

GUIDELINES FOR TCS

Spatio-Temporal Metadata Mapping

March 12, 2019

TABLE OF CONTENTS

1 INTRODUCTION	3
2 TEMPORAL METADATA.....	4
2.1 TEMPORAL INFORMATION FOR DDSS DISCOVERY	4
2.2 TEMPORAL INFORMATION FOR DDSS QUERY	5
3 SPATIAL METADATA	6
3.1 SPATIAL INFORMATION FOR DDSS DISCOVERY	6
3.2 SPATIAL METADATA FOR DDSS QUERY	7
4 METADATA QUALITY	12
5 CONTACTS AND REFERENCES.....	14

1 Introduction

This document is a guideline to help Thematic Core Services (TCS) communities to provide the spatio-temporal information through EPOS-DCAT-AP metadata standard extension.

This information will allow Integrated Core Services Metadata Team to:

1. Discover Dataset, Data products, Software and Services (DDSS) by spatial-temporal extent;
2. Query DDSS services using spatio-temporal parameters.

Section 2 describes in detail how to include temporal information and provides examples; likewise, Section 3 describes in detail how to describe spatial metadata information (with examples as well); Section 4 focuses on the provision of quality metadata and provides recommendation about how to fill the most relevant metadata elements; in Section 5, contacts for getting support together with references for further study, are provided.

All concepts, entities and attributes mentioned in this document are described and explained in detail in the EPOS-DCAT-AP model description, available in a preliminary form at https://github.com/epos-eu/EPOS-DCAT-AP/blob/EPOS-DCAT-AP-shapes/docs/EPOS-DCAT-AP_extension_v0.1%20-%202019.03.11.pdf.

2 Temporal metadata

This section shows how and where to enter the temporal information about DDSS in EPOS-DCAT-AP metadata description.

2.1 Temporal information for DDSS discovery

The temporal coverage is specified by using `dct:temporal` property in **Dataset** and **WebService** entities. In particular, this is done by using `schema:startDate` and `schema:endDate`. The date format to be used for these elements is "YYYY-MM-DDThh:mm:ssZ", as shown in the following examples.

DATASET example

```
dct:temporal [ a dct:PeriodOfTime ;  
               schema:startDate "1992-03-23T02:15:00Z"^^xsd:dateTime ;  
               schema:endDate   "2010-12-20T23:00:00Z"^^xsd:dateTime ;  
             ] ;
```

WEBSERVICE example

```
dct:temporal [ a dct:PeriodOfTime ;  
               schema:startDate "1992-03-23T02:15:00Z"^^xsd:dateTime ;  
               schema:endDate   "2010-12-20T23:00:00Z"^^xsd:dateTime ;  
             ] ;
```

Please note that `schema:endDate` may be omitted if Dataset/Webservice is continuously updated.

2.2 Temporal information for DDSS query

In order to manage, in the appropriate way, the web service parameters that allow to filter DDSS by temporal range, it is necessary to add semantic information in the **Operation** entity. This is done by adding **hydra:property** and **schema:valuePattern** elements to the appropriate parameters.

Metadata element	Description	Allowed values
hydra:property	It contains the vocabulary term which indicates the semantic description of parameter.	<ul style="list-style-type: none"> - "schema:startDate", if the parameter represents the start date; - "schema:endDate", if the parameter represents the end date;
schema:valuePattern	It contains the regular expression for testing values according to the parameter's specification.	E.g. "YYYY-MM-DD"; "YYYY-MM-DDThh:mm:ss"; "YYYY"; Please refer to the information reported in the spreadsheet: https://docs.google.com/spreadsheets/d/1n-jQVuzW0jc3xwAKl8DE3rePxxZgfhm7DtzuMByDm8Y/edit?usp=sharing

The following example shows how to use these elements in order to describe the web service parameters used to filter data by temporal range.

