Fedora and digital preservation

Chris Awre
Tackling the Preservation Challenge
12th December 2008

Background to Fedora

Computer Science project (Sandy Payette, Carl Lagoze) at Cornell University in the late 1990s

- Focus on how to organise digital objects

2002-5, Mellon funded Fedora project

- Joint-funded project between University of Virginia and Cornell University
- First release of the software attracted wide interest

2005-7, second Mellon grant to extend Fedora development

- Led to release of mature production version, 2.2.1, in early 2007

Fedora now

Overseen by Fedora Commons

- A not-for-profit foundation to oversee Fedora development
- Launched September 2007
- Underpinned by $4.9M Moore Foundation grant
- Fedora 3.0 released July 2008

Community source software

- Core development team within the Foundation
- Community input guides development
  - Adopting Eclipse Foundation model
  - Solutions communities being developed
- Collaboration with DSpace to work on common goals

Features of Fedora

Powerful digital object model

Extensible metadata management

Expressive inter-object relationships

Web service integration

Version management

Configurable security architecture

OAI-PMH conformance

Preservation worthy

Features of Fedora that support digital preservation

Powerful digital object model

Extensible metadata management

Expressive inter-object relationships

Web service integration

Version management
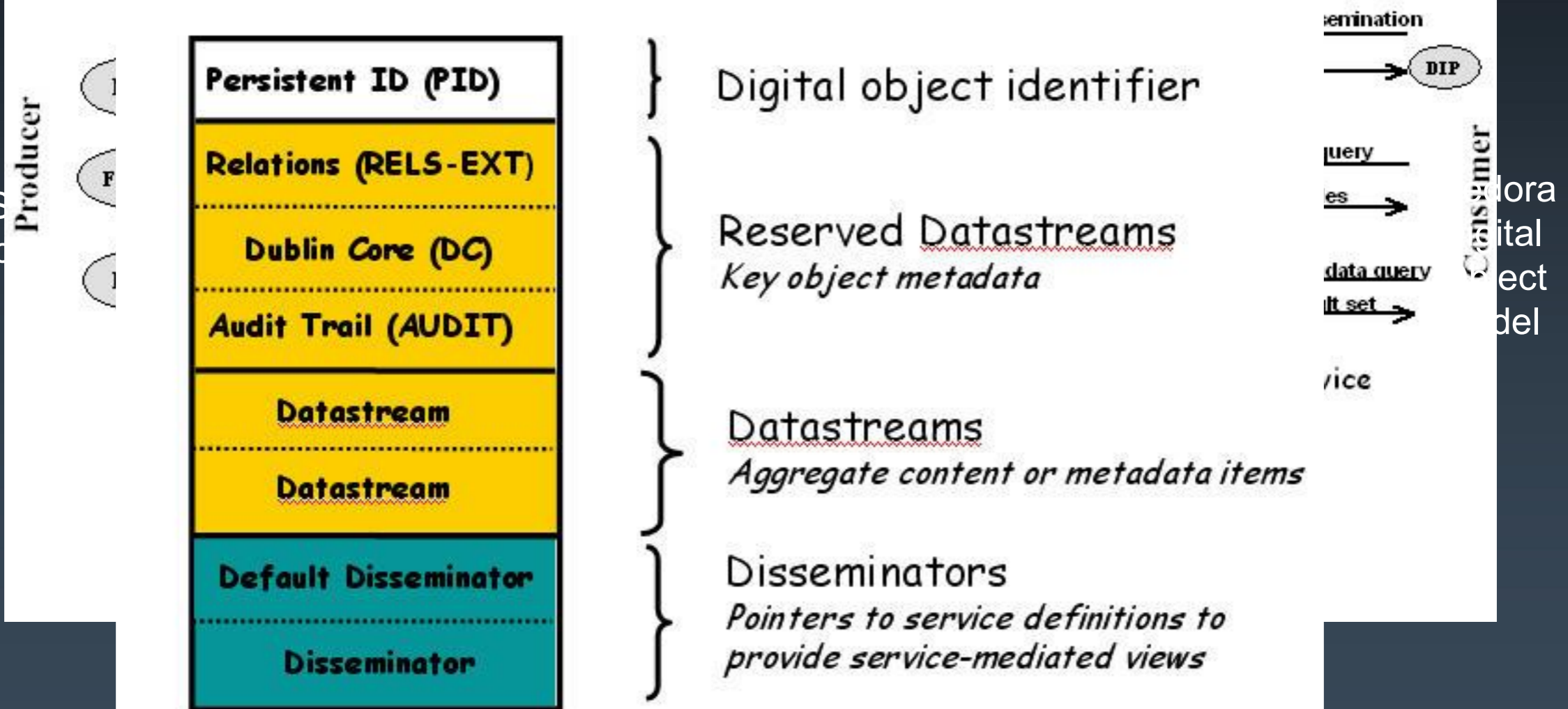
Configurable security architecture

OAI-PMH conformance

Preservation worthy

Aspects of Fedora

Flexible Extensible Digital Object Repository Architecture

Fedora S
Framewo

Fedora
ital
ect
del

Fedora started with a focus on

ORGANISATION

This has matured and evolved into an emphasis on

DURABILITY

Aspects of digital preservation

Describing the digital object

- How do we know what the object being preserved is?

Security of the digital object

- How do we ensure that what is being preserved continues to exist?

Persistence of the digital object

- How do we know what is being preserved is the same over time?

Integrity of the digital object

- How do we ensure that what is being preserved is what we think it is?

Describing the digital object

Digital object model allows for holding of metadata to describe objects for preservation

- JHOVE or PRONOM output stored alongside descriptive metadata
- PREMIS and other preservation metadata can be included

Resource Index records relationships between digital objects

- Objects are described in their context, not just in isolation
- RDF-based

Content model architecture can be used to describe how a digital object should be structured

- Describes how an object can/should be accessed and preserved

Security of the digital object

XML-based digital object storage

- Everything in the repository can be flexibly managed for preservation

XML-based ingest and export

- Conforms to OAIS, allowing digital objects to be migrated as required
  - Reduces reliance on any one system

Repository can be re-built by crawling the XML object store in case of hardware failure or corruption

- Rebuilds object registry, search index, resource index

Persistence of the digital object

Journaling module available (optional)

- Captures all API-M transactions
  - These can be replayed to one or more other repositories for replication

Unique, persistent IDs assigned to all objects

- Automatically assigned on ingest
- URIs based on PIDs conform to the info URI scheme
  - Independent of resolution protocols

XACML-based authentication policies

- Descriptions of who can access digital objects goes with the object

Integrity of the digital object

Automatic versioning of content datastreams

- Datestamp recording of exactly when versions were created

Audit trail of all modifications to objects

- Provenance and history of content development over time recorded

Digital objects record extensive object properties

- Includes created and modified dates, MIME type, format identifiers
- Checksum (MD5, SHA-1, etc.)

Looking ahead

Preservation validation and integrity service

- Datastreams - validate the bytestream format
- Digital objects - validate based on content models

Preservation monitoring and alerting service

- Listen to message broker for special events
  - Checksum failure
  - API-M modification events
- Initiate actions
  - Email preservation manager
  - Kick off an automated process (e.g., migrate)

Thank you

Chris Awre

- c.awre@hull.ac.uk

Preservation and Archiving Solution Community

- http://fedora-commons.org/confluence/display/FCCWG/Preservation+and+Archiving

Fedora-users email discussion list

- https://lists.sourceforge.net/lists/listinfo/fedora-commons-users

REMAP

- http://www.hull.ac.uk/remap/