GEodata Openness Initiative for Development and Economic Advancement in Romania

Joint Research Project under the Romanian-Swiss Research Programme

Project Number: IZERZ0-142129

Architectural and Interface Design Document

Main Authors

Nadia Panchaud, Ionut Iosifescu – ETH Zurich, Switzerland

Issue / Revision: 1 / 0

Date: 30.11.2013

IKG Institute of Cartography and Geoinformation

ETH Eidgenössische Technische Hochschule Zürich Swiss Federal Institute of Technology Zurich

Universitatea Tehnică de Construcţii Bucureşti Centrul de Cercetare Ingineria Apelor Subterane

This page is intentionally left blank.

GEOIDEA.RO Report

Grant Agreement Nr:	**Subject:**	**Project Coordinator:**
IZERZ0-142129	Architectural and Interface Design Document	ETH Zurich
Issue / Revision: 1 / 0	**Internal Reference:** geoidea_deliverable_12	

Abstract:

This document discusses the architecture and the design of the geoportal for the GEOIDEA.RO project. It first reviews the requirements for the geoportal as well as the state-of-the-art portal technologies. Then, it draws upon the experience gained at the Institute of Cartography and Geoinformation at ETH Zurich and at the Groundwater Engineering Research Centre (CCIAS) at the Technical University of Civil Engineering of Bucharest to bring additional insights into the possible solutions for a geoportal. Finally, it defines and describes the chosen architecture and design for the geoportal.

This project was funded by the Romanian-Swiss Research Programme. Responsibility for the content resides in the authors or organizations that prepared it.

Main Authors: Nadia Panchaud, Ionuț Iosifescu

This page is intentionally left blank.

TABLE OF CONTENTS

This page is intentionally left blank.

TABLE OF FIGURES

This page is intentionally left blank.

1. INTRODUCTION

1.1 Purpose

This document gives a concise description of the system architecture for the GEOIDEA.RO geoportal. It aims at providing the necessary background information about requirements for the geoportal, as well as the state-of-the-art technologies in the field of web portals and, based on those, at defining and describing the system architecture of the geoportal and its components.

1.2 Scope

This document applies to the overall architecture and interface design of the geoportal. It contains information about the functional and technical requirements for the geoportal. It also documents the system architecture and its different components, from the data management to the user interface.

1.3 Overview

The governmental agencies are major geodata providers and there is an immense potential (Ilie 2013) for data to be reused, combined, and processed for secondary purposes. However, these agencies are usually reluctant to make those geodata available for many different reasons, ranging from cultural, economic, political and infrastructural ones. Presently, the issue of open access for governmental data in general, including geodata, is a highly debated topic. Open access would mean to make publicly accessible in commonly used, machine-readable format the data that the numerous governmental agencies own and collect. There have been studies about the economic effects of open data access policy and they show that governments can expect to boost their economy by opening their data (Newbery et al. 2008 & Office of Fair Trading 2006). The main objective of the GEOIDEA.RO project is to improve the scientific basis for open geodata model adoption in Romania. Indeed, it is believed that publishing governmental geodata in Romania over the Internet, under an open license and in a reusable format can strengthen citizen engagement and yield new innovative businesses, bringing substantial social and economic gain.

One part of the project consists in creating innovative technologies and tools for geodata publishing and retrieval that can be implemented in a web geoportal. The research efforts focus not only on the technical implementation of the geoportal, but the project considers as well topics such as data harmonization, user involvement and collaboration, user-friendly navigation, data quality and quality marker visualization.

1.4 Abbreviations

CSS	Cascading Style Sheets
DBMS	DataBase Management System
DEM	Digital Elevation Model
DOM	Document Object Model
ECMA	European Computer Manufacturers Association
FE	Filter Encoding
FTP	File Transfer Protocol
GCTPC	General Cartographic Transformation Package, library written in C
GDAL/OGR	Geospatial Data Abstraction Library
GeoJSON	Geographic JavaScript Object Notation
GeoRSS	Geographic Really Simple Syndication
GIS	Geographic Information System
GLSDEM	Global Land Survey Digital Elevation Model
GML	Geographic Markup Language
GUI	Graphic User Interface
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
INSPIRE	Infrastructure for Spatial Information in the European Community
JPG	Lossy image format
KML	Keyhole Markup Language
PROJ.4	Library written in C to perform conversion between cartographic projections
Proj4js	JavaScript port of PROJ.4 and GCTPC libraries
OGC	Open Geospatial Consortium
OSM	OpenStreetMap
PNG	Portable Network Graphics
RDF	Resource Description Framework
SE	Symbology Encoding
SLD	Styled Layer Descriptor
SOA	Service Oriented Architecture
SRTM	Shuttled Radar Topography Mission
SVG	Scalable Vector Graphics
UNEP	United Nations Environment Programme
WCS	Web Coverage Service
WFS	Web Feature Service
WMS	Web Map Service
WMTS	Web Map Tile Service
WPS	Web Processing Service
W3C	World Wide Web Consortium
XML	Extensible Markup Language
WKB	Well-Known Binary
WKT	Well-Known Text

2. REQUIREMENTS FOR THE GEOIDEA.RO GEOPORTAL

To achieve one of the main goals of the project that consists in making accessible geodata to the larger public, the geoportal must be user-friendly, accessible online and offer the data on demand. These aspects support an easier access to the geodata by removing usability barriers that could prevent or delay the optimal use of the geodata. Additionally, the geoportal is to follow the principles of open, modular and scalable technologies. This facilitates the development of a stable and flexible product by allowing a stepwise approach during the implementation of the geoportal.

This section discusses the functional and technical requirements for the geoportal. First, it defines the functionality that the geoportal should offer through use cases and then, it goes into the technical aspect of standards compliance for the technologies that are used.

2.1 Functionality

The geoportal should give users the possibility to find a geospatial dataset and to download it with few interaction steps. This includes the following actions:

- Selection of a dataset using an index or a list organized meaningfully (thematic navigation);
- Visualization of the dataset in the geoportal according to cartographic rules;
- Spatial navigation using at least zooming, panning and name locations search;
- Selection of an area for download: by drawing a rectangle or by entering coordinates;
- Choice of a projection system and a data format for the download;
- Download of the dataset.

