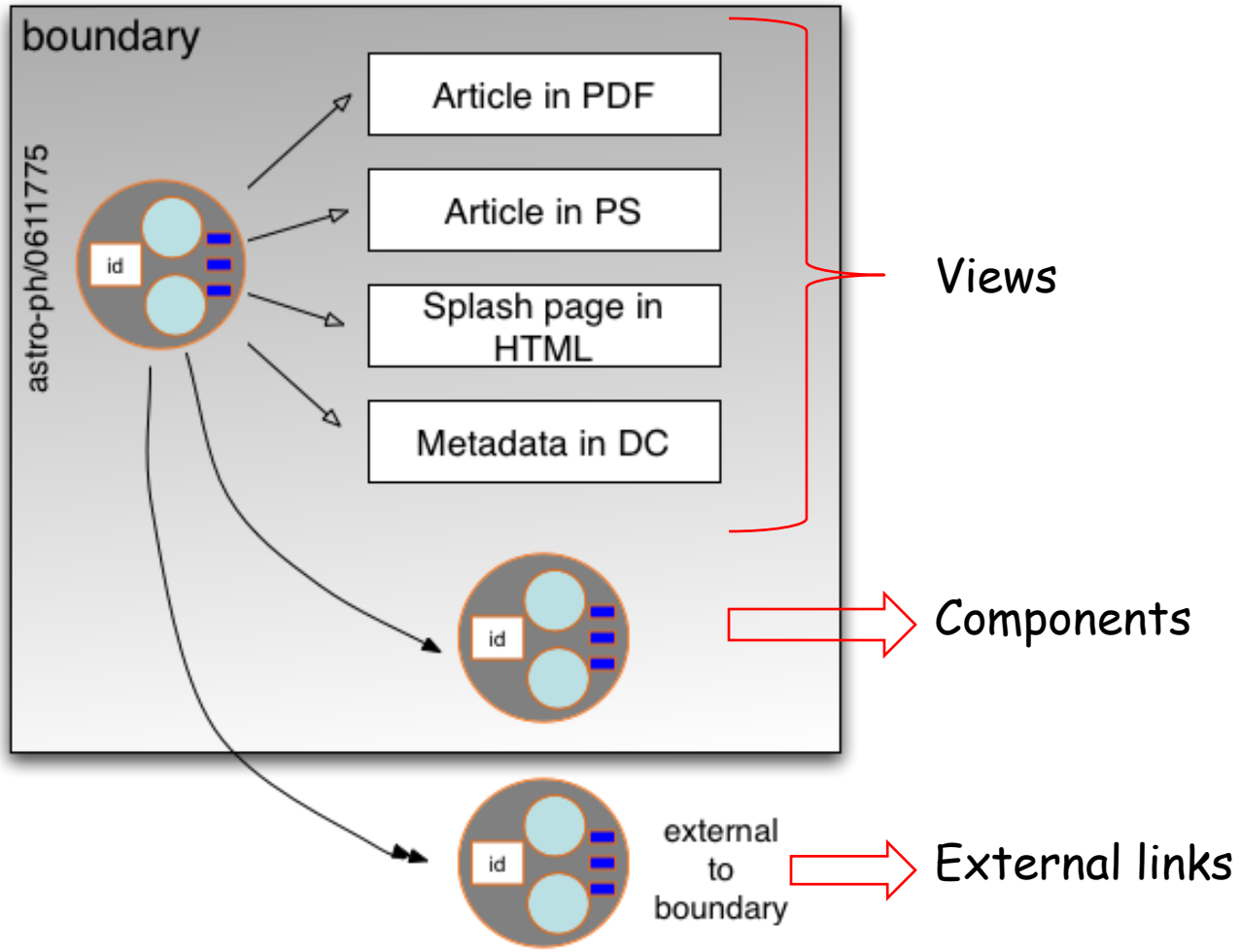
From: Simeon Warner [view email]
[v1] Thu, 5 Oct 2006 19:55:09 GMT (496kb)

Which authors of this paper are endorsers?

Simple Digital Object



More complexity…



The Fedora Project

- Fedora
  - Flexible
  - Extensible
  - Digital
  - Object
  - Repository
  - Architecture
- Open source software
  - Not Red Hat !
  - Mozilla Public License
- http://www.fedora.info

Fedora Features

- Digital Object Model
  - Aggregate multiple information streams
  - Multiple types
  - Local and Distributed
- Integrate content and web services
  - Dynamically produced representations
- Semantic Web
  - Typed relationships among objects
- Service-based
  - All functionality exposed as web services
  - Integration with other applications and interfaces

Fedora History

- **Cornell Research (1997-present)**
  - DARPA and NSF-funded research
  - First reference implementation developed
  - Distributed, Interoperable Repositories (experiments with CNRI)
  - Policy Enforcement
- **First Application (1999-2001)**
  - University of Virginia digital library prototype
  - Technical implementation: adapted to web; RDBMS storage
  - Scale/stress testing for 10,000,000 objects
- **Open Source Software (2002-present)**
  - Andrew W. Mellon Foundation grants
  - Technical implementation: XML and web services
  - Fedora 1.0 (May 2003)
  - Fedora 2.0 (Jan 2005)

