ACTA UNIVERSITATIS AGRICULTURAE ET SILVICULTURAE MENDELIANAE BRUNENSIS

Volume LVIII 38 Number 6, 2010

ON DEVELOPMENT OF SEARCH ENGINE FOR GEODATA

D. Procházka

Received: August 23, 2010

Abstract

PROCHÁZKA, D.: *On development of search engine for geodata.* Acta univ. agric. et silvic. Mendel. Brun., 2010, LVIII, No. 6, pp. 389–398

Effective management and sharing of geodata is one of the priorities of the European Union (INSPIRE activity) and companies all around the world. Many different companies and organisations publish their geodata using web mapping services. This situation leads to a multiple publishing of similar or completely same geodata. On the other hand, there is frequently a problem how to determine an appropriate mapserver with the required data. This paper presents a geodata search engine which solves the problem how to access geodata more effectively. Presented solution aggregates data from the different mapservers and provides an interface according to the Open Geospatial Consortium Web Map Server specification. This allows to use our solution in the standard GIS tools as common mapserver. Completely new feature is a request which allows to select map layers which fulfills specified criteria. Selection could be given by keywords in a map layer description and by defining a bounding box on Earth surface. Response is a list of appropriate layers sorted according to their relevance. Presented solution could be among other applications significant source of information for many data mining techniques. It allows to interconnect processed data with their space-temporal context.

geodata, mapserver, WMS, Web Map Service, searching, metadata

1 Introduction

In the area of geoinformatics an interoperability – free exchange of data between systems of different vendors – is one of the most important topics. The most significant organisation in geoinformatics focused on interoperability is Ope*n Geospatial Consortium, Inc.*[1] (OGC) which develops standards widely supported by differend geographical information systems. Representatives of large companies as well as indipendent professionals participate in the OGC. The access to geodata is solved well.

A problem appears in the area of geodata retrieval. Although we effectively search in emails, documents and pictures, the search engine for geodata which could be compared with well known services as *Google*, *Yahoo!*, etc. does not exist. Therefore, we deal with this problem. State-of-the-art geodata are necessary for many management and business tools. From decision support systems to datamining application (e.g. Štencl – Šťastný (2009)).

The first part of this work briefly presents the *Web Map Server* standard for publishing geodata. A review of approaches used for geodata and metadata aggregation and retrieval follows in section 3. Based on this review, a method for development of an engine for geodata search is proposed in section 4. This part also describes principles of our experimental implementation and the archieved results.

[1] OGC Website: http://www.opengeospatial.org/

2 Web Map Service and related methods used for geodata distrubution

Web Map Service (WMS) is the most popular of many OGC web services oriented on sharing geographical content. The basic purpose of the WMS is to provide a raster image with requested geodata within some specified location (for specification see de la Beaujardiere (cit. 20. 8. 2010)). The number of supported image formats depends on the implementation of the WMS standard (*Portable Network Graphics* (PNG), *Joint Photographic Experts Group* (JPEG) are usually supported).

WMS defines two mandatory and one optional request: *GetCapabilities*, *GetMap* and *GetFeatureInfo*. The parts 2.1 and 2.2 describe the requests *GetCapabilities* and *GetMap* in details because they are significant for our solution.

2.1 GetCapabilities

Response is usually an XML file with a description of provided map layers (names, supported formats, descriptions, etc.). The mandatory and optional parameters of each request are given by the WMS specification, but there are slight differences between implementations by different vendors (see differences between de la Beaujardiere (cit. 20. 8. 2010), Vretano (cit. 20. 8. 2010) and ESRI (cit. 10. 8. 2010)). An example of such request is:

```
http://atlas.walis.wa.gov.au/servlet/com.esri.wms.
Esrimap?
REQUEST=GetCapabilities&VERSION=1.1.1&SER
VICE=WMS&
```

This address represents a *GetCapabilities* request to a WMS service which is running on given address and is formulated using WMS specification version 1.1.1. The response is XML *Capabilities file*. The first part of this file specifies supported formats of *GetMap* response, information about owner and legal limitations and the second part contains description of provided mapsets and layers. Obviously, the request is send via HTTP protocol using GET method. Key-value pairs are separated by symbol "&".