Additionally, the user should be able to combine different datasets in the viewer of the geoportal. This demands the following additional actions:

- Selection of the datasets to be combined;
- Definition of the order of rendering;
- Simple modification of the symbology.

Moreover, the functionality of the geoportal should be intuitive and straightforward:

- The graphic user interface (GUI) should be kept simple and organized in a way that can be understood without extensive explanation;
- The different tools should be self-explanatory: they are represented with meaningful symbols and names.

2.2 Technologies and Standards

From a technical point of view, the requirements concern the use of **Free and Open Source Software** (FOSS) as building blocks for the geoportal. FOSS enables to cut costs regarding licensing and gives a greater flexibility in the implementation because the code is accessible and can be modified freely. However, this does not mean that solutions that are less than the **state-of-the-art** should be used for the sole reason that they are FOSS. The portal technologies chosen for the geoportal should be compliant with standards, such as the **OGC** or **INSPIRE** web services standards. Using standardized solutions and services supports the interoperability between the modules and the reuse of existing technologies. The combination of standardized and FOSS technologies should guarantee that the geoportal is independent from the user's platform and supported by all major browsers.

3. STATE-OF-THE-ART

The State-of-the-Art section presents an overview on a variety of key concepts, standards, software and technologies that are of outmost importance for the definition and correct understanding of the "Architectural and Interface Design Document" for the geoportal.

3.1 Data and Data Management Systems

The data and data management systems section reviews important aspects related to these topics and introduces specific technologies available to store data and to manage them. It describes the state of global open geospatial datasets and open software to process those data. Finally, it briefly reviews the use of data management systems and their advantages for geospatial data.

3.1.1 *Data*

In today's era of information abundance, finding satisfactory open datasets at a global scale is not easy. Nevertheless, there do exist global geospatial datasets that are open and that can be used without any restrictions for research and commercial purposes. However, their quality, resolution and themes vary greatly.

The datasets from **Natural Earth** covering physical and cultural themes as well as shaded relief at three different scales are probably the most complete and high quality datasets available and are very well suited for general background maps. Global datasets are also available from some **United Nations** (UN) affiliated organizations about themes that are part of their point of focus, such as **UNEP** for environmental data and the **UN Geographic Information Working Group** for SALB (Second Administrative Level Boundaries). Regarding global vector datasets, the **OpenStreetMap** (OSM) project should be mentioned as well, seeing that it aims at creating and distributing free geographic data for the world. However, the use of the raw data requires a deeper knowledge of GIS.

Regarding satellite images, the **Landsat** series is available with a resolution from 15m to 120m. There are two major sources of Digital Elevation Model (DEM): the **SRTM** datasets with a resolution of 1 to 30 arc second and the **GLSDEM** dataset of 90m. Global datasets, either vector or raster, can quickly reach a size at which they are not manageable anymore on desktop applications and thus require a server or database to be processed.

In order to use these data, they usually have to be processed with the help of GIS software. The main open source[1] GIS software options are **QGIS**[2] and **GRASS GIS**[3]. Both use an important library, namely

[1] Regarding proprietary GIS software, the major ones are ArcGIS from Esri, MapInfo, and the ERDAS suite by Intergraph.

[2] http://qgis.org/

the **GDAL/OGR** (Geospatial Data Abstraction Library), comprising a data model for the supported formats. The data need to be processed because they are rarely available in the format and extent required for GIS work and visualization. With those pieces of software, it is possible to mosaic (to put different tiles together), cut and clip data as well as to reproject them in the required projection system. Both of them also support the upload to and reading from a database. In the data preparation, for the specific purpose of fast visualization, there is also a step called tile cache preparation: it involves preparing static maps at different resolutions corresponding at different scales and dividing them into tiles, that can be then cached on a web application for fast delivery. **TileMill**[4] is an example of open source software that can perform this task by loading data, symbolizing them and then preparing the tiles.

3.1.2 *Data Management Systems*

To manage large amount of geodata, it is necessary to use a **Data Base Management System** (DBMS). A spatial database offers spatial types (geometry types such as point, line and polygon), spatial functions and allows for the creation of spatial indexes to improve the performance of the spatial queries and functions. A DBMS provides many advantages to manage and work with large amount of data, such as the support for multi-users (concurrency), data consistency and integrity, persistent storage, and query capabilities through a programming interface. The main characteristics of a DBMS are (Garcia-Molina et al. 2002):

- Atomicity (either the whole transaction succeeds or the whole transaction fails);
- Consistency (ensures that the database goes from one valid state to another valid one with every transaction);
- Isolation (by concurrent use, the resulting state of the system is as if the transactions had been executed one after the other); and
- Durability (once a transaction has been committed, it is there and remains even if there is a loss of power or a crash).

The two main solutions for open source[5] spatial DBMS are PostgreSQL[6] (in combination with the spatial extension PostGIS[7]) and MySQL[8]. Both are compliant with the four main characteristics of a

[3] http://grass.osgeo.org/

[4] http://www.mapbox.com/tilemill/

[5] Regarding proprietary DBSM, the well-known ones are Oracle Spatial and IBM DB2, as well as the ArcSDE middleware from Esri that allows storing and managing geospatial data within a DBMS package.

[6] http://www.postgresql.org/

[7] http://postgis.net/

DBMS, but the PostgreSQL/PostGIS is the most mature option and it is wider used due to better support in GIS software and libraries. There is also the SQLite[9] database with the extension SpatiaLite[10], as an example of portable database, which is better suited for smaller projects that require a local database.

3.2 Services and Standards

This section explores the field of services and standards for the Web and more specifically for the geospatial Web. Services and standards are important because they allow devices to communicate through services and using a common protocol set by standards, and therefore enabling the core aspects of the interoperability concept between devices.

