

Mapping 38 Years of Excavation: The Dissemination of Vector-data from Elephantine, Egypt

Stefan ZIEGLER¹ – Kai-Christian BRUHN²

¹Department of Geoinformation Canton Solothurn

²izmainz – Institute for Spatial Information and Surveying Technology, University of Applied Sciences Mainz

¹stefan.ziegler@bd.so.ch

²bruhn@fh-mainz.de

Abstract

The paper focuses on the implementation of an open source solution for providing vector-data through a WebGIS. It will serve as a central access to and the possibility for manipulating the graphic information of the Elephantine excavation project conducted by the German Archaeological Institute Cairo in collaboration with the Swiss Institute for Architectural and Archaeological Research Cairo.

The solution addresses common problems in multidisciplinary, international projects working in remote regions. It is neither aiming to present data to the public nor to the wider scientific community. It rather provides a solution for the widespread but internal exchange of geometric data during the phases of analysis and interpretation within such a project.

Keywords

Elephantine, dissemination, web service, wms, wfs, pdf, printing, open standards, open source, fossGIS

1. Introduction

During the last three decades the island of Elephantine became a major site for the study of settlements in ancient Egypt. Since the late 1960s fieldwork has been conducted for several months each year at this small town on the southern border of Egypt (Deutsches Archäologisches Institut 2010). Due to the specific processes of stratification of mud brick architecture, Elephantine comprises rich evidence of about four thousand years of continuous settlement in more than twenty major stratigraphic phases.

The German Archaeological Institute initiated different projects to support the presentation of the results achieved. The most outstanding is the innovative approach to “Virtual Egyptology” by the IEMAR at Vienna University of Technology (Institute of Architectural Sciences, Vienna University of Technology 2006), that was presented by Peter Ferschin at the CAA 2007 (Ferschin *et al.* 2008).

What is presented here has its seeds in the attempt to supply the scientists involved in the project with a consistent set of georeferenced vector-data containing all built structures uncovered during the excavation. The intention was to deliver the raw geometry of walls and features in order to serve for the creation of maps adapted to specific topics by the different disciplines involved.

Three major challenges had to be met during the capture of the data:

- Elephantine witnesses the ongoing development of surveying techniques applied to archaeological field work. Several grids are present on the site and the original drawings differ in their framework of reference.
- The concept of recording features changed. It has been coherent only since the middle of the 1980s.
- The existing data are widespread and their processing is in different statuses.

Not all the problems in data acquisition are solved satisfactorily yet. But already during this early stage another issue was addressed. How to later disseminate the data to the flock of scientists? Which data-format is sufficient for the forecited demands and can be handled by the individual researchers?

Due to its status as a de facto standard for GIS-data and its capabilities for storing metadata, the shapefile-format (ESRI 1998) was chosen for archiving the vector-data from Elephantine. The shapefile, however, is not suitable for graphic applications used in Egyptology, e.g. the Adobe Illustrator.

The solution was developed at the Department of Geoinformation of the Canton Solothurn (CH) and is described in the second part of the paper. It allows to search for specific data in a PostGIS-database and to

determine the scale of the print and the paper size. The output is a pdf-document with the raw geometric data originally stored in the shapefile.

2. Objectives

Web-based GIS is heavenly used for presenting geospatial data. Besides a lot of advantages over desktop GIS there are some drawbacks like poor printing support. Printing is not as functional as a in a desktop GIS and the quality of the output is disappointing. The common way of printing from a web GIS is quite simple: the system converts the produced image into a Portable Document Format file (PDF). It's possible to add several datasets but they all have to be rendered first on the screen before they can be sent to the printer.

Screen-rendered images have a resolution of 72 dpi, which is insufficient for a high quality print product. There is a workaround to obtain better quality by requesting a bigger image of the same section and scaling down the image again. The result is a graphic with 144 or 288 dpi. A huge problem by using this solution is the size of the map symbology: Line-widths and font-sizes will be scaled down too, which can make a map unreadable. This approach is ideal if the input data – like aerial photographs or a topographical map – is in a raster format but suboptimal for vector data.

The paper based map production of web GIS is unsatisfying. The printing support is too inflexible and qualitatively insufficient.

The main objectives of the “high-quality-web-printing” project are as follows:

- creating top quality print products with a web solution
- usage of open standards and interfaces
- high flexibility
- usage of open source software.

2.1. Quality

In today's printing solutions vector-data is turned into raster-data (e.g. JPEG, PNG or TIFF). To obtain ideal results it is necessary that the input vector-data will pass into vector-data in the print product. Since the Portable File Format (PDF) is able to store vector data there is no loss in quality. Further advantages of storing/printing the data as vectors are:

- scaling the print product (e.g. A4 -> A0) without any loss of quality

- postprocessing the map in a vector graphics editor.

2.2. Open Standards

The use of open standards and interfaces guarantees a fast dissemination of a software since it is possible to integrate it in one's own products. If the source code is open to the public there will be a huge community that will enhance the software and allocate it again.

By using well-known interfaces it is possible to use only one component of the whole project. The access between the modules with open standards permits the user to use only the printing tool and to embed it in his own GIS client.