2.2 GetMap

GetMap is the most significant request. Its purpose is to provide images containing map layers from described area. An example of the *GetMap* request is:

```
http://echo.mendelu.cz/cgi-bin/moebius/moebius.
py?
SERVICE=WMS&VERSION=1.1.1&REQUEST=G
etMap&
layers=europe,coast&bbox=-
10,35,6,45&srs=EPSG:4326&.
format=image/png&width=1024&height=768&
```

The request contains the address of the mapserver and basic parameters – version, service and request. Furthermore, a set of parameters is defined with specification of required map layers – their names, bounding box that speciefies place on Earth surface and a desired map projection. The last part of the request describes the specification of an output format and a size of the image. The example of the response is on Fig. 1.

3 Geodata retrieval approaches

We can transform the issue of a retrieval of an appropriate map layer (geodata) to the issue of the retrieval of an appropriate text information in metadata. Obviously, approaches based on image

1: *Result of GetMap request – coast lines and country borders of the Iberian Peninsula* processing are not suitable for this purpose. These methods are not able to find objects such as roads in some region. Shapes or contours of these object could be completely differend in various versions or interpretations.

According to the Shirky (cit. 10. 8. 2010) and Shirky (2008), there are two main approaches of information retrieval from the set of documents – *ontological classification* and *searching*. We clearly discuss the main idea about these methods in the sections 3.1, 3.2 and in section 3.3 we explain why the searching is the best method for our solution.

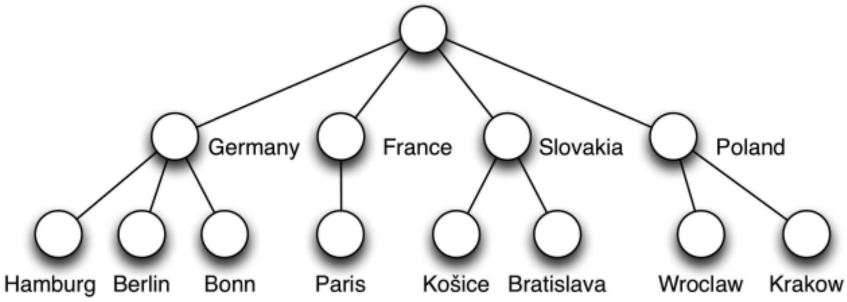
3.1 Ontological classification

A classification principle is often used in libraries and archives. The main idea is based on organizing of a set of entities into groups based on their content and possible relations. For any subject (existing or not) exists a logical place within the system. This approach is used in medicine (DSM-IV, the 4th version of the psychiatrists' *Diagnostic and Statistical Manual*), natural sciences (periodic table of the elements, taxonomy of animals, …) and also in web catalogues such as *Yahoo! Directory*[2], DMOZ[3], etc. An example of ontological classification on Fig. 2 presents origin of catalogised web pages.

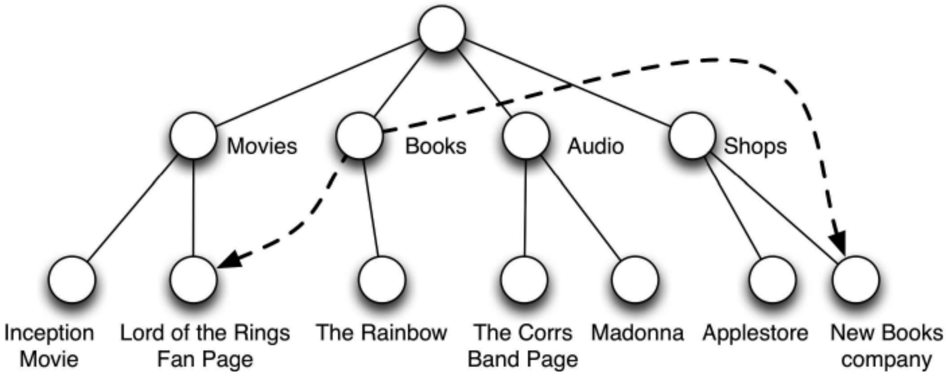
This approach should be useful but at least two conditions must be fulfilled:

- Formal categories are created by professionals – Categories must be created by some single authority because the quality of the tree with categories is crucial in this approach.
- Clear edges between categories – The Earth has seven continents. It is inevitable that a book must be written in one of them, therefore, it is possible to categorise books according to their origin. In some cases, it is not possible to formulate such clear formal categories (document should cover multiple topics).

