ESG-F F2F 2014

Towards a (Really Simple) Controlled Vocabulary Service

Mark A. Greenslade
Institut Pierre Simon Laplace (IPSL)

Introduction

Preamble

Set Theory

Convention over configuration

Definitions

Controlled Vocabulary (CV) = set of Terms

Term = Object graph + CV metadata

Domain = CV partitioning key

Type = CV identifier key

Name = Term identifier key

Dictionary = Superset of CV's

Context = Constrained subset of CV's

Sources
(climate modelling)

ESG-F Search Facets

Metafor CIM Model Components

CF Names / CSDMS Names

Agents

Server: **ESG-F Publisher**

Server: **IPSL Synchro Data**

UI: **ESG-F COG Front End**

UI: **ES-DOC Viewer**

UI: **ES-DOC Comparator**

Server: **ESG-F DRS Lib**

Example Dictionary

CV - 1

type: climate.modelling.institute

description: set of climate modelling institutes

Term - 1

type: climate.modelling.institute

name: ipsl

description: Institut Pierre Simon Laplace

url: https://www.ipsl.fr

CV - 2

type: climate.modelling.model

description: set of climate models

Term - 2

type: climate.modelling.model

name: ipsl-cm5a-lr

description: IPSL CMIP5 low resolution

synonyms: ipslcm5a-lr, ipslcm5alr

institute: ipsl

CV - 3

type: climate.modelling.model.component

description: set of climate model components

Term - 3

type: climate.modelling.model.component

name: aerosols

description: Aerosols top level component

Term - 4

type: climate.modelling.model.component

name: emission

description: Aerosols emission and concentration

parent: aerosols

Context - 1

name: cmip5

domain: climate.modelling

Context - 2

Associations

institute: ipsl,mohc,mpi-m

model: ipsl-cm5a-lr,ipsl-cm5a-mr

model.component: aerosols, aerosols>emission

Building Blocks

Building Blocks - 1

RSCV-Archive

- Version controlled file system
- https://github.com/XXX/rscv-archive

Building Blocks - 2

RSCV-Client

- Client tool for accessing / managing archive
- ISimpleControlledVocabularyClient
- Python, Java (?)
- https://github.com/XXX/rscv-client

Building Blocks - 3

RSCV-Web Service

- Web service accessing archive
- Leverages RSCV-Client
- Exposes static merged file(s)
- Exposes simple search endpoint
- https://github.com/XXX/rscv-web-service

RSCV-Archive - 1

cv-archive

dictionary

climate

modelling

institute

➡ ipsl.json (term 1)

model

➡ ipsl-cm5a-lr.json (term 2)

model.component

➡ aerosols.json (term 3)

➡ aerosols>emission.json (term 4)

Term - 2

type: climate.modelling.model

name: ipsl-cm5a-lr

description: IPSL CMIP5 low resolution

synonyms: ipslcm5a-lr

uid: 724ca179-3f95-481e-9609-6dc345a67f0d

status: accepted

associations: climate.modelling.institute.ipsl

Term - 2 - simplified

type: model

name: ipsl-cm5a-lr

description: IPSL CMIP5 low resolution

synonyms: ipslcm5a-lr

uid: 724ca179-3f95-481e-9609-6dc345a67f0d

associations: institute.ipsl

RSCV-Archive - 2

cv-archive

context

climate

modelling

cmip5

- ➡ associations-1.json
- ➡ associations-2.json
- ➡ … etc

RSCV-Client

ISimpleControlledVocabularyClient

init

retrieve

parse, validate, associate

add, update, delete, destroy, merge

commit, rollback

RSCV-Client

```
>>> import rscv

>>>  rscv.init(PATH_TO_ARCHIVE)

>>>  rscv.init(“climate.modelling”)
```