OPERATION Example

```
hydra:mapping[ a hydra:IriTemplateMapping;
  hydra:variable "epoch_start";
  hydra:property "schema:startDate";
  schema:valuePattern "YYYY-MM-DD";
  ...
];
```

```
hydra:mapping[ a hydra:IriTemplateMapping;
  hydra:variable "epoch_end";
  hydra:property "schema:endDate";
  schema:valuePattern "YYYY-MM-DD";
  ...
];
```

3 Spatial metadata

This section shows how and where to enter the spatial information in EPOS-DCAT-AP metadata description.

3.1 Spatial information for DDSS discovery

The spatial coverage is specified by using `dct:spatial` property in **Dataset** and **WebService** entities. This property allows to specify the geographic region covered by the Dataset/WebService by using one or more geometries denoted by `locn:geometry`. It is important to specify three mandatory items: coordinates, geometry type and coordinate reference system (CRS).

- The coordinates represent coordinates of the geographic area covered by the Dataset/WebService;
- geometry type is the type of geometry that characterizes the spatial object of the Dataset/WebService (e.g., polygon, point);
- CRS is the spatial reference system in which the data are represented.

The syntax encoding used in EPOS-DCAT-AP to describe geometries is Well-known text (WKT). In the WKT representation, CRS84 (<http://www.opengis.net/def/crs/OGC/1.3/CRS84>) is used as a default CRS and it shall be assumed as CRS when it is not explicitly specified. It denotes WGS84 with the order longitude, latitude.

The following examples show how to specify the spatial coverage by using WKT geometry encoding to represent more polygons and default CRS (CRS84).

DATASET example

```
dct:spatial [ a dct:Location ;

    locn:geometry "POLYGON((3.053 47.975, 7.24 47.975, 7.24
53.504, 3.053 53.504, 3.053 47.975))"^^gsp:wktLiteral ;
    locn:geometry "POLYGON((-72.2679021 42.9300036, -72.2830504
42.9263287, -72.2806247 42.9365225, -72.2679021
42.9300036))"^^gsp:wktLiteral ;

];
```

WEBSERVICE example

```
dct:spatial [ a dct:Location ;

    locn:geometry "POLYGON((3.053 47.975, 7.24 47.975, 7.24
53.504, 3.053 53.504, 3.053 47.975))"^^gsp:wktLiteral ;
    locn:geometry "POLYGON((-72.2679021 42.9300036, -72.2830504
42.9263287, -72.2806247 42.9365225, -72.2679021
42.9300036))"^^gsp:wktLiteral ;

];
```

Please note that the CRS must be always specified when it is different from WGS 84 (EPSG:4326).

The following example shows how to specify the spatial coverage by using WKT geometry encoding to represent points and explicit CRS=EPSG:23030.

```
dct:spatial [ a dct:Location ;
    locn:geometry "<http://www.opengis.net/def/crs/EPSSG/0/23030>
POINT((439930.8579 4475096.6375))"^^gsp:wktLiteral ;
];
```

3.2 Spatial metadata for DDSS query

In order to manage, in the appropriate way, the web service parameters that allow to query DDSS by spatial bounding box, it is necessary to add semantic information in the **Operation** entity. This is done by adding **hydra:property** element to the appropriate parameters, as already explained for the temporal information (Section 2).

Metadata element	Description	Allowed values
hydra:property	It contains the vocabulary term which indicates the semantic description of parameter.	<ul style="list-style-type: none"> - epos:southernmostLatitude: refers to the lower bound (min) latitude; - epos:northernmostLatitude: refers to the upper bound (max) latitude; - epos:westernmostLongitude: refers to the lower bound (min) longitude;

		- epos:easternmostLongitude: refers to the upper bound (max) longitude;
--	--	--

Currently, the web services implement the spatial query by adopting two different approaches:

- A.** by using four different parameters which represent south latitude, north latitude, west longitude, east longitude (e.g., minlatitude, maxlatitude, minlongitude, maxlongitude);
- B.** by using a single parameter which contains a list with south latitude, north latitude, west longitude, east longitude listed in a customized order (e.g., bbox=minlat,minlon,maxlat,maxlon).

The web services which adopt the **approach A** only need to add **hydra:property** element for each spatial parameter as shown in the following example.

