OpenBudgets.eu: Fighting Corruption with Fiscal Transparency

Project Number: 645833   Start Date of Project: 01.05.2015   Duration: 30 months

Deliverable 1.4

User documentation

Dissemination Level	Public
Due Date of Deliverable	Month 7, 30.11.2015
Actual Submission Date	30.11.2015
Work Package	WP 1, Data Structure Definition for Budgets and Public Spending
Task	T 1.1
Type	Report
Approval Status	Final
Version	1.0
Number of Pages	44
Filename	D1.4 User documentation.docx

Abstract: In this deliverable we present a guide to modelling datasets according to the OpenBudgets.eu data model described in deliverables D1.2 and D1.3. Since the data model is based on the RDF Data Cube Vocabulary, we start with a guide showing how the vocabulary is used throughout the data model. Next, we define IRI patterns to be adopted by the datasets published in OpenBudgets.eu, and then we explain the process of modelling a dataset through all the necessary steps and illustrate it on examples. We also include a few modelling patterns that are to be considered during dataset transformation. We briefly mention the recommended metadata and finish with a data model reference which includes descriptions and usage examples of individual classes and properties in the core OpenBudgets.eu data model.

History

Version	Date	Reason	Revised by
0.1	06.11.2015	Version for internal review	Jakub Klímek
0.9	20.11.2015	Version for external review	Tiansi Dong
1.0	30.11.2015	Final version for submission	Jakub Klímek

Author List

Organisation	Name	Contact Information
UEP	Marek Dudáš	marek.dudas@vse.cz
UEP	Linda Horáková	lin.horak@gmail.com
UEP	Jakub Klímek	klimek@opendata.cz
UEP	Jan Kučera	jan.kucera@vse.cz
UEP	Jindřich Mynarz	mynarzjindrich@gmail.com
UEP	Lucie Sedmihradská	sedmihradska@centrum.cz
UEP	Jaroslav Zbranek	zbranek.jaroslav@gmail.com
UBONN	Tiansi Dong	tdong@uni-bonn.de

Executive Summary

This deliverable is to be used as a guide to modelling datasets according to the OpenBudgets.eu data model described in deliverables D1.2 and D1.3. It contains the RDF Data Cube Vocabulary guide showing how the vocabulary is used throughout the data model. There are also IRI patterns that should be followed when creating datasets and code lists in OpenBudgets.eu. The process of modelling a dataset is described through all necessary steps and is illustrated using examples. Also included are modelling patterns that are to be considered during dataset transformation as well as the recommended metadata and validation techniques. The last part of this deliverable is a data model reference which includes descriptions and usage examples of individual classes and properties in the core OpenBudgets.eu data model.

Abbreviations and Acronyms

CSV	Comma-Separated Values
DCV	Data Cube Vocabulary
DSD	Data Structure Definition
GDP	Gross domestic product
RDF	Resource Description Framework
SDMX	Statistical Data and Metadata eXchange
XML	eXtensible Markup Language

Table of Contents

1 Introduction

This deliverable documents the data model for budget and spending data described in deliverables D1.2 and D1.3. Its primary use is to serve as a guide for users of the data model, such as those converting data to RDF represented using this data model.

Throughout this guide we use example data to illustrate the described data modelling recommendations. In the section introducing the Data Cube Vocabulary, we use data from Eurostat on government expenditure related to GDP. The following section on modelling guidelines specific for the fiscal domain uses the budget of the European Union for the year 2014[1] as a running example. Using this example dataset we will illustrate how to model fiscal data using the OpenBudgets.eu data model. This dataset was already used as an example in Deliverable D1.2 - Design of data structure definition for public budget data (Klímek et al., 2015a). In this deliverable we will delve into greater depth regarding the modelling decisions made for this dataset.

2 Data Cube Vocabulary primer

The RDF Data Cube Vocabulary[2] (DCV) is a W3C Recommendation for representing multidimensional data in RDF. Multidimensional data is any data that consists of observed values organized along a set of dimensions that describe the observed values. Statistical data is a typical representative of multidimensional data. In fact, the DCV is compatible with the cube model that forms the base of the SDMX (Statistical Data and Metadata eXchange) standard – an international standard for exchange of statistical data and metadata (Cyganiak & Reynolds, 2014).

As it is shown later in this document, budgetary and spending data also represent multidimensional data and therefore we decided to model such data as RDF data cubes using the DCV. In this primer we introduce the DCV basics for better understanding of the way the budgetary and spending is represented in the OpenBudgets.eu project.

2.1 The Data Cube Vocabulary overview

DCV represents datasets as data cubes, i.e. collections of data that comprises of observed values (observations), associated dimensions, and metadata. The DCV provides a set of classes and properties for representing the data cubes in RDF and publishing them according to the linked data principles (see Berners-Lee, 2006). Classes, properties and their relationships that are specified in the DCV are depicted on Figure 1.

---

[1] http://open-data.europa.eu/data/dataset/budget-of-the-european-union-2014

[2] http://www.w3.org/TR/vocab-data-cube/
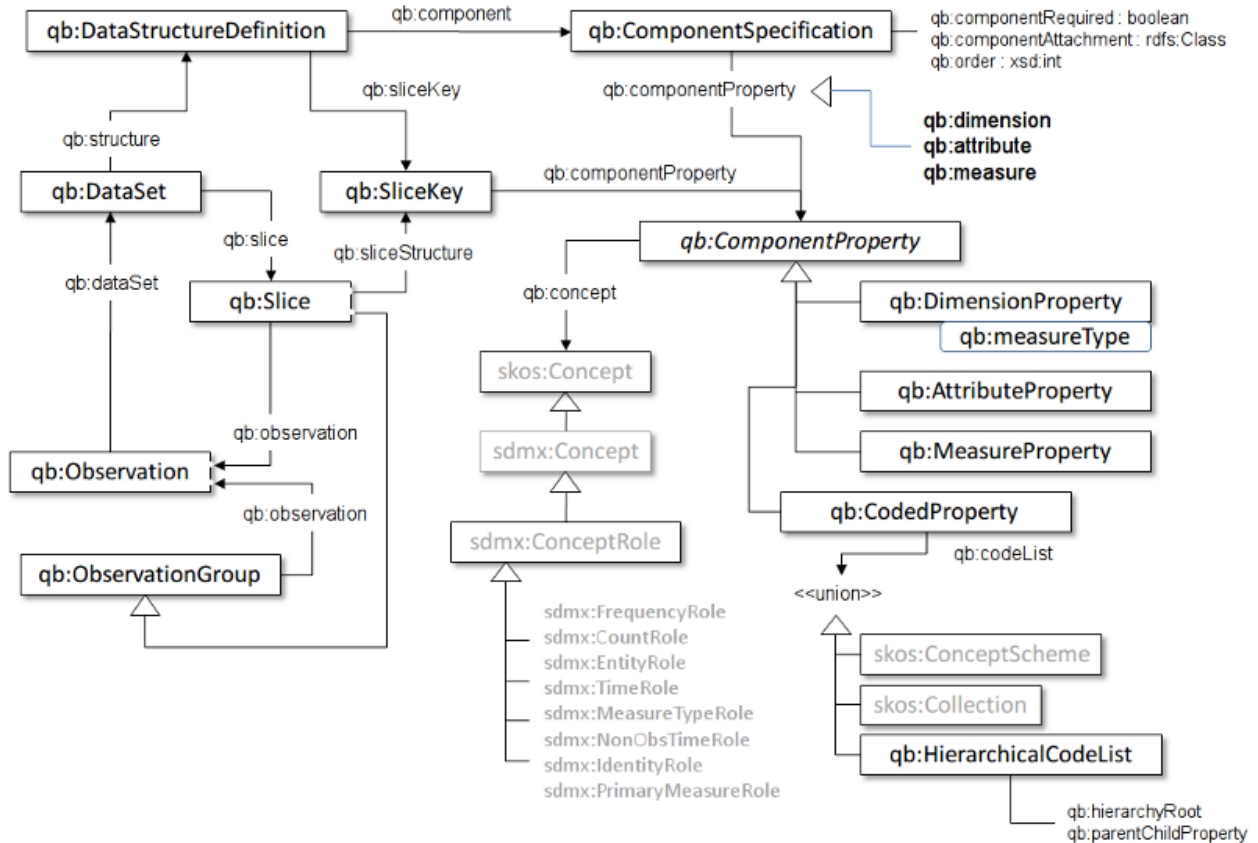
Figure 1 - Key terms and relationships in The RDF Data Cube Vocabulary, source: (Cyganiak & Reynolds, 2014)

For every dataset (`qb:DataSet`) a definition of its structure (`qb:DataStructureDefinition`) needs to be developed. This structure is made of specifications of its components properties (`qb:ComponentProperty`). There are 3 types of components properties:

- **Measures** (`qb:MeasureProperty`) – measure properties specify the types of the observed values in the dataset.
- **Dimensions** (`qb:DimensionProperty`) – dimension properties specify dimensions used in the dataset to organize the observed values in a multidimensional space.
- **Attributes** (`qb:AttributeProperty`) – attribute properties specify additional attributes of the observed values, such as currency or accuracy.

Datasets using the DCV are made of observations (`qb:Observation`). An observation might be seen as a record of measures (one or more observed values) and the respective values of the specified dimensions and attributes. By selecting specific values of one or more dimensions, a view on the data called slice (`qb:Slice`) can be defined.

We provide a more detailed discussion of the key DCV terms in the following subsections.

2.2 Observations, DataSet, and Data Structure Definition

The RDF Data Cube Vocabulary builds upon an abstract cube model, i.e. a multidimensional space where measured values are indexed by multiple dimensions. Let's illustrate this concept

using an excerpt of the total general government expenditure expressed as percentage of GDP published by Eurostat (2015) as an example.[3]

Reference area\year	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
EU (28 countries)	:	:	:	45,6	44,9	46,5	50,3	50	48,6	49	48,6	48,2
EU (27 countries)	:	:	:	45,6	44,9	46,5	50,3	50	48,6	49	48,6	48,2
Euro area (19 countries)	:	:	:	46	45,3	46,6	50,7	50,5	49,1	49,7	49,6	49,4
Euro area (18 countries)	:	:	:	46,1	45,3	46,6	50,7	50,5	49,1	49,8	49,6	49,4
Euro area (17 countries)	:	:	:	46,1	45,4	46,6	50,7	50,5	49,1	49,8	49,7	49,4

Table 1: Total general government expenditure (% of GDP, excerpt), source: excerpt from (Eurostat, 2015b)

The total general government expenditure expressed as percentage of GDP is the measured phenomenon, as illustrated in Table 1, which is indexed by two dimensions: **reference area** and **year**. The total government expenditure in EU28 in 2010 represents a single observation. The collection of observations forms a dataset, i.e. a data cube.

Any dataset represented using the DCV is an instance of the class `qb:DataSet` which contains instances of the class `qb:Observation`. In order to specify the structure of the dataset, a data structure definition needs to be developed (instance of the class `qb:DataStructureDefinition`).

A data structure definition specifies what dimensions index observations in a particular dataset and what values are measured in the observations. It might also specify what additional attributes of the observation are (or could be) provided in a dataset, such as the currency or unit of measure.

The following example shows a data structure definition for the dataset described in Table 1.

```
@prefix rdfs:           <http://www.w3.org/2000/01/rdf-schema#> .
@prefix qb:             <http://purl.org/linked-data/cube#> .
@prefix sdmx-attribute: <http://purl.org/linked-data/sdmx/2009/attribute#> .
@prefix xsd:            <http://www.w3.org/2001/XMLSchema#> .

@prefix ex-dimension:  <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-dsd:        <http://data.example.org/resource/dsd/> .
@prefix ex-measure:    <http://data.example.org/ontology/dsd/measure/> .
ex-dsd:total-general-government-expenditure a qb:DataStructureDefinition ;
  rdfs:label "Total general government expenditure"@en ;
  # Dimensions
  qb:component [
      qb:dimension ex-dimension:refPeriod ;
      qb:order 1 ;
      rdfs:label "Dimension representing a year for which the total general
government expenditure is reported"@en
  ] ;
  qb:component [
      qb:dimension ex-dimension:refArea ;
      qb:order 2 ;
      rdfs:label "Dimension representing a state or group of states for which the
total general government expenditure is reported"@en
  ] ;
  # Measure
  qb:component [
      qb:measure ex-measure:total-general-government-expenditure ;
```

---

[3] Only data for EU28, EU27 and Euro areas presented, data about the individual states omitted for the purposes of this example. Unavailable values are marked with ":". See the Eurostat website for detailed metadata:
http://ec.europa.eu/eurostat/tgm/table.do?tab=table&plugin=1&language=en&pcode=tec00023.

```
      rdfs:label "Measure representing the total general government expenditure"@en
] ;
# Attributes
qb:component [
  qb:attribute sdmx-attribute:unitMeasure ;
  qb:componentRequired true
] .
```

Example 1: Data structure definition of the dataset described in Table 1

As we can see, a data structure definition of a dataset represented using DCV consists of specifications of its components: dimensions, measures, and attributes. We introduce these components in the following section. However, you can see that we introduce dimensions `ex-dimension:refPeriod` and `ex-dimension:refArea` to represent the year, and reference area dimensions, respectively. The measure `ex-measure:total-general-government-expenditure` is introduced to represent the measured phenomenon in our example dataset. In Table 1, the total general government expenditure is expressed as % of GDP. Attribute `sdmx-attribute:unitMeasure` is used to denote the unit of measurement and declared as required component, i.e. the unit of measurement needs to be provided for every observation. Just for the purposes of this example we stick to the default attachment level of the unit of measurement attribute. However, the DCV allows to use different attachment levels. See (Cyganiak & Reynolds, 2014) for details.