Even if these conditions are fulfilled, there should appear some problems. For example we have a catalogue with web pages with these categories: *Literature*, *Shopping*, *Music*, … *Music* has these sub-categories: *Artists and groups*, *Genres*, … The question is which category is appropriate to sub-category *Music stores* – *Music* or *Shopping* category? Obviously, both options are possible. This situation is solved using a symbolic link (or alias) in the electronic catalogues. But too many links do not provide an easy survey, thus it is usually necessary to set a maximum number of such symbolic links. The described tree with structural ambiguity is on Fig. 3.



2: *Example of ontological classification: tree with predefined categories*



3: *Tree with predefined categories and symbolic links which solves ambiguity of items.*

[2] http://dir.yahoo.com/
[3] http://www.dmoz.org/

In geoinformatics we can find number of projects, usually called *metadata catalogues*, which are based on ontological classification. These catalogues are collecting information about different mapservers, their content and even about commercial geodata provided offline (*Metadata portal of the Ministry of the Environment of Czech Republic*[4], WEEMS[5], …).

The basic principle of this method is that cataloguers are anticipating what the user will be looking for. The user must browse a tree with categories to reach the desired entity. The basic problem here is that such catalogue services are usually managed by a group of administrators. Because of it this approach leads to high costs for maintenance and to inconsistency. This inconsistency is caused by delays in actualisation of the records.

Specialised kind of catalogue is a fully automated geodata aggregation engine developed by the *U. S. Naval Research Laboratory* – GIDB. The project is described in Sample et al. (Sept.-Oct. 2006) or on DMAP Team web pages[6] in details. But there is a huge difference between GIDB a previously presented approaches. This project do not classify the content, it just creates some sort of catalogue or a list of mapservers. Although this project provides just simple WMS interface to existing web services, it is one of the most successful one because it uses an automatic processing which allows aggregating large number of services. A significant similar project is *GeoBrain* (see Yue et al. (2006) or project web pages[7]).

The other method of ontological classification is focused on an interconnection of geodata (see Cruz et al. (2002), Wiegand et al. (2004), Wiegand et al. (2003) and others). The gist is obvious from the definition of the ontology – relationships between different entities in databases or mapservers. Correctly defined ontology allows to recognise that *inhabitants* in one (geo)database has the same meaning as *population* in the other one for instance. This approach is much more complex than simple catalogues. It reduces the redundancy and ambiguity in the databases. Nevertheless the basic drawback holds – these ontologies must be created by hand. However ontologies should be a supplement of complex searching methods.

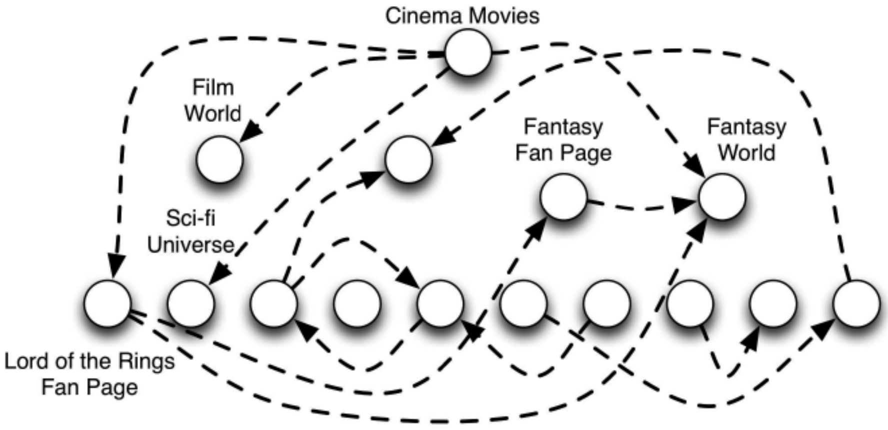
Summary of the aggregation approaches

The main idea of presented approaches is that the user is forced to go though a tree of categories or services to reveal the required map layer. All presented methods interconnect different data sources and in some cases these sources are also categorised. It is obvious that the ontological classification is not effective when it is necessary to deal with huge amount of data or ill defined ontologies.

