

---

# **skosprovider\_heritagedata Documentation**

*Release 0.3.0*

**Flanders Heritage Agency**

August 11, 2016



<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Installation . . . . .	1
1.2	Supported Heritagedata thesauri . . . . .	1
<b>2</b>	<b>Using the providers</b>	<b>5</b>
2.1	Using HeritagedataProvider . . . . .	5
2.2	Finding concepts . . . . .	5
2.3	Using expand() . . . . .	6
<b>3</b>	<b>Development</b>	<b>7</b>
<b>4</b>	<b>API Documentation</b>	<b>9</b>
4.1	Providers module . . . . .	9
4.2	Utils module . . . . .	9
<b>5</b>	<b>History</b>	<b>11</b>
5.1	0.3.0 (2016-08-11) . . . . .	11
5.2	0.2.1 (2015-03-10) . . . . .	11
5.3	0.2.0 (2014-12-19) . . . . .	11
5.4	0.1.0 (2014-10-08) . . . . .	11
<b>6</b>	<b>Glossary</b>	<b>13</b>
<b>7</b>	<b>Indices and tables</b>	<b>15</b>



---

## Introduction

---

This library offers an implementation of the `skosprovider.providers.VocabularyProvider` interface based on the [Heritagedata Vocabularies](#). These vocabularies are used by *EH*, *RCAHMS* and *RCAHMW* in their role as curators of heritage.

### 1.1 Installation

To be able to use this library you need to have a modern version of Python installed. Currently we're supporting versions 2.7, 3.3 and 3.4 of Python.

This easiest way to install this library is through **pip** or **easy install**:

```
$ pip install skosprovider_heritagedata
```

This will download and install `skosprovider_heritagedata` and a few libraries it depends on.

### 1.2 Supported Heritagedata thesauri

The webservice provided by [heritagedata.org](#) provide access to multiple vocabularies or conceptschemes. You can select one of these vocabularies by passing a `scheme_uri` to the constructor of the `HeritagedataProvider`.

[Heritagedata Vocabulary schemes](#)

An overview of all `scheme_uri` can be provided by the following service:

[www.heritagedata.org/live/services/getSchemes?pretty](http://www.heritagedata.org/live/services/getSchemes?pretty)

```
[
  {
    "uri": "http://purl.org/heritagedata/schemes/agl_et",
    "label": "EVENT TYPE (EH)",
    "label lang": "en",
    "description": "Terminology used for recording archaeological and architectural investigative",
    "attribution": "English Heritage"
  },
  {
    "uri": "http://purl.org/heritagedata/schemes/1",
    "label": "Monument Type Thesaurus (Scotland)",
    "label lang": "en",
    "description": "Monument types relating to the archaeological and built heritage of Scotland",
    "attribution": "RCAHMS"
  }
]
```

```
},
{
  "uri": "http://purl.org/heritagedata/schemes/2",
  "label": "Archaeological Objects Thesaurus (Scotland)",
  "label lang": "en",
  "description": "Objects made by human activity.",
  "attribution": "RCAHMS"
},
{
  "uri": "http://purl.org/heritagedata/schemes/3",
  "label": "Maritime Craft Thesaurus (Scotland)",
  "label lang": "en",
  "description": "Types of craft that survive as wrecks, or are documented as losses, in Scotland.",
  "attribution": "RCAHMS"
},
{
  "uri": "http://purl.org/heritagedata/schemes/11",
  "label": "PERIOD (WALES)",
  "label lang": "en",
  "description": "A list of periods for use in Wales.",
  "attribution": "RCAHMS"
},
{
  "uri": "http://purl.org/heritagedata/schemes/eh_tmt2",
  "label": "MONUMENT TYPE (EH)",
  "label lang": "en",
  "description": "Classification of monument type records by function.",
  "attribution": "English Heritage"
},
{
  "uri": "http://purl.org/heritagedata/schemes/560",
  "label": "ARCHAEOLOGICAL SCIENCES (EH)",
  "label lang": "en",
  "description": "Used for recording the techniques, recovery methods and materials associated with archaeological excavations.",
  "attribution": "English Heritage"
},
{
  "uri": "http://purl.org/heritagedata/schemes/eh_tbm",
  "label": "BUILDING MATERIALS (EH)",
  "label lang": "en",
  "description": "Thesaurus of main constructional material types (eg. the walls) for indexing buildings.",
  "attribution": "English Heritage"
},
{
  "uri": "http://purl.org/heritagedata/schemes/eh_tmc",
  "label": "MARITIME CRAFT TYPE (EH)",
  "label lang": "en",
  "description": "A thesaurus of craft types which survive as wrecks in English Heritage's maritime collection.",
  "attribution": "English Heritage"
},
{
  "uri": "http://purl.org/heritagedata/schemes/eh_period",
  "label": "PERIOD (EH)",
  "label lang": "en",
  "description": "English Heritage Periods List",
  "attribution": "English Heritage"
},
{
```

```
[
  {
    "uri": "http://purl.org/heritagedata/schemes/eh_com",
    "label": "COMPONENTS (EH)",
    "label lang": "en",
    "description": "Terminology covering divisions and structural elements of a building or monument",
    "attribution": "English Heritage"
  },
  {
    "uri": "http://purl.org/heritagedata/schemes/eh_evd",
    "label": "EVIDENCE (EH)",
    "label lang": "en",
    "description": "Terminology covering the existing physical remains of a monument, or the means of their preservation",
    "attribution": "English Heritage"
  },
  {
    "uri": "http://purl.org/heritagedata/schemes/mda_obj",
    "label": "FISH Archaeological Objects Thesaurus",
    "label lang": "en",
    "description": "Originally developed by the Archaeological Objects Working Party and published by English Heritage",
    "attribution": "English Heritage"
  },
  {
    "uri": "http://purl.org/heritagedata/schemes/10",
    "label": "MONUMENT TYPE THESAURUS (WALES)",
    "label lang": "en",
    "description": "Classification of monument types in Wales by function",
    "attribution": "RCAHMW"
  }
]
```





---

## Using the providers

---

### 2.1 Using HeritagedataProvider

The `HeritagedataProvider` is a general provider for the Heritagedata vocabularies. Its use is identical to all other `SKOSProviders`. A `scheme_uri` is required to indicate the vocabulary to be used. Please consult *Supported Heritagedata thesauri* for a complete list.

