# D1.3 - Design of data structure definition for public spending data

**Authors**

- Jakub Klímek
- Jan Kučera
- Jindřich Mynarz
- Lucie Sedmihradská
- Jaroslav Zbranek

## Executive summary

In this demonstrator deliverable, a set of OpenBudgets.eu core RDF component properties for description of spending data is presented. The deliverable is based on results achieved in a survey and knowledge elicitation performed with domain experts and reported in Deliverable D1.1 (Klímek et al., 2015). In addition to the component properties definitions, a method of creation of additional component properties and the data structure definitions themselves is presented and illustrated by an example of European Fisheries Fund from 2007 - 2013 for the Czech Republic. This deliverable has the same structure and introductory information regarding data cube modelling as Deliverable D1.2 which dealt with data structure definition for public budget data. Also some components and their descriptions are reused from Deliverable D1.2.

## Table of contents

- Executive summary
- Table of contents
- Introduction
- The RDF Data Cube Vocabulary
- OpenBudgets.eu RDF prefixes
- Definition of core component properties for spending data
  - Core dimensions
    - Organization
    - Partner
    - Date
    - Operation character
    - Budget line
    - Currency
    - Taxes included
    - Project
    - Classification
      - Functional classification
      - Programme classification
      - Economic classification

Administrative classification
Accounting record
Core measures
Amount
Core attributes
Currency
Location
Taxes included
Additional properties
Contract
Metadata
Additional component properties for specific datasets
Creating DSDs using component properties
Example: European Fisheries Fund from 2007 - 2013 for the Czech Republic
Conclusions
References
Appendix: Core component properties
Appendix: Core code lists
Appendix: Additional properties
Appendix: Example DSD and component properties
Appendix: Sample data

# Introduction

In this demonstrator deliverable, we focus on the methodology of creating data structure definitions for spending data using the RDF Data Cube Vocabulary (DCV) together with the RDF Schema (RDFS)[1] vocabulary as the formalism for describing component properties. We also provide a list of identified reusable component properties for dimensions, attributes and measures that appear frequently in spending datasets and will serve as a tool for achieving comparability among spending datasets. We decided to propose component properties (i.e. instances of `qb:ComponentProperty`) as reusable components out of which concrete data structure definitions can be built. Component properties are generic enough to be reusable across multiple datasets. Unlike component specifications, they do not describe the physical structure of the dataset (e.g., component order or attachment level). This deliverable focuses on the formalization of the data model in RDF, while the full data model documentation along with extended guidelines will be presented in Deliverable D1.4. This deliverable follows the same structure and contains the same introductory information regarding data cube modelling as Deliverable D1.2 which dealt with data structure definition for public budget data.

[1] http://www.w3.org/TR/rdf-schema/

# The RDF Data Cube Vocabulary

For representation of data cubes such as budget data in RDF the RDF Data Cube Vocabulary[2] is the most appropriate option. It is a widely used vocabulary for representing multidimensional statistical data compatible with the well-known SDMX (Statistical Data and Metadata eXchange) ISO standard. The key terms of DCV and their relationships are depicted in Figure 1.