3.2.1 *Web Services*

Web services are applications that offer functionality and that are accessible by other applications over the Web and via Internet-protocol. They support direct and remote interactions from machine to machine, using standards-based interface (Alonso et al. 2004). Thus, they allow delivering remotely available functionality from a service provider to a service consumer. Web services use the concept of request-response mechanism: a client or service consumer (a device or a person through a user interface) sends a request to the server (the service provider) and in return the server sends a response, containing the results asked for, to the client. In the field of GIS, web services are used for a wide range of tasks including the display and retrieval of data, processing operations, and search functions. In order for all these services to communicate with each other (i.e. to be interoperable), it is crucial that they operate using common standards.

3.2.2 *Standards*

Web Standards

This section shortly reviews Web standards that are useful for the development of a geoportal. Many of them are W3C standards, which stands for the World Wide Web Consortium. It is the main international organization dealing with standards for the Web. The most important standards are:

- HTML (HyperText Markup Language, W3C),
- XML (Extensible Markup Language, W3C),
- SVG (Scalable Vector Graphics, W3C),

[8] http://www.mysql.com/

[9] http://sqlite.org/

[10] https://www.gaia-gis.it/fossil/libspatialite/index

- CSS (Cascading Style Sheets, W3C), and
- JavaScript (ECMA standard).

HTML[11] is the main markup language used for the creation of web pages. It uses tags that enclose the page content and that are interpreted by the browser to display the content accordingly.

XML[12] is a markup language that encodes documents in a format readable by humans and machine. As its name says, it can be extended and used for specific purposes, such as the Geographic Markup Language (GML) and Keyhole Markup Language (KML) standards from the Open Geospatial Consortium (see below).

SVG[13] is XML-based as well and is mainly for two-dimensional vector graphics and pseudo 3D. Additionally, it supports raster file and foreign objects, such as HTML content. Most of the modern browsers can render SVG, which is in addition compliant with WMS (see below). Furthermore, the graphics are accessible by any DOM (Document Object Model) method, thus allowing for interactivity using JavaScript. Moreover, there is no problem with screen resolution as the graphics are scalable.

CSS[14] is used to encode styling (the look and feel) options for documents written in a markup language.

Finally, **JavaScript**[15] is an implementation of the scripting language standard ECAMScript[16] from Ecma International. It is widely used in client-side asynchronous programing for interactivity in the web browsers.

OGC Standards

The Open Geospatial Consortium (OGC) gathers stakeholders in the area of geo-enable Web, such as industries, government agencies and universities. They develop publicly available interface standards through consensus building (Open Geospatial Consortium 2013). The following standards are the most relevant ones for the geoportal.

- **WMS** (Web Map Service): An implementation of the WMS standard provides the client with spatially referenced 2D maps in the form of an image, such as JPG and PNG, or SVG

[11] For more information, see http://www.w3.org/standards/techs/html (accessed on 23.10.2013)

[12] For more information, see http://www.w3.org/standards/techs/xml (accessed on 23.10.2013)

[13] For more information, see http://www.w3.org/standards/techs/svg (accessed on 23.10.2013)

[14] For more information, see http://www.w3.org/standards/techs/css (accessed on 23.10.2013)

[15] For more information, see https://en.wikipedia.org/wiki/JavaScript (accessed on 30.11.2013)

[16] Fore more information, see https://en.wikipedia.org/wiki/ECMAScript (accessed on 30.11.2013)

dynamically from geographic information that can be displayed in a browser. It especially focuses on providing custom styled maps with the help of the Styled Layer Descriptor (SLD), Symbology Encoding (SE) and Filter Encoding (FE) standards (Open Geospatial Consortium 2006).

- **WMTS** (Web Map Tile Service): The WMTS standard is similar to the WMS in the fact that it renders maps as image too. But its focus is on static maps, where the extent and the scale have been constrained to discrete tiles. The service returns thus only the existing files and it allows the use of the cache capability of the browsers (Open Geospatial Consortium 2010a).
- **WFS** (Web Feature Service): The WFS standard describes a standard that should allows the sharing of geodata by allowing a client to retrieve only the geodata they are seeking and not a whole file (as it would be with FTP). In comparison with a WMS where the client receives the visualization for the data, this service delivers the actual data. (Open Geospatial Consortium 2010b).
- **WPS** (Web Processing Service): The WPS standard defines an interface that should facilitate the publication of geospatial processes, their discovery and their use by clients. Example of processing capabilities that can be offered by a WPS are reprojection algorithms, clip and mosaic functions or more complicated tasks as predictive models calculations (Open Geospatial Consortium 2007).
- **WCS** (Web Coverage Service): A WCS provides access to coverage data, representing space- or time-varying phenomena, that can either be used as inputs for another tasks or that can be directly delivered to the client (Open Geospatial Consortium 2012b).
- **GML** (Geography Markup Language): GML is an XML syntax used to model, share and store geographic information. Du to the fact that it is a text file, it is well suited to transport data over the web. (Open Geospatial Consortium 2012a).
- **KML** (Keyhole Markup Language): KML is an XML syntax to encode and transport the representations of geographic data for display in an earth browser (it was originally developed for Google Earth). Images can be included and transported under the KML-Zipped format, namely KMZ (Open Geospatial Consortium 2008).

INSPIRE

INSPIRE is a European Directive that establishes an Infrastructure for Spatial Information in the European Community[17]. It is based on the different infrastructures for spatial information operated by the Member States. The Directive goes beyond the simple coordination of spatial data themes between the Member States. It also addresses concrete measures and technical implementation

[17] http://inspire.jrc.ec.europa.eu/

specifications, such as metadata and data specifications, network services, spatial data services, as well as data and service sharing. The relevant aspects for the state-of-the-art in web services are the implementation requirements for the INSPIRE Services: the Download, View and Discovery services (European Community 2013). These services are very similar to the OGC standards regarding technologies and functionality.

Other Standards

The high complexity and sometimes inefficiency of some of the officially recognized standards (e.g. WFS, GML) lead to the development of lighter and more human-friendly formats for geodata storage and transport. Usually, the specifications for such new formats emerged from non-formal professional groups and reached the status of "de facto" standard due to their rapid adoption and spread within the international geospatial community. The most important are described bellow.