In Example 2, we provide a sample of the instance data representation of the dataset in Table 1 - the dataset and observations for the EU28 region covering years 2012-2014.

```
@prefix dcterms:            <http://purl.org/dc/terms/>.
@prefix qb:                 <http://purl.org/linked-data/cube#> .
@prefix rdfs:               <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sdmx-attribute:     <http://purl.org/linked-data/sdmx/2009/attribute#> .

@prefix ex-dataset:         <http://data.example.org/resource/dataset/> .
@prefix ex-dimension:       <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-dsd:             <http://data.example.org/resource/dsd/> .
@prefix ex-measure:         <http://data.example.org/ontology/dsd/measure/> .
@prefix ex-units-of-measure: <http://data.example.org/resource/codelist/units-of-
measure/> .
# Dataset

ex-dataset:total-general-government-expenditure a qb:DataSet ;
  rdfs:label "Total general government expenditure"@en ;
  rdfs:comment "Total general government expenditure expressed as % of GDP."@en ;
  dcterms:publisher <http://openbudgets.eu> ;
  dcterms:source
<http://ec.europa.eu/eurostat/tgm/table.do?tab=table&plugin=1&language=en&pcode=tec0
0023> ;

  qb:structure ex-dsd:total-general-government-expenditure .

# Example observations

<http://data.example.org/resource/observation/total-general-government-
expenditure/2012/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2012> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 49 .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2013/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2013> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
```

```
  ex-measure:total-general-government-expenditure 48.6 .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2014/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2014> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 48.2 .
```

Example 2: Example instance data for the dataset described in Table 1

2.3 Dimensions, Measures and Attributes

Data structure definition of a dataset represented using DCV is made of specifications of its components (qb:ComponentSpecification). There are 3 types of components: measures, dimensions, and attributes. The DCV provides specific classes to represent these components: qb:MeasureProperty, qb:DimensionProperty, qb:AttributeProperty. Component specification thus links data structure definition with instances of these classes that can be shared among multiple data structure definitions.

Measures (qb:MeasureProperty) represent types of the measured phenomenon, such as population of a given area or the total general government expenditure expressed as percentage of GDP, as illustrated in Table 1.

We have already mentioned that in the data cube model the measured values are indexed by one or more dimensions. Dimensions provide additional information to the observations such as its reference period or reference area. Dimensions that are part of the structure of the given dataset are represented as instances of the class qb:DimensionProperty.

Sometimes it might be needed to provide additional information about observations that does not form dimensions of the multidimensional space, i.e. it does not index observation, such as unit of measurement, units of currency, precision or confidentiality level of a given measurement. In DCV such additional information is called an attribute. Instances of the class qb:AttributeProperty are used to represent attributes in data structure definitions.

The DCV specification (see Cyganiak & Reynolds, 2014) sets an important integrity constraint related to measures and dimensions: values for all measures and dimensions specified in the given data structure definition need to be present in every observation of a dataset. Attributes can be either required or optional depending on the data structure definition.

In the previous section we introduced an example data structure definition for the dataset described in Table 1. Component specifications of this data structure definition reference dimension properties and a measure property that represent the total general government expenditure (measure), the reference area (dimension), and the reference year (dimension). In the following example, we provide RDF representation of these properties.

```
@prefix interval:      <http://reference.data.gov.uk/def/intervals/> .
@prefix qb:            <http://purl.org/linked-data/cube#> .
@prefix rdf:           <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:          <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sdmx-concept:  <http://purl.org/linked-data/sdmx/2009/concept#> .
@prefix sdmx-dimension: <http://purl.org/linked-data/sdmx/2009/dimension#> .
@prefix sdmx-measure:  <http://purl.org/linked-data/sdmx/2009/measure#> .
@prefix xsd:           <http://www.w3.org/2001/XMLSchema#> .

@prefix ex:            <http://data.example.org/ontology/> .
@prefix ex-codelist:   <http://data.example.org/resource/codelist/> .
@prefix ex-dimension:  <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-measure:    <http://data.example.org/ontology/dsd/measure/> .

# Dimension properties
```

```
ex-dimension:refPeriod a rdf:Property, qb:DimensionProperty ;
  rdfs:label "reference period"@en ;
  rdfs:subPropertyOf sdmx-dimension:refPeriod ;
  rdfs:range interval:Interval ;
  qb:concept sdmx-concept:refPeriod .

ex-dimension:refArea a rdf:Property, qb:DimensionProperty ;
  rdfs:label "reference area"@en ;
  rdfs:subPropertyOf sdmx-dimension:refArea ;
  rdfs:range ex:GeopoliticalEntity ;
  qb:codeList ex-codelist:geo ;
  qb:concept sdmx-concept:refArea .

# Measure properties
ex-measure:total-general-government-expenditure a rdf:Property, qb:MeasureProperty ;
  rdfs:label "total general government expenditure"@en ;
  rdfs:subPropertyOf sdmx-measure:obsValue;
  rdfs:range xsd:decimal .
```

Example 3: Vocabulary for the data structure definition of the dataset described in Table 1

We use `interval:Interval` from the Interval ontology as the range for the reference period dimension property and `ex:GeopoliticalEntity` as the range for the reference area dimension property. There needs to be a code list for every dimension property. Definition of the geopolitical entity is provided in the code lists section. See http://reference.data.gov.uk/def/intervals for intervals.

All the dimensions and the measure are modelled as subproperties of more generic concepts specified by the SDMX standard. The reason for taking this approach is that we define specific ranges and associated code lists for these dimensions.

It is possible to have more than one measure per observation. See Table 2 for an example.[4]

Year	2013	2013	2014	2014
Reference area/measure	GDP at market prices (Current prices, euro per capita)	Real GDP growth rate - volume (Percentage change on previous year)	GDP at market prices (Current prices, euro per capita)	Real GDP growth rate - volume (Percentage change on previous year)
EU (28 countries)	26600	0,2	27400	1,4
EU (27 countries)	:	:	:	:
Euro area (changing composition)	29600	-0,3	30000	0,9
Euro area (19 countries)	29400	-0,3	29800	0,9
Euro area (18 countries)	29500	-0,3	30000	0,9
Euro area (17 countries)	:	:	:	:

Table 2: GDP at market prices and real GDP growth rate, source: excerpt from (Eurostat, 2015a; Eurostat, 2015c)

There are 2 measures present in Table 2: GDP at current market prices expressed as euro per capita and volume of the real GDP growth expressed as percentage change on previous year. Both measures are indexed by the same dimensions: reference area and year. Because

---

[4] Two datasets used: (Eurostat, 2015a; Eurostat, 2015c). Only data for EU28, EU27 and Euro areas and for years 2013 and 2014 presented, rest of the data omitted for the purposes of this example. Unavailable values are marked with ":". See the Eurostat website for detailed metadata.

the dimensions are the same for both of the measures, it would be possible to represent them both as measures of the same observation.

In the DCV there are two alternative approaches to observations with multiple measures (Cyganiak & Reynolds, 2014):

1. **Multi-measure observations** – if the multi-measure observations approach is used, all specified measures are attached to a single observation.
2. **Measure dimension** – in case of the measure dimension approach a single measure is always attached to an observation and an additional dimension `qb:measureType` is used to denote which particular measure is being conveyed by the observation. Note that DCV still requires all measures to be present in the dataset for a given combination of the original dimensions. I.e. if we had 3 measures per observation using the Multi-measure approach, we need to have 3 observations using the measure dimension approach and we cannot omit any of them.

In the following example we provide data structure definition of the dataset in Table 2 that applies the multi-measure observations approach.

```
@prefix qb:               <http://purl.org/linked-data/cube#> .
@prefix rdfs:             <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sdmx-attribute:   <http://purl.org/linked-data/sdmx/2009/attribute#> .
@prefix xsd:              <http://www.w3.org/2001/XMLSchema#> .

@prefix ex-dimension:     <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-dsd:           <http://data.example.org/resource/dsd/> .
@prefix ex-measure:       <http://data.example.org/ontology/dsd/measure/> .

ex-dsd:GDP-at-market-prices-and-real-GDP-growth-rate a qb:DataStructureDefinition ;
  rdfs:label "GDP at market prices and real GDP growth rate"@en ;
  # Dimensions
  qb:component [
      qb:dimension ex-dimension:refPeriod ;
      qb:order 1 ;
      rdfs:label "Dimension representing a year for which GDP at market prices and
real GDP growth rate are reported"@en
  ] ;
  qb:component [
      qb:dimension ex-dimension:refArea ;
      qb:order 2 ;
      rdfs:label "Dimension representing a state or group of states for which GDP at
market prices and real GDP growth rate are reported"@en
  ] ;
  # Measures
  qb:component [
      qb:measure ex-measure:GDP-at-market-prices ;
      qb:order 3 ;
      rdfs:label "Measure representing the GDP at market prices"@en
  ] ;
  qb:component [
      qb:measure ex-measure:real-GDP-growth-rate ;
      qb:order 4 ;
      rdfs:label "Measure representing the real GDP growth rate"@en
  ] ;
  # Attributes
  qb:component [
      qb:attribute sdmx-attribute:unitMeasure ;
      qb:componentRequired true ;
      qb:componentAttachment qb:MeasureProperty ;
      ] .
```

Example 4: Data structure definition of the dataset described in Table 2 (multi-measure observations)

In the following example we provide the RDF representation of the dataset described in Table 2 and instance data for years 2013 and 2014 for the EU28 area. We use the data structure definition introduced above that applies the multiple measure approach.

```
@prefix dcterms:            <http://purl.org/dc/terms/> .
@prefix qb:                 <http://purl.org/linked-data/cube#> .
@prefix rdfs:               <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sdmx-attribute:     <http://purl.org/linked-data/sdmx/2009/attribute#> .

@prefix ex-dataset:         <http://data.example.org/resource/dataset/> .
@prefix ex-dsd:             <http://data.example.org/resource/dsd/> .
@prefix ex-dimension:       <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-measure:         <http://data.example.org/ontology/dsd/measure/> .
@prefix ex-units-of-measure: <http://data.example.org/resource/codelist/units-of-
measure/> .

# Dataset

ex-dataset:GDP-at-market-prices-and-real-GDP-growth-rate a qb:DataSet ;
  rdfs:label "GDP at market prices and real GDP growth rate"@en ;
  rdfs:comment "GDP at market prices (current prices, euro per capita) and real GDP
growth rate (percentage change on previous year)."@en ;
  dcterms:publisher <http://openbudgets.eu> ;
  dcterms:source
<http://ec.europa.eu/eurostat/tgm/table.do?tab=table&init=1&language=en&pcode=tec000
01&plugin=1> ;
  dcterms:source
<http://ec.europa.eu/eurostat/tgm/table.do?tab=table&init=1&language=en&pcode=tec001
15&plugin=1> ;
  qb:structure ex-dsd:GDP-at-market-prices-and-real-GDP-growth-rate .

# Unit of measure attachment

ex-measure:GDP-at-market-prices sdmx-attribute:unitMeasure ex-units-of-measure:euro-
per-capita .

ex-measure:real-GDP-growth-rate sdmx-attribute:unitMeasure ex-units-of-
measure:percentage-change-on-previous-year .

# Example observations

<http://data.example.org/resource/observation/GDP-at-market-prices-and-real-GDP-
growth-rate/2013/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:GDP-at-market-prices-and-real-GDP-growth-rate ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2012> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  ex-measure:GDP-at-market-prices 26600 ;
  ex-measure:real-GDP-growth-rate 0.2 .

<http://data.example.org/resource/observation/GDP-at-market-prices-and-real-GDP-
growth-rate/2014/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:GDP-at-market-prices-and-real-GDP-growth-rate ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2013> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  ex-measure:GDP-at-market-prices 27400 ;
  ex-measure:real-GDP-growth-rate 1.4 .
```

Example 5: Example of instance data of the dataset described in Table 2 (multi-measure observations)

The example above shows that both measures (GDP at market prices and real GDP growth rate) are part of one observation. It also demonstrates a known limitation of this approach that makes it impossible to attach attributes to a single measured value. That is why units of measurement are attached to the measure properties instead. Impact of this limitation is that the attachment of the unit of measure would apply to any dataset using that measure property.