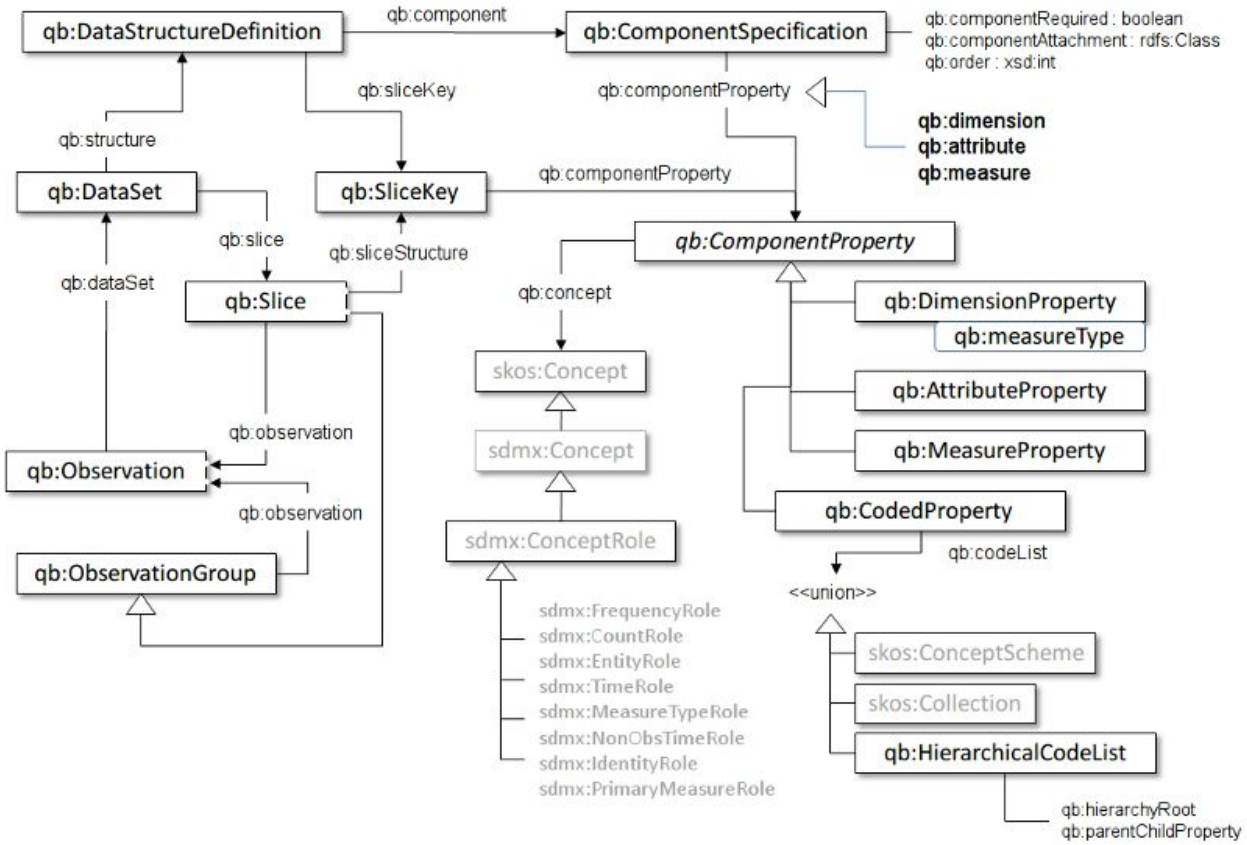


*Figure 1: Key terms and relationships in The RDF Data Cube Vocabulary, source: (Cyganiak & Reynolds, 2014)*

A data cube consists of dimensions, which describe properties of individual observations such as time period or geographical region. In the context of spending data, these are typically the transaction date, payer and payee. Then there are measures representing the observed values such as height, width, amount, etc. Again, in the context of spending data, this is typically the amount of money paid. Finally, there are attributes, which specify additional properties of the measures, such as unit of measurement or multiplier. In spending data, this is typically the currency of the transaction. Data cube can be sliced by grouping observations with the same values on selected dimensions, e.g., transactions for a selected supplier. Using the RDF Data Cube Vocabulary we model the data structure definition using components (dimensions,

[2] http://www.w3.org/TR/vocab-data-cube/

measures, and attributes) and then the defined components are used to classify individual observations.

# OpenBudgets.eu RDF prefixes

For OpenBudgets.eu we will use the following RDF prefixes based on a similar approach for SDMX:[3]

- `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/>`

# Definition of core component properties for spending data

In this section, we describe identified core component properties. Component properties to consider have been identified in Deliverable D1.1 either as occurring in datasets, although using various labels and identifiers, or they were considered important by domain experts based on knowledge elicitation. The core component properties were then selected as the most significant ones based on their frequent occurrence in datasets and the intended use of spending data based on the user requirements proceeding from the project's use cases.

## Core dimensions

Values of dimensions uniquely identify the measured value (observation). For spending data, the core dimensions are defined in the following sub-sections.

