
provneo4j Documentation

Release 0.2.0

German Aerospace Center (DLR)

Mar 23, 2017

Contents

1	Installation	3
2	Usage	5
3	Unsupported features and known issues	7
4	Contribute	9
5	API Documentation	11
5.1	provneo4j package	11
6	Indices and tables	17
	Python Module Index	19

provneo4j is a Python client for storing [PROV](#) documents in [Neo4j](#).

This library used the Python library [prov](#).

CHAPTER 1

Installation

Install by running:

```
pip install provneo4j
```


Import the API and connect to the Neo4j database:

```
import provneo4j.api
provneo4j_api = provneo4j.api.Api(base_url="http://localhost:7474/db/data", username=
↳ "neo4j", password="neo4j")
```

Import the `prov` library and generate a PROV document. As an example, you can just use the `primer example`:

```
from prov.model import ProvDocument, Namespace, Literal, PROV, Identifier
import datetime

prov_document = primer_example()
```

Now store the PROV document in Neo4j:

```
provneo4j_api.document.create(prov_document, name="Primer Example")
```

The result in Neo4j looks like this:

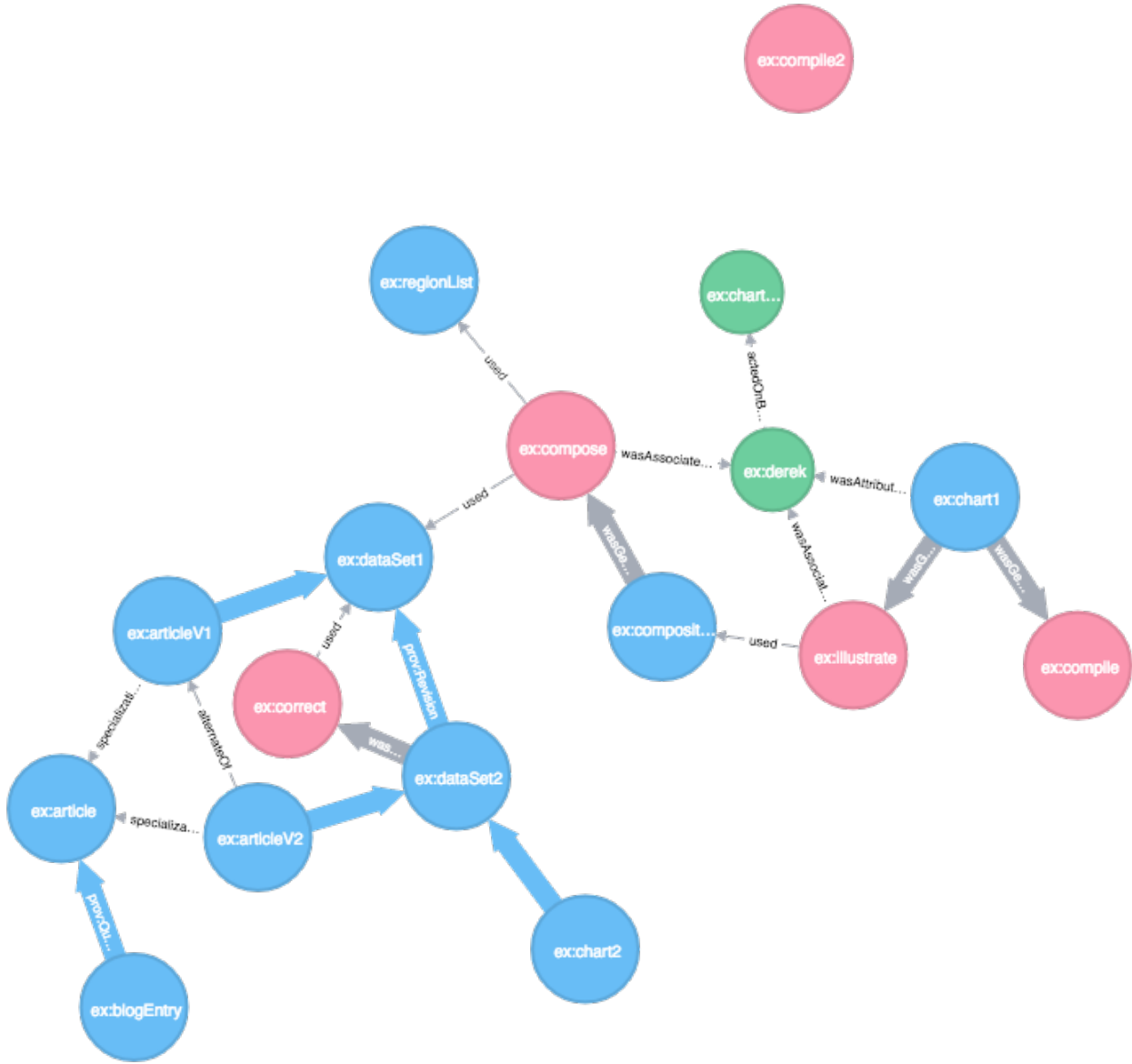


Fig. 2.1: Primer example in Neo4j.