Fedora Use Cases

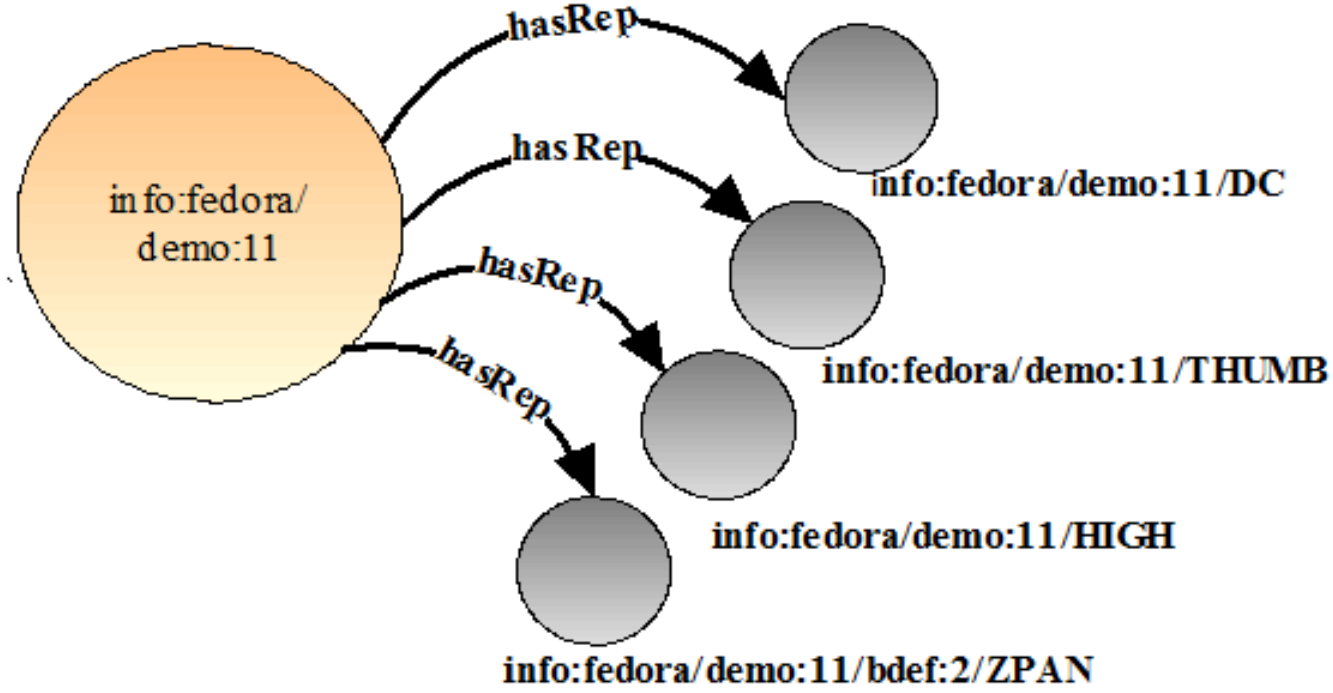
- Digital Library Collections
- Institutional Repository
- Educational Software
- Information Network Overlay
- Digital archives and preservation
- Digital Asset Management
- Content Management System
- Scholarly publishing
- eScholarship, eScience
- Data Curation

Selected Fedora Users

- **University of Virginia: digital library (image collector, EAD, e-texts)**
- **VTLS (software company): commercial product (VITAL)**
- **Tufts University: education (VUE/concept maps); digital library**
- **Northwestern: academic technologies (images, art, video, e-texts)**
- **National Science Digital Library (NSDL): Cornell Core Integration**
- **ARROW: National Library of Australia and Monash University**
- **Royal Library of Denmark and DTU**
- **Rutgers University: digital library (e-journals, numeric data)**
- **Indiana University: EVIA Digital Archive (video)**
- **American Geophysical Union: scholarly publications**
- **Max Planck Institue: Scholarly Communication**
- **Cornell University: Bear Access**
- **Yale University – electronic records**
- **New York University: humanities computing; digital library**
- **OhioLink**
- **DISA – South Africa, History of Apartheid resistance**
- **Public Library of Science (PLOS)**
- **CiteSeer – Penn State**