2.3. Flexibility

The ambition is to uncouple the presentation of the map on the screen and the printing process. To print a collection of geospatial data, the data do not have to be added to the map first. To achieve this aim there are at least two modules:

- a server module that produces the map
- a client module where the user can process some queries and request the map.

3. Programming languages, techniques and interfaces

3.1. Java

Java is an object-oriented programming language developed by Sun Microsystems (Sun Microsystems (2010)). Java applications (classes) are compiled to bytecode that can be run on any operating system with a Java virtual machine.

Java servlets are special classes that can be executed on a web server using GET- and POST-parameters, e.g.:

```
http://localhost:8080/MyFirstServlet/MyFirstServlet?forename=stefan&surname=ziegler
```

The example executes the servlet “MyFirstServlet” and passes the two GET-parameters “forename” and “surname” with the values “stefan” and “ziegler”.

3.2. Javascript, Ajax

Javascript is an object-oriented scripting language developed originally for Netscape Navigator. By now almost every browser has been supporting Javascript. Unlike PHP, Javascript is executed on the client side. By adopting Javascript the user is able to change the content of a website dynamically, e.g. the color of the text or the text itself.

For some time the usage of Javascript has not been not state of the art web-programming. By the appearance of Ajax / web 2.0 using Javascript is quite acceptable again. With Ajax (asynchronous Javascript and XML) it is possible to send and receive data without reloading the website. Complete libraries ease the use of Ajax:

```
function FirstAjaxFunction() {
    new Ajax.Request('http://localhost:8080/MyFirstServlet/MyFirstServlet', {
        method: 'get',
        parameters: {forename: „stefan“, surname: „ziegler“},
        onSuccess: function(transport){
            $('myDiv').innerHTML = "Congratulations!";
        },
        onFailure: function() {
            $('myDiv').innerHTML = "There's an error!";
        },
        onLoading: function() {
            $('myDiv').innerHTML = "Loading...";
        }
    })
}
```

In the example above the servlet “MyFirstServlet” is executed again with the same parameters. Callback routines allow to control the website against the servlet’s feedback:

- On success: “Congratulations” will be written in the “myDiv”-Element.
- On failure: “There’s an error!” will be written in the “myDiv”-Element.
- During executing the servlet: “Loading...” will be written in the “myDiv”-Element.

Unlike to a conventional request (as a normal link) the website is not reloaded and the user can continue to work without any dead time.

3.3. WMS

“An Open Geospatial Consortium Web Map Service (WMS) produces maps of spatially referenced data dynamically from geographic information.

This international standard defines a ‘map’ to be a portrayal of geographic information as a digital image file suitable for display on a computer screen. A map is not the data itself. WMS-produced maps are generally rendered in a pictorial format such as PNG, TIFF or JPEG, or occasionally as vector-based graphical elements in Scalable Vector Graphics (SVG) or Web Computer Graphics Metafile (WebCGM) formats. This is in contrast to a Web Feature Service (WFS), which returns actual vector data, and a Web Coverage Service (WCS), which returns actual raster data.“ (Wikipedia 2010).

The user receives the desired map image and does not have to pay attention to the administration of the data. Since the map is a derived product, the data provider does not need to deliver the data itself

and exercises control on the data. The user only gets a “dump” map, that is only good for visualization. It is not possible to make some queries like buffering or intersecting polygons. At least it is possible to change the presentation (e.g. change the color of all houses from black to red) of the map by using Styled Layer Descriptor (SLD).

The standard defines two mandatory requests:

- GetCapabilities: The Web Map Service delivers a well-formed and valid XML document with all the significant information: available requests, data layers and spatial reference systems.
- GetMap: With the obtained data the user is able to request a map from the server by using the GetMap-request.

Example of a GetMap-Request:

```
http://www.sogis1.so.ch/cgi-bin/sogis/sogis_uep.wms?  
SERVICE=wms&VERSION=1.1.1&REQUEST=GetMap&LAYERS=uep_gray,gemeinde&STYLES=  
&SRS=EPSG:  
21781&BBOX=607700,227800,608100,228100&WIDTH=400&HEIGHT=300&FORMAT=image/png
```

