
Catalogue Services Lite

Jan Růžička¹, Michal Šeliga²

¹Institute of geoinformatics, VSB-TU of Ostrava, 17. listopadu 15,
708 33, Ostrava – Poruba, Czech Republic
jan.ruzicka@vsb.cz

²T-MAPY spol. s r.o., Vědeckotechnologický park Ostrava, a. s., Technologická 372/2,
708 00 Ostrava Pustkovec, Czech Republic
mise@tmapy.cz

Abstract. We have discovered that a very important part of the architecture for GeoWeb services orchestration is a catalogue. It does not matter what kind of a catalogue is used and what type of technology is used for metadata storage. The most important part of the catalogue are two items. The first is a used standard for the description of GeoWeb services and the second one is a standard used for building a catalogue's interface. We spent a lot of time analysing two possible standards for building the catalogue's interface. The first standard named UDDI (Universal Description Discovery and Integration) has shown itself as not flexible enough to hold necessary geo-based attributes in communication with the catalogue. The second one, named CSW (Catalogue Services of Web), has shown itself as too complicated for implementation. We have decided to use CSW as a base for building a less-complicated standard for the catalogue used in the architecture for orchestration. The result of our research is called CSW Lite and the paper should describe this standard in a brief, but complex way. CSW Lite should be easy to implement and should have all the necessary parts for the architecture of orchestration. CSW Lite should be compatible with complete CSW specification so as to allow a general CSW client to use it in a process of an orchestration.

Keywords: catalogue, CSW Lite, CSW-L, GeoWeb

1 Introduction

The company that would like to have a CSW compatible catalogue has three possible ways of how to do this. In the commercial sphere, there are several systems that implement CSW specification. Most of them are part of general software solutions that are very expensive. There are only two known system in the area of open source software (GeoNetwork Open Source and Deegree). The third way is based on one's own CSW implementation. All of the options bring challenges. The reason why there are not enough implementations in this area (or they are costly) and why the last way is very difficult for a typical organization is the complexity of the CSW specification. CSW Lite (CSW-L) is a reaction to a quite unnoticed situation in the area of catalogue services for geodata and geoweb services. CSW-L should be a simple specification that could be implemented by a single programmer with basic knowledge of the standards and technologies. CSW-L is compatible with the full CSW 2.0.2 specification and it uses only a minimal set of capabilities of CSW 2.0.2. This simplification should make the CSW more understandable to a general developer.

2 Description of General CSW-L Capabilities

Metadata records are only based on Dublin Core (ISO 15936:2009). This could seem to be limited for searching, but according to research from the project and several other previous projects, the Dublin Core elements can be used in an efficient way. These parts are specified in CSW specification as Core returnable (query-able) elements.

Communication is based only on HTTP protocol. To make this specification simple enough, other protocols are not supported.

Supported operations are basic methods for catalogue discovery (GetCapabilities, GetRecords, DescribeRecord and GetRecordById). A distributed search is not supported.

3 GetCapabilities Operation

The GetCapabilities operation [OGC 2009a, page 131] is used as the initial handshake between a client and a service. The service responds with a document that contains a description of the service capabilities (supported operations, formats, etc.).

3.1 Request

The GetCapabilities operation is requested via the HTTP GET method with three encoded parameters using KVP (Key-Value Pairs). Values of the parameters are fixed as is shown in the following example.

```
http://gis.vsb.cz/catalogue/csw1?
REQUEST=GetCapabilities&SERVICE=CSW&VERSION=2.0.2
```