3.2 Searching

The other approach for retrieval of information – searching is based on the assumption that is not possible to unambiguously categorise all entities in some hierarchy – there are no clear borders between categories, entities belong into more categories, there is no authority to enforce a classification method or other reason. In these cases, some kind of fulltext search is usually used. Provider of a search service lets the user to formulate a question and browses some hidden catalogue (indices) by himself. This method is more user-friendly, because user is not limited whether ontology is defining appropriate combination of criteria. This method also requires much more complex catalogues (see Fig. 4) organised usually in some sort of graph.

For searching geodata, the searching method is obviously more appropriate. It is almost impossi-



4: *Entities connected with links into graph*

[4] http://mis.cenia.cz/
[5] http://weems.uwindsor.ca/
[6] http://columbo.nrlssc.navy.mil/dmap
[7] http://geobrain.laits.gmu.edu/

ble to define any universal geodata hierarchy. We could classify them by the content, temporal or space point of view and even classification based just on content would not be completely clear. In spite of this, there is currently no widely spread fulltext search engine for geodata. We could find just projects which are usually able to search on a single mapserver or a defined group of them (see *INSPIRE geoportal*[8] or previously mentioned *Metadata portal of Ministry of the Environment of Czech Rep.*[9]). Even one of widely used search engines – Google – is not able to search for geodata. According to the article *Getting KML into Google's Geo Search Engine*[10], Google experiments with searching in metadata tags in KML files only. Therefore, development of fulltext geodata search engine is a great challenge and we propose the solution described in section 4 based on the conditions in 3.3.

3.3 Analysis of the approaches

We formulate following conditions for our solution:

1. The engine for retrieving geodata must be based on fulltext search principle. Ontological searching is not effective in this case because it is more suitable for data which can be clearly divided into categories. Definition of such categories for geodata is practically impossible on account of the mentioned reasons.
2. The key part of the engine must be automatic. It is not possible to index new mapservers or administrate them manually in acceptable time. As is shown in the example of GIDB and *GeoBrain* projects, actualised simple engine is obviously more usable, than complex services with limited amount of data.
3. Indices must contain maximum amount of metadata information to provide accurate search results. Map layers distributed using OGC web mapping services are clearly described by the *GetCapabilities* files (see *LayerProperties* in de la Beaujardiere (cit. 20. 8. 2010) on page 24). The only problem is that many providers still do not use defined meta tags to describe the content, but the situation is improving. Other organization or national metadata standards could be a valuable source of information, but they are not widely spread, therefore it is not possible to use them as a primary source.

4 On development of fulltext geodata search engine

As been mentioned in the beginning of the article, large organisations (private companies, governments, etc.), which are the main producers of geodata, are usually using mapservers for their publishing. Almost all main vendors of mapservers support the standards of *Open Geospatial Consortium, Inc.* nowadays, therefore this project is focused on indexing map services using these open standards.

4.1 Indexing of mapservers

The basic component of every search engine is an indexing service which must be able to create an index – a record about a given data source. According to the conditions in 3.3, the service must work automatically. Basic element of our database is an index of single map layer.

Three groups of information in GC file must be stored in every index. The first group are attributes connected directly to the layer in GC file (name of the layer, its bounding box, etc.). The second group are information connected to the mapset in which is the layer defined or to the mapserver itself (again it is a name, description, bounding box, keywords, etc.). For example description of the mapserver describes all mapsets within the server, similarly, description of a mapset describes all layers in it. The third group of information contains technical details about the service – a list of output formats supported by the mapserver, address of the service, etc. For exact structure of the index see Procházka – Procházková (2008).