OPERATION example – approach A

```
<example/Operation/Station> a hydra:Operation;
  hydra:method "GET";
  hydra:returns "application/xml";

  hydra:property [ a hydra:IriTemplate;
    hydra:template
"http://webservices.ingv.it/fdsnws/station/1/query{?starttime,
endtime, network, station, location, channel, minlatitude,
maxlatitude, minlongitude, maxlongitude, level,
nodata}"^^xsd:string;

    hydra:mapping[ a hydra:IriTemplateMapping;
      hydra:variable "minlatitude"^^xsd:string;
      hydra:property "epos:southernmostLatitude";
      rdfs:label "Minimum Latitude";
      schema:minValue "-90.0";
      schema:maxValue "90.0";
      hydra:required "true"^^xsd:boolean;
      schema:defaultValue "36.61" ;
    ];
    hydra:mapping[ a hydra:IriTemplateMapping;
      hydra:variable "maxlatitude"^^xsd:string;
      hydra:property "epos:northernmostLatitude";
      rdfs:label "Maximum Latitude";
      schema:minValue "-90.0";
```



```

    schema:maxValue "90.0";
    hydra:required "true"^^xsd:boolean;
    schema:defaultValue "47.11" ;
  ];

  hydra:mapping[ a hydra:IriTemplateMapping;
    hydra:variable "minlongitude"^^xsd:string;
    hydra:property "epos:westernmostLongitude";
    rdfs:label "Minimum Longitude";
    schema:minValue "-180.0";
    schema:maxValue "180.0";
    hydra:required "true"^^xsd:boolean;
    schema:defaultValue "6.74" ;
  ];

  hydra:mapping[ a hydra:IriTemplateMapping;
    hydra:variable "maxlongitude"^^xsd:string;
    hydra:property "epos:easternmostLongitude";
    rdfs:label "Maximum Longitude";
    schema:minValue "-180.0";
    schema:maxValue "180.0";
    hydra:required "true"^^xsd:boolean;
    schema:defaultValue "18.48" ;
  ];

];
.

```

The web services which adopt the **approach B** need to:

1. **modify** the URI template by specifying how the single parameter accepts the geographic coordinates (i.e., north, south, west, east).

It means, looking at the snippet below, that the single parameter (e.g., called `bbox`) shall be shifted out of the curly brackets; then, the geographic coordinates (i.e., north, south, west, east) shall be added to the URI Template using the order with which they are requested by the web service (e.g., {minlatitude, maxlatitude, minlongitude, maxlongitude}). Note that the name of these parameters (`hydra:variable`) can be chosen arbitrarily because the semantic, added by using `hydra:property`, allows to manage them in the appropriate way.

```
hydra:template "https://catalog.terradue.com/gep-epos/search{?
format, pt, start, stop, bbox, geom, track}"^^xsd:string;
```

```
hydra:template "https://catalog.terradue.com/gep-epos/search{?
format, pt, start, stop, geom, track}&bbox={minlatitude, maxlatitude,
minlongitude, maxlongitude}"^^xsd:string;
```

2. **remove** the description of the single parameter (e.g., bbox);

```
hydra:mapping[ a
hydra:IriTemplateMapping,
  hydra:variable "bbox"^^xsd:string;
  rdfs:range "xsd:string";
  rdfs:label "Bounding box";
  hydra:required
  "false"^^xsd:boolean;
];
```

3. **add** the description for all spatial parameters defined in the URI template (see point 1).

The following example shows how to modify the metadata description according to the B approach.