To demonstrate both of the possible approaches to handling datasets with multiple measures, we provide the following example where the measure dimensions approach is applied.

```
@prefix qb:              <http://purl.org/linked-data/cube#> .
@prefix rdfs:            <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sdmx-attribute:  <http://purl.org/linked-data/sdmx/2009/attribute#> .
@prefix xsd:             <http://www.w3.org/2001/XMLSchema#> .

@prefix ex-dimension:    <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-dsd:          <http://data.example.org/resource/dsd/> .
@prefix ex-measure:      <http://data.example.org/ontology/dsd/measure/> .

ex-dsd:GDP-at-market-prices-and-real-GDP-growth-rate a qb:DataStructureDefinition ;
  rdfs:label "GDP at market prices and real GDP growth rate"@en ;
  # Dimensions
  qb:component [
    qb:dimension ex-dimension:refPeriod ;
    qb:order 1 ;
    rdfs:label "Dimension representing a year for which GDP at market prices and
real GDP growth rate are reported"@en
  ] ;
  qb:component [
    qb:dimension ex-dimension:refArea ;
    qb:order 2 ;
    rdfs:label "Dimension representing a state or group of states for which GDP at
market prices and real GDP growth rate are reported"@en
  ] ;
  qb:component [
    qb:dimension qb:measureType ;
    qb:order 3 ;
    rdfs:label "Measure type"@en
  ] ;
  # Measures
  qb:component [
    qb:measure ex-measure:GDP-at-market-prices ;
    qb:order 4 ;
    rdfs:label "Measure representing the GDP at market prices"@en
  ] ;
  qb:component [
    qb:measure ex-measure:real-GDP-growth-rate ;
    qb:order 5 ;
    rdfs:label "Measure representing the real GDP growth rate"@en
  ] ;
  # Attributes
  qb:component [
    qb:attribute sdmx-attribute:unitMeasure ;
    qb:componentRequired true
  ] .
```

Example 6: Data structure definition of the dataset described in Table 2 (measure dimension)

In the following example we provide the RDF representation of the dataset described in Table 2 and instance data for years 2013 and 2014 for the EU28 area. We use the data structure definition introduced above that applies the measure dimension approach.

```
@prefix dcterms:            <http://purl.org/dc/terms/> .
@prefix qb:                 <http://purl.org/linked-data/cube#> .
@prefix rdfs:               <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sdmx-attribute:     <http://purl.org/linked-data/sdmx/2009/attribute#> .

@prefix ex-dataset:         <http://data.example.org/resource/dataset/> .
@prefix ex-dimension:       <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-dsd:             <http://data.example.org/resource/dsd/> .
@prefix ex-measure:         <http://data.example.org/ontology/dsd/measure/> .
@prefix ex-units-of-measure: <http://data.example.org/resource/codelist/units-of-
measure/> .

# Dataset
```

```
ex-dataset:GDP-at-market-prices-and-real-GDP-growth-rate a qb:DataSet ;
  rdfs:label "GDP at market prices and real GDP growth rate"@en ;
  rdfs:comment "GDP at market prices (current prices, euro per capita) and real GDP
growth rate (percentage change on previous year)."@en ;
  dcterms:publisher <http://openbudgets.eu> ;
  dcterms:source
<http://ec.europa.eu/eurostat/tgm/table.do?tab=table&init=1&language=en&pcode=tec000
01&plugin=1> ,
http://ec.europa.eu/eurostat/tgm/table.do?tab=table&init=1&language=en&pcode=tec0011
5&plugin=1> ;
  qb:structure ex-dsd:GDP-at-market-prices-and-real-GDP-growth-rate .

# Example observations

<http://data.example.org/resource/observation/GDP-at-market-prices-and-real-GDP-
growth-rate/GDP-at-market-prices/2013/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:GDP-at-market-prices-and-real-GDP-growth-rate ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2012> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  qb:measureType ex-measure:GDP-at-market-prices;
  sdmx-attribute:unitMeasure ex-units-of-measure:euro-per-capita ;
  ex-measure:GDP-at-market-prices 26600 .

<http://data.example.org/resource/observation/GDP-at-market-prices-and-real-GDP-
growth-rate/real-GDP-growth-rate/2013/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:GDP-at-market-prices-and-real-GDP-growth-rate ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2012> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  qb:measureType ex-measure:real-GDP-growth-rate ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percentage-change-on-previous-year
;
  ex-measure:real-GDP-growth-rate 0.2 .

<http://data.example.org/resource/observation/GDP-at-market-prices-and-real-GDP-
growth-rate/GDP-at-market-prices/2014/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:GDP-at-market-prices-and-real-GDP-growth-rate ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2013> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  qb:measureType ex-measure:GDP-at-market-prices;
  sdmx-attribute:unitMeasure ex-units-of-measure:euro-per-capita ;
  ex-measure:GDP-at-market-prices 27400 .

<http://data.example.org/resource/observation/GDP-at-market-prices-and-real-GDP-
growth-rate/real-GDP-growth-rate/2014/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:GDP-at-market-prices-and-real-GDP-growth-rate ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2013> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  qb:measureType ex-measure:real-GDP-growth-rate ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percentage-change-on-previous-year
;
  ex-measure:real-GDP-growth-rate 1.4 .
```

Example 7: Example of instance data of the dataset described in Table 2 (measure dimension)

As indicated above, when the measure type approach is applied there is always only one measure per observation. Due to this feature it is possible to denote the unit of the measured value at the observation level which allows measure properties to be reused across multiple datasets with different units of measurement for the same measure type. In our example GDP in market prices is expressed in euros per capita. However it would be also possible to express the GDP in market prices in millions of euro.[5]

---

[5] See the dataset (Eurostat, 2015a).

2.4 Code lists

Possible values of dimensions are limited to items of the used code lists. Code list can be defined as *"a predefined list from which some statistical coded concepts take their values."*[6] It is recommended to represent code lists as SKOS concept schemes,[7] however the DCV provides an alternative approach to definition of code lists via `qb:HierarchicalCodeList` – see (Cyganiak & Reynolds, 2014) for details. Existing code lists that might be used in budgetary and spending datasets are analysed in Deliverable D1.6 (Ioannidis et al., 2015).

In the above examples reference area is one of the dimensions. Various groups of European countries represent values of this dimension. We provide an excerpt of the code list of geopolitical entities as an example of a code list.

```
@prefix nuts:        <http://ec.europa.eu/eurostat/ramon/rdfdata/nuts2008/> .
@prefix rdfs:        <http://www.w3.org/2000/01/rdf-schema#> .
@prefix schema:      <http://schema.org/> .
@prefix skos:        <http://www.w3.org/2004/02/skos/core#> .

@prefix ex:          <http://data.example.org/ontology/> .
@prefix ex-codelist: <http://data.example.org/resource/codelist/> .
@prefix ex-geo:      <http://data.example.org/resource/codelist/geo/> .

# Geopolitical entity
ex:GeopoliticalEntity a rdfs:Class, s:Place ;
  rdfs:label "Geopolitical entity"@en ;
  rdfs:isDefinedBy <http://data.example.org/ontology> .

ex-codelist:geo a skos:ConceptScheme ;
  rdfs:label "Code list of geopolitical entities"@en .

ex-geo:EU28 a skos:Concept, ex:GeopoliticalEntity, schema:Place ;
  skos:prefLabel "European Union (28 countries)"@en ;
  skos:definition "This aggregate covers following countries: BE, BG, CZ, DK, DE,
EE, IE, EL, ES, FR, HR, IT, CY, LV, LT, LU, HU, MT, NL, AT, PL, PT, RO, SI, SK, FI,
SE, UK"@en ;
  skos:notation "EU28" ;
  skos:inScheme ex-codelist:geo ;
  skos:narrower nuts:AT, nuts:BE, nuts:BG, nuts:CY, nuts:CZ, nuts:DE, nuts:DK,
nuts:EE, nuts:ES, nuts:FI, nuts:FR, nuts:GR, nuts:HR, nuts:HU, nuts:IE, nuts:IT,
nuts:LT, nuts:LU, nuts:LV, nuts:MT, nuts:NL, nuts:PL, nuts:PT, nuts:RO, nuts:SE,
nuts:SI, nuts:SK, nuts:UK ;
  rdfs:seeAlso
<http://ec.europa.eu/eurostat/ramon/nomenclatures/index.cfm?TargetUrl=DSP_NOM_DTL_VI
EW&StrNom=CL_GEO&StrLanguageCode=EN&IntPcKey=34617515&IntKey=34617522&StrLayoutCode=
HIERARCHIC&IntCurrentPage=1> .

ex-geo:EA19 a skos:Concept, ex:GeopoliticalEntity, s:Place ;
  skos:prefLabel "Euro area (19 countries)"@en ;
  skos:definition "This aggregate covers following countries since 2015: BE, DE, EE,
IE, EL, ES, FR, IT, CY, LV, LT, LU, MT, NL, AT, PT, SI, SK, FI"@en ;
  skos:notation "EA19" ;
  skos:inScheme ex-codelist:geo ;
  skos:narrower nuts:AT, nuts:BE, nuts:CY, nuts:DE, nuts:EE, nuts:ES, nuts:FI,
nuts:FR, nuts:GR, nuts:IE, nuts:IT, nuts:LT, nuts:LU, nuts:LV, nuts:MT, nuts:NL,
nuts:PT, nuts:SK, nuts:SI ;
  rdfs:seeAlso
<http://ec.europa.eu/eurostat/ramon/nomenclatures/index.cfm?TargetUrl=DSP_NOM_DTL_VI
EW&StrNom=CL_GEO&StrLanguageCode=EN&IntPcKey=34617515&IntKey=34617613&StrLayoutCode=
HIERARCHIC&IntCurrentPage=1> .
```

Example 8: Geopolitical entity code list

---

[6] https://stats.oecd.org/glossary/detail.asp?ID=3371

[7] See http://www.w3.org/TR/skos-reference/

See (Miles & Bechhofer, 2009) for more detailed reference of the Simple Knowledge Organization System (SKOS).

2.5 Slices and SliceKeys

The DCV allows a set of observations with one or more dimensions fixed to be grouped into a slice and associated with a slice key. This allows the group of observations to be referenced and provided with additional metadata. Using the data in Table 1 as an example it is possible to fix the area dimension and group all observations for EU (28 countries), forming a time-series slice for this reference area (the only free dimension is the time dimension).