3.2 Response

The response is an XML formatted document as is described in the following example. Values marked by -- should be modified, the others should stay untouched. This document specifies which part of the CSW capabilities must be implemented.

```
<?xml version="1.0" encoding="UTF-8"?>
<ows:Capabilities version="2.0.2" updateSequence="0"
xmlns:ows="http://www.opengis.net/ows"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:csw="http://www.opengis.net/cat/csw/2.0.2"
xmlns:xlink="http://www.w3.org/1999/xlink">
  <ows:ServiceIdentification>
    <ows:ServiceType>CSW</ows:ServiceType>
    <ows:ServiceTypeVersion>2.0.2</ows:ServiceTypeVersion>
    <ows:Title>--CSW Lite example--</ows:Title>
    <ows:Abstract>--This is example of CSW Lite specification--
  </ows:Abstract>
    <ows:Keywords>
      <ows:Keyword>--CSW Lite--</ows:Keyword>
      <ows:Keyword>--GeoWeb--</ows:Keyword>
    </ows:Keywords>
    <ows:Fees>NONE</ows:Fees>
    <ows:AccessConstraints>NONE</ows:AccessConstraints>
  </ows:ServiceIdentification>
  <ows:ServiceProvider>
    <ows:ProviderName>--VSB-TUO--</ows:ProviderName>
    <ows:ProviderSite xlink:href="--http://gis.vsb.cz--"/>
    <ows:ServiceContact>
      <ows:IndividualName>--Jan Ruzicka--</ows:IndividualName>
      <ows:PositionName>--Lecturer--</ows:PositionName>
      <ows:ContactInfo>
        <ows:Address>
          <ows:ElectronicMailAddress>--jan.ruzicka@vsb.cz--
        </ows:ElectronicMailAddress>
        </ows:Address>
        <ows:OnlineResource
xlink:href="mailto:--jan.ruzicka@vsb.cz--"/>
        </ows:ContactInfo>
      </ows:ServiceContact>
    </ows:ServiceProvider>
  <ows:OperationsMetadata>
    <ows:Operation name="GetCapabilities">
      <ows:DCP>
        <ows:HTTP>
```

```

        <ows:Get
xlink:href="http://gis.vsb.cz/catalogue/cswl?"/>
        </ows:HTTP>
    </ows:DCP>
</ows:Operation>
<ows:Operation name="DescribeRecord">
    <ows:DCP>
        <ows:HTTP>
            <ows:Post
xlink:href="http://gis.vsb.cz/catalogue/cswl?"/>
            </ows:HTTP>
        </ows:DCP>
        <ows:Parameter name="typeName">
            <ows:Value>csw:Record</ows:Value>
        </ows:Parameter>
        <ows:Parameter name="outputFormat">
            <ows:Value>application/xml</ows:Value>
        </ows:Parameter>
        <ows:Parameter name="schemaLanguage">
            <ows:Value>XMLSCHEMA</ows:Value>
        </ows:Parameter>
    </ows:Operation>
<ows:Operation name="GetRecords">
    <ows:DCP>
        <ows:HTTP>
            <ows:Post
xlink:href="http://gis.vsb.cz/catalogue/cswl?"/>
            </ows:HTTP>
        </ows:DCP>
        <ows:Parameter name="TypeName">
            <ows:Value>csw:Record</ows:Value>
        </ows:Parameter>
        <ows:Parameter name="outputFormat">
            <ows:Value>application/xml</ows:Value>
        </ows:Parameter>
        <ows:Parameter name="outputSchema">
<ows:Value>http://www.opengis.net/cat/csw/2.0.2</ows:DefaultValue>
        </ows:Parameter>
        <ows:Parameter name="resultType">
            <ows:Value>hits</ows:DefaultValue>
            <ows:Value>results</ows:Value>
        </ows:Parameter>
        <ows:Parameter name="ElementSetName">
            <ows:Value>brief</ows:Value>
            <ows:Value>summary</ows:Value>
            <ows:Value>full</ows:Value>
        </ows:Parameter>
        <ows:Parameter name="CONSTRAINTLANGUAGE">
            <ows:Value>Filter</ows:Value>
        </ows:Parameter>
    </ows:Operation>
<ows:Operation name="GetRecordById">
    <ows:DCP>
        <ows:HTTP>
            <ows:Get
xlink:href="http://gis.vsb.cz/catalogue/cswl?"/>
            </ows:HTTP>
        </ows:DCP>
        <ows:Parameter name="ElementSetName">
            <ows:Value>brief</ows:Value>
            <ows:Value>summary</ows:Value>