OPERATION example – B approach

```
<example/Operation/Station> a hydra:Operation;
  hydra:method "GET"^^xsd:string;
  hydra:returns "application/atom+xml";
  hydra:property[ a hydra:IriTemplate;
```

```
    hydra:template "https://catalog.terradue.com/gep-epos/search{?
```

```
format, pt, start, stop, geom, track}&bbox={minlatitude,
maxlatitude, minlongitude, maxlongitude} "^^xsd:string;
```

```
hydra:mapping[ a hydra:IriTemplateMapping;
  hydra:variable "minlatitude"^^xsd:string;
  hydra:property "epos:southernmostLatitude";
  rdfs:label "Minimum Latitude";
  schema:minValue "-90.0";
  schema:maxValue "90.0";
  hydra:required "true"^^xsd:boolean;
  schema:defaultValue "36.61" ;
];
hydra:mapping[ a hydra:IriTemplateMapping;
  hydra:variable "maxlatitude"^^xsd:string;
  hydra:property "epos:northernmostLatitude";
  rdfs:label "Maximum Latitude";
  schema:minValue "-90.0";
  schema:maxValue "90.0";
  hydra:required "true"^^xsd:boolean;
  schema:defaultValue "47.11" ;
];
hydra:mapping[ a hydra:IriTemplateMapping;
  hydra:variable "minlongitude"^^xsd:string;
  hydra:property "epos:westernmostLongitude";
  rdfs:label "Minimum Longitude";
  schema:minValue "-180.0";
  schema:maxValue "180.0";
  hydra:required "true"^^xsd:boolean;
  schema:defaultValue "6.74" ;
];
hydra:mapping[ a hydra:IriTemplateMapping;
  hydra:variable "maxlongitude"^^xsd:string;
  hydra:property "epos:easternmostLongitude";
  rdfs:label "Maximum Longitude";
  schema:minValue "-180.0";
  schema:maxValue "180.0";
  hydra:required "true"^^xsd:boolean;
  schema:defaultValue "18.48" ;
];
];
```

Please note that at this stage only web services that support the spatial query by using **rectangle** will be considered and integrated into ICS search. Other services will be included at a later stage, but before the end of EPOS-IP.

4 Metadata Quality

This section contains a list of metadata element that play a crucial role in the DDSS discovery phase. It is important to fill in the information about DDSS accurately and with all required details.

Dataset entity

- **dct:title** "It contains a name given to the Dataset";
- **dct:description** "It contains a free-text description of the Dataset";
- **adms:identifier** "It contains the DDSS-ID according to the master table";
- **dct:spatial** "It refers to a geographical area covered by the Dataset";
- **dct:temporal** "It refers to a temporal period that the Dataset covers";
- **dcat:keyword** "It contains the keywords used to describe the Dataset";

Distribution entity

- **dct:title** "It contains a name given to the Distribution";
- **dct:description** "It contains a free-text account of the Distribution";
- **dct:license** "It refers to the license under which the Distribution is made available";

WebService entity

- **schema:name** "It contains a name given to the Web Service";
- **schema:description** "It contains a free-text description of the Web Service";

- **schema:keywords** "It contains the keywords used to describe the Web Service";
- **dct:spatial** "It refers to a geographical area covered by the Web Service";
- **dct:temporal** "It refers to a temporal period that the Web Service covers";

Operation entity

- **rdfs:label** "It contains a short string used to describe the meaning of the parameter";
- **hydra:returns** "It is used to specify the output format of the Operation. The possible values are listed here: <https://www.iana.org/assignments/media-types/media-types.xhtml>";

5 Contacts and references

Contacts:

- **Rossana Paciello**, rossana.paciello@ingv.it, for Metadata description issues;
- **Jan Michalek**, jan.michalek@uib.no, for DDSS-ID and other issue.

References:

- **EPOS-DCAT-AP documentation:** https://github.com/epos-eu/EPOS-DCAT-AP/blob/EPOS-DCAT-AP-shapes/docs/EPOS-DCAT-AP_extension_v0.1%20-%202019.03.11.pdf
- **EPOS-DCAT-AP RDF/Turtle example:** https://github.com/epos-eu/EPOS-DCAT-AP/blob/EPOS-DCAT-AP-shapes/examples/EPOS-DCAT-AP_example.ttl
- **SPATIO-TEMPORAL information by TCS:** <https://docs.google.com/spreadsheets/d/1n-jQVuzW0jc3xwAKl8DE3rePxhZgfHm7DtzuMByDm8Y/edit?usp=sharing>