GeoRSS[18] is an extension to the common RSS (Really Simple Syndication) used on websites to notify readers of new articles or updates. GeoRSS adds geographic coordinates and features to RSS and Atom items (Turner 2006). There are currently three encodings of GeoRSS:

- Simple (lightweight format that developers and users can quickly and easily add to their existing feeds with little effort),
- GML (supports a greater range of features, notably coordinate reference systems other than WGS-84 latitude/longitude),and
- W3C Geo (RDF vocabulary for representing the latitude and longitude of spatially-located objects).

GeoJSON[19] (Geometry JavaScript Object Notation) is a recently developed format based on JavaScript Object Notation (JSON). GeoJSON is geared toward consumption by Ajax oriented applications (such as OpenLayers) because its output notation is in JavaScript format. JSON is the standard object representation in JavaScript data structures. GeoJSON extends JSON by defining a format for geometry storage within the JSON format (Obe & Hsu 2011).

One important extension, called **TopoJSON**, brings topology support and some other advanced features to GeoJSON. Rather than representing geometries discretely, geometries in TopoJSON files are stitched together from shared line segments called arcs. In addition, TopoJSON facilitates applications that use topology, such as topology-preserving shape simplification, automatic map colouring, and cartograms.

[18] http://georss.org

[19] http://geojson.org/

3.2.3 *Web Servers Implementations*

There exist several implementations of geodata servers that offer web services. The major open source[20] ones are GeoServer[21], MapServer[22], deegree[23] and QGIS Server[24]. All of them are compliant with WMS and WFS OGC standards and can work with databases for input data. GeoServer additionally offers a user interface for the administrative tasks. The QGIS server works with the desktop version, allowing the users to define and style their projects in the desktop version before publishing it on the server. MapServer is less user-friendly because it based on configuration files. MapServer, GeoServer and QGIS are projects of the Open Source Geospatial Foundation, which represents a sign of maturity and stability for an open source project. Regarding geoprocessing services, there are several open source projects such as the ZOO project[25], 52 North WPS[26] and pyWPS[27], to quote just a few of them. They implement the OGC WPS standard and allow to create and chain WPS web services in a developer-friendly framework.

3.3 Web Frameworks for Georeferenced Information

This first part of this section gives an overview of the concept of web framework and of the available options for web mapping and geoportal. These frameworks allow presenting the geospatial data to the users in a simple and coherent manner. The second part of the section reviews some well-known open source web frameworks that are available and relatively well documented[28]. There exist also many frameworks that are not as developed as the ones mentioned below or with very limited documentation and thus there are not explicitly introduced here.

3.3.1 *Basic Functionality*

Web frameworks support the development of dynamic web applications by providing libraries, packages and templates into a single connected piece of software and, thus, alleviating the tasks of

[20] Regarding proprietary solutions for WMS, one can mention the ArcGIS Server from ESRI

[21] http://geoserver.org

[22] http://www.mapserver.org/

[23] http://www.deegree.org/

[24] http://qgis.org/en/site/about/features.html#qgis-server

[25] http://zoo-project.org/

[26] http://52north.org/communities/geoprocessing/wps/

[27] http://pywps.wald.intevation.org/

[28] The following proprietary frameworks can be mentioned, but are not taken into account due to the nature of the project: Google API, Microsoft Bing, MapQuest and ESRI's ArcGIS API

dealing with low-level details. More specifically in the geospatial world, web frameworks allow to easily combine the visualization of geodata and to publish them online without having to deal with every single detail regarding projection and interactivity functions. For instance, basic functions to navigate the map, such as zooming and panning, are already built in. State-of-the-art web frameworks in the field of geographic information are able to deal with different projection systems and different data sources, from classical web map services and tile services to vector data, in JSON format, for instance. They also offer basic GUI objects that can be extended and modified with additional functions and/or plugins. They usually offer an API (Application Programming Interface) to allow an easier use and access to the libraries.

3.3.2 *Available Technologies*

OpenLayers

OpenLayers[29] is a free and open source JavaScript framework that is licensed under a FreeBSD license and is an Open Source Geospatial Foundation project. It offers the possibility to publish dynamic maps in any webpage. Moreover, it is supported by most modern browsers because it is pure JavaScript and that there are no server dependencies. OpenLayers 2 supports many different data formats, such as WMS, WMTS, GeoJSON, GML, KML, GeoRSS and WFS which enables this framework to be compatible with the major data formats that can be found in the geospatial web field. It offers spatial navigation tools and other built-in functions such as switch layer, show coordinates, pan, draw and select functions. The default projection system used is the Spherical Mercator, but there is support for reprojection, however, an additional library might be needed. On a side note, a WMS should be able to reproject automatically the layers when sending them to Open Layers. The version 3, will offer a comprehensive rewrite of the code and will integrate the WebGL library (for 3D visualization) and the Cesium library (3D globe), as well as target more precisely HTML5 and CSS3 (web standards for web page and styling). OpenLayers is also widely used by well-known institutions[30], such as The White House, the Swiss Federal Office of Topography and the Ordnance Survey, and some open source mapping projects, such as OpenCycleMap.

Leaflet

Leaflet[31] is a free and open source software and is very similar to OpenLayers. However, as said in their motto “simplicity, performance and usability”, it aims at being simpler and smaller in its core

[29] http://www.openlayers.org/

[30] see http://www.whitehouse.gov/change/ , http://map.geo.admin.ch , https://www.ordnancesurvey.co.uk/ and http://www.opencyclemap.org/

[31] http://leafletjs.com/

functions. Then, it is extended by a plugin system offering the more complex and specialized functions. It supports WMS, WMTS, GeoJSON, and vector data and allows for spatial navigation functions and layer control. Alike OpenLayers, the important standards that are HTML5 and CSS3 are also integrated. Dealing with different projection systems also requires the use of an additional library. Many significant institutions[32] use Leaflet, such as the Los Angeles Time, Flickr, foursquare, craigslist, OSM, Mapbox, CartoDB and the Institut Géographique National (France).