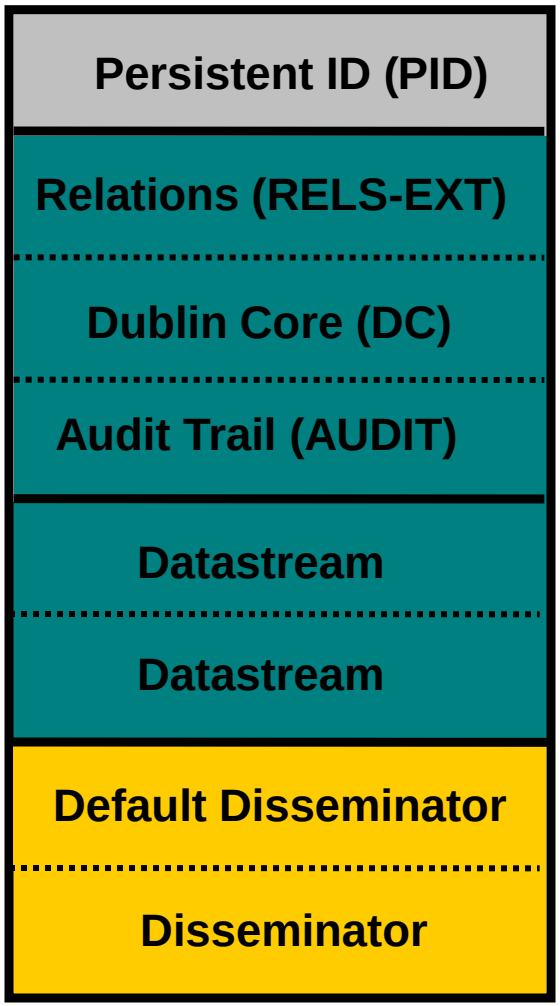
Digital Object Model

"Graph" View of Fedora Objects



Fedora Digital Object Model

Component View



Digital object identifier

Reserved Datastreams
Key object metadata

Datastreams
Set of content or metadata items

Disseminators
Pointers to service definitions to provide service-mediated views

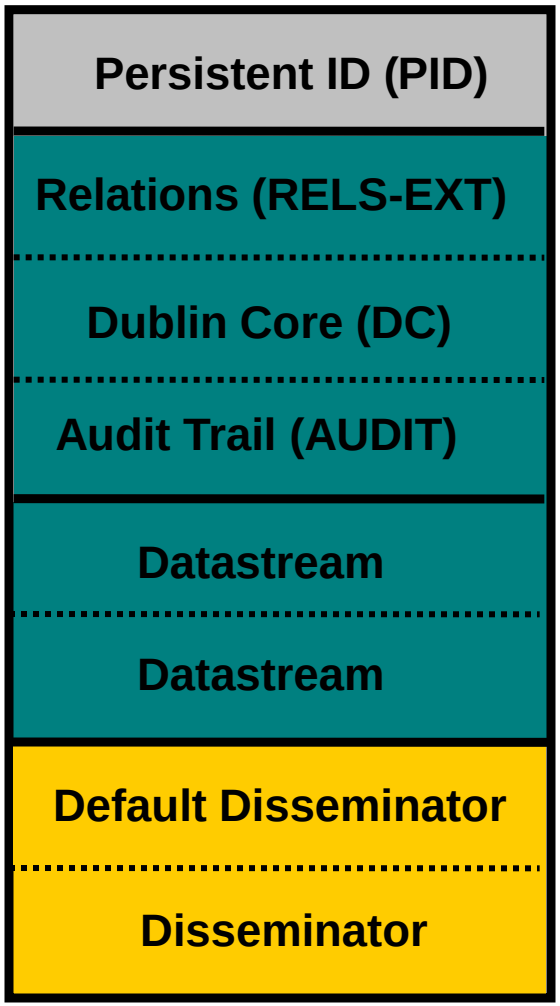
Fedora – XML for digital objects

- **FOXML (Fedora Object XML)**
  - Simple XML format directly expresses Fedora object model
  - Easily adapts to Fedora new and planned features
  - Easily translated to other well-known formats
  - Internal storage format for objects in repository

```
<?xml version="1.0" encoding="UTF-8"?>
<foxml:digitalObject xmlns:foxml="info:fedora/fedora-system:def/foxml#"
        xmlns:fedoraxsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:audit="info:fedora/fedora-system:def/audit#"
        fedoraxsi:schemaLocation="info:fedora/fedora-system:def/foxml# http://www.fedora.info/definitions/1/0/foxml1-0.xsd"
        PID="demo:1">
   <foxml:objectProperties>
      <foxml:property NAME="http://www.w3.org/1999/02/22-rdf-syntax-ns#type" VALUE="FedoraBDefObject"/>
      <foxml:property NAME="info:fedora/fedora-system:def/model#state" VALUE="Active"/>
      <foxml:property NAME="info:fedora/fedora-system:def/model#label" VALUE="Behavior Definition Object for UVA Simple Image Contract"/>
      <foxml:property NAME="info:fedora/fedora-system:def/model#ownerId" VALUE="fedoraAdmin"/>
      <foxml:property NAME="info:fedora/fedora-system:def/model#createdDate" VALUE="2007-04-10T20:55:40.969Z"/>
      <foxml:property NAME="info:fedora/fedora-system:def/view#lastModifiedDate" VALUE="2007-04-10T20:55:40.969Z"/>
      <foxml:property NAME="info:fedora/fedora-system:def/model#contentModel" VALUE="fedora:BDEF"/>
   </foxml:objectProperties>
   <foxml:datastream ID="DS1" STATE="A" CONTROL_GROUP="E" VERSIONABLE="true"> [5 lines]
   <foxml:datastream ID="DC" STATE="A" CONTROL_GROUP="X" VERSIONABLE="true"> [10 lines]
   <foxml:datastream ID="METHODMAP" STATE="A" CONTROL_GROUP="X" VERSIONABLE="true"> [13 lines]
   <foxml:disseminator ID="DISS1" BDEF_CONTRACT_PID="fedora-system:1" STATE="A" VERSIONABLE="true"> [7 lines]
</foxml:digitalObject>
```