```

```

    <ows:Value>full</ows:Value>
  </ows:Parameter>
</ows:Operation>
</ows:OperationsMetadata>
<ogc:Filter_Capabilities>
  <ogc:Spatial_Capabilities>
    <ogc:Spatial_Operators>
      <ogc:BBOX/>
    </ogc:Spatial_Operators>
  </ogc:Spatial_Capabilities>
  <ogc:Scalar_Capabilities>
    <ogc:Logical_Operators/>
    <ogc:Comparison_Operators>
      <ogc:Simple_Comparisons/>
      <ogc:Like/>
    </ogc:Comparison_Operators>
  </ogc:Scalar_Capabilities>
</ogc:Filter_Capabilities>
</csw:Capabilities>

```

4 DescribeRecord Operation

The DescribeRecord operation [OGC 2009a, page 134] returns the description of a metadata record returned in a response to GetRecords or GetRecordById operations.

4.1 Request

The request is XML encoded and transferred via the HTTP POST method. The request is encoded as is described in the following example.

```

<?xml version="1.0" encoding="UTF-8"?>
<DescribeRecord service="CSW" version="2.0.2"
outputFormat="application/xml"
  schemaLanguage="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.opengis.net/cat/csw/2.0.2">
  <TypeName>csw:Record</TypeName>
</DescribeRecord>

```

4.2 Response

The response for this operation is an XML schema that describes (specifies) the structure of the metadata record. The CSW-L compatible service must return document available at <http://schemas.opengis.net/csw/2.0.2/record.xsd> [OGC 2009b].

5 GetRecords Operation

The GetRecords operation [OGC 2009a, page 142] returns metadata records according to specified conditions in a request.

5.1 Request

The request is XML encoded and delivered via the HTTP POST method. The request consists of four parts:

- the GetRecords element
- the Query element

- the ElementSetName element
- the Constraint element

The structure of a request is described in the following example.

```
<?xml version="1.0" encoding="UTF-8"?>
<GetRecords
  service="CSW"
  version="2.0.2"
  maxRecords="5"
  startPosition="1"
  resultType="results"
  xmlns="http://www.opengis.net/cat/csw/2.0.2"
  xmlns:csw="http://www.opengis.net/cat/csw/2.0.2"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:gml="http://www.opengis.net/gml">
  <Query typeNames="csw:Record">
    <ElementSetName typeNames="csw:Record">full</ElementSetName>
    <Constraint version="1.1.0">
      <ogc:Filter>
        <ogc:And>
          <ogc:PropertyIsLike escapeChar="\ " singleChar="?"
wildCard="*">
            <ogc:PropertyName>dc:title</ogc:PropertyName>
            <ogc:Literal>*Elevation*</ogc:Literal>
          </ogc:PropertyIsLike>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>dc:type</ogc:PropertyName>
            <ogc:Literal>Service</ogc:Literal>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsGreaterThanOrEqualTo>
            <ogc:PropertyName>dct:modified</ogc:PropertyName>
            <ogc:Literal>2004-03-01</ogc:Literal>
          </ogc:PropertyIsGreaterThanOrEqualTo>
          <ogc:BBOX>
            <ogc:PropertyName>ows:BoundingBox</ogc:PropertyName>
            <gml:Envelope>
              <gml:lowerCorner>14.05 46.46</gml:lowerCorner>
              <gml:upperCorner>17.24 48.42</gml:upperCorner>
            </gml:Envelope>
          </ogc:BBOX>
        </ogc:And>
      </ogc:Filter>
    </Constraint>
  </Query>
</GetRecords>
```

The GetRecords Element

The CSW-L compatible service must implement five attributes that are described in the following table.

Table 1. Attributes for the GetRecords element

Attribute	Description
service	Fixed value "CSW"
version	Fixed value "2.0.2"
maxRecords	Maximal count of returned records. Default is 10. 0 means resultType = hits.
startPosition	Index of the first returned record. Starts at 1.
resultType	Hits or results. Option hits must return an empty result set with an identified number of records.

The Query Element

The Query element is fixed as is shown in the previous example.