MapFish

MapFish[33] is based on the Pylons Python web framework and extends it with geospatial functionality. MapFish offers a JavaScript toolbox composed of the ExtJS, OpenLayers and GeoExt JavaScript libraries. It is compliant with the OGC standards, such as WMS, WFS, KML and GML. It is distributed under a BSD license and is a project of the Open Source Geospatial Foundation.

Carto.net

Carto.net[34] is a SVG framework for web mapping using as well JavaScript for the interactivity. It is free and open source under the GNU Lesser General Public License (LGPL) license, but not supported by a large community, as are Open Layers and Leaflet. The framework itself is more complicated and has not been maintained and updated as conscientiously as the two above-mentioned frameworks. However, it has notable advantages because it does not use HTML, but SVG for the GUI. For instance, the GUI automatically rescales with different screen sizes and resolutions, as well as opens possibilities for native GIS vector data rendering with SVG. As all major browsers now support SVG, this option is becoming more attractive than before.

[32] see http://maps.latimes.com/crime/ , http://www.flickr.com/map , https://foursquare.com/ , http://www.craigslist.org , http://www.openstreetmap.org , https://www.mapbox.com/, https://cartodb.com/ and http://www.ign.fr/

[33] http://mapfish.org/

[34] http://www.carto.net/papers/svg/samples/

4. STATE OF OWN RESEARCH

This section describes the actual state of research at the Institute of Cartography and Geoinformation (IKG) at ETH Zurich and at the Groundwater Engineering Research Centre (CCIAS) at the Technical University of Civil Engineering in Bucharest in the field of cartographic web technologies and geoportals. It draws on our experience from previous and current projects to show what are possible successful implementations of geoportal technologies and where the drawbacks are.

4.1 Three-Tier Architecture and Service-Driven Web Mapping

Trends in web mapping applications are going from monolithic architecture toward more modular approaches such as service-oriented architectures (SOA). These architectures use the mechanism of request-response between a service provider and a service consumer (often the web browser or another service) to serve cartographic content to the web through the HTTP protocol. The generic architecture for service-driven web mapping is a three-tier architecture. It divides the application into three tiers: a data tier, a logic tier and a presentation tier (see Figure 1). The data tier hosts the data in DBSM where they are organized in a collection of tables. Spatial databases allow to manage geospatial data using geometry type, spatial indexes and spatial functions to increase performance and be independent from GIS proprietary formats. The logic tier holds the services and makes the link between the databases (data tier) and the web browser (presentation tier). The services perform the tasks that are needed to bring the data and/or their visualization to the user, such as processing, symbolizing and sending the map content to the presentation tier. The presentation tier is usually located on the client machine, where the users can interact with the application, most of the time through a GUI.

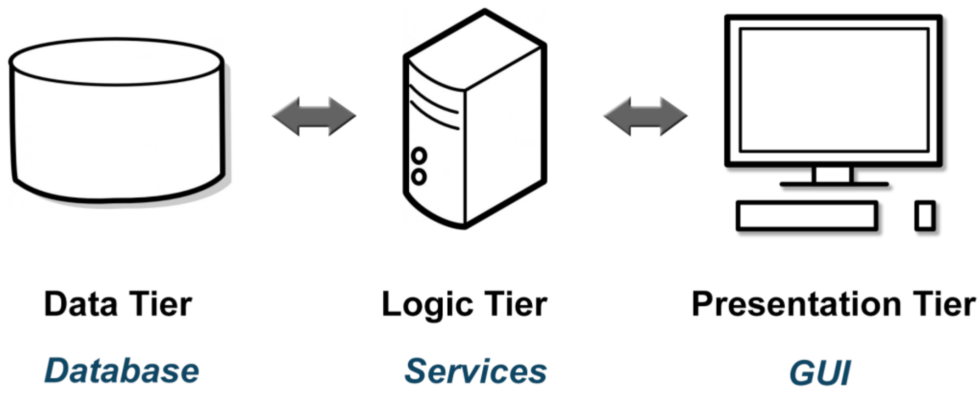

Figure 1: Three-tier architecture

The advantages of using a three-tier architecture are found in its modularity; each tier is separated from the other one and can be located on different devices and thus allowing those components to be easily scalable and updatable. It also means that they are independent from each other, which simplify updates and modifications on each of the tier.

The three-tier architecture brings additional advantages regarding interoperability and platform independency, because it uses a service-driven approach. The services that are loosely coupled and highly modular can perform most of the tasks in a chain process before delivering the final map visualization to the GUI. Thus, it avoids software-hardware incompatibilities, as well as limitations in processing power and storage capacity on the end-user side.

4.2 GeoVITe Platform

The GeoVite platform has been continuously developed at the Institute of Cartography and Geoinformation (IKG) since 2004. The initial goal for the geoportal was to provide geodata and tutorials in the framework of the teaching functions for the geomatics studies at IKG. The name comes from that initial purpose as an E-learning platform: GEOvisualization and Interactive Training Environment. But it soon proved to be able to support more daring tasks, namely to offer the possibility of directly downloading geodata through the online GUI in a user-friendly manner. The Beta version was released in 2010 and has been continuously updated and enriched with new content. It now serves as the main distribution channel for geodata (instead of a file system) within the ETH network.

The GeoVITe project has offered the opportunity to develop knowledge in the areas of web services and graphic user interface design. The geoportal is built using a three-tier architecture and the platform uses several web services for the visualization, the geoprocessing functions and the support functions (see Figure 2). The project started with proprietary technologies and then gradually moved toward open sources solutions. As a result, there are now two parallel solutions, a proprietary and an open source one, that are used in an interoperable manner. Both solutions use the WMS standard for the visualization. For the other web services, the proprietary solution option is built upon an ArcGIS Server and an ESRI ArcSDE database. The geoprocessing services are defined using the functionality in the ArcGIS Toolbox, which include extraction functions and export to other formats, orchestrated through geoprocessing workflows built with the Model Builder. The geocoding service is also done using ArcGIS. Regarding the open source solutions, it is built upon a PostgreSQL/PostGIS database coupled with QGIS Server for the WMS. The geoprocessing and support web services are created using a rapid service composition framework called JOpera[35]. All the geoprocessing tools are done with Java and the C++ GDAL/OGR library and are service enabled through the framework.