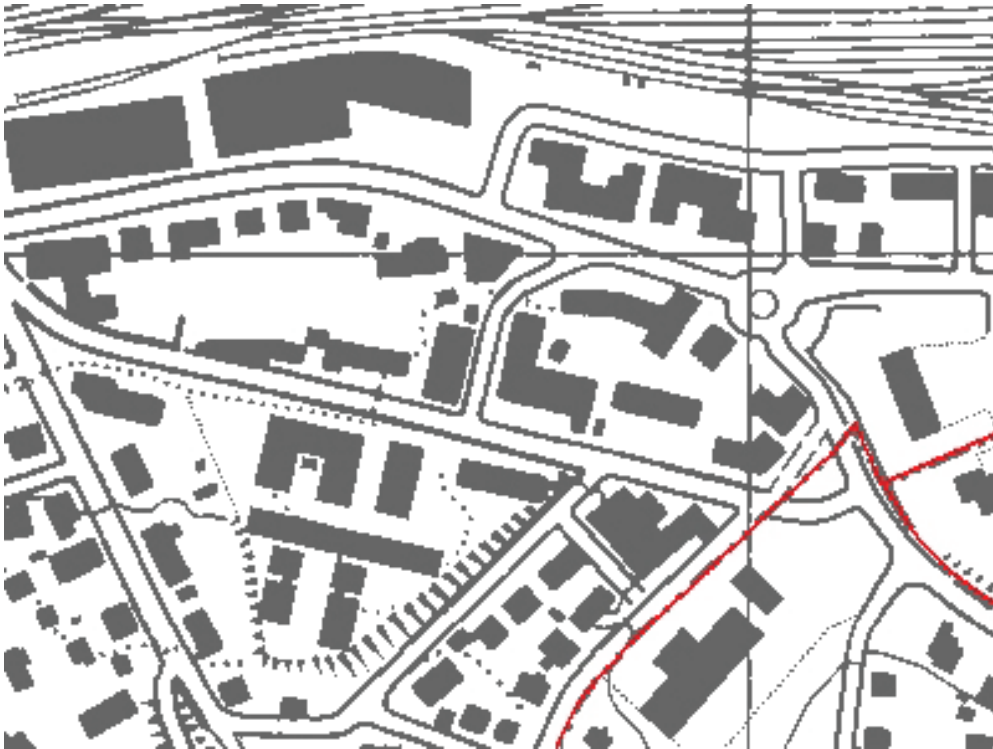


Fig. 1. Result of a GetMap-request.

In the request above the user demands for two data layers: “uep_gray” and “gemeinde”. The parameter BBOX defines the displayed area in the spatial reference system (SRS=EPSG:21781). The size of the image is controlled by the parameters HEIGHT and WIDTH.

3.2. WFS

Web Feature Service provides an interface allowing requests for geospatial features across the web. Instead of getting a “dump” map the user is now able to request a copy of the data (= feature) itself and also query the features. The standard output format of a Web Feature Service is GML (Geography Markup Language) but other formats like ESRI shapefiles are also imaginable. WFS-T (Web Feature Service Transactional) is a derived standard, that allows to create, delete and update features.

The standard defines three mandatory requests:

- GetCapabilities: Analog to the WMS GetCapabilities-request.
- GetFeatures: The service returns the requested feature(s).

- DescribeFeatureType: The service returns information about a layer as a XML schema.

4. Software

4.1. PostgreSQL / Postgis

Postgis (Refractions 2010) enables the open source database server PostgreSQL (PostgreSQL Global development Group 2010) to store geographical datatypes. A two-dimensional point is no longer stored as an x- and y-attribute but can be defined as point datatype. Lines and polygons can be handled in an analogous manner.

Postgis supports Simple Features according to the specification of the OpenGIS Consortium (OGC). Popular GIS analysis functions (buffer, intersection) and spatial indexing are provided too. The spatial indexing increases the speed of queries.

4.2. Apache Tomcat

As a servlet container Apache Tomcat (The Apache Software Foundation 2010) can execute Java Servlets on a web server. The software is bundled with its own

HTTP server but the better known Apache web server can also be used as a web server. In the latter case Tomcat works as a plugin in the Apache web server.

4.3. GeoServer

GeoServer (GeoServer 2010) provides server software services like Web Map Service (WMS), Web Feature Service (WFS) and Web Coverage Service (WCS). GeoServer is written in Java and runs in a Apache Tomcat environment. As a server software GeoServer delivers – in case of WMS – only the map as an image through standardized requests. A graphical user interface for zooming and panning is not part of the development.

One of the main advantages over other WMS servers is the amount of output formats. Besides the usual raster images formats – like JPEG, PNG or TIFF – GeoServer supports native PDF output, SVG, KML and GeoRSS. The native PDF output does not convert JPEG to PDF but it produces PDF vector graphics. The output has neither a frame nor a title or north arrow.

4.4. iText

iText (iText Software Corp 2009) is a java written open source library for creating and manipulating PDF, RTF and HTML files.

4.5. MapFish

MapFish (MapFish 2010) is an extensible web GIS application that is composed of MapFish Client and MapFish Server. MapFish Client is based on OpenLayers for the mapping part and ExtJS for the graphical user interface.

4.6. OpenLayers

OpenLayers (OpenLayers 2010) is an open source, completely written in JavaScript library for visualization of geospatial data in the browser. OpenLayers is mostly used as WMS client but can also be used to present KML or GML. An interface to include Google Maps is also provided.

4.7. ExtJS

ExtJS (Ext LLC 2010) is a JavaScript written cross-browser library for building rich internet applications (RIA).

5. Implementation

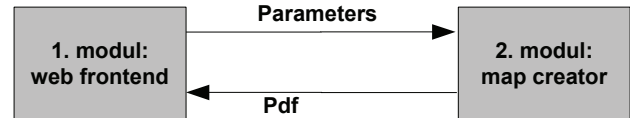


Fig. 2. Basic application workflow.

To achieve all the demands two modules are developed. The first one is a web frontend where the user can choose the desired section of the map, scale, data layers, paper size etc. These informations will be passed to the second module which will return the map.

5.1. Web frontend

The web frontend is developed with MapFish by creating two new widgets: a) the printing widget and b) the query widget.

The printing widget lets the user choose:

- the paper format
- the paper orientation
- the map scale

After assigning values to the three parameters, a map frame appears on the map and the user can move the frame to the desired section of the map (Fig. 3).