Unsupported features and known issues

- Authentication: The library only support the authentication with username / password. No public access
- [Primer_example](#): The test for the primer_example from the prov library fails on the first run
- Public access: The library don't support the public / private access flag

CHAPTER 4

Contribute

- Issue Tracker: [GitHub issues](#).
- Source Code: [GitHub](#).

provneo4j package

provneo4j.api module

class `provneo4j.api.Api` (*username=None, password=None, base_url=None*)

Bases: `object`

Main Neo4J PROV API client object

Most functions are not used directly but are instead accessed by functions of the Document, BundleManager and Bundle objects.

To create a new Api object:

```
>>> from provneo4j.api import Api
>>> api = Api(username="your_neo4j_username" password="your_neo4j_password")
```

Note: The username and api_key parameters can also be omitted in which case the client will look for `NEO4J_USERNAME` and `NEO4J_PASSWORD` environment variables.

add_bundle (*document_id, prov_bundle, identifier*)

delete_document (*document_id*)

document

get_bundle (*document_id, bundle_id, prov_format=<class 'prov.model.ProvDocument'>*)

get_bundles (*document_id*)

get_document_meta (*document_id*)

get_document_prov (*document_id, prov_format=<class 'prov.model.ProvDocument'>*)

post_document (*prov_document, prov_format, name, public=False*)

provneo4j.bundle module

class `provneo4j.bundle.Bundle` (*api, document, bundle*)

Bases: `object`

created_at

Returns When the bundle was added

Return type `datetime.datetime`

identifier

Returns Identifier of the document, used as index on `provneo4j.bundle_manager.BundleManager`.

Return type `str`

prov

Returns This bundle's provenance

Return type `prov.model.ProvDocument`

provneo4j.bundle_manager module

class `provneo4j.bundle_manager.BundleManager` (*api, document*)

Bases: `object`

A document's bundle manager.

This is an iterable and will iterate through all of a document's bundles.

Note: Iteration is expensive, consider using `provneo4j.document.Document.prov.bundles` instead!

Example getting and adding bundles:

```
>>> api = Api()
>>> api.document.create(prov_document, name="name")
>>> api.bundles
A BundleManager object for this document
>>> api.bundles['ex:bundle']
A Bundle with the identifier given (if exists)
>>> api.bundles['ex:new_bundle'] = prov_bundle
Saves a new bundle with the identifier specified
```

refresh()

Reload list of bundles from the store

Returns `self`

provneo4j.document module

exception `provneo4j.document.AbstractDocumentException`

Bases: `provneo4j.document.DocumentException`

class provneo4j.document.Document (*api*)

Bases: object

ProvStore Document model.

Note: This class should not be instantiated manually but should be accessed via `provneo4j.api.Api.document()` like so:

```
>>> from provneo4j.api import Api
>>> api = Api()
>>> api.document
<provneo4j.document.Document at ...>
```

abstract

True if this document doesn't reference a ProvStore document yet

add_bundle (*prov_bundle*, *identifier*)

Verbose method of adding a bundle.

Can also be done as:

```
>>> api = Api()
>>> document = api.document.get(148)
>>> document.bundles['identifier'] = prov_bundle
```

Parameters

- **prov_bundle** (*prov.model.ProvDocument* or *str*) – The bundle to be added
- **identifier** (*str*) – URI or QName for this bundle

bundles

Returns This document's bundle manager

Return type `provstore.bundle_manager.BundleManager`

create (*prov_document*, *prov_format=None*, *refresh=False*, ***props*)

Create a document on ProvStore.

Parameters

- **prov_document** (*prov.model.ProvDocument* or *str*) – The document to be stored
- **prov_format** (*str* or *None*) – The format of the document provided
- **refresh** (*bool*) – Whether or not to load back the document after saving
- **props** (*dict*) – Properties for this document [**name** (required), **public** = False]

Returns This document itself but with a reference to the newly stored document

Return type `provstore.document.Document`

Raises `ImmutableDocumentException` – If this instance already refers to another document

created_at

Returns When the document was created

Return type `datetime.datetime`

delete ()

Remove the document and all of its bundles from ProvStore.

Warning: Cannot be undone.

get (document_id)

Associate this model with a document on ProvStore.