The ElementSetName Element

The element ElementSetName contains the fixed attribute typeNames as is shown in the previous example. There are three options of the possible value; they are described in the following table.

Table 2. Values for the ElementSetName element

Value	Description
full	Returns a full set of elements for each record in the response.
summary	Returns a selected set of elements described in the Response subchapter for each record in the response.
brief	Returns a selected set of elements described in the Response subchapter for each record in the response.

The Constraint Element

The Constraint element is based on the OGC Filter Implementation Specification [OGC 2009a, page 152]. It has one fixed attribute and one fixed element ogc:Filter. Specified conditions for searching in a catalogue are found inside ogc:Filter. The structure is described in the previous example. The CSW-L compatible service must support operators described in the following table.

Table 3. Mandatory constraint operators

	Description
And	Conditions can be in a logical And relationship.
Or	Conditions can be in a logical Or relationship.
Not	Conditions can be negated.
PropertyIsEqualTo	Simple equal (==) comparison.
PropertyIsNotEqualTo	Simple not-equal (!=) comparison.
PropertyIsLessThan	Simple less than (<) comparison.
PropertyIsGreaterThan	Simple greater than (>) comparison.
PropertyIsLessThanOrEqualTo	Simple less than or equal to (<=) comparison.
PropertyIsGreaterThanOrEqualTo	Simple greater than or equal to (>=) comparison.
PropertyIsLike	Like operator including information for: <ul style="list-style-type: none"> ● escaping special characters such as quotes (attribute escapeChar), ● wild-card for one letter (attribute singleChar), ● wild-card for more than one letter (attribute wildCard).
BBOX	Spatial operator that selects all overlapping envelopes.

Properties can be specified from all metadata elements listed in the Response subchapter.

5.2 Response

The response structure [OGC 2009a, page 120] depends on two parameters in the request resultType and ElementSetName. When the requested resultType is hit, only then a count of records is returned. When the requested resultType is results, then metadata records are returned.

Structure of each record depends on the ElementSetName parameter. The richest record is returned when the value is full.

Response on hits

Response on resultType=hits returns basic information about the search result, but without any records included. The structure of the response is described in the following example.

```
<?xml version="1.0" encoding="UTF-8"?>
<csw:GetRecordsResponse xmlns="http://www.opengis.net/cat/csw/2.0.2"
  xmlns:csw="http://www.opengis.net/cat/csw/2.0.2">
  <csw:SearchStatus timestamp="2009-10-04T09:05:25"/>
  <csw:SearchResults
    nextRecord="0"
    numberOfRecordsMatched="5"
    numberOfRecordsReturned="5"
  />
</csw:GetRecordsResponse>
```

All elements and attributes included in the example are mandatory. The following table describes attributes.

Table 4. Attributes for response on hits

Value	Description
timestamp	Date and time when the response was created.
nextRecord	Index of the first record from the search result that was not included in the response because of some limits (data size) of the communication. Zero means that all records have been returned in the response.
numberOfRecordsMatched	Number of records that have been matched by specified search criteria.
numberOfRecordsReturned	Number of records that have been returned in the response.

Response on full

Response on resultType=results and ElementSetName=full is the same as response on resultType=hits described above, but it includes records (elements Record) in the SearchResults element. The structure of the Record element is described in the following example. All elements are mandatory. If the information for the element is not available, it must be filled with NULL. Elements dc:subject, dc:contributor, dc:relation, dc:source, and dc:publisher can occur more than once.

```
<csw:Record
  xmlns:csw="http://www.opengis.net/cat/csw/2.0.2"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dct="http://purl.org/dc/terms/"
  xmlns:ows="http://www.opengis.net/ows"
  >
  <dc:creator>VSB-TUO</dc:creator>
  <dc:contributor>T-Mapy</dc:contributor>
  <dc:publisher>VSB-TUO</dc:publisher>
  <dc:subject>erosion</dc:subject>
  <dct:abstract>Dataset that describes erosion risk in the Czech
republic</dct:abstract>
  <dc:identifier>http://gis.vsb.cz/datasets/1567890</dc:identifier>
  <dc:relation>contact:mailto:jan.ruzicka@vsb.cz</dc:relation>
  <dc:source>http://gis.vsb.cz/datasets/0015600</dc:source>
  <dc:rights>free to use</dc:rights>
```

```
<dc:type>Dataset</dc:type>
<dc:title>Erosion risk in the Czech republic</dc:title>
<dct:modified>2006-05-01</dct:modified>
<dc:language>en</dc:language>
<ows:BoundingBox>
  <ows:LowerCorner>12.10 48.27</ows:LowerCorner>
  <ows:UpperCorner>18:87 51.32</ows:UpperCorner>
</ows:BoundingBox>
</csw:Record>
```