[35] http://www.jopera.org/

The GUI itself has been implemented with FOSS technologies from the beginning, by using the Carto.net framework. However, many extensions were made to the original framework to support the specific needs and characteristics of the project. The GeoVITe platform has also been successfully used in different ways for other projects at the IKG such as the OSPER - Open Support Platform for Environmental Research (for the visual management and exploration of sensor data available in Switzerland) or GeoHistory project (for the online access to historical maps of the Canton of Zurich, Switzerland).

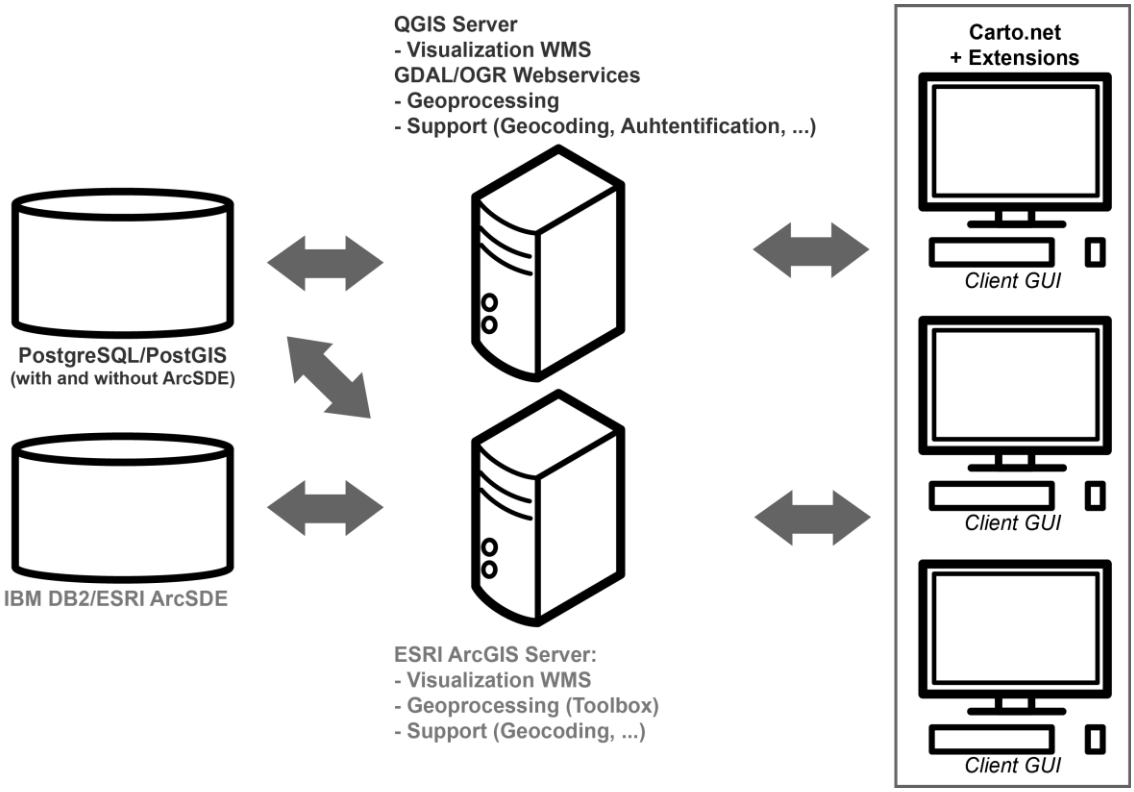

Figure 2: Schematic architecture of the GeoVITe platform

4.3 eHarta platform

eHarta is an online web mapping platform, developed by the CCIAS team members, that offers free access to thousands of old map sheets and atlases in georeferenced form (Craciunescu et al 2011). The system was designed following a distributed architecture, entirely based on free and open source software. The content is managed by Textpattern, a powerful and flexible open source content management (CMS) application. For supplementary and specific functionality, custom modules were built. Other free applications are providing server-side functionality: MySQL (relational database management system), PHP, Python, Java (serverside scripting languages), Apache (webserver), Tomcat (servlet container), phpMyAdmin, phpPgAdmin (web clients for database management). For geospatial data management, top open source applications were also integrated in the website: PostGIS (vector geospatial data storage), GeoNetwork (geospatial data catalog and metadata editor), GeoServer (standard geospatial server for serving data via WMS), GeoWebCache (Java based WMTS/TMS server, with pluggable caching mechanisms and rendering backends), OpenLayers and GeoExt (client webmapping application), GDAL (data processing). The information flow between the various server side applications and the front-end graphical interface is determined by the interaction with the portal users and their requests (see Figure 3).