RSCV-Client

```
>>>  all = rscv.retrieve()
>>>  print(type(all))
>>>  dict

>>>  models = rscv.retrieve(“model”)
>>>  print(type(models))
>>>  list

>>>  model = rscv.retrieve(“model.IPSL-Cm5A-lr”)
>>>  print(type(model))
>>>  dict

>>>  model = rscv.retrieve(“model.IPSL-Cm5A-lr”, “namedtuple”)
>>>  print(type(model))
>>>  namedtuple

>>>  model = rscv.retrieve(“model.IPSL-Cm5A-lr”, “json”)
>>>  print(type(model))
>>>  unicode

>>>  model = rscv.retrieve(“model.IPSL-Cm5A-lr”, “xml”)
>>>  print(type(model))
>>>  unicode

>>>  model = rscv.retrieve(“model.IPSL-Cm5A-lr”, “rdf”)
>>>  print(type(model))
>>>  unicode
```

RSCV-Client

```
>>>  model_name = rscv.parse(“model”, “IPSL-Cm5A-lr”)
>>>  print(model_name)
>>>  ipsl-cm5a-lr

>>>  institute_name = rscv.parse(“climate.modelling.institute”, “IPSL”)
>>>  print(institute_name)
>>>  ipsl

>>>  is_valid = rscv.validate(“climate.modelling.institute”, “ipsl”)
>>>  print(is_valid)
>>>  True

>>>  is_valid = rscv.validate(“climate.modelling.institute”, “IPSL”)
>>>  print(is_valid)
>>>  False
```

RSCV-Client

```
>>>  model = {
          "description": “ESGF-F2F-2014 temporary model”,
          "name": “ESGF-2014-F2F“
        }
>>>  model = rscv.add(model)
>>>  rscv.associate(model, “institute.pcmdi”)

>>>  print(model)
{
    "description": "ESGF-F2F-2014 temporary model",
    "meta": {
        "associations": [
            "institute.pcmdi"
        ],
        "create_date": "2014-12-10T14:16:58.917769+00:00",
        "governance": “new",
        "name": “esgf-2014-f2f”,
        "type": “climate.modelling.model”,
        "uid": "724ca179-3f95-481e-9609-6dc345a67f0d"
    },
    "name": "ESGF-2014-F2F"
}
```

RSCV-Client

```
>>>  model = {
            "description": “ESGF-F2F-2014 temporary model”,
            "name": “ESGF-2014-F2F“,
             “owner”: “Karl Taylor”
          }
>>>  model = rscv.update(model)

>>>  print(model)
{
    "description": "ESGF-F2F-2014 temporary model",
    "meta": {
        "associations": [
            "institute.pcmdi"
        ],
        "create_date": "2014-12-10T14:16:58.917769+00:00",
        "governance": “new",
        "name": “esgf-2014-f2f”,
        "type": “climate.modelling.model”,
        "uid": “724ca179-3f95-481e-9609-6dc345a67f0d",
        "update_date": "2014-12-10T14:16:58.917769+00:00"
    },
    "name": “ESGF-2014-F2F”,
    “owner”: “Karl Taylor”
}
```

RSCV-Client

```
>>>  rscv.commit()

>>>  print(model)
{
    "description": "ESGF-F2F-2014 temporary model",
    "meta": {
        "associations": [
            "institute.pcmdi"
        ],
        "create_date": "2014-12-10T14:16:58.917769+00:00",
        "governance": “awaiting_approval”,
        "name": “esgf-2014-f2f”,
        "type": “climate.modelling.model”,
        "uid": “724ca179-3f95-481e-9609-6dc345a67f0d",
        "update_date": "2014-12-10T14:16:58.917769+00:00"
    },
    "name": “ESGF-2014-F2F”,
    “owner”: “Karl Taylor”
}
```

RSCV-Client

```
>>>  rscv.destroy(“model.esgf-f2f-2014”)
>>>  rscv.commit()

>>>  model = rscv.retrieve(“model..esgf-f2f-2014”)
>>>  print(model)
None
```

RSCV-Web-Service

Simple (tornado ???) web application

Exposes static files and a search endpoint

Leverages RSCV-Archive and RSCV-Client

Hooks into GitHub notifications to keep local copy of CV-Archive upto date

HTTP Caching & E-Tags

Mirrored ?

Finally

Must be <u>brain-dead simple</u>

IPSL Prodiguer project being built upon this concept - reuse this work

GO-ESSP 2015 Meeting in UK