In order to be able to use slices it is necessary to define data structure of the required slices and associate them with the respective slice keys. We illustrate this step with the following example that builds upon the dataset described in Table 1.

```
@prefix qb:               <http://purl.org/linked-data/cube#> .
@prefix rdfs:             <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sdmx-attribute:   <http://purl.org/linked-data/sdmx/2009/attribute#> .
@prefix xsd:              <http://www.w3.org/2001/XMLSchema#> .

@prefix ex-dsd:           <http://data.example.org/resource/dsd/> .
@prefix ex-dimension:     <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-measure:       <http://data.example.org/ontology/dsd/measure/> .
@prefix ex-slicekey:      <http://data.example.org/resource/slicekey/> .

# Slice key definition

ex-slicekey:slice-by-ref-period a qb:SliceKey ;
  rdfs:label "slice by reference period"@en ;
  rdfs:comment "Slice by reference period, fixes the reference area forming a time
series."@en ;
  qb:componentProperty ex-dimension:refArea .

# Data structure definition

ex-dsd:total-general-government-expenditure a qb:DataStructureDefinition ;
  rdfs:label "Total general government expenditure"@en ;
  # Dimensions
  qb:component [
      qb:dimension ex-dimension:refPeriod ;
      qb:order 1 ;
      rdfs:label "Dimension representing a year for which total general government
expenditure is reported"@en
  ] ;
  qb:component [
      qb:dimension ex-dimension:refArea ;
      qb:order 2 ;
      rdfs:label "Dimension representing a state or group of states for which total
general government expenditure is reported"@en
  ] ;
  # Measure
  qb:component [
      qb:measure ex-measure:total-general-government-expenditure ;
      rdfs:label "Measure representing the total general government expenditure"@en
  ] ;
  # Attributes
  qb:component [
      qb:attribute sdmx-attribute:unitMeasure ;
      qb:componentRequired true
  ] ;
  qb:sliceKey ex-slicekey:slice-by-ref-period .
```

Example 9: Slice key and updated data structure definition of the dataset described in Table 1

We use the same data structure definition as in Example 1, we only update it with a link to the defined slice key using the `qb:sliceKey` property. Slice in the example above groups together observations with the same reference area. Based on the data described in Table 1

the second dimension in this example is the reference period which remains the free dimension, i.e. the slice should contain all yearly observations for a specific area. Example of the RDF representation of the instance data is provided below - slice for the EU28 area (we provide examples of only three observations to keep the the example to a reasonable length).

```
@prefix dcterms:            <http://purl.org/dc/terms/> .
@prefix qb:                 <http://purl.org/linked-data/cube#> .
@prefix rdfs:               <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sdmx-attribute:     <http://purl.org/linked-data/sdmx/2009/attribute#> .

@prefix ex-dataset:         <http://data.example.org/resource/dataset/> .
@prefix ex-dsd:             <http://data.example.org/resource/dsd/> .
@prefix ex-dimension:       <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-measure:         <http://data.example.org/ontology/dsd/measure/> .
@prefix ex-slice:           <http://data.example.org/resource/slice/> .
@prefix ex-slicekey:        <http://data.example.org/resource/slicekey/> .
@prefix ex-units-of-measure: <http://data.example.org/resource/codelist/units-of-
measure/> .

# Dataset

ex-dataset:total-general-government-expenditure a qb:DataSet ;
  rdfs:label "Total general government expenditure"@en ;
  rdfs:comment "Total general government expenditure expressed as % of GDP."@en ;
  dcterms:publisher <http://openbudgets.eu> ;
  dcterms:source
<http://ec.europa.eu/eurostat/tgm/table.do?tab=table&plugin=1&language=en&pcode=tec0
0023> ;
  qb:structure ex-dsd:total-general-government-expenditure ;
  qb:slice ex-slice:EU28 .

# Slice

ex-slice:EU28 a qb:Slice ;
  rdfs:label "Time series slice for the EU28 area"@en ;
  qb:sliceStructure ex-slicekey:slice-by-ref-period ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  qb:observation <http://data.example.org/resource/observation/total-general-
government-expenditure/2012/EU28>,
              <http://data.example.org/resource/observation/total-general-
government-expenditure/2013/EU28>,
              <http://data.example.org/resource/observation/total-general-
government-expenditure/2014/EU28>
  .

# Example observations

<http://data.example.org/resource/observation/total-general-government-
expenditure/2012/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2012> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 49 .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2013/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2013> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 48.6 .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2014/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2014> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
```

```
    sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
    ex-measure:total-general-government-expenditure 48.2 .
```

Example 10: Example of instance data using a defined slice over the dataset described in Table 1

In Example 10 data about the reference area is provided at both the observation level as well as at the slice level. This allows working with the observations independently on the defined slice. However, slices can also be used to reduce verbosity of a dataset, as the values of the fixed components (dimensions, attributes) can be specified only once at the slice level (in combination with the `qb:componentAttachment qb:Slice` property and value). This can save a number of triples. On the other hand, usage of components attached to slices complicates the data usage, as some of the dimensions for a given observation can be attached to the observation directly and some of them can be attached to the slice itself. We illustrate this with the following examples.

The following example shows a modified data structure definition used in Example 9, where the attachment level for the reference area dimension is changed to the slice level.

```
@prefix qb:              <http://purl.org/linked-data/cube#> .
@prefix rdfs:            <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sdmx-attribute:  <http://purl.org/linked-data/sdmx/2009/attribute#> .
@prefix xsd:             <http://www.w3.org/2001/XMLSchema#> .

@prefix ex-dimension:    <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-dsd:          <http://data.example.org/resource/dsd/> .
@prefix ex-measure:      <http://data.example.org/ontology/dsd/measure/> .
@prefix ex-slicekey:     <http://data.example.org/resource/slicekey/> .

# Slice key definition
ex-slicekey:slice-by-ref-period a qb:SliceKey ;
  rdfs:label "slice by reference period"@en ;
  rdfs:comment "Slice grouping observations of the same region forming a time
series."@en ;
  qb:componentProperty ex-dimension:refArea .

# Data structure definition
ex-dsd:total-general-government-expenditure a qb:DataStructureDefinition ;
  rdfs:label "Total general government expenditure"@en ;
  # Dimensions
  qb:component [
    qb:dimension ex-dimension:refPeriod ;
    qb:order 1 ;
    rdfs:label "Dimension representing a year for which total general government
expenditure is reported"@en
  ] ;
  qb:component [
    qb:componentAttachment qb:Slice ;
    qb:dimension ex-dimension:refArea ;
    qb:order 2 ;
    rdfs:label "Dimension representing a state or group of states for which total
general government expenditure is reported"@en
  ] ;
  # Measure
  qb:component [
    qb:measure ex-measure:total-general-government-expenditure ;
    rdfs:label "Measure representing the total general government expenditure"@en
  ] ;
  # Attributes
  qb:component [
    qb:attribute sdmx-attribute:unitMeasure ;
    qb:componentRequired true
  ] ;
  qb:sliceKey ex-slicekey:slice-by-ref-period .
```

Example 11: Slice key and data structure definition of the dataset described in Table 1 with changed reference area dimension attachment level

Changing the attachment level of the reference area dimension would allow to provide this dimension only at the slice level, as shown in the following example.

```
@prefix dcterms:           <http://purl.org/dc/terms/> .
@prefix qb:                <http://purl.org/linked-data/cube#> .
@prefix rdfs:              <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sdmx-attribute:    <http://purl.org/linked-data/sdmx/2009/attribute#> .

@prefix ex-dataset:        <http://data.example.org/resource/dataset/> .
@prefix ex-dimension:      <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-dsd:            <http://data.example.org/resource/dsd/> .
@prefix ex-measure:        <http://data.example.org/ontology/dsd/measure/> .
@prefix ex-slice:          <http://data.example.org/resource/slice/> .
@prefix ex-slicekey:       <http://data.example.org/resource/slicekey/> .
@prefix ex-units-of-measure: <http://data.example.org/resource/codelist/units-of-
measure/> .

# Dataset

ex-dataset:total-general-government-expenditure a qb:DataSet ;
  rdfs:label "Total general government expenditure"@en ;
  rdfs:comment "Total general government expenditure expressed as % of GDP."@en ;
  dcterms:publisher <http://openbudgets.eu> ;
  dcterms:source
<http://ec.europa.eu/eurostat/tgm/table.do?tab=table&plugin=1&language=en&pcode=tec0
0023> ;
  qb:structure ex-dsd:total-general-government-expenditure ;
  qb:slice ex-slice:EU28 .

# Slice

ex-slice:EU28 a qb:Slice ;
  rdfs:label "Time series slice for the EU28 area"@en ;
  qb:sliceStructure ex-slicekey:slice-by-ref-period ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  qb:observation  <http://data.example.org/resource/observation/total-general-
government-expenditure/2012/EU28>,
  <http://data.example.org/resource/observation/total-general-government-
expenditure/2013/EU28>,
  <http://data.example.org/resource/observation/total-general-government-
expenditure/2014/EU28> .

# Example observations

<http://data.example.org/resource/observation/total-general-government-
expenditure/2012/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2012> ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 49 .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2013/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2013> ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 48.6 .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2014/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2014> ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 48.2 .
```

Example 12: Example of instance data using a defined slice over the dataset described in Table 1 with slice level attachment of the reference period dimension

3 OpenBudgets.eu RDF prefixes

For OpenBudgets.eu we will use the following RDF prefixes based on a similar approach for SDMX:[8]

- `obeu: <http://data.openbudgets.eu/ontology/>`
- `obeu-dsd: <http://data.openbudgets.eu/resource/dsd/>`
- `obeu-dimension:`
  `<http://data.openbudgets.eu/ontology/dsd/dimension/>`
- `obeu-measure:`
  `<http://data.openbudgets.eu/ontology/dsd/measure/>`
- `obeu-attribute:`
  `<http://data.openbudgets.eu/ontology/dsd/attribute/>`
- `obeu-codelist: <http://data.openbudgets.eu/resource/codelist/>`
- `obeu-metadata: <http://data.openbudgets.eu/ontology/metadata/>`
- `obeu-currency:`
  `<http://data.openbudgets.eu/resource/codelist/currency/>`

4 IRI patterns

Internationalized Resource Identifiers (IRIs) (Duerst & Suignard, 2004) should be treated as opaque,[9] but following consistent IRI patterns improves human understanding of data, which is especially important for application developers and data analysts. Moreover, when source data identifiers are used in IRI patterns, IRIs can be programmatically constructed by simple string concatenation. In this way, it is straightforward to create links to external datasets. However, nothing should be inferred from the IRI's constituent parts and IRIs should be treated as meaningless identifiers. Note that we use IRIs instead of URIs, so that international character sets are supported as valid parts of identifiers.

When designing IRI patterns, start by choosing a base namespace on a domain you own. Consider using a dedicated subdomain for the namespace of IRIs in order to separate them from the rest of your domain. In the following example, we will use `http://data.openbudgets.eu/` as our base namespace. All your IRIs will start with this namespace. The IRIs in this namespace can be partitioned into a logical space by the types of resources they identify. First, we propose to distinguish the IRIs of the terminological entities from the data structure definition by appending `ontology/` to the base namespace and append `resource/` for the resources instantiating the terminological entities. Subsequently, we recommend to append a label of the type of the identified resource, such as `codelist/`. You can structure the types of the resource further, such as first adding `dsd/` for a data structure definition followed by `measure/` for an IRI identifying measure property. The last part of an IRI must uniquely identify the resource within its namespace. We recommend to reuse identifiers from the source data, such as codes of code list concepts. Make sure the characters of these identifiers are allowed in IRIs by converting them into URI slugs (see the chapter on identifier patterns in Dodds, Davis, 2012). If such identifiers are unavailable, use a randomly generated UUID that guarantees uniqueness. We recommend avoiding auto-incremented integer identifiers, since they are too brittle and in RDF they do not provide the usual benefit of fast index access.

---

[8] https://github.com/UKGovLD/publishing-statistical-data/tree/master/specs/src/main/vocab

[9] http://www.w3.org/DesignIssues/Axioms.html#opaque

For example, if data describing the budget of the EU for the year 2014 was published directly by its maintainers using the data model of OpenBudgets.eu, the base namespace can be defined as `http://open-data.europa.eu`, which is the URL of the European Union Open Data Portal. In order to distinguish elements of the data model and data described with the model, we can append `ontology/` or `resource/` respectively to the base namespace.

Regarding word case, the path parts of IRIs use `kebab-case`. The IRI's local name (ID) that comes as its last part should use `camelCase` for properties or classes and `kebab-case` for instances (instances). In addition, local names of properties start with a lowercase letter and local names of classes start with an uppercase letter. Local names of instances start with a lowercase letter. An exception to this rule should be applied when an identifier from the source dataset is used as a local name. For example, suppose we create IRI for code list concepts using their codes as local names. In that case we recommend using the identifier literally, subjected only to IRI-encoding. For example, currency code "EUR" should be kept in uppercase. This way you can avoid potential IRI collisions caused by identifier normalization.