4.2 Structure of the search engine

The first step is to propose the structure of request and response. The search request must contain keywords defining the subject of search, definition of area and optionally, further parameters (required output format, etc.). For sending these parameters, we should use a similar principle as other WMS requests do. A more complex problem is how to design a method for returning the results of query. The basic approach is to return a web page with a list of layers matching to the query.

Each layer should be represented with a link the corresponding mapserver. These links should be *GetMap* requests using GET method (leads to appropriate raster pictures or GML files). This concept is similar to the approach which is used by fulltext search engines such as Google, Yahoo!, … A significant drawback is that it is virtually impossible to integrate such engine with some GIS tools, therefore a completely different architecture is used in this project. Our solution is called virtual mapserver and it is described in next paragraph.

Virtual mapserver

The key part of our search engine is a virtual WMS mapserver called *Moebius*. *Moebius* is able to process information about map layers stored in indices. On

[8] http://www.inspire-geoportal.eu/

[9] http://mis.cenia.cz/

[10] http://www.gearthblog.com/blog/archives/2008/05/getting_kml_into_googles_geo_ search.html

GetCapabilities request, *Moebius* generates own GC file which contains all indexed map layers. From the user point of view, *Moebius* is a standard WMS mapserver with huge number of layers. It creates a single interface to many different map services.

Obviously, to fulfill the WMS specification, it is necessary to implement also *GetMap* request. If the user requests our virtual map layer with name *world@nasa.gov* (this is in fact **NickName**), *Moebius* accepts this request, creates a new request on a real layer *world* and sends this request to the real mapserver (e.g. **http: //someserver.nasa.gov/cgi-bin/someservice?**). The response – a raster image file – is redirected from *Moebius* to the user. The communication scheme on Fig. 5 shows very simplified principle.

Many problems arise during the solution. The most important one is the merging layers from different mapservers. For example GIDB (Sample et al. (Sept.–Oct. 2006)) uses many virtual mapservers and layers from different mapservers cannot be merged. In our project all layers are in one mapserver and can be composed without any limitations. Related problems are different supported output formats and coordinate systems. This is solved using libraries such as GDAL[11] and ImageMagick[12].

Example

The example presents a request on two layers from different real mapservers (radarsat 1000 m coverage over Canada from *Natural Resources Canada* and background is from the *Blue Marble* project).

http://echo.mendelu.cz/cgi-bin/moebius/moebius.py?service=wms&version=1.1.1& request=getmap &layers=Radarsat_1000@cgkn.net,bluemarble_1@iceds.ge.ucl.ac.uk& crs=EPSG:4326&bbox=-180,10,-30,90&styles=&format=image/jpeg&width=800& height=600&

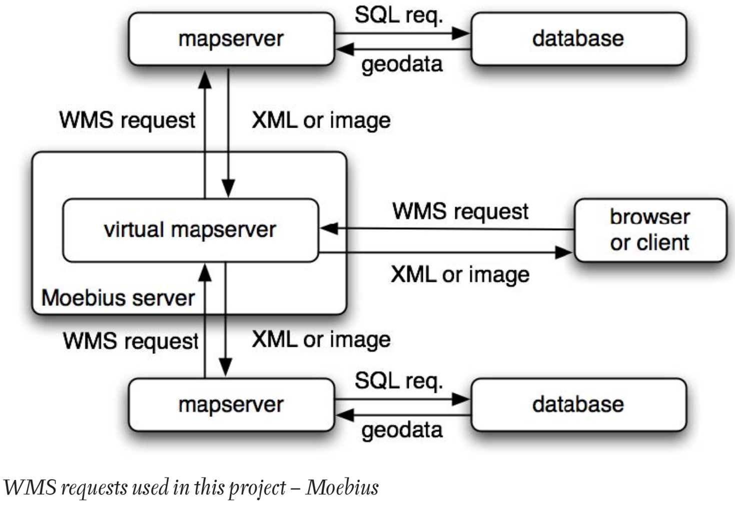
Translated requests for the real mapservers:

http://cgkn2.cgkn.net/cgi-bin/cgknwms?VERSION=1.1.1& REQUEST=GetMap &LAYERS=Radarsat_1000&bbox=-110,15,15,80& SRS=EPSG:4326&FORMAT=image/png&TRANSPARENT=true&WIDTH=800&HEIGHT=600&

http://iceds.ge.ucl.ac.uk/cgi-bin/icedswms?SERVICE=wms&VERSION=1.1.1& REQUEST=getmap&LAYERS=bluemarble_1&CRS=EPSG:4326&BBOX=-180,-90,180,90& STYLES=&TRANSPARENT=true&FORMAT=image/png&WIDTH=800&HEIGHT=600&

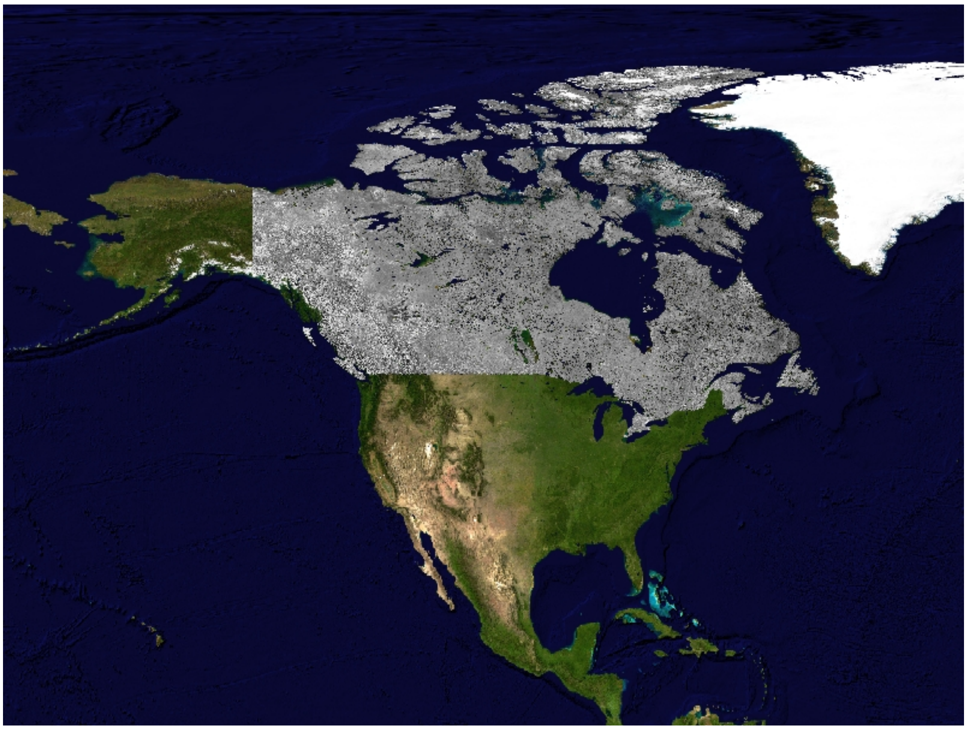
These two PNG images with transparent background are on *Moebius* merged and converted into JPEG file. The resulting image is on Fig. 6.

Merging of requested layers on the server side has two important benefits. The first one is simplicity on integration into current information infrastructure. It is not necessary to deal with this problem on the client side. Other benefit is reduction of transferred data. This is especially important in mobile applications such as *WhateverMap* (see Kamínek – Klimánek (2010)). As an example, it is possible to take a map layer composition given by *Moebius* composed of five layers from different mapservers. This composition is transferred as a single image. Without *Moebius* will be necessary to transfer all five map layers independently. Therefore the amount of transferred data will be five times higher. This could also lead to significantly longer transfer times. Generally, with complexity of requested composition grows the advantage of *Moebius* usage, especially on mobile devices with limited resources.