### Organization

Organization is the entity which made the payment or received revenue.

```
obeu-dimension:organization a rdf:Property, qb:DimensionProperty,
qb:CodedProperty, owl:ObjectProperty ;
  rdfs:label "organization"@en ;
  rdfs:comment "The entity which made the payment or collected
revenue."@en ;
  rdfs:range org:Organization .
```

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

### Partner

A partner is the entity to which the payment was made or from which the revenue was collected. The partner can be either legal or physical person. If the partner is a legal person, we expect it to be uniquely identified by a company identifier from an official business register.

```
obeu-dimension:partner a rdf:Property, qb:DimensionProperty,
qb:CodedProperty, owl:ObjectProperty ;
  rdfs:label "partner"@en ;
  rdfs:comment "The entity to which the payment was made or from
which the revenue was collected."@en .
```

If partners are not disclosed as part of a spending dataset, this dimension is omitted from the data structure definition of the dataset. If identities of only some partners are withheld (e.g., because of confidentiality restrictions), then the dimension is included in the data structure definition and blank nodes[4] are used as placeholders for withheld partners.

### Date

Spending data creates a historical record of government's financial management. In order to organize the spending history it is necessary to be able to situate financial transactions in time when they took place. To describe this time we provide the date dimension property. Because spending is cash-based, it is associated with the date when an expense was paid or revenue received. We define the range of the date property as `time:Interval` to maintain compatibility with the definition of `obeu-dimension:fiscalPeriod` from Deliverable D1.2. This allows to express hierarchical links between dates (e.g., a day is contained within a quarter) and guide data aggregation.

```
obeu-dimension:date a rdf:Property, qb:DimensionProperty,
qb:CodedProperty ;
  rdfs:label "date"@en ;
  rdfs:comment "Date when expense was paid or revenue received."@en ;
  rdfs:range time:Interval ;
  rdfs:subPropertyOf obeu-dimension:fiscalPeriod ;
  qb:concept sdmx-concept:refPeriod .
```

### Operation character

There are spending datasets that contain revenues and expenditures of the organization (i.e. `obeu-dimension:organization`). To distinguish among those, a dimension property and a corresponding code list are defined:

```
obeu-dimension:operationCharacter a rdf:Property,
qb:DimensionProperty, qb:CodedProperty ;
```

[4] http://www.w3.org/TR/rdf11-concepts/#section-blank-nodes

```
      rdfs:label "operation character"@en ;
      rdfs:comment "Distinguishes among expenditure and revenue."@en
;
      rdfs:range obeu:OperationCharacter ;
      qb:codeList obeu-codelist:operationCharacter .
```

The supplied code list has 2 items: `obeu-operation:Expenditure`, `obeu-operation:Revenue`.

This dimension is paired with the supplied codelist and from knowledge elicitation it seems that these two operation characters are widely adopted. However, some datasets distinguish transactions in financial assets and liabilities (such as loans) as a separate operation character. In such case, a dataset-specific version of this dimension property with an extended code list can be devised.

### Budget line

Each expenditure or revenue must be linked to a single budget line, from which it is funded. In practice this link is often not released as part of spending data. The reason why we define this dimension as part of the core data model is twofold. First, we want to promote a good practice that includes publishing links to budget lines to which spending is associated. Second, even though links to budget lines may be provided only in a minority of spending datasets, not defining this dimension would require the publishers of spending datasets containing links to budgets to define their own dimensions describing the same relationship. Having multiple and possibly incompatible definitions of the budget line dimension would hinder interoperability and make reuse of data more difficult. Hence, we provide the budget line component property that can be readily reused.

```
obeu-dimension:budgetLine a rdf:Property, qb:DimensionProperty,
qb:CodedProperty, owl:ObjectProperty ;
  rdfs:label "budget line"@en ;
  rdfs:comment "Budget line from which the payment draws its
funds"@en ;
  rdfs:range qb:Observation .
```

### Currency

Currency of transactions is a very important fact that is often left implicit in spending datasets. To improve comparability and machine readability, in OpenBudgets.eu all datasets must have their currency specified. There are 2 common types of modelling of currency. In the first case, the spending dataset contains a transaction amount in a single currency and we want to specify this currency. For this case we use only the currency attribute described below, not the currency dimension. The currency dimension will be used in the second case where we have one amount of money expressed in multiple currencies. Here, the currency dimension will distinguish

between the individual currencies in addition to the currency attribute. This is also illustrated by the example attached to this deliverable.

```
obeu-dimension:currency a rdf:Property, qb:DimensionProperty,
qb:CodedProperty ;
    rdfs:label "currency"@en ;
    rdfs:comment "The currency of the financial amount"@en ;
    rdfs:range obeu:Currency ;
    qb:concept sdmx-concept:currency .
```