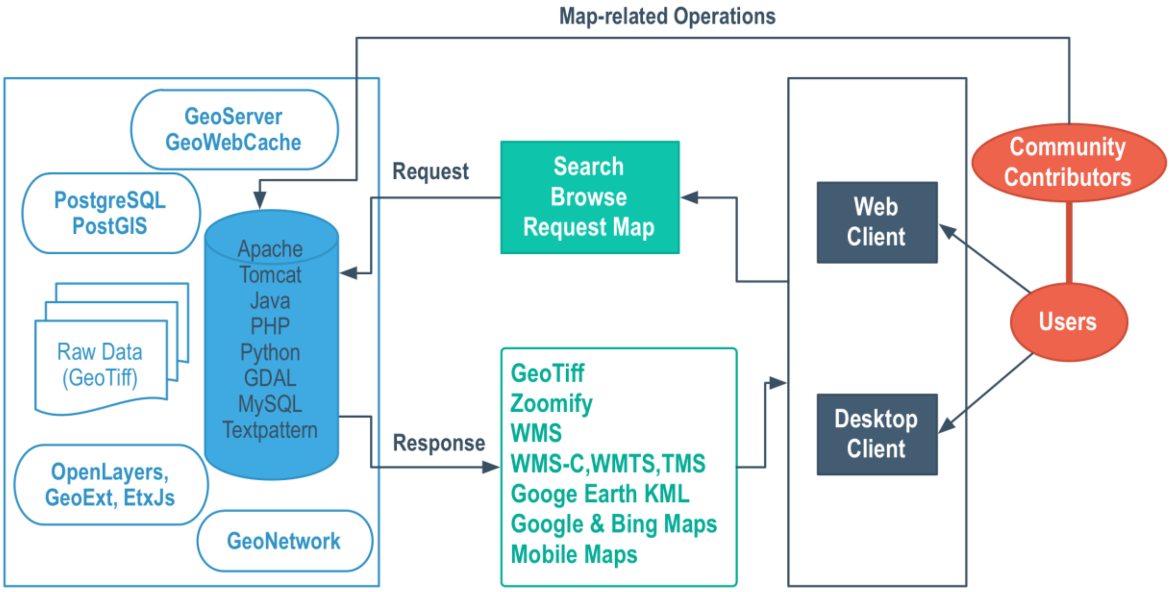

Figure 3: eHarta Architecture

4.4 Prototype of GEOIDEA.RO – The Geoportal

The prototype of the geoportal for the GEOIDEA.RO project is based on the GeoVITe framework, taking into consideration the generic architectural elements of the eHarta platform and using exclusively an open source solution (in order to be consistent with the openness values promoted by the project). It required the adaptation of several aspect of the framework, for instance the projection system and the geocoding service that were originally developed for the Swiss territory. Thus, for the name search, the Geonames.org service was integrated with the help of the proj4js library[36]. Additionally, because it is a prototype and that the project has not yet advanced to the point that actual data of the government agencies are made available, basis open geodata had to be found to populate the geospatial database: the choice went to the OSM data. The OSM data required a complex processing before being put online for download, but they proved to be adequate. The prototype allowed to test a fully open source pipeline and to show that it was not only possible, but also that the performance was not lower.

[36] The proj4js library is a port to JavaScript of the Proj.4 and GCTPC libraries (written in C) and enables the transformation of points from one coordinate system to another, including datum transformations.

The GUI is organized similarly to the GeoVITe GUI and offers distinct zones with distinct functions regarding the spatial and thematic navigation, the area for selection, name search and the download tools (see Figure 4).

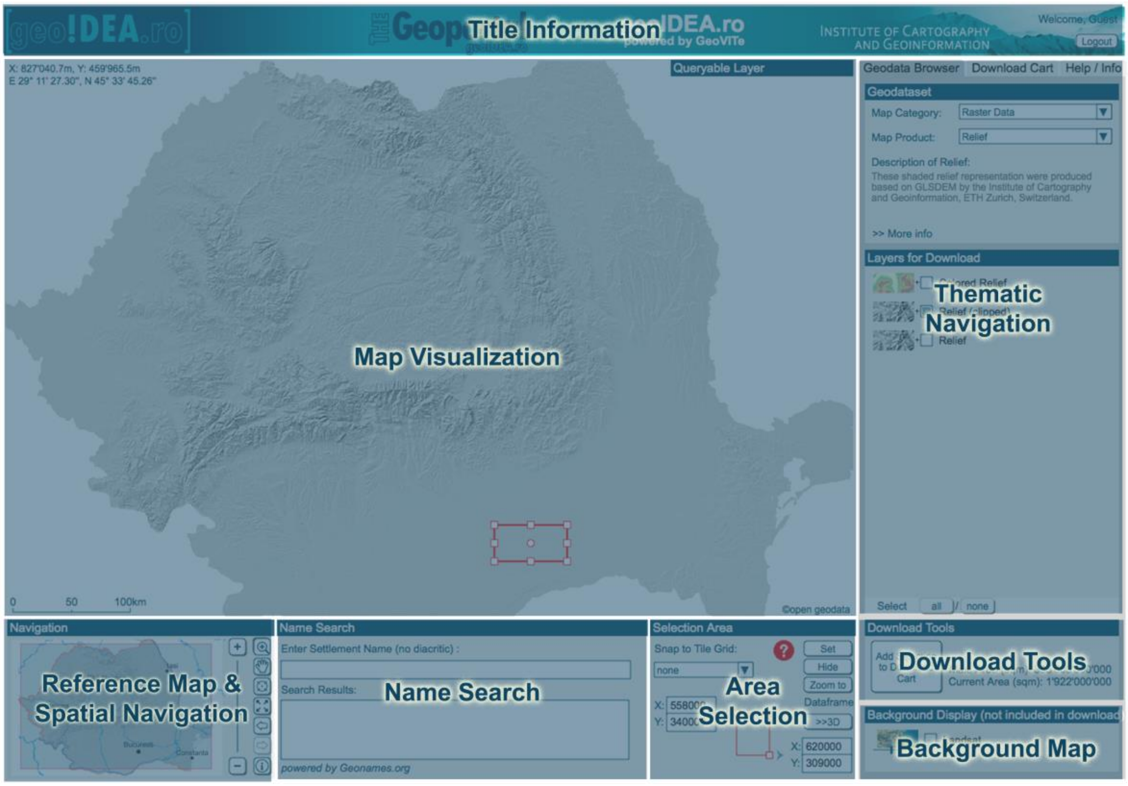

Figure 4: GUI organization of the GEOIDEA.RO geoportal

5. DEFINITION OF THE GEOIDEA.RO GEOPORTAL ARCHITECTURE

This section describes the chosen system architecture of the geoportal for the GEOIDEA.RO project. It first gives a description of the schematic architectural representation and then explains the different main components of the geoportal. Finally, it describes in more details the planned implementation and how the different components communicate and how the flow of information goes from the client to the geospatial database hosting the data and back.

