

API-M


Unable to render {include} The included page could not be found.

- [Introduction](#)
- [Methods](#)
 - [Datastream Management](#)
 - [addDatastream](#)
 - [compareDatastreamChecksum](#)
 - [getDatastream](#)
 - [getDatastreamHistory](#)
 - [getDatastreams](#)
 - [modifyDatastreamByReference](#)
 - [modifyDatastreamByValue](#)
 - [setDatastreamState](#)
 - [setDatastreamVersionable](#)
 - [purgeDatastream](#)
 - [Relationship Management](#)
 - [addRelationship](#)
 - [getRelationships](#)
 - [purgeRelationship](#)
 - [Object Management](#)
 - [modifyObject](#)
 - [purgeObject](#)
 - [export](#)
 - [getNextPID](#)
 - [getObjectXML](#)
 - [ingest](#)
 - [validate](#)
- [WSDL](#)


Introduction

The Fedora Management service defines an open interface for administering the repository, including creating, modifying, and deleting digital objects, or components within digital objects. The Management service interacts with the underlying repository system to read content from and write content to the digital object and datastream storage areas. The Management service exposes a set of operations that enable a client to view and manipulate digital objects from an abstract perspective, meaning that a client does not need to know anything about underlying storage formats, storage media, or storage management schemes for objects. Also, the underlying repository system handles the details of storing datastream content within the repository, as well as mediating connectivity for datastreams that reference external content.

Ensure DC, RELS-EXT and RELS-INT are versionable if using Managed Content

 Due to an outstanding bug [FCREPO-849](#), if you use Managed Content for DC, RELS-EXT or RELS-INT then please make sure these datastreams are versionable (the default setting for versionable is "true", so if you haven't specified this datastream property then you are safe). Ensure that you don't inadvertently set this property to "false" for these datastreams when using the API methods.

Deprecation of "force" parameter

 The "force" parameter on modify/purge operations is deprecated, and will be removed in a future release. This parameter has never been used in practice.

Methods

Datastream Management

addDatastream

Creates a new Datastream in the object.

Input parameters:

- *String pid* The PID of the object.
- *String dsID* The datastream ID (64 characters max). If null, Fedora will generate the value.
- *String[] altIDs* Alternate identifiers for the datastream. Can be null.
- *String dsLabel* The label for the datastream. Can be null.
- *boolean versionable* Enable versioning of the datastream.
- *String MIMETYPE* The mime-type of the datastream. Can be null.
- *String formatURI* The format URI of the datastream. Can be null.
- *String dsLocation* Location of managed, redirect, or external referenced datastream content.
- *String controlGroup* One of "X", "M", "R", or "E" (Inline XML, Managed Content, Redirect, or External Referenced). Required
- *String dsState* One of "A", "D", or "I" (Active, Deleted, or Inactive).
- *String checksumType* The algorithm used to compute the checksum. Possible values are MD5, SHA-1, SHA-256, SHA-386, SHA-512, DISABLED
- *String checksum* The value of the checksum represented as a hexadecimal string. Can be null.
- *String logMessage* A log message.

Returns:

- *String* The datastreamID of the newly added datastream.

compareDatastreamChecksum

Verifies that the Datastream content has not changed since the checksum was initially computed.

Input parameters:

- *String pid* The PID of the object.
- *String dsID* The datastream ID.
- *String versionDate* A date/time indicating the version of the datastream to verify. If null, Fedora will use the most recent version.

Returns:

- *String* The checksum if there is no difference, a message indicating checksum failure otherwise.

getDatastream


Gets the specified datastream.

Input parameters:

- *String pid* The pid of the object.
- *String dsID* The datastream ID.
- *String asOfDateTime* The date/time stamp specifying the desired version of the object. If null, the current version of the object (the most recent time) is assumed.

Returns:

Managed Content datastream size

 Since Fedora 3.4, the size of datastreams is returned for inline XML and managed content datastreams (but not for external or referenced content). Prior to this the size is valid only for inline XML datastreams. Please note that a valid size for managed content datastreams will only be returned for datastreams that are new or have been modified after installing Fedora 3.4. Datastreams created in earlier versions will continue to return a size of zero. A tool for calculating and storing the size of these datastreams will be provided in a future release.