### Taxes included

Taxes included similarly to currency can be both a dimension and an attribute. It indicates whether the reported monetary amount includes taxes. It is a boolean property that can be either true or false. Examples of common taxes included in expenditure are value-added tax or excise duty (e.g., for fuel). In some cases, payments of taxes are reported as dedicated expenditure lines aggregated by the tax type. In datasets containing for each line both the amount with and without taxes, this will be used as a dimension distinguishing between the two observations. In addition, it will be used as an attribute qualifying the amount, which is described below.

```
obeu-dimension:taxesIncluded a rdf:Property, qb:DimensionProperty,
qb:CodedProperty ;
  rdfs:label "tax included"@en ;
  rdfs:comment "Indicates whether the reported amount includes
taxes."@en ;
  rdfs:range xsd:boolean .
```

### Project

Expenditure may be associated with a project. In particular, this is common in data about subsidies, such as EU structural and cohesion funds.

```
obeu-dimension:project a rdf:Property, qb:DimensionProperty,
qb:CodedProperty, owl:ObjectProperty ;
  rdfs:label "project"@en ;
  rdfs:comment "Project associated with the payment"@en ;
  rdfs:range foaf:Project .
```

### Classification

Classifications of spending data are inherited from the budget lines the spending is linked to. For example, in many cases, if a need to reclassify spending arises, the approved budget is amended and the budget lines from which the expenditure is made is reclassified. However, we are aware that in practice the link between spending and budget lines may be missing. Therefore, we reuse budget classifications defined in Deliverable D1.2 for spending data. Budget line classifications should be attached to spending data only if links to budget lines are

unavailable. Otherwise, it is recommended to attach classifications to budget lines. However, in some cases, there are classifications specific for spending, e.g., in cases of subsidies there can be specific programme classifications.

```
obeu-dimension:classification a rdf:Property, qb:DimensionProperty,
qb:CodedProperty ;
      rdfs:subPropertyOf dcterms:subject ;
      rdfs:label "classification"@en .
```

In our survey in D1.1 we have identified 4 recurring types of classifications:

### Functional classification

The functional classification categorizes expenditures according to the purposes and objectives for which they are intended. When there is only one functional classification used in the given dataset, this property should be used. All other functional classification dimensions should be subproperties of this property.

```
obeu-dimension:functionalClassification a rdf:Property,
qb:DimensionProperty, qb:CodedProperty ;
      rdfs:label "functional classification"@en ;
      rdfs:comment "Categorizes expenditures according to the
purposes and objectives for which they are intended."@en ;
      rdfs:subPropertyOf obeu-dimension:classification .
```

### Programme classification

The programme classification is a grouping of expenditures by common objective. When there is only one programme classification used in the given dataset, this property should be used. All other programme classification dimensions should be subproperties of this property.

```
obeu-dimension:programmeClassification a rdf:Property,
qb:DimensionProperty, qb:CodedProperty ;
      rdfs:label "programme classification"@en ;
      rdfs:comment "Grouping of expenditure by common objective."@en
;
      rdfs:subPropertyOf obeu-dimension:classification .
```

### Economic classification

The economic classification identifies the type of expenditure incurred. When there is only one economic classification used in the given dataset, this property should be used. All other economic classification dimensions should be subproperties of this property.

```
obeu-dimension:economicClassification a rdf:Property,
qb:DimensionProperty, qb:CodedProperty ;
      rdfs:label "economic classification"@en ;
      rdfs:comment "Identifies the type of expenditure incurred."@en
;
      rdfs:subPropertyOf obeu-dimension:classification .
```

### Administrative classification

The administrative classification identifies the entity responsible for managing the public funds concerned. This can be for example a specific department of a municipality, whereas the organization is the municipality itself. When there is only one administrative classification used in the given dataset, this property should be used. All other administrative classification dimensions should be subproperties of this property.

```
obeu-dimension:administrativeClassification a rdf:Property,
qb:DimensionProperty, qb:CodedProperty ;
      rdfs:label "administrative classification"@en ;
      rdfs:comment "Identifies the entity responsible for managing
the public funds concerned."@en ;
      rdfs:subPropertyOf obeu-dimension:classification .
```

### Accounting record

Most governing bodies require by law that for every spending operation there is an accounting record. For example, it is common that invoice IDs are provided for expenditure lines. This data is important for ensuring correct handling of expenditures in accounting.