Fedora Digital Object Model

Component View



Digital object identifier

Reserved Datastreams
Key object metadata

Datastreams
Set of content or metadata items

Disseminators
Pointers to service definitions to provide service-mediated views

The Datastream Component

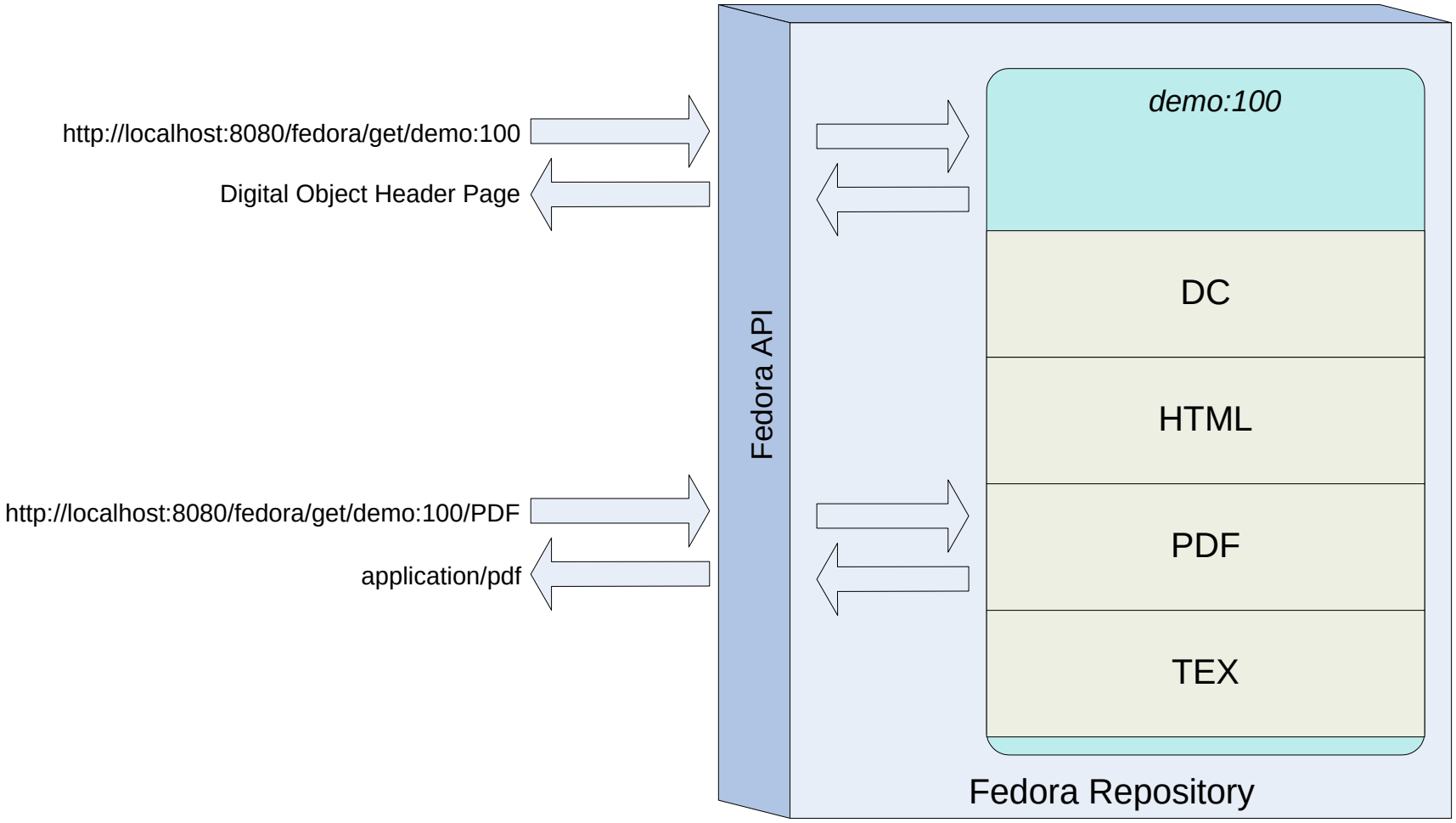
4 Classifications for Datastreams

Inline XML	***Fedora stores a name-spaced block of XML content within the Fedora digital object XML file.***
Managed Content	***Fedora stores and manages the content bytestream (non-XML content)***
External Referenced	***Fedora stores a reference (URL) to the content***
External Redirected	***Fedora stores a reference (URL) to the content, but will not mediate access to content. (Optimized for streaming)***

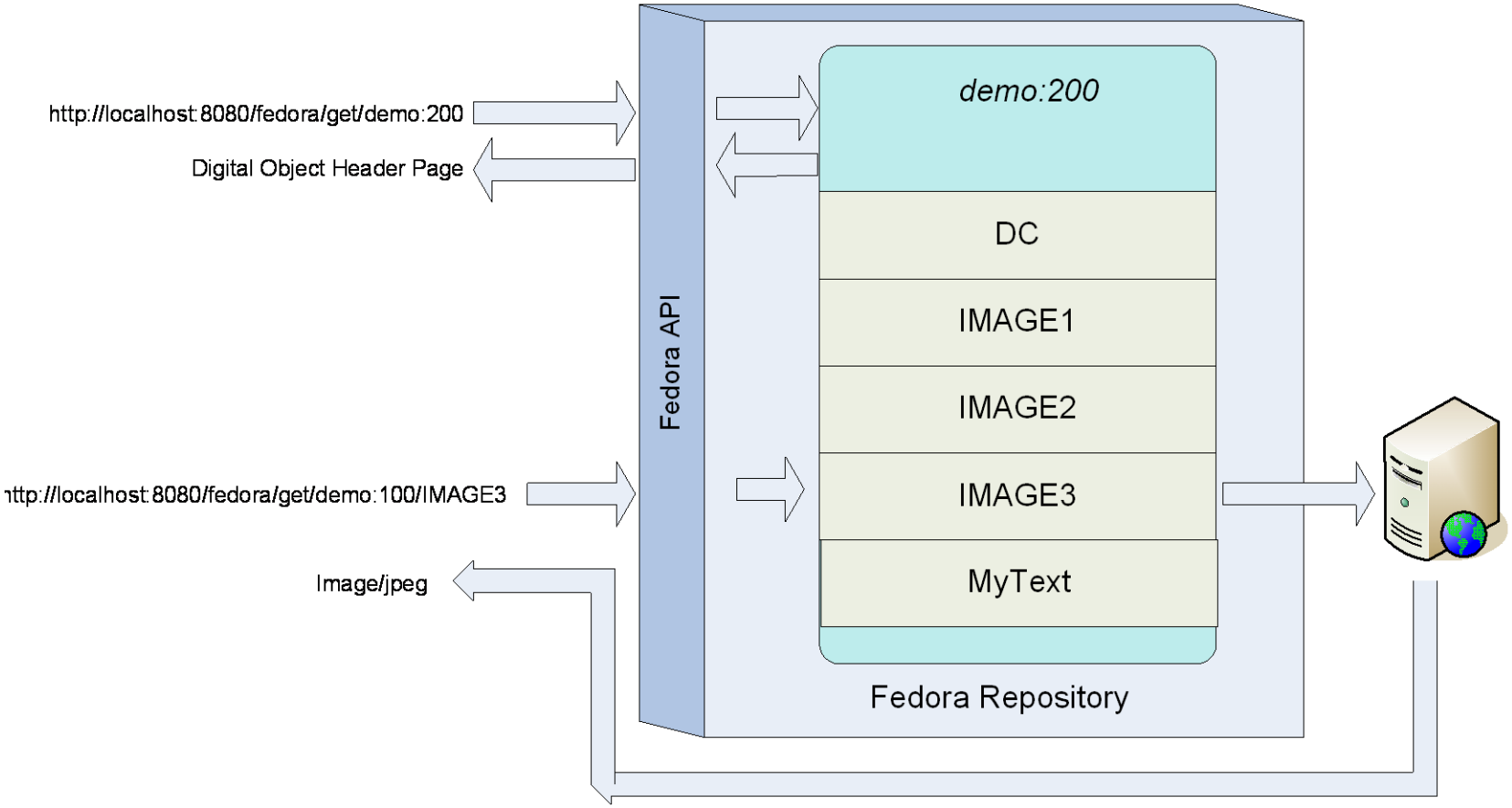
Simple Fedora model for aggregating static content

- Representations map to datastreams
- Datastreams may be local or surrogates (redirect) to remote data
- REST URL's give client access to representations

Digital Object Aggregating Local Content



Digital Object for Local and Remote Content



Integrating Content and Web Services

What is a Web Service

- A software application
- Identified by a URI
- Interfaces and bindings described by XML
- Supports direct interactions with other software applications
- Messaging (request and response) based on XML
- Uses Internet protocols (e.g., HTTP)

Web Services Compents (SOAP)

- Simple Object Access Protocol
- XML-based RPC
- Uses XML for data encoding
- Defines
  - Message envelope
  - Encoding Rules
  - RPC Convention
  - Binding with underlying protocol

SOAP RPC Request Example

```
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="..."
    SOAP-ENV:encodingStyle="...">
  <SOAP-ENV:Header>
    <!-- Optional context information -->
  </SOAP-ENV:Header>

  <SOAP-ENV:Body>
    <m:GetLastTradePrice xmlns:m="some_URI">
      < tickerSymbol> SUNW</tickerSymbol >
    </m:GetLa stTra dePric e>
  </SOA P-ENV:Body>

</SOAP-ENV:Envelope>
```