5: *Scheme of translation WMS requests used in this project – Moebius*

[11] http://www.gdal.org/
[12] http://www.imagemagick.org/



6: *Result of GetMap request on two layers from different mapservers provided by Moebius*

Integration of searching into Moebius

As been previously mentioned, the result of the search request is a list of relevant layers. The basic idea of our approach is that *GetCapabilities* request means returning all available layers, and the purpose of the search request (e.g. *FindMap*) is to return all layers fulfilling given criteria. Therefore the *FindMap* is very similar to *GetCapabilities*, hence also the format of the response should be the same – *GetCapabilities* file. This approach allows to use this searching engine easily in current GIS tools. The search engine is in fact a standard mapserver returning GC files which should be automatically processed. It is necessary to emphasize that virtual mapserver could contain thousands of map layers so effective access to map layers is possible only through *FindMap* request.

The structure of the query language is similar to other WMS requests because of easy use and potential integration into WMS standard. Query has the following mandatory parameters: **REQUEST=FindMap** – identification of the request, **WORDS=keyword,keyword,...** – list of keywords which are searched in the indices and **BBOX=minx,miny,maxx,maxy** – bounding box for searching. These parameters allows to define a unique area on the Earth surface and required content. Furthermore, every query should be optionally extended by definition of relation between keywords **(OPERATOR={and|or})** and setting an integer number that represents the significance of instaces of keywords in given part of the index **(DESCRIPTION=0, KEYWORDS=9** – high significance of keywords, insignificant content of the description).

Layers in the resulting file are sorted by relevance. In a calculation of relevance instances of searched keywords in different elements of index and their proximity are counted (see Procházka – Procházková (2008)).

Example

The task is to find layers with information about "Iberian peninsula". We formulate following request:

http://echo.mendelu.cz/cgi-bin/moebius/search.py? bbox=-180,-90,180,90&words=iberian,peninsula&operator=and

As been previously mentioned, response will be the capabilities file with map layers which describes Iberian peninsula (coastlines, roads, etc.).

5 Summary

This article presents new approach to development of a fulltext geodata search engine. The presented solution allows to index any WMS mapserver automatically. Indices contain original names, descriptions, bounding boxes, addresses and other information necessary for the construction of *GetMap* requests. The presented searching engine is based on the idea of a virtual mapserver called *Moebius*. *Moebius* provides a WMS compliant interface to all indexed map layers. This feature allows to browse easily all layers from any GIS tool. A significant advantage of this architecture also the ability to combine layers from different real mapservers. For easier access to geodata we designed a new request called *FindMap*. This request is in fact a supplement of *GetCapabilities* request. *FindMap* returns part of *GetCapabilities* file (in this case all indexed layers) which is fullfiling the requirements given by parameters of the request. This project could be a valuable source of information for many business tools, especially data mining (Štencl – Šťastný (2009)). From other applications, it is necessary to mention modelling of natural phenomena (Machalová (2009)), etc. Key aspect of these models is their precision which is infeasible without state-of-the-art geodata.

Results of our experimental implementation of this search tool are very promising. It was proven that presented concept is viable and provides an easy access to number of worldwide map services. Current development is focused on improvement of performance, automatic actualisation of map services. We believe that the free software could encourage other developers to participate on the solution of this problem as is discussed e. g. in Čepek – Pytel (2005). Therefore, the source codes of the experimental implementation are available under the GNU General Public License (http:\\echo.mendelu.cz).

SOUHRN

Vývoj nástroje pro vyhledávání geodat

Velké množství firem a institucí v současné době při své práci využívá geodata a značná část z nich geodata přímo vytváří. Aktuální geodata jsou také zásadním zdrojem informací pro řadu aplikací z oblasti managementu a obchodu – systémy pro podporu rozhodování, dolování dat atp. Jedním z klíčových problémů současné geoinformatiky je efektivní správa a sdílení těchto geodat. Převažující metodou jejich publikování jsou webové mapové služby. Na jedné straně je nevyhovující, že značná část mapových služeb obsahuje velmi podobná nebo zcela duplicitní data. Na druhé straně je často problém nalézt jakoukoliv službu, která požadovaná data obsahuje. Jedná se o obdobnou situaci jako v případě webových stránek před nástupem vyhledávacích nástrojů. Nicméně v oblasti webových mapových služeb žádný podobný vyhledávací engine neexistuje. Tento článek navrhuje novou podobu vyhledávacího nástroje pro geodata (nazvaného Moebius) a na základě analýzy popisuje jeho strukturu a kostru elementárního vyhledávacího jazyka odvozeného z používaných standardů. Princip funkce je ilustrován na příkladech a podložen experimentální implementací.

geodata, mapový server, WMS, Web Map Service, vyhledávání, metadata

The paper is written as a part of a solution of a project IGA FBE MENDELU 12/2010 and a FBE MENDELU research plan: MSM 6215648904.