A link to any accounting record, such as an invoice, associated with spending. The kind of record is determined by `rdf:type` of the linked resource. For example, invoices can be represented as instances of `schema:Invoice` from Schema.org. In most cases only an identifier of the accounting record is available in spending datasets, which allows to create an explicit link to the accounting record.

```
obeu-dimension:accountingRecord a rdf:Property, qb:DimensionProperty,
qb:CodedProperty ;
  rdfs:label "accounting record"@en ;
   rdfs:comment "Link to an accounting record (e.g., invoice, credit
note) associated with expenditure or revenue."@en ;
  rdfs:range foaf:Document .
```

## Core measures

Measures contain the actual measured values identified by values on dimensions. In case of spending data, the core measure is the monetary amount.

### Amount

Amount is a monetary amount represented as a non-negative number. Instead of using minus or plus sign to indicate the payment direction, it must be specified by the `obeu-dimension:operationCharacter` dimension.

```
obeu-measure:amount a rdf:Property, qb:MeasureProperty ;
      rdfs:label "amount"@en ;
      rdfs:comment "The monetary amount."@en ;
```

```
    rdfs:subPropertyOf sdmx-measure:obsValue ;
    rdfs:range xsd:decimal ;
    qb:concept sdmx-concept:obsValue .
```

Note that we specify `xsd:decimal` as the data type to be used with the amount measure. This is because we want to avoid loss of precision that could happen when `xsd:float` or `xsd:double` are used.

When amount is not disclosed and kept confidential, 0 should be used as its value and `sdmx-attribute:obsStatus` attribute should be used to describe the reason why the amount is unavailable. For example, if the amount is missing, `sdmx-code:obsStatus-M`[5] should be used.

## Core attributes

Attributes further specify the numerical values of measures, such as currency or the inclusion of taxes.

### Currency

Currency of the transactions is a very important fact that is often omitted in spending datasets. To improve comparability and machine readability, in OpenBudgets.eu all datasets must have their currency specified. The currency is modelled here as an attribute because it specifies a property of the measure (amount) rather than identifying it.[6] Note that according to the RDF Data Cube Vocabulary, the attribute components can be attached at observation, slice or dataset level. This allows the datasets to have one or even multiple currencies.

```
obeu-attribute:currency a rdf:Property, qb:AttributeProperty,
qb:CodedProperty ;
    rdfs:label "currency"@en ;
    rdfs:comment "The currency of the financial amount"@en ;
    rdfs:subPropertyOf sdmx-attribute:currency ;
    rdfs:range obeu:Currency ;
    qb:concept sdmx-concept:currency .
```

### Location

In some cases spending is associated with a particular physical location. For example, it may be used to pay for a road repair.

```
obeu-attribute:location a rdf:Property, qb:AttributeProperty,
owl:ObjectProperty ;
```

---

[5] `sdmx-code` prefix corresponds to the http://purl.org/linked-data/sdmx/2009/code# namespace.

[6] Currency could be modelled as a dimension when we would model a dataset like currency exchange rates, where the currency would expressed as a dimension identifying the measure.

```
  rdfs:label "location"@en ;
  rdfs:comment "Physical location affected by the payment"@en ;
  rdfs:range schema:Place .
```

### Taxes included

Taxes included attribute indicates whether the reported monetary amount includes taxes. It is a boolean property that can be either true or false. Examples of common taxes included in expenditure are value-added tax or excise duty (e.g., for fuel). In some cases, payments of taxes are reported as dedicated expenditure lines aggregated by the tax type.

```
obeu-attribute:taxesIncluded a rdf:Property, qb:AttributeProperty,
qb:CodedProperty ;
  rdfs:label "tax included"@en ;
  rdfs:comment "Indicates whether the reported amount includes
taxes."@en ;
  rdfs:range xsd:boolean .
```

## Additional properties

Aside from the dimension, measure and attribute properties which are used in data structure definitions directly, there can be more information attached to the individual observations of the data cube which is not part of the data structure.