SOAP RPC Response Example

```
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="..."
    SOAP-ENV:encodingStyle="...">
  <SOAP-ENV:Header>
    <!-- Optional context information -->
  </SOAP-ENV:Header>

  <SOAP-ENV:Body>
    <m:GetLastTradePriceResponse xmlns:m="some_URI">
      <price>30 .5</price>
    </m:GetLa stTra dePric eResponse>
  </SOA P-ENV:Body>

</SOAP-ENV:Envelope>
```

Web Service Components - WSDL

- Web Services Description Language
- Components
  - Abstract Definition of operations and messages
  - Concrete binding to network protocol and endpoint address

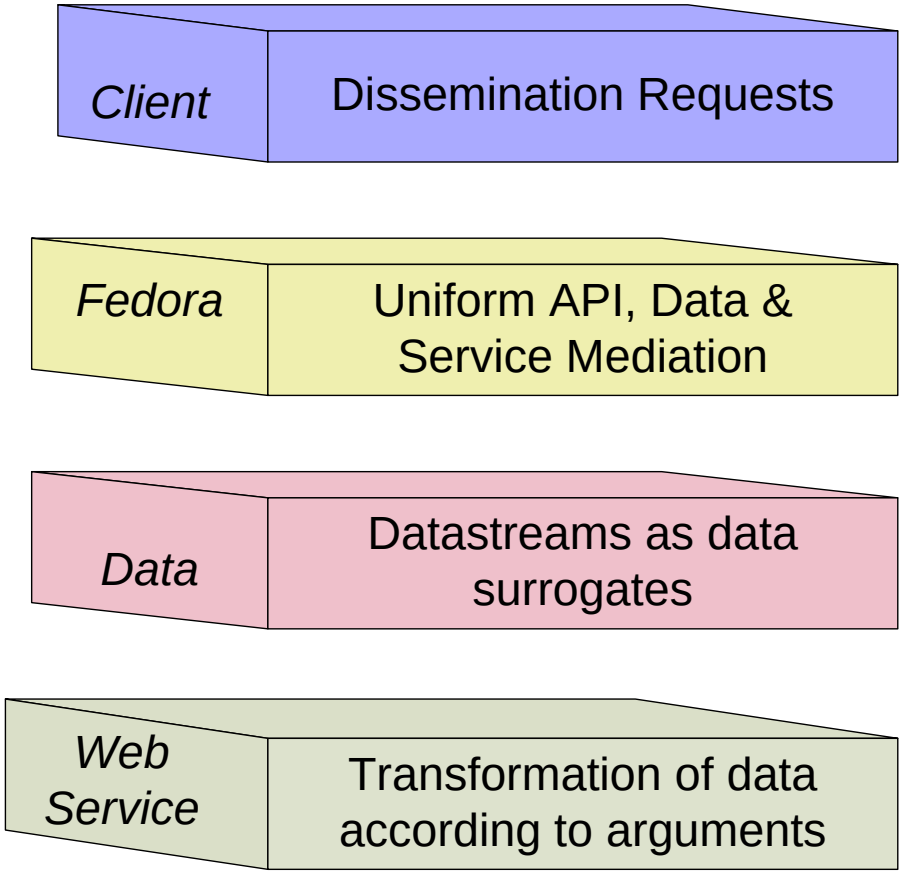
Fedora for dynamic content

- Representations map to service-based transforms of data (in addition to static datastreams)
- Opaque to REST based access (client see only representations, not how they are produced)
- Motivating examples
  - Canonical XML metadata format – XSLT to Dublin Core
  - Document source in TeX, programmatic transform to PDF, PS, HTML, etc.

Dynamic Content – The Big Picture

- From the client perspective they are just representations of the object
- 1 or more datastreams for the foundation of the dissemination
- These are parameters to a web service that produces the actual representation

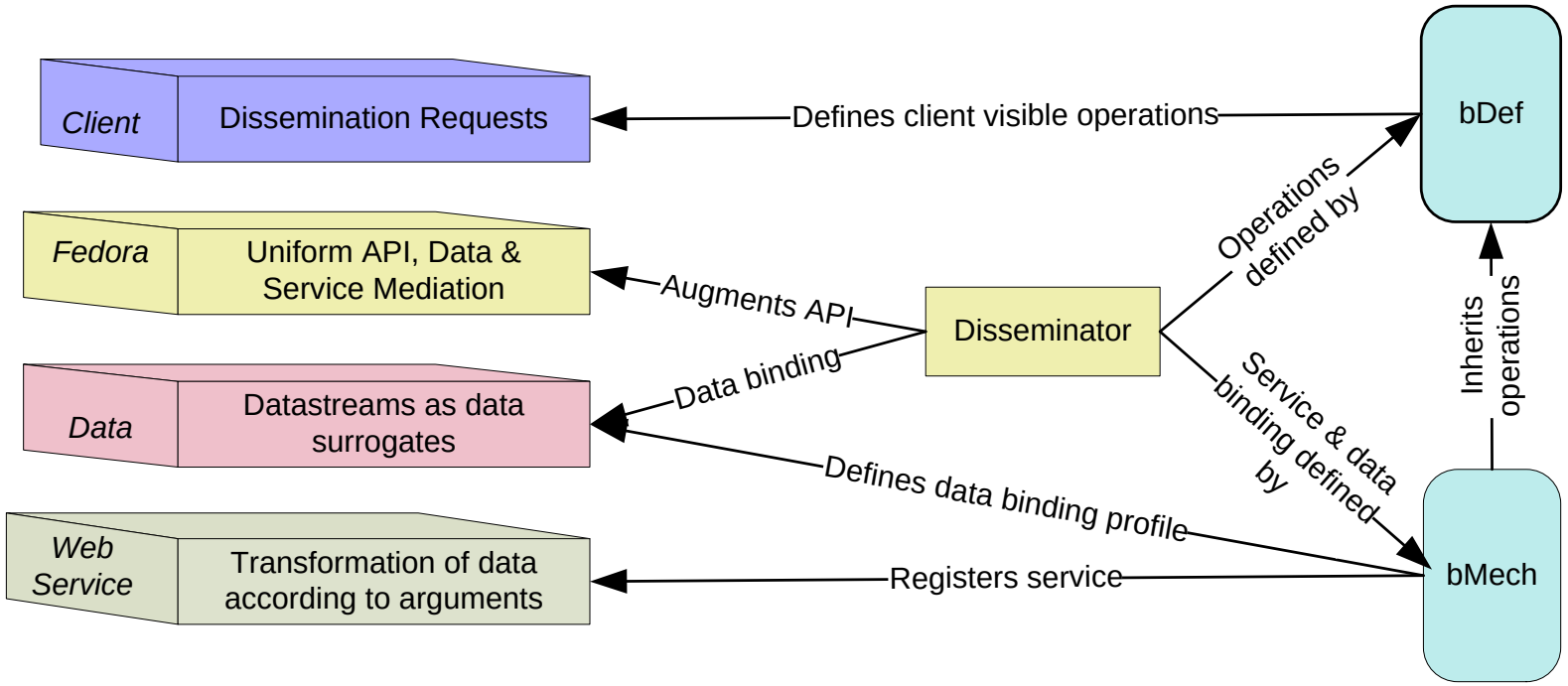
Understanding Dynamic Disseminations (1)