Examples of property and entity IRIs:

- Core OpenBudgets.eu dimension property (fiscalPeriod): `http://data.openbudgets.eu/ontology/dsd/dimension/fiscalPeriod`
- Core OpenBudgets.eu measure property (amount): `http://data.openbudgets.eu/ontology/dsd/measure/amount`
- Core OpenBudgets.eu attribute property (taxesIncluded): `http://data.openbudgets.eu/ontology/dsd/attribute/taxesIncluded`
- Core OpenBudgets.eu codelist (operationCharacter): `http://data.openbudgets.eu/resource/codelist/operation-character`
- Core OpenBudgets.eu codelist item (Expenditure from operationCharacter codelist): `http://data.openbudgets.eu/resource/codelist/operation-character/expenditure`
- Non OpenBudgets.eu dimension property (catpol from the EU Budget dataset): `http://example.openbudgets.eu/ontology/dsd/eu-budget-2014/dimension/catpol`
- Non OpenBudgets.eu attribute property (reserve from EU Budget dataset): `http://example.openbudgets.eu/ontology/dsd/attribute/reserve`
- Non OpenBudgets.eu codelist (EU Budget dataset operation character codelist): `http://example.openbudgets.eu/resource/eu-budget-2014/codelist/operation-character`
- Non OpenBudgets.eu codelist item (Commitment from EU Budget dataset operation character codelist): `http://example.openbudgets.eu/resource/eu-budget-2014/codelist/operation-character/commitment`

As already illustrated, there is a special IRI pattern for observations of a data cube. The IRI starts with the domain as usual, followed by `/resource` and `/observation` and a URI slug of the name of the data cube. Observations of a data cube are distinguished by values of dimensions. These values are taken from code lists where each code list item (usually `skos:Concept`) should also have a machine readable code (`skos:notation`). These codes can be then used in the observation IRI, as they should guarantee uniqueness of the IRI and provide some insight to the nature of the observation. An example is:

```
<http://data.example.org/resource/observation/total-general-government-
expenditure/2013/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2013> ;
  ex-dimension:refArea <http://data.example.org/resource/codelist/geo/EU28> ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 48.6 .
```

Here, `total-general-government-expenditure` is the name of the data cube and `2013` and `EU28` are the dimension values (codes of codelist items) identifying the observation.

5 Budget data modelling guide

Having introduced the core underpinnings of the DCV we will move on to a concrete application of the vocabulary into the domain of public finance. Data model of the OpenBudgets.eu is a specific application of the DCV designed to represent the core concepts of this domain. We will walk through a sequence of steps in modelling fiscal datasets using the proposed data model.

5.1 Data identification

The first step in modelling budget data is to identify what kind of dataset you have. The OpenBudgets.eu data model recognizes 2 principal kinds of datasets: budget and spending. It may be difficult to tell them apart, in part because budget data may contain aggregated expenditure for previous fiscal periods. This is why we provide several checkpoints to help distinguish the nature of fiscal datasets.

Budget datasets:

- Budget data is a plan for a future fiscal period, which is aggregated by classifications, i.e. rarely include individual transactions
- Budget data does not contain specific partners who receive or pay the expenditures.

Spending datasets:

- Spending data contains records of realized financial transactions, i.e. really collected revenue and paid expenditure are shown.
- Spending data may contain specific partners who received or paid the reported expenditures.

Alternatively, source datasets can be combined for presentation purposes. For example, budgeted appropriations may be shown along with disbursed subsidies. In this case, it is advised to split the source dataset into multiple logical datasets. If needed, joins can be made over the datasets in queries to get at the same view as is provided by the source dataset.

5.2 Data interpretation

Prior to formalizing the data model we need to understand the data we have. Unlike RDF, most data formats are not self-descriptive, so that the data itself is often insufficient for deriving a correct interpretation. Therefore, it is necessary to have access to out-of-band information explaining the data. This information is typically embedded in schemata, documentation or metadata. For example, there can be a document explaining what column names in a dataset refer to. Alternatively, one can have a structured metadata descriptor of a dataset, such as the JSON descriptor format used by Fiscal Data Package. Without documentation, basing the interpretation of data on the column labels only should be used only when you have a strong confidence that you are able to interpret them correctly.

Understanding schemata is tied to the understanding of the language they are written in. This is often a natural language, since the schema is embodied only in column names. In other cases, a formal schema language is used, such as XML Schema. Understanding the language of data is the minimum prerequisite for understanding the data. First, users need to understand the natural language used in the descriptions of data. In case of fiscal data this aim often entails understanding the domain-specific jargon and terminology. If terminological confusion arises, we recommend to consult domain experts to help clarify the intended meaning of the employed terms. The second step is to understand the schema of the dataset at hand. Dataset schema can be explicitly formalized using a schema language or be left implicit, such as implied relations between columns in a table. This understanding is subsequently projected into the data structure definition of the dataset.

5.3 Mapping source data structure to the target (DCV) data model structure

Let us demonstrate the process of mapping the source data structure to the target OpenBudgets.eu DCV data structure definition using an example of a CSV file. A CSV file is composed of columns, some of which will become dimensions and others will become measures. Generally speaking, dimensions are usually columns representing classifications, time, area, etc., while measures are usually the numeric values like monetary amounts, numbers of persons etc. In addition, there are attributes like currency, which are often not specified in the source data, or which are specified only in documentation of the source data and need to be added during data transformation. Some CSVs can be more complicated, especially when they represent a direct transcript of a table originally formatted for visualizations, such as Table 1, where a more natural way of representing the same data would use 3 columns: reference area, time period (dimensions), and observed value (measure), instead of encoding the values of the time dimension in column names.

Not every column from the source data needs to be mapped to a component property. Some columns specify attributes of entities, which are already related to the described observation by another component property. For example, the source dataset may contain project names. Projects are already related to the described observations via `obeu-dimension:project`, so that their names can be represented as values of `foaf:name` property of the linked entities.

5.3.1 Reusing OpenBudgets.eu core component properties

When the dimension, measure and attribute roles are identified in the source dataset, we should look in the list of OpenBudgets.eu core component properties for corresponding ones to reuse. See the reference section below for a comprehensive overview of the component properties defined in the data model of OpenBudgets.eu. Typically, for datasets that OpenBudgets.eu is mainly focused on, there will be a monetary amount measure, for which we have the `obeu-measure:amount` measure property and also often there will be measurements in different time periods, for which we can reuse the `obeu-dimension:fiscalPeriod` property in our new data structure definition. The remaining parts of data structure definitions typically vary among datasets and may require dataset-specific extensions of the OpenBudgets.eu data model.

5.3.2 Extending the core data model

If the core data model of OpenBudgets.eu does not suffice for your modelling needs, you can extend it. The primary way of extending the data model is to derive a more specific component property from a more generic core component property. With a specific component property the representation of your dataset can be more descriptive. For example, the core data model contains the component property `obeu-dimension:fiscalPeriod` to represent time intervals associated with fiscal data:

```
obeu-dimension:fiscalPeriod a rdf:Property, qb:DimensionProperty, qb:CodedProperty
;
  rdfs:label "fiscal period"@en ;
  rdfs:comment "The period of time reflected in financial statements."@en ;
  rdfs:subPropertyOf sdmx-dimension:refPeriod ;
  rdfs:range time:Interval ;
  qb:concept sdmx-concept:refPeriod .
```

In order to derive a more specific component properties use the `rdfs:subPropertyOf` property from RDF Schema (Brickley, Guha, 2014) to link the specific property to its parent and more generic property. In this way, tools that understand the core data model can treat data using the specific property as if it used the core property, from which the specific one is derived. Each derived component property should be described well enough to be able to distinguish it clearly from its parent component property. Property's description should include a label and a definition at least. Additionally, each property can link to a concept it represents

via the `qb:concept` property. For example, a subproperty can link a narrower concept of the concept linked by its parent property.

The time intervals used in budget data often last for a year, which is why the core data model also includes the `obeu-dimension:fiscalYear` component property as a sub-property of `obeu-dimension:fiscalPeriod`:

```
obeu-dimension:fiscalYear a rdf:Property, qb:DimensionProperty, qb:CodedProperty ;
  rdfs:label "fiscal year"@en ;
  rdfs:comment "The year reflected in financial statements."@en ;
  rdfs:subPropertyOf obeu-dimension:fiscalPeriod ;
  rdfs:range interval:Year ;
  qb:concept sdmx-concept:refPeriod .
```

Similarly, component properties for other sub-intervals may be created, such as for a quarter of a year.

An important part of defining a component property is specifying its code list. Code lists enumerate the values that are allowed to be used with a given component property. All dimension properties are coded, that is, there is a code list restricting the range of their values. Code lists can be optionally defined for attribute properties as well. In the DCV you associate a code list with a component property using the `qb:codeList` property that links the IRI of the code list. If you derive a coded component property, it would typically define a different code list to its parent property. However, this code list may include concepts from the parent property's code list. You can include external concepts into your code list by linking them to the code list IRI via the `skos:inScheme` property. This way, you can directly reuse code list concepts instead of duplicating them. Code lists can be extended in a similar fashion as component properties. You can create a mode specific code list concept and link it to its parent concept using the `skos:broader` property. Other semantic relations defined by SKOS[10], such as `skos:related`, can be used as well.

An example use of the described code list extension can be seen in Appendix: Codelist extension example. For the purpose of modelling the European Union budget dataset we extended the `obeu-codelist:operation-character` code list enumerating the characters of operations for which budget is allocated. For the same purpose we created the `eu-dimension:operationCharacter` subproperty. The extended code list directly reuses the top concepts of the `obeu-codelist:operation-character` code list: `obeu-operation:expenditure` and `obeu-operation:revenue`. It defines 2 additional concepts that are narrower to the concept of `obeu-operation:expenditure`: `eu-operation:commitment` and `eu-operation:payment`. These concepts are specific for the budget of the European Union.

5.3.3 Composing a data structure definition

Now that we are familiar with our source data and we have the necessary dimension, measure and attribute properties ready (either reused from OpenBudgets.eu core properties or newly defined), it is time to compose the data structure definition (DSD).[11] DSD specifies mainly the logical structure (e.g., what dimensions are used) of a dataset, but can also contain usage hints and optimisations (e.g., component ordering and component attachment). Understanding of the dataset's structure should be captured in a DSD. Let us demonstrate the composition of a DSD out of component properties on an example of the budget of the European Union:

```
<http://example.openbudgets.eu/ontology/dsd/eu-budget-2014> a
qb:DataStructureDefinition ;
  rdfs:label "Data structure definition for the budget of the European Union of the
year 2014"@en ;
```

---

[10] http://www.w3.org/TR/skos-reference/#semantic-relations

[11] http://www.w3.org/TR/vocab-data-cube/#dsd

```
qb:component [ qb:dimension obeu-dimension:budgetaryUnit ;
               qb:componentAttachment qb:DataSet ],
             [ qb:dimension obeu-dimension:budgetPhase ],
             [ qb:dimension eu-dimension:operationCharacter ],
             [ qb:dimension obeu-dimension:fiscalYear ],
             [ qb:dimension eu-dimension:budgetNomenclature ],
             [ qb:dimension eu-dimension:catpol ],

             [ qb:attribute obeu-attribute:currency ;
               qb:componentRequired true ;
               qb:componentAttachment qb:DataSet ],
             [ qb:attribute eu-attribute:reserve ;
               qb:componentRequired false ],

             [ qb:measure obeu-measure:amount ] .
```

Here, the DSD is composed of 3 reused dimensions (`obeu-dimension:budgetaryUnit`, `obeu-dimension:budgetPhase`, `obeu-dimension:fiscalYear`), 3 newly defined dimensions (`eu-dimension:operationCharacter`, `eu-dimension:budgetNomenclature`, `eu-dimension:catpol`), 1 reused attribute (`obeu-attribute:currency`), 1 newly defined attribute (`eu-attribute:reserve`), and 1 reused measure (`obeu-measure:amount`). The `obeu-dimension:budgetaryUnit` dimension has the `qb:componentAttachment` property set to `qb:DataSet`. This is because its value will be the European Union for each observation in the dataset and therefore it is not necessary to specify it for each observation separately. The same goes for the `obeu-attribute:currency` attribute which, in addition, has the `qb:componentRequired` property set to `true`, because every dataset in OpenBudgets.eu should have the currency specified. Not every observation in the EU budget dataset has to have a `eu-attribute:reserve` specified though, and therefore this attribute is not required to be specified for each observation.

Once the DSD is set, the thing left to do is the actual transformation of the source data to the observations in RDF which form the target data cube.

6 Modelling patterns

Having described the core mechanisms of building DSDs, we continue with a description of more high-level data modelling patterns. Following these patterns influences design of DSDs.

6.1 Lossless mapping

We recommend to attempt a lossless data conversion when mapping source data to RDF. Even when the source dataset contains measures that can be derived from other measures, it is better to preserve them in the RDF mirror of this dataset. Recomputing measures may be complicated in case several data points need to be used as input or the result of computation may be skewed by rounding error. By preserving the source data you preserve the authoritative values present in it.

6.2 Multi-currency datasets

For datasets that capture financial amounts in multiple currencies we recommend using both the `obeu-dimension:currency` dimension and the `obeu-attribute:currency` attribute. The currency dimension distinguishes between observations in different currencies such as the amount in euros (EUR) and the amount in Czech crowns (CZK), while the attribute specifies the currency for each observation consistent with single currency datasets, which improves consistency across datasets.