The query widgets has implemented some hardcoded queries. As a prototype for archaeologists the user can query for walls by choosing the house number or the occupation level. The results of the query are presented in a table where every single wall can be selected to show up on the map (Fig. 4). The queries are realized as WFS requests with a filter argument, there is no need to make use of PHP that connects the database. A big advantage of this approach is the independence of the data storage type of the data provider. As long as the data is served as WFS the provider can switch from shapefiles to an Oracle Spatial database to Postgis.

By hitting the “Print”-button the parameters are passed as a HTTP-request to the second module:

```
http://localhost:8080/MapCreator/GetMap?mode=3&size=A2&orientation=portrait&scale=500&title=CAA2008&timestamp=122262186229&datalayers=caa:isolines,caa:areas&BBOX=600000,200000,600100,200200
```

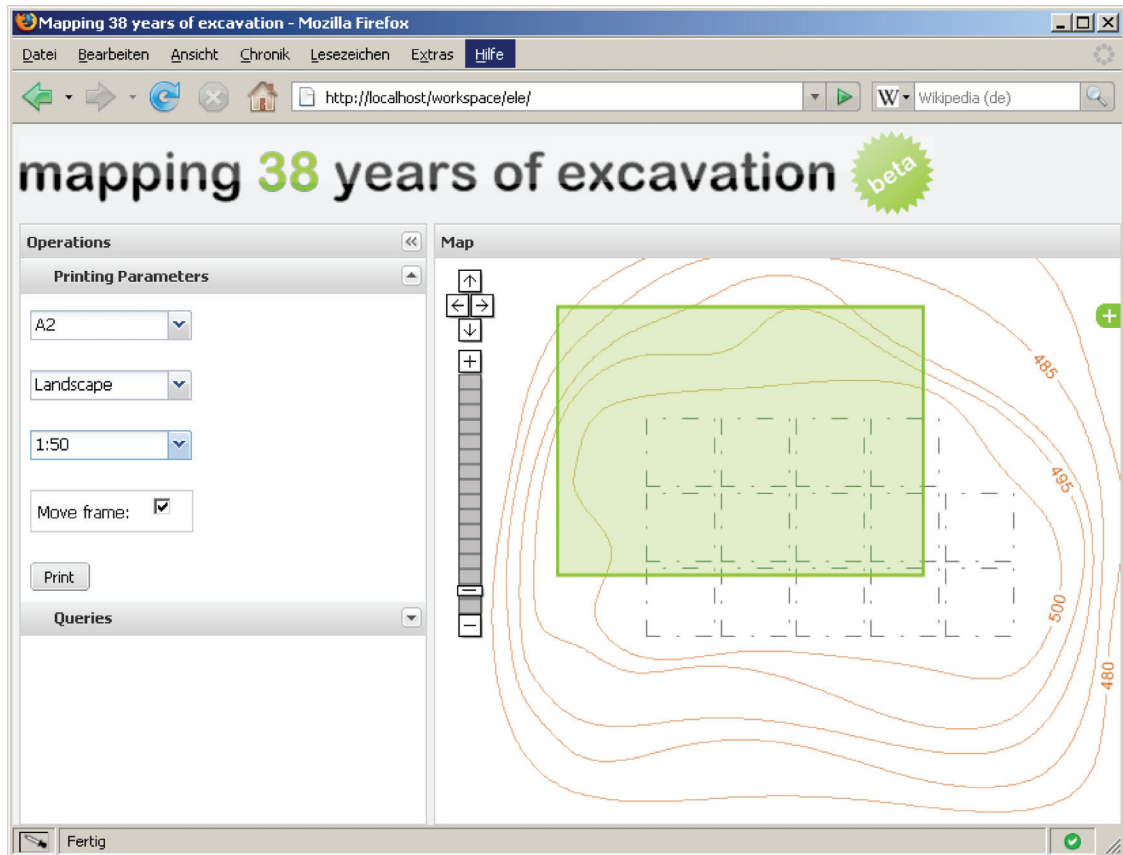



Fig. 3. Printing widget.

