

Purpose

Provide a basis for developing and implementing a RESTful API for the management of digital object identifiers, DOI's. Management includes creation, updating, deletion, and validation of DOI's. Finally, making these services easily available and simply addressable for use with multiple RUCore applications is a core requirement when developing this API.

Base URI

The proposed base URI is 'purl' (Persistent Uniform Resource Locator).

`/api/purl/`

From this base URI the 'DOI' (Digital Object Identifier) portion of this service would be located at:

`/api/purl/doi/`

In the future other persistent URL technologies could be supported by branching off from the `/api/purl/` base URI.

API Methods

Create

Description	An identifier will be "created" based on the information provided. If a suffix is not provided, a suffix will be minted or generated. By default all identifiers will be created with a state set to "reserved" and "export" set to no.	
Method	POST	
Restricted	Yes, passkey and IP based authentication	
Parameters	<u>Repository Object ID</u>	string : required
	<u>Suffix</u>	string : optional

Update

Description	Updates metadata at the registrar for a given repository resource	
Method	POST	
Restricted	Yes, passkey and IP based authentication	
Parameters	<u>Repository Object ID</u>	string : required
	<u>Profile</u>	enumerated value (datacite, dc, all) :optional, uses default otherwise. 'all' registers all available metadata profiles with the DOI

Verify

Description	Tests a repository resource is resolving properly by comparing the DOI stored in the objects metadata with the actual DOI response.	
Method	GET	
Restricted	No	
Parameters	<u>Repository Object ID</u>	string : required

Metadata

Description	If the source is set to “repository” this method generates metadata suitable for the DOI registrar. When the source is set to “registrar” the currently stored metadata at the registrar is retrieved and returned.	
Method	POST	
Restricted	Yes, passkey and IP based authentication	
Parameters	<u>Identifier</u>	string : required (Repository ID when source is “repository”, DOI when the source is set to “registrar”)
	<u>Source</u>	enumerated value (registrar, repository) : required
	<u>Profile</u>	enumerated value (datacite, dc, all):optional, uses default otherwise. Only relevant when source is set to “repository.” When source is set to “registrar” the profile the metadata is stored in is returned.

State

Description	Either reports the current state of an identifier or changes the state of an identifier. When changing the state of an identifier to “public” the export flag will be set to “yes.” When setting an identifier to “unavailable” the export flag will be set to “no.”	
Method	POST	
Restricted	Yes, passkey and IP based authentication	
Parameters	<u>Identifier</u>	string : required (Can be a DOI or Repository Object ID)
	<u>Action</u>	enumerated value (current, update) : required
	<u>State</u>	enumerated value (unavailable, public) : required when action is set to update
	<u>Reason</u>	string : optional (When marking a DOI “unavailable” an optional reason can be supplied)

Status

Description	Reports the current status of the DOI provider and/or a subsystem	
Method	GET	
Restricted	No	
Parameters	<u>Subsystem</u>	string : optional (the name of a subsystem or * for all subsystems statuses)

Delete

Description	Deletes an identifier if its state is “reserved.”	
Method	DELETE	
Restricted	Yes, passkey and IP based authentication	
Parameters	<u>Identifier</u>	string : required (Can be a DOI or Repository Object ID)

EZID Identifier Service

The California Digital Library offers an identifier management service called EZID. This service will be used when creating and managing DOIs. The following are some initial findings after research the EZID identifier service.

- 1) When creating a DOI a corresponding ARK (Archival Resource Key) is created automatically. This cannot be prevented and presents us an option to also store and/or use the ARK with the object.
 - a. Should the generated ARK be ignored?
 - b. Should it be stored in the objects metadata, but not displayed in any public interface?
 - c. Should the ARK be stored and displayed alongside the DOI?

Source: <http://n2t.net/ezid/doc/apidoc.html#shadow-arks>

- 2) EZID provides the capability to supply metadata about the resource when the DOI is created. The metadata can follow one of three profiles.
 - I. ERC (Electronic Resource Citations)
 - II. Datacite
 - III. DC (Dublin Core)

Metadata Working Group will need to be informed of these options and they must offer a mapping of our data model to the preferred profile.

Source: <http://n2t.net/ezid/doc/apidoc.html#internal-metadata>

Implementation

Once the new API is implemented all applications that either create or manage persistent URL's will need to have this functionality added. These applications include:

- 1) Workflow management System
- 2) DLR/EDIT
- 3) RUanalytic Tool