As examples of observations of a multi-currency dataset we picked the EU fishing subsidies fund 2007-2013 for the Czech Republic which indicates amounts both in EUR and CZK. Note that the measure property and the `qb:measureType` dimensions are the same and the only thing distinguishing between the two observations is the value of the currency dimension. The measure `eu-measure:amountCZ` indicates an amount paid by the Czech Republic, i.e. "CZ" in the identifier of the measure does not denote the currency. The currency attribute is then provided to interpret the value in the same way as that in the single currency case:

```
<http://data.openbudgets.eu/resource/observation/eu-fishing-subsidies-CS-2007-
2013/expenditure/Ministry_of_Agriculture_(Czech_Republic)/2.1a/CZ25170538/EUR/amoun
tCZ> a qb:Observation ;
  obeu-dimension:currency obeu-currency:EUR ;
  obeu-attribute:currency obeu-currency:EUR ;
  qb:measureType eu-measure:amountCZ ;
  eu-measure:amountCZ 13829.87 ;
  qb:dataSet <http://data.openbudgets.eu/resource/dataset/eu-fishing-subsidies-CS-
2007-2013> .

<http://data.openbudgets.eu/resource/observation/eu-fishing-subsidies-CS-2007-
2013/expenditure/Ministry_of_Agriculture_(Czech_Republic)/2.1a/CZ25170538/CZK/amoun
tCZ> a qb:Observation ;
  obeu-dimension:currency obeu-currency:CZK ;
  obeu-attribute:currency obeu-currency:CZK ;
  qb:measureType eu-measure:amountCZ ;
  eu-measure:amountCZ 345000.0 ;
  qb:dataSet <http://data.openbudgets.eu/resource/dataset/eu-fishing-subsidies-CS-
2007-2013> .
```

6.3 Data normalization

There are 2 key ways to normalize DCV data cubes.

According to DCV, a data cube is called normalized, if all its components are attached at the level of `qb:Observation`.[12] This is not always the case as DCV also supports other types of component attachment, i.e. observations, slices, measure properties or the dataset entity. One normalization way is therefore to reattach values of all components that do not have `qb:componentAttachment` set to observations (`qb:Observation`), which simplifies querying the data, while increasing data redundancy. We illustrate data normalization via component attachment in examples 9 to 11. As each implementation of RDF store leads to specific querying behaviors for different ways of component attachment, the choice of the component attachment matters especially for larger datasets. Different ways of attachments represent the same meaning. What changes are only the number of triples and the complexity of queries.

The second normalization way is to reattach data about linked entities. Linked data can be structured using the star schema (see Appendix: Star schema for example) or the fully denormalized schema (see Appendix: Fully denormalized schema for example). In this sense, the representation that is favoured by DCV and linked data principles is the normalized star or snowflake schema. However, as with component attachment the choice of normalization schema can affect queries both in terms of complexity and performance. For instance, (Jakobsen et al., 2015) found that data following the snowflake pattern is around 6 times slower to query using Openlink Virtuoso[13] RDF store than the same data denormalized. However, the contrary holds for the Apache Jena[14] RDF store, in which the snowflake pattern is generally faster. Data denormalization is thus recommended for Openlink Virtuoso, while it should be avoided in Jena that does not cope that well with the increased data size. The denormalized pattern is better for static data. If data changes frequently, then the cost of updates may

---

[12] http://www.w3.org/TR/vocab-data-cube/#h2_normalize

[13] https://github.com/openlink/virtuoso-opensource

[14] https://jena.apache.org/

surpass the benefits gained from denormalization. However, in the context of fiscal data we suggest using immutable snapshots of data, so that data does not change in place.

6.4 Slices as views

If you want to model a subset of a dataset, you can describe it as a dataset's slice (instance of `qb:Slice`). Data publishers may decide to split a dataset into multiple slices to ease consumption. For example, all dimensions except a temporal one can be grouped into the temporal slice to produce time series. Similarly, publishers may decide to reduce dimensionality of their datasets in order to make them fit the tabular format (e.g., an Excel file). If datasets views are published as slices of a single dataset, it simplifies integration of this dataset. Since the structure of the dataset is explicitly described in a DSD and the structure of its slices is described using instances of `qb:SliceKey`, slices can be automatically merged to form a unified dataset. Data publishers may also use slices to explicitly convey that only a particular subset of data is disclosed, while the remaining data is kept withheld. Consumers can infer this by comparing components included in the dataset's DSD and the components included in the slice's slice key.

Conversely, when data consumer recognizes that some published non-RDF data belongs to a single dataset, they can represent it in RDF using slices to maintain the identity and separation of the published data, while integrating the data in a single dataset.

6.5 Versioning via snapshots

In the course of budget formulation several versions of budget are created. We recommend using immutable snapshots of DCV datasets to represent versions of the same data. Newer snapshots of a dataset should link the `qb:DataSet` instance by `dcterms:replaces` to the `qb:DataSet` instance in the previous snapshot.

Snapshots should be used for versions of budget during its life cycle. For example, there can be a snapshot for a proposed budget and an approved budget. This technique of versioning should not be used for correction of minor errors. If each fix required a new snapshot of data to be produced, the volume of data would quickly become unwieldy. Instead, data corrections mutate data in place. Since this way of changing data is not explicit and cannot be observed, dataset metadata should document what changes were made using provenance information (e.g., using the PROV-O Ontology[15]).

7 Validation

When you have an RDF dataset following the proposed data model, there are several ways to test whether it is valid. Besides manual scrutiny, there are few automated tests that can help you to ascertain that the dataset is well-formed. The tests check either syntax or semantics of the dataset.

First, you should verify that the syntax of your dataset is correct. In order to do so, you can use any of the RDF validators available. Most RDF parsers offer syntax validation. For example, Riot from Apache Jena[16] can be invoked with the `--validate` parameter to test syntactical validity (e.g., `riot --validate path/to/file.ttl`).

Semantic validity with respect to the integrity constraints defined by DCV can be checked by the Data Cube Validator.[17] However, note that this tool is intended to be used for small datasets. If you have a larger dataset you can test the integrity constraints using any SPARQL

---

[15] http://www.w3.org/TR/prov-o/

[16] https://jena.apache.org/documentation/io/

[17] http://www.w3.org/2011/gld/validator/qb/qb-validator

endpoint that exposes the dataset thanks to the constraints being expressed as SPARQL ASK queries.[18] If your datasets passes all the constraints, it is considered to be well-formed. Alternatively, you can employ more sophisticated tools such as RDFUnit[19] to perform the validation.

8 Recommended metadata

We recommend the budget and spending datasets to be described by the metadata proposed in the DCV specification[20] and in DCAT-AP.[21] While we aim for the data to be as self-descriptive as possible, some information required for correct interpretation of data is beyond what can be explicitly formalized. This is why fiscal datasets should link to a textual documentation explaining how the data was created and how it can be used.

An important prerequisite for data reuse is an explicitly specified open licence. We adopt the Open Definition[22] to define what an open licence must conform to. When choosing which licence to use, we recommend following the Publisher's Guide to Open Data Licensing.[23]

9 Data model reference

In this section we present a comprehensive reference of the OpenBudgets.eu core data model. We list the core component properties defined for budget and spending data along with the core entities that are described using these properties. Additionally, we describe the linked entities that are modelled outside of the DCV model. These entities are linked via the component properties from DCV datasets.

9.1 Core properties

The core data model of OpenBudgets.eu defines 18 dimensions, 3 attributes, and 1 measure. Additionally, the model defines 2 extra properties not included in the data cube model.

9.1.1 Dimensions

9.1.1.1   accounting record

IRI: `obeu-dimension:accountingRecord`

Description: Link to an accounting record (e.g., invoice, credit note) associated with expenditure or revenue.

Allowed values: `foaf:Document`

Example value:

```
:document a foaf:Document;
  dcterms:issued "2015-11-04"^^xsd:date .
```

---

[18] http://www.w3.org/TR/vocab-data-cube/#h3_wf-rules

[19] http://aksw.org/Projects/RDFUnit.html

[20] http://www.w3.org/TR/vocab-data-cube/#metadata

[21] https://joinup.ec.europa.eu/asset/dcat_application_profile/description

[22] http://opendefinition.org/

[23] https://theodi.org/guides/publishers-guide-open-data-licensing

9.1.1.2 administrative classification

IRI: `obeu-dimension:administrativeClassification`

Description: Identifies the entity responsible for managing the public funds concerned.

Allowed values: `skos:Concept`

Super-property: `obeu-dimension:classification`

Example value:
```
<http://publications.europa.eu/resource/authority/corporate-body/ECHA> a
skos:Concept ;
    skos:notation "ECHA" ;
    skos:broader <http://publications.europa.eu/resource/authority/corporate-
body/EURAG> ;
    skos:inScheme atold:corporate-body ;
    skos:prefLabel "Европейска агенция по химикали"@bg,
        "Evropská agentura pro chemické látky"@cs,
        "Det Europæiske Kemikalieagentur"@da,
        "Europäische Chemikalienagentur"@de,
        "Ευρωπαϊκός Οργανισμός Χημικών Προϊόντων"@el,
        "European Chemicals Agency"@en,
        "Agencia Europea de Sustancias y Preparados Químicos"@es,
        "Euroopa Kemikaaliamet"@et,
        "Euroopan kemikaalivirasto"@fi,
        "Agence européenne des produits chimiques"@fr,
        "An Ghníomhaireacht Eorpach Ceimiceán"@ga,
        "Europska agencija za kemikalije"@hr,
        "Európai Vegyianyag-ügynökség"@hu,
        "Agenzia europea per le sostanze chimiche"@it,
        "Europos cheminių medžiagų agentūra"@lt,
        "Eiropas Ķīmisko vielu aģentūra"@lv,
        "L-Aġenzija Ewropea għas-Sustanzi Kimiċi"@mt,
        "Europees Agentschap voor chemische stoffen"@nl,
        "Europejska Agencja Chemikaliów"@pl,
        "Agência Europeia dos Produtos Químicos"@pt,
        "Agenţia Europeană pentru Produse Chimice"@ro,
        "Európska chemická agentúra"@sk,
        "Evropska agencija za kemikalije"@sl,
        "Europeiska kemikaliemyndigheten"@sv .
```

9.1.1.3 budget line

IRI: `obeu-dimension:budgetLine`

Description: Budget line from which the payment draws its funds.

Allowed values: `qb:Observation`

Example value: `<http://data.openbudgets.eu/resource/observation/eu-fishing-subsidies-CS-2007-2013/EUR/amountCZ>`

9.1.1.4 budget phase

IRI: `obeu-dimension:budgetPhase`

Description: Major event or stage in the budget cycle.

Allowed values: `obeu:BudgetPhase`

Example value: `obeu-budgetphase:Draft`

9.1.1.5 budgetary unit

IRI: `obeu-dimension:budgetaryUnit`

Description: An economic entity that is capable, in its own right, of owning assets, incurring liabilities, and engaging in economic activities and in transactions with other entities.

Allowed values: `org:Organization`

Example value: `<http://reference.data.gov.uk/id/department/justice>`

9.1.1.6   classification

IRI: `obeu-dimension:classification`

Description: Category to which observation belongs.

Allowed values: `skos:Concept`

Example value: This property is abstract, so it is not expected to be used directly. Either use a more specific property or create your own subproperty of this one.

9.1.1.7   currency

IRI: `obeu-dimension:currency`

Description: Currency of a financial amount.

Allowed values: `obeu:Currency`

Example value: `obeu-currency:EUR`

9.1.1.8   date

IRI: `obeu-dimension:date`

Description: Date when expense was paid or revenue received.

Allowed values: `time:Interval`

Example value:

```
:2015-11-01 a time:Interval ;
  time:hasBeginning :2015-11-01T00-00-00Z ;
  time:hasEnd :2015-11-01T00-00-00Z .


:2015-11-01T00-00-00Z a time:Instant ;
  time:inXSDDateTime "2015-11-01T00:00:00Z"^^xsd:dateTime .
```

9.1.1.9   economic classification

IRI: `obeu-dimension:economicClassification`

Description: Identifies the type of expenditure incurred or source of revenues.

Allowed values: `skos:Concept`

Super-property: `obeu-dimension:classification`

Example value:

```
<http://data.openbudgets.eu/resource/codelist/esa2010-distributive-
transactions/D.211> a skos:Concept ;
  skos:prefLabel "Value added type taxes (VAT)"@en ;
  skos:notation "D.211" ;
  skos:broader <http://data.openbudgets.eu/resource/codelist/esa-2010-
classification-of-transactions-and-other-flows/D.21> ;
  skos:inScheme <http://data.openbudgets.eu/resource/codelist/esa-2010-
classification-of-transactions-and-other-flows> .
```

9.1.1.10 fiscal period

IRI: `obeu-dimension:fiscalPeriod`

Description: The period of time reflected in financial statements.