- *Datastream*:
 - *DatastreamControlGroup controlGroup* - String restricted to the values of "X", "M", "R", or "E" (InlineXML, Managed Content, Redirect, or External Referenced).
 - *String ID* - The datastream ID (64 characters max).
 - *String versionID* - The ID of the most recent datastream version
 - *String[] altIDs* - Alternative IDs for the datastream, if any.
 - *String label* - The Label of the datastream.
 - *boolean versionable* - Whether the datastream is versionable.
 - *String MIMEType* - The mime-type for the datastream, if set.
 - *String formatURI* - The format uri for the datastream, if set.
 - *String createDate* - The date the first version of the datastream was created.
 - *long size* - The size of the datastream in Fedora. Only valid for inline XML metadata and managed content datastreams.
 - *String state* - The state of the datastream. Will be "A" (active), "I" (inactive) or "D" (deleted).
 - *String location* - If the datastream is an external reference or redirect, the url to the contents. TODO: Managed?
 - *String checksumType* - The algorithm used to compute the checksum. One of "DEFAULT", "DISABLED", "MD5", "SHA-1", "SHA-256", "SHA-385", "SHA-512".
 - *String checksum* - The value of the checksum represented as a hexadecimal string.

getDatastreamHistory

Gets all versions of a datastream, sorted from most to least recent.

Input parameters:

- *String pid* The pid of the object.
- *String dsID* The datastream ID

Returns:

- *Datastream[]* See [#getDatastream](#)

getDatastreams

Gets all datastreams in the object

Input parameters:

- *String pid* The pid of the object.
- *String asOfDateTime* The date/time stamp specifying the desired version of the object. If null, the current version of the object (the most recent time) is assumed.
- *String dsState* One of "A", "D", or "I" (Active, Deleted, or Inactive).

Returns:

- *Datastream[]* See [#getDatastream](#)

modifyDatastreamByReference

Change the referenced location for a datastream. This operation is only relevant for managed, redirect or external reference datastreams (controlgroup E, M,R).

Input parameters:

- *String pid* The PID of the object.
- *String dsID* The datastream ID.
- *String[] altIDs* Alternate identifiers for the datastream, if any.
- *String dsLabel* The label for the datastream.
- *String MIMEType* The mime type.
- *String formatURI* Optional format URI of the datastream.
- *String dsLocation* Location of managed, redirect, or external referenced datastream content.
- *String checksumType* The algorithm used to compute the checksum. One of "DEFAULT", "DISABLED", "MD5", "SHA-1", "SHA-256", "SHA-385", "SHA-512".
- *String checksum* The value of the checksum represented as a hexadecimal string.
- *String logMessage* A log message.
- *boolean force* ~~Deprecated. Force the update even if it would break a data contract.~~

Returns:

- *String* The timestamp of the operation according to the server, in ISO8601 format.

modifyDatastreamByValue

Modifies an existing Datastream in an object, by value. This operation is only valid for Inline XML Datastreams (i.e. controlGroup "X").

Input parameters:

- *String pid* The PID of the object.
- *String dsID* The datastream ID.
- *String[] altIDs* Alternate identifiers for the datastream, if any.
- *String dsLabel* The label for the datastream.
- *String MIMEType* The mime type.
- *String formatURI* Optional format URI of the datastream.
- *String dsContent* The content of the datastream.
- *String checksumType* The algorithm used to compute the checksum. One of "DEFAULT", "DISABLED", "MD5", "SHA-1", "SHA-256", "SHA-385", "SHA-512".
- *String checksum* The value of the checksum represented as a hexadecimal string.
- *String logMessage* A log message.
- *boolean force* ~~Deprecated. Force the update even if it would break a data contract.~~

Returns:

- *String* The timestamp of the operation according to the server, in ISO8601 format.

setDatastreamState

Sets the state of a Datastream to the specified state value.

Input parameters:

- *String pid* The PID of the object.
- *String dsID* The datastream ID.
- *String dsState* One of "A", "D", or "I" (Active, Deleted, or Inactive).
- *String logMessage* A log message.

Returns:

- *String* The timestamp of the operation according to the server, in ISO8601 format.

setDatastreamVersionable

Selectively turn versioning on or off for selected datastream. When versioning is disabled, subsequent modifications to the datastream replace the current datastream contents and no versioning history is preserved. To put it another way: No new datastream versions will be made, but all the existing versions will be retained. All changes to the datastream will be to the current version.

Input parameters:

- *String pid* The PID of the object.
- *String dsID* The datastream ID.
- *Boolean versionable* Enable versioning of the datastream.
- *String logMessage* A log message.