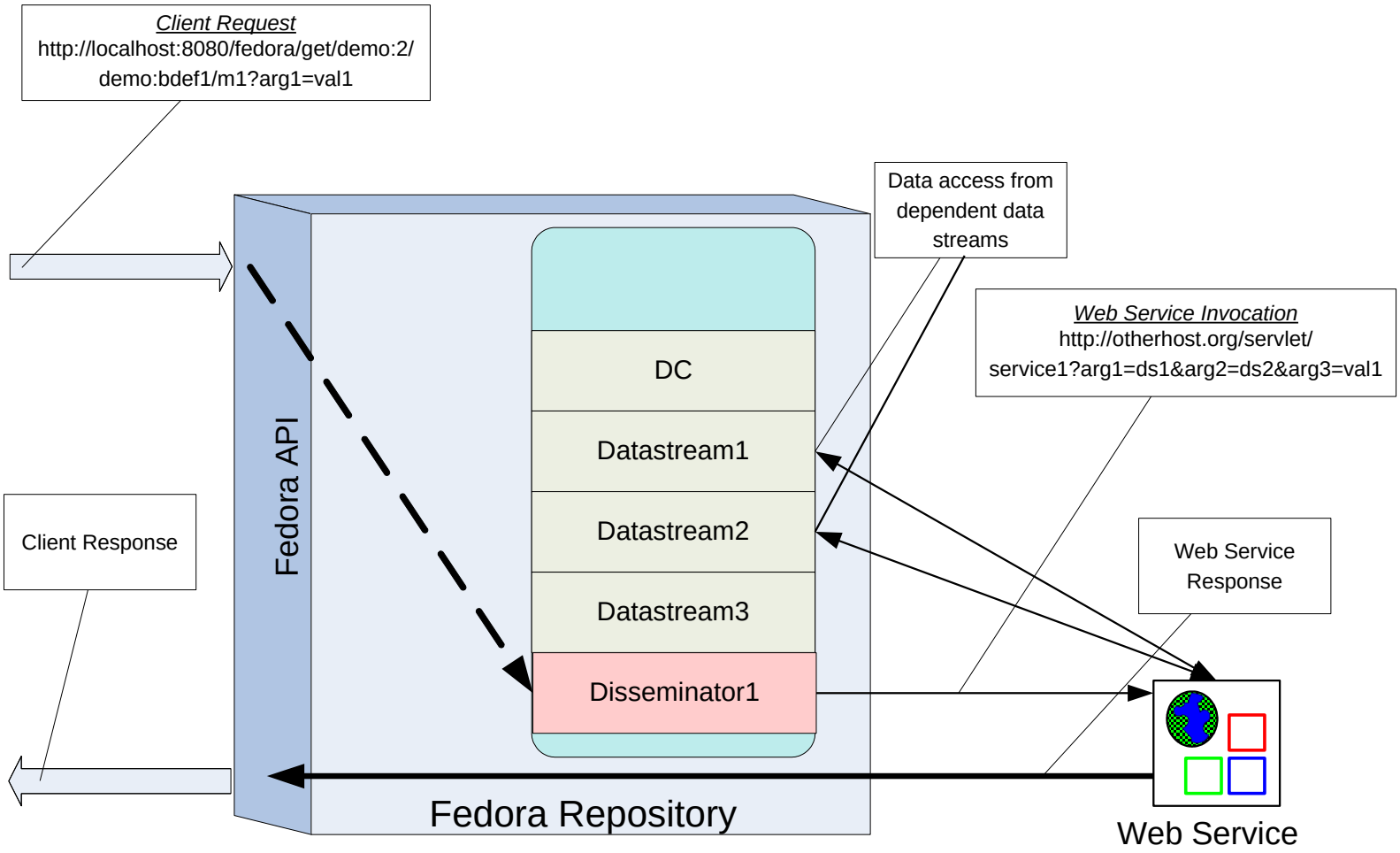
Understanding Dynamic Disseminations (2)

- Behavior Definitions (bDef)
  - Special digital object defining client side functionality (method template)
- Behavior Mechanism (bMech)
  - Special digital object that refines a bDef by defining:
    - Data profile: set of datastreams required for execution
    - Service binding: where the work is performed
  - May be many bMechs for a bDef
- Disseminator
  - Association of a bMech/bDef with a digital object endowing it with bDef-defined functionality (methods)
  - A digital object may have multiple disseminators (polymorphic typing)