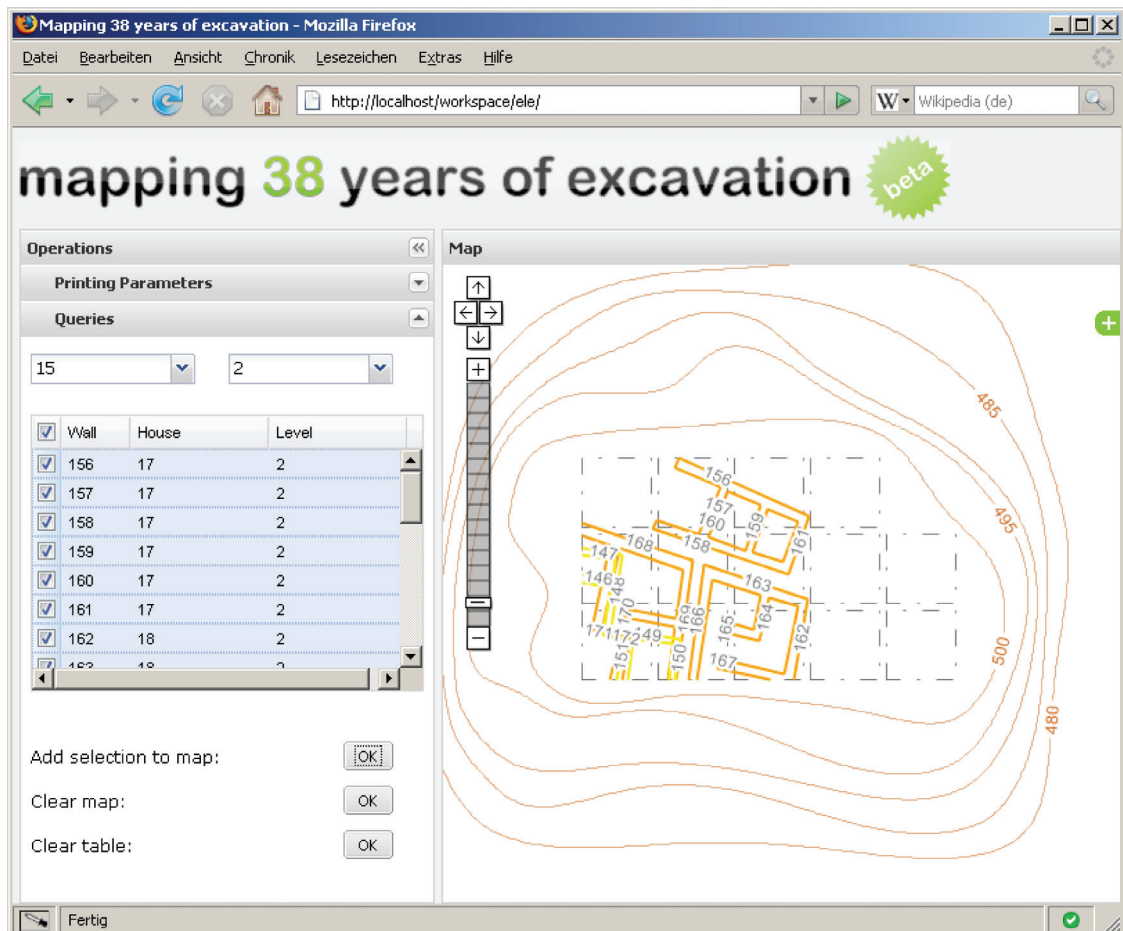


Fig. 4. Query widget.

The sample above is requesting an A3 sized PDF with portrait orientation and map scale 1:500 and two data layers: “caa:isolines” and “caa:areas”.

5.2. Map creator

The second module is a Java Servlet that works as a wrapper between the web frontend and GeoServer. (Fig. 5).

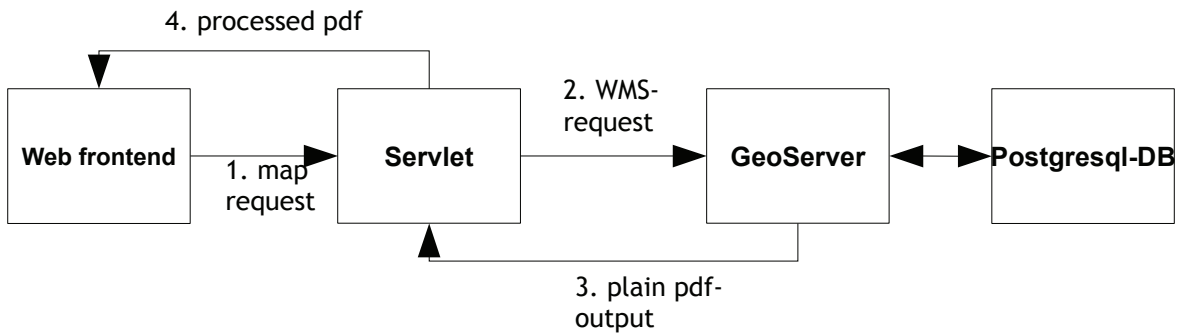


Fig. 5. Application workflow.