Example::

```
>>> api = Api()
>>> api.document.get(148)
https://provenance.ecs.soton.ac.uk/store/api/v0/documents/148
>>> api.id
148
>>> api.name
ex:bundles1-sep
```

Parameters document_id – The document ID on ProvStore

Returns `self`

id

Unique ID of the document as defined by ProvStore. Used in `get ()` and `set ()`

name

Name of document as seen on ProvStore

owner

Username of document creator

prov

Provenance stored for this document as `prov.model.ProvDocument`

public

Is this document visible to anyone?

read (document_id=None)

Load the document contents and metadata from the server.

The following are equivalent::

```
>>> api = Api()
>>> api.set(148).read()
>>> api.get(148)
```

Parameters document_id – (optional) Set the document ID if this is an abstract document.

Returns `self`

read_meta (document_id=None)

Load metadata associated with the document

Note: This method is called automatically if needed when a property is first accessed. You will not normally have to use this method manually.

Parameters `document_id` – (optional) set the document id if this is an `abstract()` document

Returns `self`

read_prov (`document_id=None`)

Load the provenance of this document

Note: This method is called automatically if needed when the `prov()` property is accessed. Manual use of this method is unusual.

Parameters `document_id` – (optional) set the document id if this is an `abstract()` document

Returns `:py:class:`prov.model.ProvDocument`

refresh ()

Update information about the document from ProvStore.

Returns `self`

save (`prov_document`, `prov_format=None`, `refresh=False`, `**props`)

Create a document on ProvStore.

Parameters

- **prov_document** (`prov.model.ProvDocument` or `str`) – The document to be stored
- **prov_format** (`str` or `None`) – The format of the document provided
- **refresh** (`bool`) – Whether or not to load back the document after saving
- **props** (`dict`) – Properties for this document [**name** (required), **public** = `False`]

Returns This document itself but with a reference to the newly stored document

Return type `provstore.document.Document`

Raises `ImmutableDocumentException` – If this instance already refers to another document

set (`document_id`)

Associate this document with a ProvStore document without making any calls to the API. `:param int document_id`: ID of the document on ProvStore `:return: self`

url

URL of document on ProvStore

Example

```
>>> stored_document.url
'https://provenance.ecs.soton.ac.uk/store/documents/148'
```

views

Number of views this document has received on ProvStore

exception `provneo4j.document.DocumentException`

Bases: `exceptions.Exception`

exception `provneo4j.document.EmptyDocumentException`

Bases: `provneo4j.document.DocumentException`

exception `provneo4j.document.ImmutableDocumentException`

Bases: `provneo4j.document.DocumentException`

CHAPTER 6

Indices and tables

- `genindex`
- `modindex`
- `search`

p

provneo4j.api, 11
provneo4j.bundle, 12
provneo4j.bundle_manager, 12
provneo4j.document, 12

A

abstract (provneo4j.document.Document attribute), 13
AbstractDocumentException, 12
add_bundle() (provneo4j.api.Api method), 11
add_bundle() (provneo4j.document.Document method), 13
Api (class in provneo4j.api), 11

B

Bundle (class in provneo4j.bundle), 12
BundleManager (class in provneo4j.bundle_manager), 12
bundles (provneo4j.document.Document attribute), 13

C

create() (provneo4j.document.Document method), 13
created_at (provneo4j.bundle.Bundle attribute), 12
created_at (provneo4j.document.Document attribute), 13

D

delete() (provneo4j.document.Document method), 14
delete_document() (provneo4j.api.Api method), 11
Document (class in provneo4j.document), 12
document (provneo4j.api.Api attribute), 11
DocumentException, 16

E

EmptyDocumentException, 16

G

get() (provneo4j.document.Document method), 14
get_bundle() (provneo4j.api.Api method), 11
get_bundles() (provneo4j.api.Api method), 11
get_document_meta() (provneo4j.api.Api method), 11
get_document_prov() (provneo4j.api.Api method), 11

I

id (provneo4j.document.Document attribute), 14
identifier (provneo4j.bundle.Bundle attribute), 12
ImmutableDocumentException, 16

N

name (provneo4j.document.Document attribute), 14

O

owner (provneo4j.document.Document attribute), 14

P

post_document() (provneo4j.api.Api method), 11
prov (provneo4j.bundle.Bundle attribute), 12
prov (provneo4j.document.Document attribute), 14
provneo4j.api (module), 11
provneo4j.bundle (module), 12
provneo4j.bundle_manager (module), 12
provneo4j.document (module), 12
public (provneo4j.document.Document attribute), 14

R

read() (provneo4j.document.Document method), 14
read_meta() (provneo4j.document.Document method), 14
read_prov() (provneo4j.document.Document method), 15
refresh() (provneo4j.bundle_manager.BundleManager method), 12
refresh() (provneo4j.document.Document method), 15

S

save() (provneo4j.document.Document method), 15
set() (provneo4j.document.Document method), 15

U

url (provneo4j.document.Document attribute), 15

V

views (provneo4j.document.Document attribute), 15