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
'''
This script demonstrates using the HeritagedataProvider to get the concept of
'POST MEDIEVAL'.
'''

from skosprovider_heritagedata.providers import HeritagedataProvider

periodprovider = HeritagedataProvider(
    {'id': 'Heritagedata'},
    scheme_uri='http://purl.org/heritagedata/schemes/eh_period'
)

pm = periodprovider.get_by_id('PM')

print('Labels')
print('-----')
for l in pm.labels:
    print(l.language + ': ' + l.label + ' [' + l.type + ']')

print('Notes')
print('-----')
for n in pm.notes:
    print(n.language + ': ' + n.note + ' [' + n.type + ']')
```

### 2.2 Finding concepts

See the `skosprovider_heritagedata.providers.HeritagedataProvider.find()` method for a detailed description of how this works.

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
```

```
'''
This script demonstrates using the HeritagedataProvider to find the concepts with 'iron' in their label
'''

from skosprovider_heritagedata.providers import HeritagedataProvider

periodprovider = HeritagedataProvider(
    {'id': 'Heritagedata'},
    scheme_uri='http://purl.org/heritagedata/schemes/eh_period'
)

results = periodprovider.find(
    {
        'label': 'iron',
        'type': 'concept'
    }
)

print('Results')
print('-----')
for result in results:
    print(result)
```

## 2.3 Using expand()

The expand methods return the id's of all the concepts that are narrower concepts of a certain concept or collection.

See the `skosprovider_heritagedata.providers.HeritagedataProvider.expand()` method for a detailed description of how this works.

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
'''
This script demonstrates using the HeritagedataProvider to expand a concept
'''

from skosprovider_heritagedata.providers import HeritagedataProvider

periodprovider = HeritagedataProvider(
    {'id': 'Heritagedata'},
    scheme_uri='http://purl.org/heritagedata/schemes/eh_period'
)

results = periodprovider.expand('PM')

print('Results')
print('-----')
for result in results:
    print(result)
```

---

## Development

---

Skosprovider\_heritagedata is being developed by the [Flanders Heritage Agency](#).

Since we place a lot of importance on code quality, we expect to have a good amount of code coverage present and run frequent unit tests. All commits and pull requests will be tested with [Travis-ci](#). Code coverage is being monitored with [Coveralls](#).

Locally you can run unit tests by using [pytest](#) or [tox](#). Running pytest manually is good for running a distinct set of unit tests. For a full test run, tox is preferred since this can run the unit tests against multiple versions of python.

```
# Setup for development
$ python setup.py develop
# Run unit tests for all environments
$ tox
# No coverage
$ py.test
# Coverage
$ py.test --cov skosprovider_heritagedata --cov-report term-missing tests
# Only run a subset of the tests
$ py.test skosprovider_heritagedata/tests/test_providers.py
```

Please provide new unit tests to maintain 100% coverage. If you send us a pull request and this build doesn't function, please correct the issue at hand or let us know why it's not working.



---

**API Documentation**

---

**4.1 Providers module**

**4.2 Utils module**



### 5.1 0.3.0 (2016-08-11)

- Compatible with `SkosProvider 0.6.0`.
- Allow passing a custom requests Session to a provider. (#14)

### 5.2 0.2.1 (2015-03-10)

- Fix an issue where calls that include a *language* parameter would fail because certain methods were not expecting extra keyword parameters. (#12)
- Some documentation clarifications. (#11)

### 5.3 0.2.0 (2014-12-19)

- Compatible with `SkosProvider 0.5.0`.
- Each Concept or Collection now also provides information on the ConceptScheme it's part of.
- Fix some issues with UTF-8 encoding.
- Fixed some issues with Python 2.x/3.x compatibility.
- Provider now throws a `ProviderNotAvailableException` when the underlying service is down.

### 5.4 0.1.0 (2014-10-08)

- Initial version
- Compatible with `SkosProvider 0.3.0`.





---

## Glossary

---

**EH** English Heritage.

**RCAHMS** The Royal Commission on the Ancient and Historical Monuments of Scotland.

**RCAHMW** The Royal Commission on the Ancient and Historical Monuments of Wales.

**RDF** *Resource Description Framework*. A very flexible model for data definition organised around *triples*. These triples forms a directed, labeled graph, where the edges represent the named link between two resources, represented by the graph nodes.

**SKOS** *Simple Knowledge Organization System*. An general specification for Knowledge Organisation Systems (thesauri, word lists, authority files, ...) that is commonly serialised as *RDF*.

**URI** A *Uniform Resource Identifier*.

**URN** A URN is a specific form of a *URI*.



---

## Indices and tables

---

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



**E**

EH, **13**

**R**

RCAHMS, **13**

RCAHMW, **13**

RDF, **13**

**S**

SKOS, **13**

**U**

URI, **13**

URN, **13**