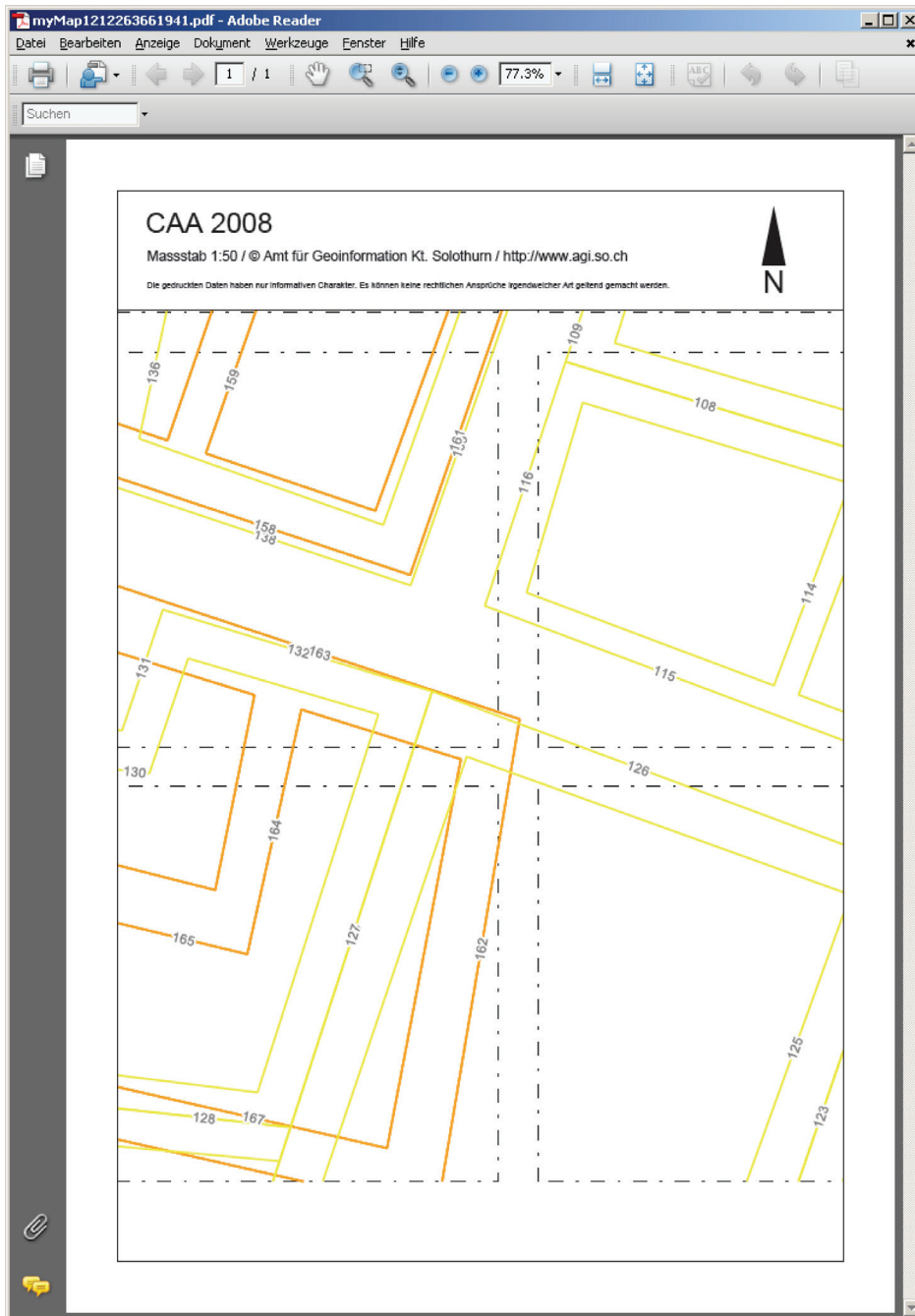


Fig. 6. Resulting print product.

The servlet generates WMS requests from the parameters passed by the web frontend and sends them to a Web Map Service. GeoServer returns a plain PDF which is postprocessed by the servlet by adding a frame and additional information, e.g. title, scale, north arrow (Fig. 6).

6. Conclusion

High quality web printing is feasible with open source components. The developed solution is reliable and fast (no noticeable delay even for an AO sized print). It is used in the administration of the Canton Solothurn to produce cadastral surveying maps. It's also possible to combine vector data with raster data (aerial photographs), but speed and hardware will be an issue since it requests an approximately 100'000'000 pixel image for an AO plot. The web frontend is not used in the administration, but thanks to the easy interface, it was not hard to implement the printing part into the existing web GIS.

Bibliography

- Deutsches Archäologisches Institut (2010). "Elephantine." http://www.dainst.org/index_56_en.html (Online, accessed 3 – February – 2010).
- Ext LLC (2010). "ExtJS." <http://www.extjs.com> (Online, accessed 3 – February – 2010).
- ESRI (1998). "ESRI Shapefile Technical Description." <http://esri.com/library/whitepapers/pdfs/shapefile.pdf> (Online, accessed 3 – February – 2010).
- Ferschin, Peter, Iman Kulitz, Andreas Jonas and Dietrich Raue (2007) "Spatial and Temporal Visualization in Archaeology. Examples from the Excavation on Elephantine, Egypt". In: A. Posluschny *et al.* (eds.) *Layers of Perception. Proceedings of the 35th International Conference on Computer Applications and Quantitative Methods in Archaeology (CAA). Berlin, Germany, April 2–6, 2007.* Kolloquien zur Vor- und Frühgeschichte, Vol. 10.
- GeoServer (2010). "GeoServer." <http://geoserver.org> (Online, accessed 3 – February – 2010).
- Institut of Architectural Sciences Vienna University of Technology (2006). "Digital Architectural and Planning / Egypt." <http://www.iemar.tuwien.ac.at/content/egypt.aspx> (Online, accessed 3 – February - 2010).
- iText Software Corp (2009). "iText PDF." <http://www.lowagie.com/iText/> (Online, accessed 3 – February – 2010).
- MapFish (2010). "MapFish." <http://www.mapfish.org> (Online, accessed 3 – February – 2010)
- OpenLayers (2010). "OpenLayers." <http://www.openlayers.org> (Online, accessed 3 – February – 2010).
- PostgreSQL Global development Group (2010). "PostgreSQL." <http://www.postgresql.org> (Online, accessed 3 – February – 2010).
- Refractions (2010). "Postgis." <http://www.postgis.org/> (Online, accessed 3 – February – 2010)
- Sun Microsystems (2010). "Java Technology." <http://www.sun.com/java/> (Online, accessed 3 – February – 2010).
- The Apache Software Foundation (2010). "Apache Tomcat." <http://tomcat.apache.org> (Online, accessed 3 – February – 2010).
- Wikipedia (2010). "Web Map Service." http://en.wikipedia.org/wiki/Web_Map_Service (Online, accessed 3 – February - 2010).