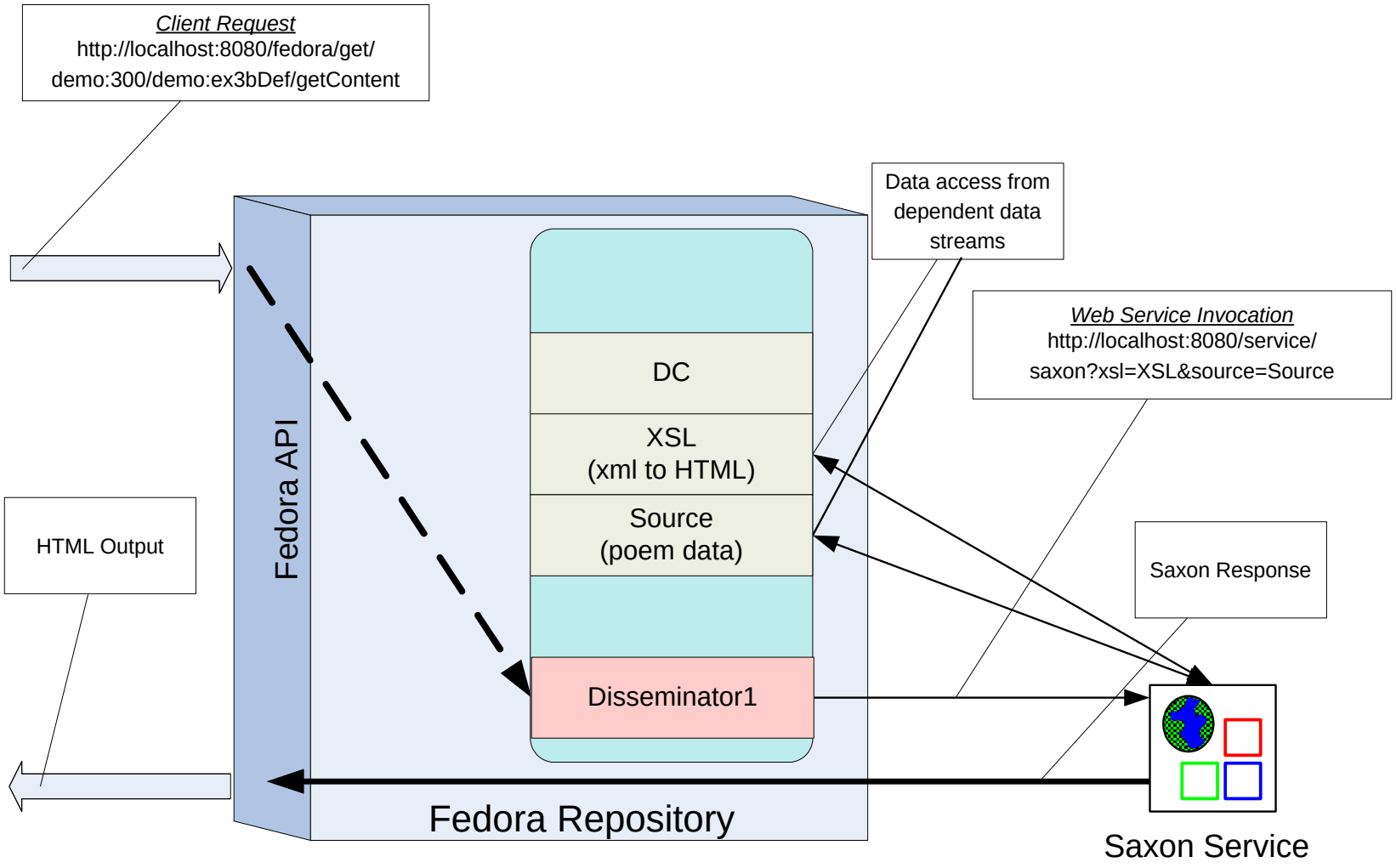
Understanding Dynamic Disseminations (3)



Dynamic Dissemination Access



Dynamic Dissemination Example

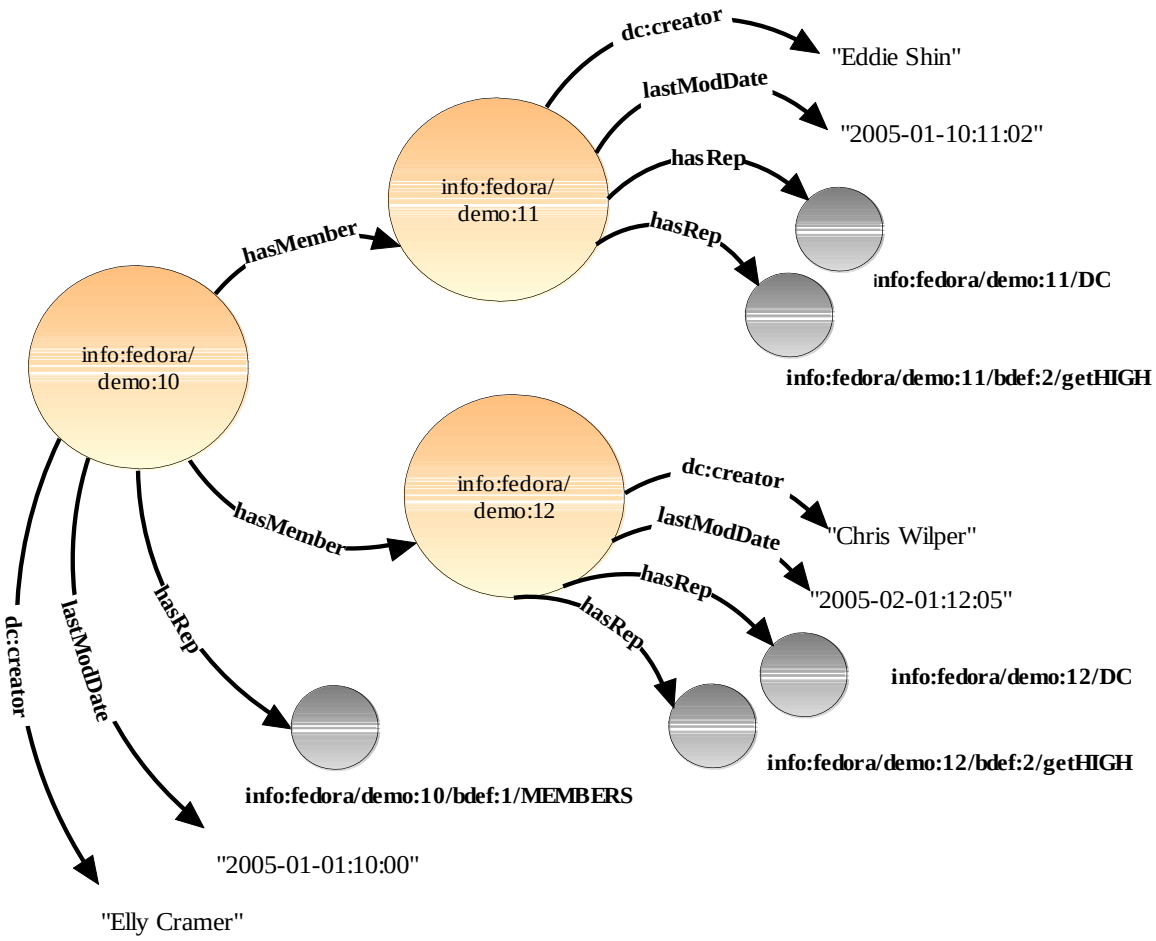


Fedora and Semantic Web: Resource Index

Using RDF and ontologies

Fedora Digital Objects

Resource Index View



Fedora and RDF

- **Object-to-object and object-to-literal Relationships**
  - Ontology of common relationships (RDF schema)
  - Relationships stored in special datastream (RELS-EXT)