Allowed values: `time:Interval`

Example value:

```
<http://reference.data.gov.uk/id/quarter/2012-Q1> a interval:Quarter ;
  time:hasBeginning <http://reference.data.gov.uk/id/gregorian-instant/2012-01-
01T00:00:00> ;
  time:hasEnd <http://reference.data.gov.uk/id/gregorian-instant/2012-04-
01T00:00:00> .
```

9.1.1.11 fiscal year

IRI: `obeu-dimension:fiscalYear`

Description: The year reflected in financial statements.

Allowed values: `interval:Year`

Super-property: `obeu-dimension:fiscalPeriod`

Example value: `<http://reference.data.gov.uk/id/year/2012>`

9.1.1.12 functional classification

IRI: `obeu-dimension:functionalClassification`

Description: Classifies expenditures by general government sector and by the purpose of the expenditure.

Allowed values: `skos:Concept`

Super-property: `obeu-dimension:classification`

Example value:

```
<http://unstats.un.org/unsd/cr/references/cofog/version1/09> a skos:Concept ;
  skos:notation "09" ;
  skos:prefLabel "Educación"@es, "Education"@en,
                 "Enseignement"@fr, "Образование"@ru ;
  skos:inScheme <http://unstats.un.org/unsd/cr/references/cofog/version1/> .
```

9.1.1.13 operation character

IRI: `obeu-dimension:operationCharacter`

Description: Distinguishes among expenditure and revenue.

Allowed values: `obeu:OperationCharacter`

Example value: `obeu-operation:Expenditure`

9.1.1.14 organization

IRI: `obeu-dimension:organization`

Description: The entity which made the payment or collected revenue.

Allowed values: `org:Organization`

Example value:

```
<http://reference.data.gov.uk/id/department/dclg> a org:Organization ;
  rdfs:label "Department for Communities and Local Government" .
```

9.1.1.15 partner

IRI: `obeu-dimension:partner`

Description: The entity to which the payment was made or from which the revenue was collected.

Allowed values: `org:Organization`

Example value:

```
:organization a org:Organization;
  rdfs:label "ACME Corp." .
```

9.1.1.16 programme classification

IRI: `obeu-dimension:programmeClassification`

Description: Groups budget lines by common objective for budgeting purposes.

Allowed values: `skos:Concept`

Super-property: `obeu-dimension:classification`

Example value:

```
eff:2 a skos:Concept ;
  skos:notation "2" ;
  skos:prefLabel "Aquaculture, processing and marketing of fishery and aquaculture products"@en ;
  skos:narrower eff:2.1 ;
  skos:topConceptOf <http://example.openbudgets.eu/resource/eu-fishing-subsidies-2007-2013/codelist/eu-fishing-subsidies-2007-2013> .
```

9.1.1.17 project

IRI: `obeu-dimension:project`

Description: Project associated with the payment.

Allowed values: `foaf:Project`

Example value:

```
:project a foaf:Project ;
  foaf:name "Renovation of playgrounds" .
```

9.1.1.18 taxes included

IRI: `obeu-dimension:taxesIncluded`

Description: Indicates whether the reported amount includes taxes.

Allowed values: `xsd:boolean`

Example value: `false`

9.1.2 Attributes

9.1.2.1 currency

IRI: `obeu-attribute:currency`

Description: Currency of a financial amount.

Allowed values: `obeu:Currency`

Example value: `obeu-currency:CZK`

9.1.2.2   location

IRI: `obeu-attribute:location`

Description: Physical location affected by a payment.

Allowed values: `schema:Place`

Example value:
```
:place a schema:Place ;
  schema:geo [
    a schema:GeoCoordinates ;
    schema:latitude 50.088382 ;
    schema:longitude 14.403665
  ] .
```

9.1.2.3   taxes included

IRI: `obeu-attribute:taxesIncluded`

Description: Indicates whether the reported amount includes taxes.

Allowed values: `xsd:boolean`

Example value: `true`

9.1.3 Measures

9.1.3.1   amount

IRI: `obeu-measure:amount`

Description: Monetary amount.

Allowed values: `xsd:decimal`

Example value: `3141.59`

9.1.4 Extra properties
The extra properties do not fit the DCV model, so they are defined as regular RDF properties.

9.1.4.1   contract

IRI: `obeu:contract`

Description: Public contract for which a payment is made.

Compatible with: `qb:Observation`

Allowed values: `pc:Contract`

Example value:
```
:contract a pc:Contract ;
  pc:contractingAuthority :authority ;
  pc:awardedTender [
```

```
   pc:bidder :supplier ;
   pc:offeredPrice [
     a gr:PriceSpecification ;
     gr:hasCurrencyValue 1000000.0 ;
     gr:hasCurrency "EUR"
   ]
 ] .
```

9.1.4.2 Methodology used

IRI: `obeu-metadata:methodologyUsed`

Description: A link to the document describing the methodology that was used a data structure definition.

Compatible with: `qb:DataStructureDefinition`

Allowed values: `foaf:Document`

Example value:

```
:document a foaf:Document;
  foaf:name "A budget methodology" ;
  foaf:homepage <http://example.org/budget/methodology/> .
```

9.2 Core entities

The core entities of the OpenBudgets.eu data model are represented as instances of `qb:Observation` from DCV. The observations can form a part of either budget or spending data. The OpenBudgets.eu directly reuses `qb:Observation` instead of a specific subclass[24] to maintain compatibility with existing tools for processing DCV data.

9.2.1 Budget line (`qb:Observation`)

Budget line is an identified amount allocated for a specific purpose.

```
<http://data.openbudgets.eu/resource/dataset/eu-budget-
2014/observation/Executed/Expenditure/2012/XX-01-01-01-02/5.2.3X> a qb:Observation
;
  qb:dataSet <http://data.openbudgets.eu/resource/dataset/eu-budget-2014> ;
  obeu-dimension:budgetPhase obeu-budgetphase:executed ;
  eu-dimension:operationCharacter obeu-operation:expenditure ;
  obeu-dimension:fiscalYear <http://reference.data.gov.uk/id/year/2012> ;
  eu-dimension:budgetNomenclature <http://data.openbudgets.eu/resource/codelist/eu-
budget-nomenclature/XX-01-01-01-02> ;
  eu-dimension:catpol <http://data.openbudgets.eu/resource/codelist/catpol/5.2.3X>
;
  obeu-measure:amount 13301985.81 .
```

9.2.2 Expenditure line (`qb:Observation`)

An item of expenditure that can be classified or assigned to a cost centre.[25]

```
<http://data.openbudgets.eu/resource/observation/eu-fishing-subsidies-CS-2007-
2013/Expenditure/Ministry_of_Agriculture_(Czech_Republic)/2.1a/CZ25170538/1/EUR/amo
untCZ> a qb:Observation ;
  obeu-dimension:currency obeu-currency:EUR ;
```

---

[24] For example, the Payments Ontology provides the class `pay:Payment` as a specific subclass of `qb:Observation`.

[25] Definition reused from the Payments Ontology (https://data.gov.uk/resources/payments).

```
obeu-attribute:currency obeu-currency:EUR ;
qb:measureType eu-measure:amountCZ ;
eu-measure:amountCZ 13829.87 ;
qb:dataSet <http://data.openbudgets.eu/resource/dataset/eu-fishing-subsidies-CS-
2007-2013> .
```

9.3 Linked entities

9.3.1 Code list concept (`skos:Concept`)

The core code list concepts are represented as SKOS concepts and the code lists themselves are SKOS Concept schemes. For each concept scheme a class is also defined and each concept of the concept scheme belongs to this class.

9.3.1.1 Budget phase

Budget phase distinguishes among phases of the budget. We specify 4 core budget phases, Draft, Revised, Approved and Executed.

```
obeu:BudgetPhase a rdfs:Class ;
  rdfs:label "Budget phase"@en ;
  rdfs:isDefinedBy <http://data.openbudgets.eu/ontology> .

obeu-codelist:budget-phase a skos:ConceptScheme ;
  rdfs:label "Code list that distinguishes among phases of the budget."@en ;
  skos:hasTopConcept obeu-budgetphase:draft, obeu-budgetphase:revised, obeu-
budgetphase:approved, obeu-budgetphase:executed .

obeu-budgetphase:draft a skos:Concept, obeu:BudgetPhase ;
  skos:prefLabel "Draft"@en ;
  skos:topConceptOf obeu-codelist:budget-phase ;
  skos:inScheme obeu-codelist:budget-phase .

obeu-budgetphase:revised a skos:Concept, obeu:BudgetPhase ;
  skos:prefLabel "Revised"@en ;
  skos:topConceptOf obeu-codelist:budget-phase ;
  skos:inScheme obeu-codelist:budget-phase .

obeu-budgetphase:approved a skos:Concept, obeu:BudgetPhase ;
  skos:prefLabel "Approved"@en ;
  skos:topConceptOf obeu-codelist:budget-phase ;
  skos:inScheme obeu-codelist:budget-phase .

obeu-budgetphase:executed a skos:Concept, obeu:BudgetPhase ;
  skos:prefLabel "Executed"@en ;
  skos:topConceptOf obeu-codelist:budget-phase ;
  skos:inScheme obeu-codelist:budget-phase .
```

9.3.1.2 Classification

Revenue and expenditure are grouped based on common characteristics. Several different criteria may be used for grouping revenue and expenditure via classifications. Classifications constitute a basic information system that enables an objective breakdown of the operations performed by the public sector.[26]

There are 4 main types of budget and spending classifications: administrative, economic, functional, and programme. Usually, classifications are organized hierarchically, so that major categories break down into narrower categories. Several guiding principles can be used when you try to recognize what kind of classification is used in a fiscal dataset:

---

[26] http://www.mecon.gov.ar/consulta/ingles/glosario.html

- In case of administrative classification each category of the most detailed breakdown will be equal to a single organization or its units.
- In case of economic classification the major categories for revenue are taxes and other revenues and for expenditure these categories include current (operational) and capital (investment) expenditure. The major categories of current expenditures are wages, purchases, and transfers.
- The distinction between functional and programme classification is not always clear: functional classification organizes government activities according to their purpose and programme classification according to government policy objectives. While functional classification can be in place for years, programme classification should reflect current policy documents (Allen & Tommasi, 2001, p. 126).

9.3.1.3   Currency

Currency is specified by an entity and its label, it is connected to qb:Observations through the `obeu-attribute:currency` and `obeu-dimension:currency` component properties.

```
:observation obeu-attribute:currency obeu-currency:czk .

obeu-currency:czk skos:prefLabel "Czech koruna" ;
                  skos:notation "CZK" .
```

9.3.1.4   Operation character

Operation character distinguishes among characters of fiscal operation. We specify two core operation characters, Expenditure and Revenue.

```
obeu:OperationCharacter a rdfs:Class ;
  rdfs:label "Operation character"@en ;
  rdfs:isDefinedBy <http://data.openbudgets.eu/ontology> .

obeu-codelist:operation-character a skos:ConceptScheme ;
  rdfs:label "Code list that distinguishes among characters of fiscal
operation."@en ;
  skos:hasTopConcept obeu-operation:expenditure, obeu-operation:revenue .

obeu-operation:expenditure a skos:Concept, obeu:OperationCharacter ;
  skos:prefLabel "Expenditure"@en ;
  skos:definition "Decrease in net worth resulting from a transaction and the net
investment in nonfinancial assets"@en ;
  skos:topConceptOf obeu-codelist:operation-character ;
  skos:inScheme obeu-codelist:operation-character .

obeu-operation:revenue a skos:Concept, obeu:OperationCharacter ;
  skos:prefLabel "Revenue"@en ;
  skos:definition "An increase in net worth resulting from a transaction"@en ;
  skos:topConceptOf obeu-codelist:operation-character ;
  skos:inScheme obeu-codelist:operation-character .
```

9.3.2 Interval (time:Interval)

Temporal intervals are represented using the Time Ontology in OWL.[27] Intervals are delimited by 2 instants (`time:Instant`). Each instant is represented using the `xsd:dateTime` data type associated via the `time:inXSDDateTime` property. If source data contains only dates expressed by the `xsd:date` data type, you can coerce them into `xsd:dateTime` by appending `T00:00:00`.

---

[27] http://www.w3.org/TR/owl-time/

In case a single point in time is associated with a fiscal data item, both instants delimiting the interval are the same. To prevent data duplication in such case, IRIs should be used to identify instances of `time:Instant`. This way the instant can be described once and reused many times.

For longer intervals representing fiscal periods, such as quarter or year, established IRIs from the `http://reference.data.gov.uk/id/` namespace (e.g., `http://reference.data.gov.uk/id/year/2014` for the year 2014) should be reused.