Index of Authors

A			
AGAPIOU, Athos	CD 17	DI LUDOVICO, Alessandro	CD 135
AKO, Takayuki	CD 332	DI TONDO, Sergio	326, CD 147
ALEXAKIS, Dimitrios	15, CD 15	DOULAMIS, Nicolas	CD 17
ASTARAS, Theodoros	15, CD 15	DUCKE, Benjamin	CD 580
AUSTIN, Anthony	CD 285		
B		E	
BALCISOV, Selim	CD 307	ECKKRAMMER, Florian	80, CD 153
BARBER, John	25, CD 74	ECKKRAMMER, Tobias	80, CD 153
BARNES, Adam	CD 88, 92	EITEL, Bernhard	CD 525
BARTON, Justin	173, CD 35	EKE, István	202, CD 159
BATTINI, Carlo	75, CD 41	ELIUK, Steven	332, CD 163
BEALE, C. Gareth	CD 46	ERNENWEIN, Eileen	CD 92
BENAZZI, Stefano	311, CD 52	EVE, Stuart	129, CD 441
BEUSING, Ruth	318, CD 59		
BINDING, Ceri	111, CD 402	F	
BONACCINI, Federico	51, CD 485	FANTINI, Filippo	340, CD 171
BONETTI, Costanza	311, CD 52	FARJAS, Mercedes	35, CD 177
BOULANGER, Pierre	332, CD 163	FASLER, Daniela	295, CD 512
BÖDŐCS, András	CD 67	FELDBACHER, Rainer	80, CD 153
BRUHN, Kai-Christian	CD 623	FERNÁNDEZ PERIS, Josep	288, CD 596
BRUNNER, David	156, CD 592	FERRARI, Ivan	353, CD 200
BRUNETT, Guido	156, CD 592	FIZ, Ignazio	CD 184
BURILLO, Francisco	35, CD 177	FLATEN, Arne R.	346, CD 193
		FREDRICK, David	CD 88
		FRISCHER, Bernard	413, CD 229
C		G	
CANALS, Antoni	288, CD 596	GABELLONE, Francesco	259, 353, CD 200, 476
CANO, María Ascensión	35, CD 177	GALLOTTI, Rosalia	51, CD 485
CAO, Yiwei	CD 550	GEORGOPOULOS, Andreas	CD 17
CAVERS, Graeme	25, CD 74	GEORGOULA, Olga	42, CD 298
CELLARY, Wojciech	406, CD 599	GILL, Alyson A.	361, CD 208
CHAPMAN, Sam	CD 285	GIURI, Francesco	353, CD 200
CHARNO, Michael	141, CD 471	GNESI, Diego	279, CD 476
CIRAVEGNA, Fabio	CD 285	GOODMASTER, Christopher	CD 88, 92
CORNS, Anthony	66, CD 518	GÖTTING, Marcel	376, CD 312
COTHREN, Jackson	CD 88, 92	GREEN, Chris	206, CD 213
CREMA, Enrico R.	179, CD 99	GRUPPIONI, Giorgio	311, CD 52
CSIPPÁN, Péter	CD 107	GRUPPIONI, Giulia	51, CD 485
CZAJLIK, Zoltán	CD 114	GUCCINI, Giovanni	CD 219
D		H	
DANESE, Mari	279, CD 476	HANKE, Klaus	CD 243
DELLA CASA, Philippe	298, CD 512	HEALD, Andrew	25, CD 74
DE NOBLE, Tim	CD 88	HECHT, Stefan	CD 525
DE RUNZ, Cyril	187, CD 120	HEIDEN, René	376, CD 312
DE SALVO, Marco	249, CD 380	HELLING, Harry	413, CD 229
DE SILVA, Michele	192, CD 125	HENDERSON, Jon	25, CD 74
DESJARDIN, Eric	187, CD 120	HERBIN, Michel	187, CD 120

