Finding OGC Web Services in the Digital Earth

Aneta J. Florczyk[1], Patrick Maué[2],
Francisco J. López-Pellicer[1], and Javier Nogueras-Iso[1]

[1] Departamento de Informática e Ingeniería de Sistemas,
Universidad de Zaragoza, Spain
{florczyk,fjlopez,jnog}@unizar.es
[2] Institute for Geoinformatics (ifgi),
University of Muenster, Germany
pajoma@uni-muenster.de

Abstract. The distribution of OGC Web Catalogues (CSW) across professional communities, the expert profile of the catalogue user and also the low coverage of OGC Web Services (OWS) in standard search engines reduce the possibility of discovery and sharing geographic information. This paper presents an approach to simple spatio-temporal search of OWS retrieved from the web by a specialized crawler. The Digital Earth could benefit from this solution as it overcomes technical boundaries and solves the limitation of CSW acceptance only within professional community.

Keywords: OWS, discovery, open search

1 Introduction

The concept of the Digital Earth envisions easy-to-use applications to support non-expert users to interact with various kinds of geographic information. Users are expected to search, load, and visualize spatial data without being challenged by the plethora of file formats and methods to access and download the data. The Open Geospatial Consortium (OGC) standards specify well-defined interfaces for spatial data services to ensure interoperability across information communities. Embedded in Spatial Data Infrastructures (SDI), OGC services are mainly (but not only) used by the public administration to publish their data on the Web. On-going initiatives such as the Global Earth Observation System of Systems (GEOSS) or the European Shared Environmental Information Space (SEIS) form an important backbone to the vision of the Digital Earth. They aim to make an abundance of geographic information about our environment available through OGC services.

The main sources of OWSs are distributed OGC Web Catalogues (CSW) [1] which functionality is usually exposed by a web application on community geoportals. Both, the distributional character of CSW and the fact that it is mainly focused on the expert user, difficult its access to mass market. There are also OWSs that are not registered in any catalogue, and (1) might be announced via human-readable description and usage examples or (2) may operate behind a web application. Standard web search engines (e.g. Google) are not prepared to deal with OGC specification (i.e. OWS oriented indexing and ranking). Moreover, only a half of the OWSs deployed in Europe are indexed by the main search engines [2], i.e. Google, Yahoo and Bing.

Digital Earth requires the integration of service description across technical boundaries and solving the limitation of CSW acceptance only within professional community. Therefore, we propose a simple search engine based on Open Search recommendation [7] in order to (1) enable simple spatio-temporal search and (2) discover existing OWSs. It requires an OWS-focused crawler capable of finding and indexing OGC-compliant services beyond the usual capacity of traditional search engines [5]. The gathered information is exposed as linked data by using a simple service model which covers common aspects of spatial data services. Applications which need more sophisticated query capabilities can directly use the SPARQL endpoint.

The rest of the paper is organized as follows. First, the OWS crawler is introduced. Section 3 outlines the simple service model. The Open Search specification is characterized in section 4 and section 5 presents briefly the proposed architecture. Conclusions and future work are included at the end.

2 OWS-focused Retrieva

An OWS-focused crawler is a Web crawler designed to find and index OWSs, which are not necessarily findable through existing catalogues. Development of such a crawler raises several challenges. It is necessary to identify and extract the explicit and implicit URLs from documents of different format (html, pdf, etc.), identify and harvest OGC CSW, identify the same OWS instance and manage the provenance and service versioning.

The work [2] presents a proposal for a crawler which uses OWS request patterns to extract URL from online documents, applies heuristics to explore existing search engines and harvests CSWs. The content is published as linked data which might be broswered and requested using SPARQL. The used model is rich and complex since it is thought for long term purposes (e.g., monitoring) and validation (e.g., conformance with an OGC or INSPIRE specification).

Simple announcement on the Web does not require such complex and rich description and even might be a performance disadvantage. The proposal of minimal service publishing model is described in the next section.