Returns:

- *String* The timestamp of the operation according to the server, in ISO8601 format.

purgeDatastream

Permanently removes one or more versions of a Datastream from an object.

Input parameters:

- *String pid* The PID of the object.
- *String dsID* The datastream ID.
- *String startDT* The (inclusive) start date-time stamp of the range. If null, this is taken to be the lowest possible value, and thus, the entire version history up to the endDT be purged.
- *String endDT* The (inclusive) ending date-time stamp of the range. If null, this is taken to be the greatest possible value, and thus, the entire version history back to the startDT will be purged.
- *String logMessage* A log message.
- *Boolean force* ~~Deprecated. Force the update even if it would break a data contract.~~

Returns:

- *String* The timestamp of the operation according to the server, in ISO8601 format.

Relationship Management

pid parameter changes



From version 3.3 onwards, the "pid" parameter of the relationships methods has been changed to "subject". The parameter identifies the subject of a relationship and should be either a Fedora object URI (eg info:fedora/demo:333) or a Fedora object's datastream URI (eg info:fedora/demo:333/DS1). Subjects that identify a Fedora object will result in operations on RELS-EXT, and subjects that identify a datastream will result in operations on RELS-INT. The pid format for a subject (eg demo:333) is still accepted, but is deprecated as of version 3.3.

WSDL note: The WSDL for versions 3.3 onwards continues to identify this parameter as "pid" to avoid breaking any existing SOAP clients. The parameter is interpreted as a relationship's subject as above. The SOAP parameter name will be changed to "subject" in a future release.

The relationship management methods manipulate the content of the RELS-EXT and RELS-INT datastreams. The datastream to be modified is determined from the subject of the relationship, ie a subject of info:fedora/demo:333 will result in changes to RELS-EXT in demo:333, and a subject of info:fedora/demo:333/DS1 will result in changes to RELS-INT in demo:333. These modifications will also be propagated to the Resource Index if it is enabled.

addRelationship

Creates a new relationship in the object. Adds the specified relationship to the object's RELS-EXT or RELS-INT Datastream. If the Resource Index is enabled, the relationship will be added to the Resource Index.

A rdf tuple consist of an object or datastream (the subject), having a predicate relating it to a target (the object). The object can either be a literal value, or a URI (which can identify for example a Fedora object or a datastream).

Input parameters:

- *String subject* The subject. Either a Fedora object URI (eg info:fedora/demo:333) or a datastream URI (eg info:fedora/demo:333/DS1).
- *String relationship* The predicate.
- *String object* The object (target).
- *boolean isLiteral* A boolean value indicating whether the object is a literal.
- *String datatype* The datatype of the literal. Optional.

Returns:

- *boolean* True if and only if the relationship was added.

getRelationships

Get the relationships asserted in the object's RELS-EXT or RELS-INT Datastream that match the given criteria.

Input parameters:

- *String subject* The subject. Either a Fedora object URI (eg info:fedora/demo:333) or a datastream URI (eg info:fedora/demo:333/DS1).
- *String relationship* The predicate to match. A null value matches all predicates.

Returns:

- *RelationshipTuple[]* *String subject* - The subject of the relation. Either a Fedora object URI (eg info:fedora/demo:333) or a datastream URI (eg info:fedora/demo:333/DS1).
 - *String predicate* - The predicate relating the subject and the object. Includes the namespace of the relation.
 - *String object* - The URI of the object (target) of the relation
 - *boolean isLiteral* - If true, the subject should be read as a literal value, not a URI
 - *String datatype* - If the subject is a literal, the datatype to parse the value as. Optional.

purgeRelationship

Delete the specified relationship. This method will remove the specified relationship(s) from the RELS-EXT or RELS-INT datastream. If the Resource Index is enabled, this will also delete the corresponding triples from the Resource Index.

Input parameters:

- *String subject* The subject. Either a Fedora object URI (eg info:fedora/demo:333) or a datastream URI (eg info:fedora/demo:333/DS1).
- *String relationship* The predicate, null matches any predicate.
- *String object* The object, null matches any object.
- *boolean isLiteral* A boolean value indicating whether the object is a literal.
- *String datatype* The datatype of the literal. Optional.

Returns:

- *boolean* True if and only if the relationship was purged.

Object Management

modifyObject

Modify an object.

Input parameters:

- *String pid* The PID of the object.
- *String state* The new state, "A", "I" or "D". Null leaves unchanged
- *String ownerId* The ownerId for the object.
- *String logMessage* A log message.

