

Glossary

Fedora 4 is a [Linked Data Platform server](#) as defined by the [Linked Data Platform 1.0 specification](#). As such, many of the terms in this glossary are drawn directly from this specification.

- [Access Role](#)
- [Binary](#)
- [Checksum](#)
- [Children](#)
- [Compact Node Definition \(CND\)](#)
- [Container \(formerly known as: Object\)](#)
- [Datastream](#)
- [Dissemination](#)
- [Federation \(aka Projection\)](#)
- [Fixity](#)
- [Indexer](#)
- [jcr/xml](#)
- [Linked Data Platform \(LDP\)](#)
- [Managed External Content](#)
- [Namespace](#)
- [NonRdfSourceDescription \(formerly known as: datastream\)](#)
- [Object](#)
- [Policy Enforcement Point \(aka PEP\)](#)
- [Predicate](#)
- [Prefix](#)
- [Projection \(aka: Federation\)](#)
- [Properties](#)
- [rdf:type](#)
- [Repository](#)
- [Resource](#)
- [Tombstone](#)
- [Transactions](#)
- [Triple](#)
- [Uniform resource identifier \(URI\)](#)
- [Universally Unique Identifier \(UUID\)](#)
- [Version](#)

Access Role

A named role, such as "writer", that is assigned to a user, group or some other identifying principal within part of the repository. Roles may be used by the policy enforcement point (PEP) to authorize actions taken in the repository.

Binary

A binary in a Fedora repository is a `nonRdfSource` *resource*. In other contexts, binaries would sometimes be described as bit-streams or files. Binaries are always accompanied by a *nonRdfSourceDescription*.

Checksum

A computed fingerprint for binary content, used to ensure a complete transfer or the fixity of stored information. Fedora supports the SHA-1 checksum algorithm for safely uploading content.

Children

The resources which are immediate children of a given container. Properties are not counted as children.

Compact Node Definition (CND)

A file format defined by JCR [here](#) that allows for the definition of new *resource* types and *namespaces*. It is highly recommended that users do not modify Fedora 4's CND file, as such modifications may limit the ability for subsequent Fedora version upgrades of the repository installation.

Container (formerly known as: Object)

A container is a *resource* that represents intellectual entities and can also be used to aggregate other *resources* in a Fedora repository. Containers may contain other containers or *binaries* and their *nonRdfSourceDescriptions*.

Datastream

See: *NonRdfSourceDescription*

Dissemination

An approach previously available in Fedora 3 that allowed for binding a service to a content model. Note: *disseminators*, as such, do not exist in Fedora 4.

Federation (aka Projection)

See: *Projection*

Fixity

Fixity is the integrity of stored information over time. Fedora performs fixity checks on demand by comparing a stored checksum with one that is newly computed.

Indexer

Creating, modifying or deleting *resources* in the repository generates events. The indexer monitors and processes these events; by ingesting relevant RDF to an external triplestore, for example.

jcr/xml

An XML document, the schema of which is defined by the [JCR specification](#), which has a format that represents JCR resources. It can be used to export /import a set of *resources* and *properties* into a repository.

Linked Data Platform (LDP)

The W3C [Linked Data Platform](#) (LDP) specification describes a set of best practices and simple approach for a read-write Linked Data architecture, based on HTTP access to web resources that describe their state using the RDF data model. Fedora 4 implements the LDP specification for create, read, update and delete (CRUD), allowing HTTP, REST, and linked data clients to make requests to Fedora 4.

Managed External Content

Refers to content that sits outside the Fedora-configured datastore, but is managed by Fedora through a *projection*. Managed external content can be accessible via the Fedora API as well as have Fedora-managed audits.

Namespace

A namespace is a container for a set of identifiers (also known as symbols, names). In Fedora 4, *resource* properties may belong to any namespace providing semantic assertions that support interoperable metadata. Namespaces are restricted to being an empty string or to a URI as defined in [section 3](#) of RFC3986. An example of a namespace would be "<http://purl.org/dc/elements/1.1/>".

NonRdfSourceDescription (formerly known as: datastream)

A nonRdfSourceDescription is a *resource* that describes a binary *resource* within the Fedora repository. A nonRdfSourceDescription is always accompanied by a *binary*, and vice versa.

Object

See: *Container*

Policy Enforcement Point (aka PEP)

This is a pluggable component in the Fedora framework that is responsible for authorizing all actions take on *resources*.

Predicate

A predicate expresses a relationship between the subject and the object of a *triple*.

Prefix

Typically a short string representation of an associated *namespace*. For example the prefix "dc" could be used to represent the namespace "<http://purl.org/dc/elements/1.1/>". If we wanted to access an element (e.g. "contributor") within that namespace we could do it with the prefix "dc:contributor" or in the full form "<http://purl.org/dc/elements/1.1/contributor>".

Prefixes are limited to the following grammar:

```
LocalName ::= ValidString - SelfOrParent
/* Any ValidString except SelfOrParent */
SelfOrParent ::= '.' | '..'
ValidString ::= ValidChar {ValidChar}
ValidChar ::= XmlChar - InvalidChar
/* Any XmlChar except InvalidChar */
InvalidChar ::= '/' | ':' | '[' | ']' | '|' | '*'
XmlChar ::= /* Any character that matches the Char production at http://www.w3.org/TR/xml/#NT-Char */
```

Projection (aka: Federation)

The process by which a repository may present *resources* through the API that are actually stored in a different system, such as a file system or database.

Properties

Properties are name/value pairs that belong to *resources*. The name of a property can be any term from a *namespaced* vocabulary. When [RDF](#) is generated in response to a request for a *resource* that contains properties, the [RDF](#) will contain *triples* for each property where the subject of the *triple* is the resource itself, the *predicate* of the *triple* is the property name, and the object of the triple is the value of the property. Property values can be of any valid *rdf:type*.

rdf:type

Containers are defined by one or more *rdf:types* that describe the nature of the *container*. Similarly, every *property* will be an instance of a single *rdf:type*. The official definition can be found [here](#).

Repository

A repository is a single, self-contained persistent store of information plus the software that is used to access and update that information. Fedora is a repository.

Resource

Resources are the primary organizational structure in the repository. A resource is any web-addressable entity, such as a *container*, a *nonRdfSourceDescription*, or a *binary*. Every resource has a name and a unique identifier, and can also be identified by a path. They are comprised of zero or more properties / child resources.

Tombstone

A tombstone is a repository *resource* indicating that a *container*, *binary*, or *nonRdfSourceDescription* used to exist at a given URL. A tombstone is created when a resource is deleted or moved.

Transactions

A transaction represents a series of changes to the repository that must execute successfully and completely or not at all. Transactions should be used to ensure consistency. Each transaction succeeds or fails as a complete unit; it cannot remain in an intermediate state.

Triple

A triple is a fundamental building block of RDF. It consists of: a subject, *predicate*, and an object. In this way, a triple can describe a relationship (via the *predicate*) of the subject to the object. The official definition can be found [here](#).

Uniform resource identifier (URI)

A string of characters used to uniquely identify a *resource*. It is defined in [RFC3986, section 3](#). An example of a URI would be "<http://purl.org/dc/elements/1.1/>".

Universally Unique Identifier (UUID)

A "practically" unique identifier that is used to identify a *resource*.

Version

A snapshot of a *resource* that is saved in version history for later access.