Index of Authors

HERZOG, Irmela	212, CD 236	MAY, Keith	111, CD 402
HESHIKI, Inne	226, CD 332	MCKEAGUE, Peter	117, CD 409
HIEBEL, Gerald	CD 243	MECHELKE, Klaus	376, CD 312
HOLL, Balázs	219, CD 114, 251	MEGARRY, Will	415
HÖRR, Christian	156, 366, CD 258, 268, 592	MERICO, Davide	123, CD 423
HUSI, Philippe	86, CD 276	MILLARD, Andrew	301, CD 580
I		MITCHAM, Jenny	CD 429
IOANNIDES, Marinos	CD 17	MONTI, Alberto	273, CD 435, 491
IOANNIDIS, Charalampos	CD 17	MOSCA, Alessandro	256, CD 373
		MOSTAZA, Teresa	35, CD 177
J		N	
JANSEN, Michael	CD 550	NICOLI, Silvia	326, CD 147
JARKE, Mathias	CD 550	O	
JEFFREY, Stuart	CD 285	OIKONOMIDIS, Dimitrios	15, CD 25
JOHNSON, Ian	93, CD 291	OLSEN, Henriette Roued	129, CD 441
K		ORENGO, Hector A.	CD 184
KAIMARIS, Dimitris	42, CD 298	ORLANDI, Marco	311, CD 52
KAKARGIAS, Antonis	147, CD 559	P	
KAKOULAKI, Georgia	57, CD 503	PALET, Josep M.	CD 184
KAMERMANS, Hans	301, CD 580	PANAGIOTAKIS, Nikos	57, CD 503
KAMPEL, Martin	163, CD 606	PAYNE, Angelia	CD 92
KAYALAR, Ceren	CD 307	PESCARIN, Sofia	CD 446
KARADEDOS, George	42, CD 298	PIANTONI, Frederic	187, CD 120
KAVLAK, Ahmet Emrah	CD 307	PINI, Stefania	340, CD 171
KERSTEN, Thomas P.	376, CD 312	PIPERNO, Marcello	51, CD 485
KLAMMA, Ralf	CD 550	PIRAS, Federico	CD 219
KLEBER, Florian	100, CD 320	PIZZIOLO, Giovanna	390, CD 454
KOLLER, David	384, CD 326	POSŁUSCHNY, Axel	212, CD 236
KONDO, Yasuhisa	226, CD 332	POUNCETT, John	240, CD 363
KOROBOV, Dmitry	CD 339	PRINKE, Andrzej	406, CD 599
KOZCIAK, Simone	51, CD 485	R	
KVASSAY, Judit	202, CD 159	RABINOWITZ, Adam	134, CD 463
L		REJAS, Juan Gregorio	35, CD 177
LANGÓ, Péter	CD 348	RÉTI, Zsolt	CD 348
LAURENT, Amélie	233, CD 357	RICHARDS, Julian D.	141, CD 285, 429, 470
LIMONCELLI, Massimo	353, CD 200	RODIER, Xavier	86, CD 278
LIMP, Fredrick W.	CD 88, 92	RONDELLI, Bernardo	256, 263, CD 373, 387
LINDINGER, Elisabeth	366, CD 258, 268	ROUBIS, Dimitris	279, CD 476
LINDSTAEDT, Maren	376 CD 312	S	
LOCK, Gary	240, CD 364	SABLATNIG, Robert	100, CD 320
LÓPEZ, Raul	35, CD 177	SÁIZ, María Esperanza	35, CD 177
M		SALVI, Maria Cristina	51, CD 485
MANTEGARI, Glauco	123, 249, 256, CD 373, 380, 423	SALVINI, Riccardo	51, CD 485
MANTELLINI, Simone	263, CD 387	SANTORO, Sara	CD 491
MARTÍNEZ-CARRILLO, Ana L.	106, CD 397	SAÑUDO, Pablo	288, CD 596
MATSUMOTO, Go	226, CD 332	SARRIS, Apostolos	15, 57, CD 25, 503

Index of Authors

SAUERBIER, Martin	295, CD 512	V	
SCHUKRAFT, Gerd	CD 525	VAN LEUSEN, Martijn	301, CD 580
SEDIKOVA, Larissa	134, CD 463	VAQUERO, Manuel	288, CD 596
SEINO, Yoichi	226, CD 332	VARYTIMIADIS, Savvas	147, CD 559
SHAW, Robert	66, CD 84, 518	VERDIANI, Giorgio	CD 564
SIART, Christoph	CD 525	VERDONCK, Lieven	CD 571
DE SILVA, Michele	192, CD 125	VERHAGEN, Philip	152, 301, CD 580, 588
SOGLIANI, Francesca	279, CD 476	VITI, Sabina	390, CD 454
SOLOMON, Eric	413, CD 229	VRANICH, Alexei	CD 92
SORO, Laura	CD 533		
SPANIOL, Marc	CD 550	W	
STEINMETZ, Charlie	413, CD 229	WAGNER, Stefan	156, CD 592
STERGIOPOULOU, Eleni	147, CD 559	WALCZAK, Krzysztof	406, CD 599
STEVENS, Caitlin	CD 88	WALLER, Stewart	CD 285
STRIDE, Sebastian	263, CD 387	WANSLEEBEN, Milco	152, CD 588
SUEUR, Chris	CD 588	WINTERS, Judith	141, CD 470
SUSCA, Filippo	CD 219	WORDSWORTH, Paul	173, CD 35
SZENTPÉTERI, József	219, CD 251	YAMAGUCHI, Hiroshi	226, CD 332
T		Z	
TAKEDA, Yoshimasa	226, CD 332	ZAMBANINI, Sebastian	163, CD 606
TIMÁR, Lőrinc	399, CD 543	ZAMORA, Mar	CD 614
TIOLI, Francesco	CD 564	ZANCAJO, José Julio	35, CD 177
TOUBEKIS, Georgios	CD 550	ZHANG, Zigi	CD 285
TUDHOPE, Doug	111, CD 402	ZIEGLER, Stefan	CD 623
TÜRK, Attila A.	CD 348		