Returns:

- *String* The timestamp of the operation according to the server, in ISO8601 format.

purgeObject

Permanently removes an object from the repository.

Parameters:

Input parameters:

- *String pid* The PID of the object.
- *Boolean force* Force the purge, even if it would break a dependency
- *String logMessage* A log message.

Returns:

- *String* The timestamp of the operation according to the server, in ISO8601 format.

export

Exports the entire digital object in the specified XML format, and encoded appropriately for the specified export context.

Parameters:

- *pid*: The pid of the object.
 - *format*: The XML format to export, one of "info:fedora/fedora-system:FOXML-1.1", "info:fedora/fedora-system:FOXML-1.0", "info:fedora/fedora-system:METSFedoraExt-1.1", "info:fedora/fedora-system:METSFedoraExt-1.0", "info:fedora/fedora-system:ATOM-1.1", or "info:fedora/fedora-system:ATOMZip-1.1".
 - *context*: The export context, which determines how datastream URLs and content are represented. One of "public", "migrate", or "archive".
- Returns: the digital object in the requested XML format.

getNextPID

Retrieves the specified number of next available pid(s) for a given pid namespace.

Parameters:

- *numPIDs*: The number of pids to retrieve.
- *pidNamespace*: The namespace of the requested pid(s). Returns: An array of the requested next available pid(s).

getObjectXML

Gets the serialization of the digital object to XML appropriate for persistent storage in the repository, ensuring that any URLs that are relative to the local repository are stored with the Fedora local URL syntax. The Fedora local URL syntax consists of the string "local.fedora.server" standing in place of the actual "hostname:port" on the URL. Managed Content (M) datastreams are stored with internal identifiers in dsLocation. Also, within selected inline XML datastreams (i.e., WSDL and SERVICE_PROFILE) any URLs that are relative to the local repository will also be stored with the Fedora local URL syntax.

Parameters:

- *pid*: The PID of the object. Returns: The digital object in Fedora's internal storage format.

ingest

Creates a new digital object in the repository. If the XML document does not specify the PID attribute of the root element, the repository will generate and return a new pid for the object resulting from this request. That pid will have the namespace of the repository. If the XML document specifies a pid, it will be assigned to the digital object provided that 1. it conforms to the Fedora pid Syntax, 2. it uses a namespace that matches the "retainPIDs" value configured for the repository, and 3. it does not collide with an existing pid of an object in the repository.

Parameters:

- *byte[] objectXML* The digital object in an XML submission format.
- *String format* The XML format of objectXML, one of "info:fedora/fedora-system:FOXML-1.1", "info:fedora/fedora-system:FOXML-1.0", "info:fedora/fedora-system:METSFedoraExt-1.1", "info:fedora/fedora-system:METSFedoraExt-1.0", "info:fedora/fedora-system:ATOM-1.1", or "info:fedora/fedora-system:ATOMZip-1.1".
- *String logMessage* A log message.

Returns:

- *String* The pid of the newly created object.
- *String asOfDateTime* A dateTime indicating the version of the datastream to verify. If null, Fedora will use the most recent version.

validate

Validates an object against it's content models

Input parameters:

- *String pid* The PID of the object.
- *String asOfDateTime* indicates that the result should be relative to the digital object and the repository as it existed at the given date and time. Formatted like yyyy-MM-dd or yyyy-MM-ddTHH:mm:ssZ. If null is given, the most recent version will be used.

Returns:

- *Validation* The validation datastructure
 - *String pid* The pid of the object
 - *Boolean valid* true if the object was valid
 - *Calendar asOfDateTime* equals the asOfDateTime that was given as a parameter
 - *String[] contentModels* The content models of the object
 - *String[] objectProblems* List of problems with the object itself
 - *DatastreamProblem[] datastreamProblems* list of datastream problems
 - *String datastreamID* The datastream ID
 - *String[] problem* List of problems with this datastream


WSDL

When running your own Fedora server, the API-M WSDL is available at **/wsdl?api=API-M**.

Example:

<http://localhost:8080/fedora/wsdl?api=API-M>

Warning

 The Apache Axis library also automatically provides an INCORRECT WSDL DOCUMENT at /fedora/services/management?wsdl. Please DO NOT USE this; it is automatically generated and will result in unexpected behavior for your clients. The recommended URL, /fedora/wsdl?api=API-M produces a copy of the API-M WSDL exactly as it was intended.

Unable to render {include} The included page could not be found.