3 Simple Service Models for Spatial Data Services

The proposed model is based on the simple service model WSML-Lite. The Web Service Modelling Language (or WSML), is the formalization language for the Web Service Modelling Ontology (WSMO). The latter had in its origins a rather complex, but high-potential, approach to the capture of the semantics within the Web services. Even though it is tightly bound to W3C-compliant Web services, it can also be applied to OWSs with minor modifications. The flexibility of data models can be considered as one of the flaws of linked data. Integration requires from application developers to analyze and understand the models, and adapt their implementations accordingly. Our model extends the WSML-Lite model, so all applications which are already able to deal with it, can use our model as well. The RDF-based approach supports very flexible extensions of the existing model.

We can distinguish between the functional and non-functional Web service properties. The former contains all information needed to invoke the service, and load and visualize the retrieved data (possible protocol, a list of exposed operations, output data, etc.). The non-functional properties required for the discovery (in particular for evaluating if the Web service matches the client needs) cover aspects such as data provenance (e.g. the procedure for data acquisition), data quality (e.g. uncertainty), and descriptive metadata (e.g. title, keywords).

Some functional properties are important for all Web services, regardless of the chosen standard. Others depend on the specific OGC standard (e.g. the schema of the output). The *common model* comprises information common across all services, whereas the *extended model* covers specific aspects related to the nature of the provided data [8]. The suggested approach based on simple graph models encoded in RDF can be extended to capture both types of information.

Simple queries based on the Open Search recommendation (introduced briefly in the next section) support horizontal search (i.e. the search across the whole index unaware of the underlying service models) and SPARQL-based retrieval support vertical search, thus allowing the search regarding all aspects on particular services (e.g. specific aspects in the sensor model described in the metadata of a Sensor Observation Service (SOS)).

4 Simple Search of Spatial Data Services

Publishing contents as linked data has been proved useful as it supports building mashups and facilite integration. However, linked data and especially SPARQL is not adequate for a simple service search.

The Open Search (OS) specification proposal enables the coupling of OS clients and OS engines by providing the "description document" to the client, i.e., the structured and machine-readable description of its capabilities. The main advantage of this specification is simplicity combined with maintenance of extensibility. The OS-Geo extension has a potential for discovering and sharing both OWSs (e.g. GeoNetworks and GI-cat [3]) and neogeograpy web services [4]. It ensures simplicity by limiting the set of retrieval area definitions (a minimal bounding box, a polygon or a location approximation by combining lat/lon point with radius), and (2) the CRS to WGS84. Also, it is possible to use place names instead of explicit spatial objects. The OS-Time permits defining the time slice ("start" and "end") encoded according to RFC-33391[3]. The OS-Semantic allows

[3] http://www.ietf.org/rfc/rfc3339.txt

adding in description document the reference ontologies (taxonomies in OWL, SKOS or WSML), the sources of concept URIs which can be used in queries. The "classifiedAs" parameter defines classification (traduced into "subconcept of" and "equivalent to") of searched data; similarly the "relation" and "related" parameters allow performing similar search to the SPARQL predicate-object request.

```
<?xml version="1.0" encoding="UTF-8"?>
<GetRecords service="CSW" version="2.0.2" resultType="results"
outputFormat="application/xml" outputSchema="http://
www.opengis.net/cat/csw/2.0.2"
  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:ows="http://www.opengis.net/ows"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/cat/csw/2.0.2
                        ../../../csw/2.0.2/CSW-discovery.xsd">
<Query typeNames="csw:Record">
<ElementSetName typeNames="csw:Record">full</ElementSetName>
<Constraint version="1.1.0">
<ogc:Filter><ogc:And>
<ogc:PropertyIsEqualTo>
<ogc:PropertyName>csw:AnyText</ogc:PropertyName>
<ogc:Literal>river</ogc:Literal>
</ogc:PropertyIsEqualTo>
<ogc:BBOX><ogc:PropertyName>ows:BoundingBox</ogc:PropertyName>
<gml:Box><gml:coordinates>1.0,0.0 19.0,56.0</gml:coordinates></
gml:Box></ogc:BBOX>
</ogc:And></ogc:Filter>
</Constraint>
</Query>
</GetRecords>
```

Fig. 1. An example of query consumed by OGC Catalogue Service (POST).

The figure 1 shows an example of OGC CSW query which produces full-text search with additional spatial restriction. The Open Search specification allows to hide complexity of such query. The figure 2 shows the equivalent query which complies with the OS specification.

```
http://example.com?service=csw
&request=GetRecords
&version=2.0.2
&anyText=river
&box=1.0,0.0 19.0,56.0
&format=vnd.ogc.csw_xml
```

Fig. 2. An example of query consumed by OS-enabled catalogue.