### Contract

Expenditure may be made for a public contract. Links to contracts are optional. There are legitimate cases in which spending is not warranted by a contract, such pensions. In other cases links to contracts are confidential and thus are not disclosed, such as for wages. Expenditure may be associated with at most one contract. If it relates to multiple contracts, it is typically split into multiple expenditures. In order to address the nature of the links to contracts, we decided to represent them outside of the core data cube. The `obeu:contract` property is therefore not defined as a DCV's component property, but as a plain `rdf:Property`. This mirrors the approach to linking contracts in the Payments Ontology (Tennison, 2011). As a consequence of defining the property not as a DCV's component property, it is likely that it will be ignored by most off-the-shelf tools that work with data represented using DCV. However, it can be exploited by applications made aware of its existence. The range of this property is `pc:Contract` from the Public Contracts Ontology.[7] In practice, links to contracts are often missing in spending datasets. However, we followed the same reasoning as with the `obeu-dimension:budgetLine` property and decided to include this property in the core data model.

```
obeu:contract a rdf:Property, owl:FunctionalProperty,
owl:ObjectProperty ;
  rdfs:label "contract"@en ;
```

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

```
rdfs:comment "Public contract for which the payment is made"@en ;
rdfs:domain qb:Observation ;
rdfs:range pc:Contract .
```

## Metadata

Data Cube Vocabulary recommends describing datasets with basic metadata,[8] which we endorse. The purpose of providing metadata include helping users to discover the dataset via subject classifications, learning about conditions for reuse, guiding interpretation of its content by additional explanations, assessing dataset's dynamics based on modification timestamps, or establishing trustworthiness from provenance metadata. An especially important part of metadata is link to documentation that describes how the datasets was created, including details about the methodology or aggregation methods used.

## Additional component properties for specific datasets

As spending datasets can be variable in terms of the number and the nature of component properties (dimensions, measures and attributes), in OpenBudgets.eu we only specify a core set of components which should be used whenever appropriate. However, often it will be the case that additional data needs to be represented. In this case, additional component properties are to be defined using DCV similarly to how the core components were defined. For instance, in the case of classifications, new classification dimensions may be subproperties of the core classification dimension types.

## Creating DSDs using component properties

Using the provided core component properties and optional additional component properties a data structure definition (DSD) is to be created according to DCV to describe the structure of a given dataset. A data structure definition comprises components that specify which component properties are used. For attribute components, the `qb:componentAttachment` property can be useful. It defines whether a component will be specified for each observation or for the dataset as a whole. For example, the currency attribute will be typically attached to the dataset, meaning that each observation (transaction) from the dataset uses the same currency. While missing values of dimensions are invalid in DCV, values of attributes are optional. However, there may be attributes that are necessary for correct interpretation of data, such as currency. DCV allows to define such attributes as required using the `qb:componentRequired` property. This approach can be further extended to other attributes when designing DSDs for concrete datasets. Moreover, DSDs may optionally specify the order of components via the `qb:order` property. Doing so can be useful for presentation purposes and thus, similarly to ordering columns in a table, components may be highlighted by bringing them to the front or ordered in a logical way.

The mandatory components are:

- Organization

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

- Operation character
- Date
- Amount
- Currency (attribute)

An example of an OpenBudgets.eu compliant spending DSD is:

```
obeu-dsd:Spending1 a qb:DataStructureDefinition ;
  rdfs:label "Sample spending Data Structure Definition"@en ;
  rdfs:comment "Made for D1.3 of OpenBudgets.eu"@en ;
  qb:component [
      rdfs:label "Organization"@en ;
      qb:dimension obeu-dimension:organization
  ], [
      rdfs:label "Transaction date"@en ;
      qb:dimension obeu-dimension:date
  ], [
      rdfs:label "Partner"@en ;
     qb:dimension obeu-dimension:partner
  ], [
      rdfs:label "Currency"@en ;
      qb:attribute obeu-attribute:currency ;
      qb:componentAttachment qb:DataSet ;
      qb:componentRequired true
  ], [
      rdfs:label "The monetary amount"@en ;
      qb:measure obeu-measure:amount
  ].
```

We can have a dataset which is missing a dimension value for some expenditure lines. In that case, either the data is incomplete and needs to be fixed or the component is in fact an attribute, not a dimension. Values of dimensions should together uniquely identify a measure. With a missing value of a dimension, it is not possible to uniquely identify a measure (incomplete data) or it is an attribute - additional information related to the measure, but not necessary to identify it.

## Example: European Fisheries Fund from 2007 - 2013 for the Czech Republic

We chose a dataset describing the subsidies disbursed from the European Fisheries Fund (EFF) during 2007 - 2013 in the Czech Republic. This fund supports sustainable management of fisheries across the EU member states. The source data is available in an Excel spreadsheet.