Response on summary

Response on resultType=results and ElementSetName=summary is the same as response on ElementSetName=full described above, but the record element is named SummaryRecord and it includes only selected metadata items (dc:identifier, dc:title, dc:type, dc:subject, dc:format, dc:relation, dct:modified, dct:abstract, and ows:BoundingBox).

Response on brief

Response on resultType=results and ElementSetName=brief is the same as response on ElementSetName=full described above, but the record element is named BriefRecord and it includes only selected metadata items (dc:identifier, dc:title, dc:type, and ows:BoundingBox).

6 GetRecordById Operation

The GetRecordById operation [OGC 2009a, page 160] is KVP encoded and requested via HTTP GET method. It returns a record (records) according to specified id (ids). The same possibility exists to request a full, summary, or brief record from the catalogue.

6.1 Request

```
http://gis.vsb.cz/catalogue/csw1?
REQUEST=GetRecordById&SERVICE=CSW&VERSION=2.0.2&ElementSetName=full&id=
http://gis.vsb.cz/datasets/0015600,http://gis.vsb.cz/datasets/0115600
```

Parameters REQUEST, SERVICE, and VERSION are mandatory and fixed. The parameter ElementSetName is optional (default is summary). Parameter Id can contain a comma to separate ids and is mandatory.

6.2 Response

The response is the same as in the GetRecords operation.

7 Description of Services

The CSW-L specification comes from the GeoWeb services orchestration project. To make CSW-L specifications more clear in a context of GeoWeb services orchestration, it should include information on how to describe specific information about services using Dublin Core.

7.1 Service type

One of the most important issues that is connected with service description is the identification of the service type. When searching in the catalogue, a user (client) must have the possibility to find similar services based not only on service interface (WMS), but on the data used as well.

For purposes of an orchestration, it is necessary to be able find similar services. This can be managed through several ways. Some of them are described in [Růžička 2008]. The most suitable seems to be

the concept based on ontology, where similar services are described as related terms in one branch of the ontology. The service type consists of interface definition, used data, and used algorithms. The service type is defined by the identifier in the thesaurus. The identifier is constructed from parts of the service description to make the id more human readable. Examples of two identifiers follow.

```
ogc:WMS:1.1.1/datasets:ZABAGED,DMÚ200/algorithms:simplify
gis.vsb.cz:erosion:0.0.1/datasets:FreeGEOCZ/algorithms:RUSLE
```

What is still not defined is the dependency of the similarity on a context of a provided process (running orchestra). This should be defined later and included in the CSW-L specification.

7.2 Monitoring

For purposes of an orchestration, there are very important measures in service monitoring [Kaszper 2008]. This information can be directly written into `csw:Record`, but it could make `csw:Record` too complex. That is why this kind of information is only referenced via the `dc:relation` element. One of the `dc:relation` elements in `csw:Record` should (necessary for dynamic orchestration) include a reference to an XML document (static or dynamic). The reference must be a URI that starts with a monitoring prefix and continues with the document's URL. The two following examples describe how the reference must be constructed.

```
<dc:relation>monitoring:http://gis.vsb.cz/services/monitoring/0015647.xml</
dc:relation>
<dc:relation>monitoring:http://gis.vsb.cz/catalogue/monitor?
id=0015647</dc:relation>
```

The monitoring document should contain results from monitoring. We have specified three monitoring parameters that should be included, but others can be added to the results. An example of the XML file that describes results from monitoring follows.

```
<?xml version="1.0" encoding="UTF-8"?>
<Service xmlns="http://gis.vsb.cz/monitoring/0.0.1"
id="http://gis.vsb.cz/services/0015647">
  <stability>0.95</stability>
  <lastState>ssOK</lastState>
  <speed>1000</speed>
</Service>
```

The `stability` element contains a float value that describes the long time stability of the service measured in percentage of valid responses according to all valid requests.