REFERENCES

BEAUJARDIERE, J., 2010: OpenGIS Web Map Server Implementation Specification [on-line]. Available: http://www.opengeospatial.org/standards/wms, cit. 20. 8. 2010.

CRUZ, I. F. et al., 2002: Handling semantic heterogeneities using declarative agreements. In: *GIS '02: Proceedings of the 10th ACM international symposium on Advances in geographic information systems*, p. 168–174, New York, NY, USA, 2002. ACM Press. ISBN 1-58113-591-2.

ČEPEK, A., PYTEL, J., 2005: Software Freedom as an Academic Motivator. *50 Years of Research Institute of Geodesy Topography and Cartography: Jubilee Proceedings, 1954–2004*. 2005, Vol. 50, 36, p. 71–77. ISBN 978-80-85881-23-3.

ESRI, 2010: WMS and WFS Connector Documentation [on-line]. Available: http://interops.esri.com/, cit. 10. 8. 2010.

IEEE International Conference on Geoscience and Remote Sensing Symposium 2006. 2006, p. 3486–3489.

KAMÍNEK, J., KLIMÁNEK, M., 2010: Mobile usage of digital geographical data in the Apple iPhone device. *Acta Universitatis Agriculturae et Silviculturae Mendeleianae Brunensis*. 2009, LVIII, 4, p. 89–95. ISSN 1211-8516.

MACHALOVÁ, J., 2009: Space modeling in management of landscape flood-protection measures. *Acta Universitatis Agriculturae et Silviculturae Mendeleianae Brunensis* 2009, LVII, 6, p. 133–142. ISSN 1211-8516.

PROCHÁZKA, D., PROCHÁZKOVÁ, J., 2008: Moebius: An interface to web map services. *Geoinformatics FCE CTU*. 2008, III, p. 39–50. ISSN 1802-2669.

SAMPLE, J. et al., 2006: Enhancing the US Navy's GIDB Portal with Web Services. *Internet Computing, IEEE*. Sept.-Oct. 2006, 10, 5, p. 53–60. ISSN 1089-7801.

SHIRKY, C., 2010: Ontology is Overrated: Categories, Links, and Tags [on-line]. Available: http://www.shirky.com, cit. 10. 8. 2010.

SHIRKY, C., 2008: *Here Comes Everybody: The Power of Organizing Without Organizations*. London: Penguin Press HC, 2008. ISBN 1-5942-0153-6.

ŠTENCL, M., ŠŤASTNÝ, J., 2009: Advanced approach to numerical forecasting using artificial neural networks. *Acta Universitatis Agriculturae et Silviculturae Mendeleianae Brunensis*. 2009, LVII, 6, p. 297–304. ISSN 1211-8516.

VRETANO, P. A., 2010: Web Feature Service Implementation Specification [online]. Available: http://www.opengeospatial.org/standards/wfs, cit. 20. 8. 2010.

WIEGAND, N. et al., 2003: Extending XML web querying to heterogeneous geospatial information. In: *dg.o '03: Proceedings of the 2003 annual national conference on Digital government research*. Digital Government Research Center, 2003.

WIEGAND, N. et al., 2004: A web query system for heterogeneous government data. In: *dg.o '04: Proceedings of the 2004 annual national conference on Digital government research*, p. 1–10. Digital Government Research Center, 2004.

YUE, P. et al., 2006: Semantic Augmentations for Geospatial Catalogue Service.

Address

Ing. David Procházka, Ph.D., Ústav informatiky, Mendelova univerzita v Brně, 613 00 Brno, Česká republika, email: david.prochazka@mendelu.cz