[9] We mapped a sample line of this spreadsheet to a representation using the component properties proposed in this deliverable.

The general approach that we adopted when modelling the data was to achieve a lossless conversion. The dataset contains 6 measures, 3 of which are in Czech crowns, while the other 3 represent the same monetary amounts converted to Euro by using exchange rates averaged over a year. In each group of measures in a single currency one represents the aggregated total contribution. Consequently, there are only 2 primary measures out of which the others are derived either by summation or currency conversion. Viewed this way, it suffices to capture the 2 primary measures and recompute the remaining measures. However, exact reconstruction of some source measures may not be straightforward. For instance, recomputing the exchange rate averaged over a year and applying it for currency conversion may introduce a loss of precision due to rounding. Therefore we take the source data as is and mirror it in RDF using the proposed data model. This helps to preserve the authoritative nature of the source data. The derived non-authoritative data needs to be provided with a provenance trail to be able to examine how it was created.

In order to capture all measures present in the source dataset we use currency as a dimension and devise 3 measure properties derived from the `obeu-measure:amount` core property, one for the contribution from the budget of the European Union, one for the contribution from the budget of the Czech Republic, and one for the total contribution. Note that the relationship of the measures aggregated in the total is left implicit. Additionally, we created a sub-property of `obeu-dimension:programmeClassification` to describe the classification of subsidies from the EFF. Besides these component properties payments in the dataset are indexed with properties reused from the data model's core. The reused properties include `obeu-dimension:organization`, `obeu-dimension:partner`, `obeu-dimension:project`, `obeu-dimension:currency`, and `obeu-attribute:currency`. Currency is duplicated as a dimension and an attribute. While the dimension is used because the dataset contains multiple currencies, the attribute is used to maintain compatibility with single-currency datasets. Multiple measures are distinguished by the `qb:measureType` dimension. Alternative syntactic approaches to multi-measure datasets are described in the DCV's specification.[10] To avoid duplication of data and to give each expenditure line a referenceable identity via URI we group observations for the same partner, project, and programme classification into slices.

# Conclusions

In this deliverable, we specified the core component properties of OpenBudgets.eu spending RDF data representation identified based on the survey and knowledge elicitation with domain experts presented in Deliverable D1.1. We presented means of creating additional component

[9] http://eagri.cz/public/web/mze/dotace/operacni-program-rybarstvi-na-obdobi/opatreni-osy-ii/schvalene-projekty/

[10] http://www.w3.org/TR/vocab-data-cube/#h3_dsd-mm

properties where necessary as well as means of creating data structure definitions out of these component properties. We illustrated the process with an example of the European Fisheries Fund from 2007 - 2013 for the Czech Republic.

## References


- Klímek J., Kučera J., Mynarz J., Sedmihradská L., Zbranek J.: OpenBudgets.eu - Deliverable D1.1 - Survey of modelling public spending data & Knowledge elicitation report, 2015, https://openbudgets.atlassian.net/browse/OB-12
- Tennison, J. *Payments Ontology: Reference*. Last updated 2011-03-28 [cit. 2015-10-07]. Available from WWW: https://data.gov.uk/resources/payments/reference


## Appendix: Core component properties

This appendix (`spending-components.ttl`) contains the definition of core component properties in RDF (Turtle) using the RDFS and DCV vocabularies.

## Appendix: Core code lists

This appendix (`spending-codelists.ttl`) contains a preliminary definition of core code lists in RDF (Turtle) using the RDFS and DCV vocabularies.

## Appendix: Additional properties

This appendix (`spending.ttl`) contains the additional properties describing spending data defined outside of the Data Cube Vocabulary in RDF (Turtle).

## Appendix: Example DSD and component properties

This appendix (`eu-fishing-subsidies-dsd.ttl`) contains an example DSD and component properties for the European Fisheries Fund from 2007 - 2013 for the Czech Republic in RDF (Turtle). It shows the reuse of core component properties, definition of new component properties and the creation of the data structure definition for a given dataset.

## Appendix: Sample data

This appendix (`eu-fishing-subsidies-data.ttl`) contains sample data in RDF (Turtle) using the example data structure definition from Appendix: Example DSD and component properties.