The `lastState` element describes the state of the service from the last monitoring. The possible values are: `ssOK`, `ssWarning`, `ssCritical`, and `ssUnknown`.

The `speed` element contains the average time spent on one request-response operation. It is measured in milliseconds.

7.3 Full example

The following example describes how a service description should look using DublinCore. There are five `dc:relation` elements, one for contact, one for monitoring, and three for binding (where a service is accessible) information.

```
<csw:Record
  xmlns:csw="http://www.opengis.net/cat/csw/2.0.2"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dct="http://purl.org/dc/terms/"
  xmlns:ows="http://www.opengis.net/ows"
  >
  <dc:creator>VSB-TUO</dc:creator>
  <dc:contributor>T-Mapy</dc:contributor>
```

```

    <dc:publisher>VSB-TUO</dc:publisher>
    <dc:subject>erosion</dc:subject>
    <dct:abstract>Service that returns erosion risk in a specified
location</dct:abstract>
    <dc:identifier>http://gis.vsb.cz/services/1567890</dc:identifier>
    <dc:relation>contact:mailto:jan.ruzicka@vsb.cz</dc:relation>
    <dc:relation>monitoring:http://gis.vsb.cz/catalogue/monitor?
id=0015647</dc:relation>
    <dc:relation>binding:http://gis.vsb.cz/services/http/erosion</
dc:relation>
    <dc:relation>binding:soap:http://gis.vsb.cz/services/soap/erosion?
wsdl</dc:relation>
    <dc:relation>binding:rest:http://gis.vsb.cz/services/rest/erosion</d
c:relation>
    <dc:source>NULL</dc:source>
    <dc:rights>free to use</dc:rights>
    <dc:type>Service/gis.vsb.cz:erosion:0.0.1/datasets:FreeGEOCZ/algori
thms:RUSLE</dc:type>
    <dc:title>Service for erosion risk in the Czech republic</dc:title>
    <dct:modified>2006-05-01</dct:modified>
    <dc:language>en</dc:language>
    <ows:BoundingBox>
        <ows:LowerCorner>12.10 48.27</ows:LowerCorner>
        <ows:UpperCorner>18:87 51.32</ows:UpperCorner>
    </ows:BoundingBox>
</csw:Record>

```

8 Conclusion

The prepared CSW-L specification should be simple enough to be easily implemented by any developer. There are still some open problems with the specification, mainly in the area of a service description using Dublin Core, but this first proposal of the CSW-L specification should be discussed in 2010 and developed into a better solution. CSW-L will be proved in a real situation when the dynamic orchestration will be used. This testing should bring results that can improve the specification.

References

- [Kaszper 2008]: Kaszper, R. (2008) *Sledování stavu služeb pro GeoWeb*. Diploma thesis. VŠB-TUO, Ostrava. 2008.
- [OGC 2009a]: OGC. http://portal.opengeospatial.org/files/?artifact_id=20555, CSW 2.0.2.
- [OGC 2009b]: OGC. <http://schemas.opengis.net/csw/2.0.2/>.
- [Růžička 2008]: Růžička, J. (2008a) *ISO 19115 for GeoWeb services orchestration. Proceedings from Geoinformatics on CTU 2008*. CTU, Prague. Available at: http://geoinformatics.fsv.cvut.cz/wiki/index.php/ISO_19115_for_GeoWeb_services_orchestration

Support

The article is supported by the Grant agency of the Czech Republic (GACR) as a part of the project **GA 205/07/0797 GeoWeb services orchestration**.