5 Architecture

The figure 3 presents the main elements of an architecture for finding OWSs on the web. A dedicated crawler collects and indexes information on existing OWSs, and finally deploys it in a RDF repository ("LD") according to Linked Data principles. The OS engine describes its search capabilities in a description document ("OS DD") and transforms incoming OS queries into correspondent SPARQL requests by applying a translator ("OS2Sparql Translator"). In consequence, the OS clients can learn from the description document how to request the OS engine. For example, OS client might be a component of a simple web page for searching for OGC web services (a simple search box and a map to visualize the results). Applications which need sophisticated query capabilities can, in any case, directly use the SPARQL endpoint.

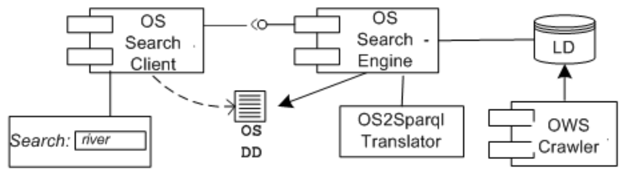

Fig. 3. Discovering and publishing OGC web services on web.

Since the crawler might gather more information about service instances (e.g. validation, monitoring elements), the proposed service model would be a sub-graph of the crawler model. Such extraction is quite straightforward in case of graph databases. The maintenance of the URI link to the original service instance in the crawler model might be useful for a potential user but raises integrity challenges (e.g. updating procedures). The translator might be seen as a separated component that maps OS requests into SPARQL queries. The translator has to apply advanced strategies to manage the OS extensions (*geo*, *time* and *semantic*) to minimalize number of SPARQL queries. OS client receives ordered results according to the data nature (e.g., ranking higher "dc:title" than "dc:abstract" content) in a previously selected format (RSS, KML, etc.).

6 Conclusions and Future Work

This paper introduces a simple approach which enables to find OGC Web services on the web. The proposed service model, based on the simple service model WSML-Lite, comprises characteristics of different OGC standards. Both, simple queries based on the Open Search recommendation support horizontal search, and SPARQL-based retrieval, enable to search for all aspects on particular services. Future work will consist on developing and implementing a specification for mapping between a proposed service model and Open Search specification.

Acknowledgments. This work has been partially supported by Spanish Government (projects "España Virtual" ref. CENIT 2008-1030, TIN2007-65341 and PET2008_0026), the Aragón Government (project PI075/08), the National Geographic Institute (IGN) of Spain, and GeoSpatiumLab S.L. The work of Aneta Jadwiga Florczyk has been partially supported by a grant (ref. AP2007-03275) from the Spanish government.

References

1. Nebert, D.: OpenGIS® Catalogue Services Specification. Version 2.0.2. Open Geospatial Consortium Inc. Ref. OGC 07-006r1 (2007)
2. López-Pellicer, F.J., Bejar, R., Florczyk, A.J., Muro-Medrano, P., Zarazaga-Soria, J.: State of Play of OGC Web Services across the Web. INSPIRE Conference (2010)
3. Baldini, A., Boldrini, E., Santoro, M., Mazzetti, P.: GeoNetwork powered GI-cat: a geoportal hybrid solution. Poster on EGU General Assembly (2010)
4. Fonts, O., Huerta, J., Daz, L., Granell, C.: OpenSearch-geo: The simple standard for geographic web search engines. IV Jornadas SIG Libre (2010)
5. López-Pellicer, F.J., Florczyk, A.J., Nogueras-Iso, J., Muro-Medrano, P.R., Zarazaga-Soria, F.J.: Exposing CSW Catalogues as Linked Data. 13th AGILE Conference. LNCS, pp. 183–200. (2010)
6. Cox, S., Schade, S., Portele, C.: Linked Data in SDI. INSPIRE Conference (2010)
7. Clinton, D.: OpenSearch 1.1 Specification (draft 4). http://www.opensearch.org
8. Maué, P.: An extensible semantic catalogue for geospatial web services. International Journal of Spatial Data Infrastructures Research, (2008)