9.3.3 Organization (`org:Organization`)

Organizations, including budgetary units or project partners, are represented as instances of the `org:Organization` class from the Organization Ontology.[28] You can use the means provided by this ontology to further describe the organizations.

9.3.4 Place (schema:Place)

Locations where money is spent are represented as instances of the `schema:Place` class from the Schema.org.[29]

9.3.5 Accounting record (`foaf:Document`)

Accounting records are represented as instances of the `foaf:Document` class from the Friend of a Friend vocabulary.[30] They can be further described by the Dublin Core[31] vocabulary.

9.3.6 Project (foaf:Project)

Projects are represented as instances of the `foaf:Project` class from the Friend of a Friend vocabulary.

9.3.7 Contract (`pc:Contract`)

Public contracts are represented as instances of the `pc:Contract` class from the Public Contracts Ontology.[32]

10 References

- Allen R., Tommasi D. (eds.) (2001): Managing public expenditure: a reference book for transition countries.
  http://www1.worldbank.org/publicsector/pe/oecdpemhandbook.pdf
- Berners-Lee, T. (2006): Linked Data - Design Issues,
  http://www.w3.org/DesignIssues/LinkedData.html
- Brickley D., Guha R. (2014): RDF Schema 1.1, http://www.w3.org/TR/rdf-schema/
- Cyganiak R, Reynolds D. (2014): The RDF Data Cube Vocabulary,
  http://www.w3.org/TR/vocab-data-cube/

---

[28] http://www.w3.org/TR/vocab-org/

[29] http://schema.org

[30] http://xmlns.com/foaf/spec/

[31] http://dublincore.org/documents/dcmi-terms/

[32] https://github.com/opendatacz/public-contracts-ontology

- Dodds L., Davis I. (2012): Linked data patterns: a pattern catalogue for modelling, publishing, and consuming linked data. http://patterns.dataincubator.org/book/
- Duerst M., Suignard M. (2004): Internationalized Resource Identifiers (IRIs). https://www.ietf.org/rfc/rfc3987.txt
- European Union: public finance (2014). 5th ed. Luxembourg: Publications Office of the European Union. http://ec.europa.eu/budget/library/biblio/publications/2014/EU_pub_fin_en.pdf. ISBN 978-92-79-35004-7.
- Eurostat (2015a): Gross domestic product at market prices, http://ec.europa.eu/eurostat/tgm/table.do?tab=table&init=1&language=en&pcode=tec00001&plugin=1
- Eurostat (2015b): Total general government expenditure, http://ec.europa.eu/eurostat/tgm/table.do?tab=table&plugin=1&language=en&pcode=tec00023
- Eurostat (2015c): Real GDP growth rate – volume, http://ec.europa.eu/eurostat/tgm/table.do?tab=table&init=1&language=en&pcode=tec00115&plugin=1
- Ioannidis, L., Philippides, P.-M., Bratsas, C., Koupidis, K. (2015): OpenBudgets.eu – Deliverable D1.6 – Survey of code lists for the data model's coded dimensions, https://openbudgets.atlassian.net/browse/OB-17
- Jakobsen K., Andersen A., Hose K., Pedersen T. (2015): Optimizing RDF data cubes for efficient processing of analytical queries, http://ceur-ws.org/Vol-1426/paper-02.pdf
- Klímek J., Kučera J., Mynarz J., Sedmihradská L., Zbranek J. (2015a): OpenBudgets.eu - Deliverable D1.2 - Design of data structure definition for public budget data, https://openbudgets.atlassian.net/browse/OB-13
- Klímek J., Kučera J., Mynarz J., Sedmihradská L., Zbranek J. (2015b): OpenBudgets.eu - Deliverable D1.3 - Design of data structure definition for public spending data, https://openbudgets.atlassian.net/browse/OB-14
- Miles A., Bechhofer S. (2009): SKOS Simple Knowledge Organization System Reference, http://www.w3.org/TR/skos-reference/

11 Appendix: Codelist extension example

```
# ----- DSD-specific namespaces -----

@prefix eu-attribute:       <http://example.openbudgets.eu/ontology/dsd/eu-budget-
2014/attribute/> .
@prefix eu-dimension:       <http://example.openbudgets.eu/ontology/dsd/eu-budget-
2014/dimension/> .
@prefix eu-measure:         <http://example.openbudgets.eu/ontology/dsd/eu-budget-
2014/measure/> .
@prefix eu-codelist:        <http://example.openbudgets.eu/resource/eu-budget-
2014/codelist/> .
@prefix eu-operation:       <http://example.openbudgets.eu/resource/eu-budget-
2014/codelist/operation-character/> .

# ----- OpenBudgets.eu namespaces -----

@prefix obeu:               <http://data.openbudgets.eu/ontology/> .
@prefix obeu-attribute:     <http://data.openbudgets.eu/ontology/dsd/attribute/> .
@prefix obeu-dimension:     <http://data.openbudgets.eu/ontology/dsd/dimension/> .
@prefix obeu-measure:       <http://data.openbudgets.eu/ontology/dsd/measure/> .
@prefix obeu-operation:     <http://data.openbudgets.eu/resource/codelist/operation-
character/> .

# ----- Generic namespaces ------

@prefix qb:                 <http://purl.org/linked-data/cube#> .
@prefix rdf:                <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:               <http://www.w3.org/2000/01/rdf-schema#> .
@prefix skos:               <http://www.w3.org/2004/02/skos/core#> .
@prefix xsd:                <http://www.w3.org/2001/XMLSchema#> .

eu-dimension:operationCharacter a rdf:Property, qb:CodedProperty,
qb:DimensionProperty ;
  rdfs:label "Operation character"@en ;
  rdfs:comment "EU budget's specific operation characters"@en ;
  rdfs:subPropertyOf obeu-dimension:operationCharacter ;
  qb:codeList eu-codelist:operation-character ;
  rdfs:range obeu:OperationCharacter ;
  rdfs:isDefinedBy <http://example.openbudgets.eu/ontology/dsd/eu-budget-2014> .

eu-codelist:operation-character a skos:ConceptScheme ;
  rdfs:label "An extended code list of operation characters for the EU budget"@en ;
  skos:hasTopConcept obeu-operation:expenditure, obeu-operation:revenue .

eu-operation:commitment a skos:Concept, obeu:OperationCharacter ;
  skos:prefLabel "Commitment"@en ;
  skos:definition "Total cost of the legal commitments during the current fiscal
year."@en ;
  skos:broader obeu-operation:expenditure ;
  skos:inScheme eu-codelist:operation-character .


eu-operation:payment a skos:Concept, obeu:OperationCharacter ;
  skos:prefLabel "Payment"@en ;
  skos:definition "Payments made to honour the legal commitments entered into in
the current fiscal year and/or earlier fiscal years."@en ;
  skos:broader obeu-operation:expenditure ;
  skos:inScheme eu-codelist:operation-character .
```

12 Appendix: Star schema

```
@prefix interval:          <http://reference.data.gov.uk/def/intervals/> .
@prefix qb:                <http://purl.org/linked-data/cube#> .
@prefix rdfs:              <http://www.w3.org/2000/01/rdf-schema#> .
@prefix scv:               <http://purl.org/NET/scovo#> .
@prefix sdmx-attribute:    <http://purl.org/linked-data/sdmx/2009/attribute#> .
@prefix skos:              <http://www.w3.org/2004/02/skos/core#> .
@prefix xsd:               <http://www.w3.org/2001/XMLSchema#> .

@prefix ex-codelist:       <http://data.example.org/resource/codelist/> .
@prefix ex-dataset:        <http://data.example.org/resource/dataset/> .
@prefix ex-dimension:      <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-dsd:            <http://data.example.org/resource/dsd/> .
@prefix ex-geo:            <http://data.example.org/resource/codelist/geo/> .
@prefix ex-measure:        <http://data.example.org/ontology/dsd/measure/> .
@prefix ex-units-of-measure: <http://data.example.org/resource/codelist/units-of-
measure/> .

# Data structure definition

ex-dsd:total-general-government-expenditure a qb:DataStructureDefinition ;
  rdfs:label "Total general government expenditure"@en ;
  qb:component [ qb:dimension ex-dimension:refPeriod ],
               [ qb:dimension ex-dimension:refArea ],
               [ qb:measure ex-measure:total-general-government-expenditure ],
               [ qb:attribute sdmx-attribute:unitMeasure ;
                 qb:componentRequired true ] .

# Dataset

ex-dataset:total-general-government-expenditure a qb:DataSet ;
  rdfs:label "Total general government expenditure"@en ;
  qb:structure ex-dsd:total-general-government-expenditure .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2012/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2012> ;
  ex-dimension:refArea ex-geo:EU28 ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 49 .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2013/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2013> ;
  ex-dimension:refArea ex-geo:EU28 ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 48.6 .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2014/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriod <http://reference.data.gov.uk/id/gregorian-year/2014> ;
  ex-dimension:refArea ex-geo:EU28 ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 48.2 .

<http://reference.data.gov.uk/id/gregorian-year/2014> a interval:Year ;
  scv:max "2014-12-31"^^xsd:date ;
  scv:min "2014-01-01"^^xsd:date .

<http://reference.data.gov.uk/id/gregorian-year/2013> a interval:Year ;
  scv:max "2013-12-31"^^xsd:date ;
  scv:min "2013-01-01"^^xsd:date .

<http://reference.data.gov.uk/id/gregorian-year/2012> a interval:Year ;
```

```
    scv:max "2012-12-31"^^xsd:date ;
    scv:min "2012-01-01"^^xsd:date .

ex-geo:EU28 a skos:Concept ;
    skos:prefLabel "European Union (28 countries)"@en ;
    skos:notation "EU28" ;
    skos:inScheme ex-codelist:geo .
```

```
    scv:max "2012-12-31"^^xsd:date ;
```

13 Appendix: Fully denormalized schema

```
@prefix qb:                 <http://purl.org/linked-data/cube#> .
@prefix rdfs:               <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sdmx-attribute:     <http://purl.org/linked-data/sdmx/2009/attribute#> .
@prefix xsd:                <http://www.w3.org/2001/XMLSchema#> .

@prefix ex-dataset:         <http://data.example.org/resource/dataset/> .
@prefix ex-dimension:       <http://data.example.org/ontology/dsd/dimension/> .
@prefix ex-dsd:             <http://data.example.org/resource/dsd/> .
@prefix ex-measure:         <http://data.example.org/ontology/dsd/measure/> .
@prefix ex-units-of-measure: <http://data.example.org/resource/codelist/units-of-
measure/> .

# Data structure definition

ex-dsd:total-general-government-expenditure a qb:DataStructureDefinition ;
  rdfs:label "Total general government expenditure"@en ;
  qb:component [ qb:dimension ex-dimension:refPeriodStart ],
               [ qb:dimension ex-dimension:refPeriodEnd ],
               [ qb:dimension ex-dimension:refAreaCode ],
               [ qb:dimension ex-dimension:refAreaLabel ],
               [ qb:measure ex-measure:total-general-government-expenditure ],
               [ qb:attribute sdmx-attribute:unitMeasure ;
                 qb:componentRequired true ] .

# Dataset

ex-dataset:total-general-government-expenditure a qb:DataSet ;
  rdfs:label "Total general government expenditure"@en ;
  qb:structure ex-dsd:total-general-government-expenditure .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2012/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriodStart "2012-01-01"^^xsd:date ;
  ex-dimension:refPeriodEnd "2012-12-31"^^xsd:date ;
  ex-dimension:refAreaCode "EU28" ;
  ex-dimension:refAreaLabel "European Union (28 countries)"@en ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 49 .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2013/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriodStart "2013-01-01"^^xsd:date ;
  ex-dimension:refPeriodEnd "2013-12-31"^^xsd:date ;
  ex-dimension:refAreaCode "EU28" ;
  ex-dimension:refAreaLabel "European Union (28 countries)"@en ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 48.6 .

<http://data.example.org/resource/observation/total-general-government-
expenditure/2014/EU28> a qb:Observation ;
  qb:dataSet ex-dataset:total-general-government-expenditure ;
  ex-dimension:refPeriodStart "2014-01-01"^^xsd:date ;
  ex-dimension:refPeriodEnd "2014-12-31"^^xsd:date ;
  ex-dimension:refAreaCode "EU28" ;
  ex-dimension:refAreaLabel "European Union (28 countries)"@en ;
  sdmx-attribute:unitMeasure ex-units-of-measure:percent-of-GDP ;
  ex-measure:total-general-government-expenditure 48.2 .
```