- **Resource Index (RI)**
  - RDF-based index of repository (Kowari triple-store)
  - Graph-based index includes:
    - **Object properties and Dublin Core**
    - **Object Relationships**
    - **Object Disseminations**

- **RI Search**
  - Powerful querying of graph of inter-related objects
  - REST-based query interface (using RDQL or ITQL)
  - Results in different formats (triples, tuples, sparql)

Uses of Object Relationships

- Define collections (e.g., collection objects)
- Assert critical relationships among object for management purposes
- Enable network overlay
  - Surrogate objects referring to external entities
  - Assert relationships among them
  - Assert other relationships (e.g., annotations)
- Enable navigation of repository (as tree or graph)

Fedora Relationship Ontology (RDFS)

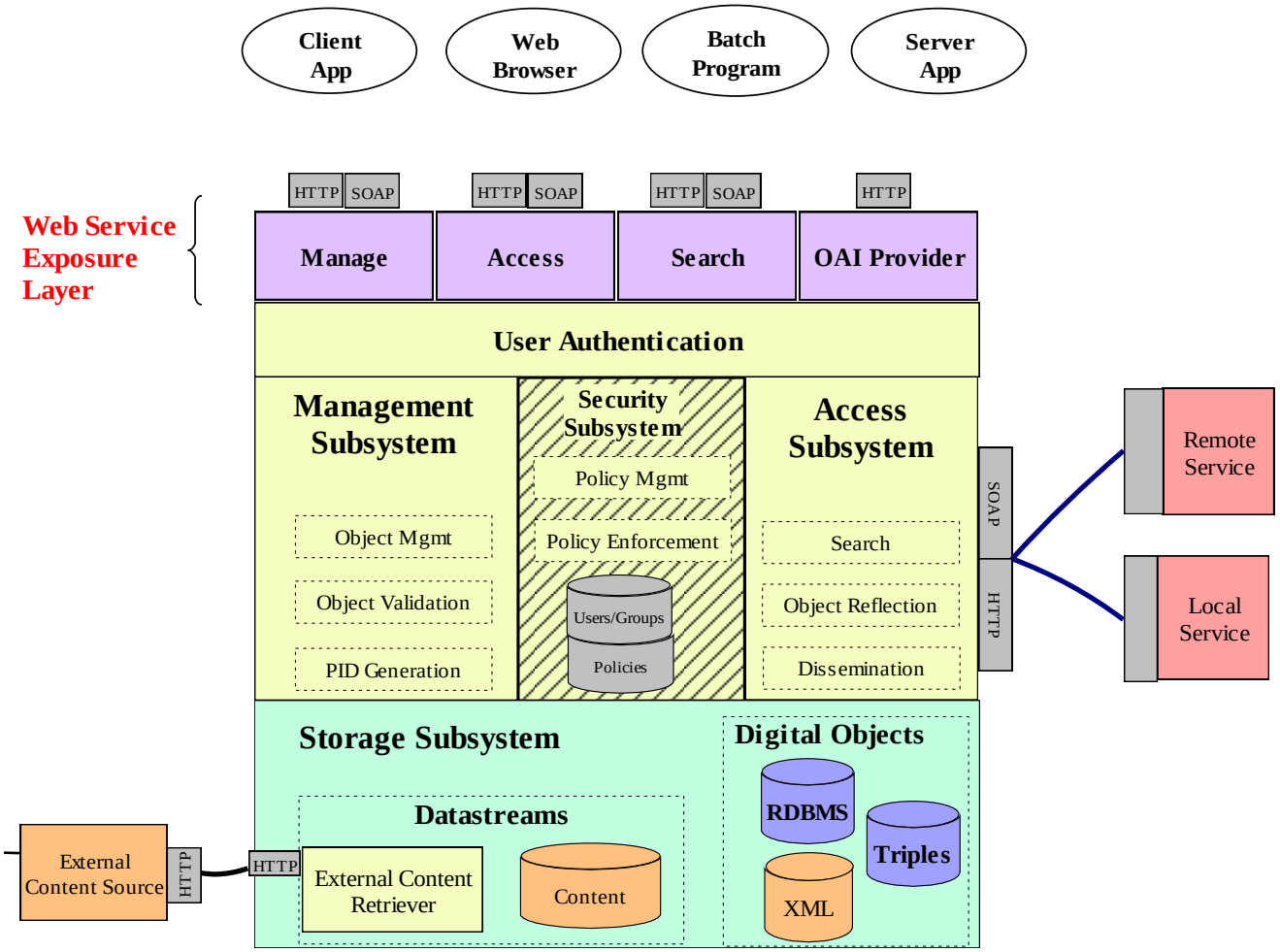
- isPartOf / hasPart
- isMemberOf / hasMember
- isDescriptionOf / hasDescription
- hasEquivalent
- … others

Demo:
Collection – Member Relationships

- Collection Object [smiley]
  - Datastream containing a query to Resource Index for all members of collection
- Image Objects [brush]
  - Use RELS-EXT datastream to assert relationship to collection object

Fedora Repository Service

Fedora Repository Service



Fedora Software Distribution

- **Open Source (Mozilla Public License)**
- **100% Java (Sun Java J2SDK1.4)**
- **Supporting Technologies**
  - Apache Tomcat and Apache Axis (SOAP)
  - Xerces for XML parsing and validation
  - Saxon for XSLT transformation
  - Schematron for validation
  - MySQL and Mckoi relational database
  - Oracle 9i support
  - Kowari for triple-store
- **Deployment Platforms**
  - Windows 2000, NT, XP
  - Solaris
  - Linux
  - Mac OSX