5.1 Architectural Representation

The system is designed using a three-tier architecture comprising a Presentation Tier, a Logic Tier and a Data Tier (see Figure 5). The Presentation Tier contains two Graphic User Interfaces (the GUI Pro and the GUI Lite, i.e. the clients) and manages the input from and output to the user. The Logic Tier handles the geospatial web services and provides the communication between the two other tiers; it is the logic of the application. The Data Tier consists of the geodata stored in tables and managed by one or more databases.

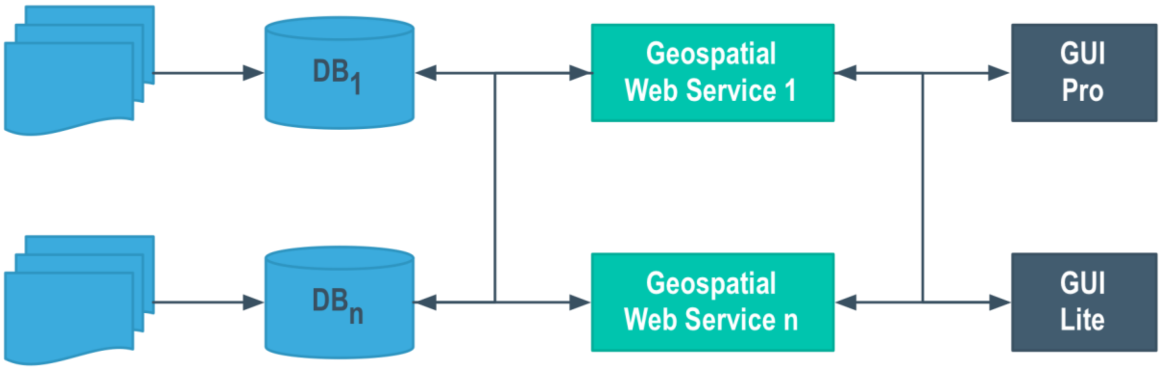

Figure 5: Schematic architecture of the geoportal

5.2 Logical View

This section presents the main components of the geoportal and their role in the geoportal. The Figure 6 shows the main components, into which the system is broken down, organized according to the three tiers. These main components are explained below.

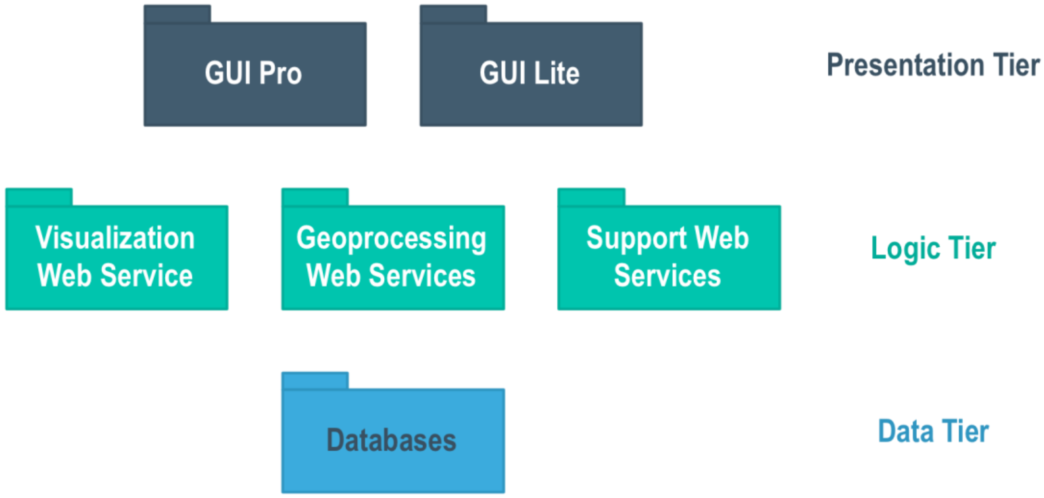

Figure 6: Main components of the geoportal

5.2.1 *Main Components*

Data Tier

The **databases** store the geodata in tables and files that can be queried and accessed by the different web services and that are on different servers. The databases guarantee the consistency of the data and their availability to several web services simultaneously (for instance, for visualization and download).

Logic Tier

The Logic Tier contains the WMS and other web services that perform geoprocessing tasks. The different web services receive input from the Presentation Tier through user's input or from other web services that the user solicited and then, they will run their task and send the output back to the user through the Presentation Tier. The **WMS** provides a representation of the geodata to the client. The other web services fulfil **geoprocessing tasks** as well as other **support tasks** needed for the geoportal (login, compressing files, etc.). For instance, the clip, reproject and export web services are responsible for preparing the data for download.

Presentation Tier

The Presentation Tier is where the visualization of the available data is presented to the users and it is the component through which the user can interact with the system. The geoportal offers two different GUIs: The Pro and the Lite. The **GUI Pro** is a complex GUI offering all the functionality of the GeoVITe framework, allowing to access metadata, download data and combine datasets in a single visualization directly in the geoportal, whereas the **GUI Lite** offers only the most basic functionality (visualization, spatial and thematic navigation) and demonstrates how to integrate services, such as WMS, beyond the main geoportal. This tier also contains a **welcome page**, where the access to the system is granted, based on the acceptance of a disclaimer (no registration needed).

5.3 Implementation View

This section gives a more detailed description about the plans to implement the architecture described above. It details the relation between the different architectural components and describes the flow of information according to the functional requirements (see 2.1).

The entire architecture is based on modular and scalable components that are interoperable with each other and external systems. As a consequence, they should allow to easily increase the complexity of the services and the amount of geodata that can be served without disrupting the rest of the architecture. It must be mentioned that because the project deals with the development of new technologies for the publication of governmental geodata in an open framework, the architecture is based solely on thopen source solutions. Furthermore, it is compliant, at the service and client levels, with most international standard, such as OGC and W3C and the web-based user interface should be supported by all major browsers. The following diagram (see Figure 7) illustrates the system architecture to be implemented.

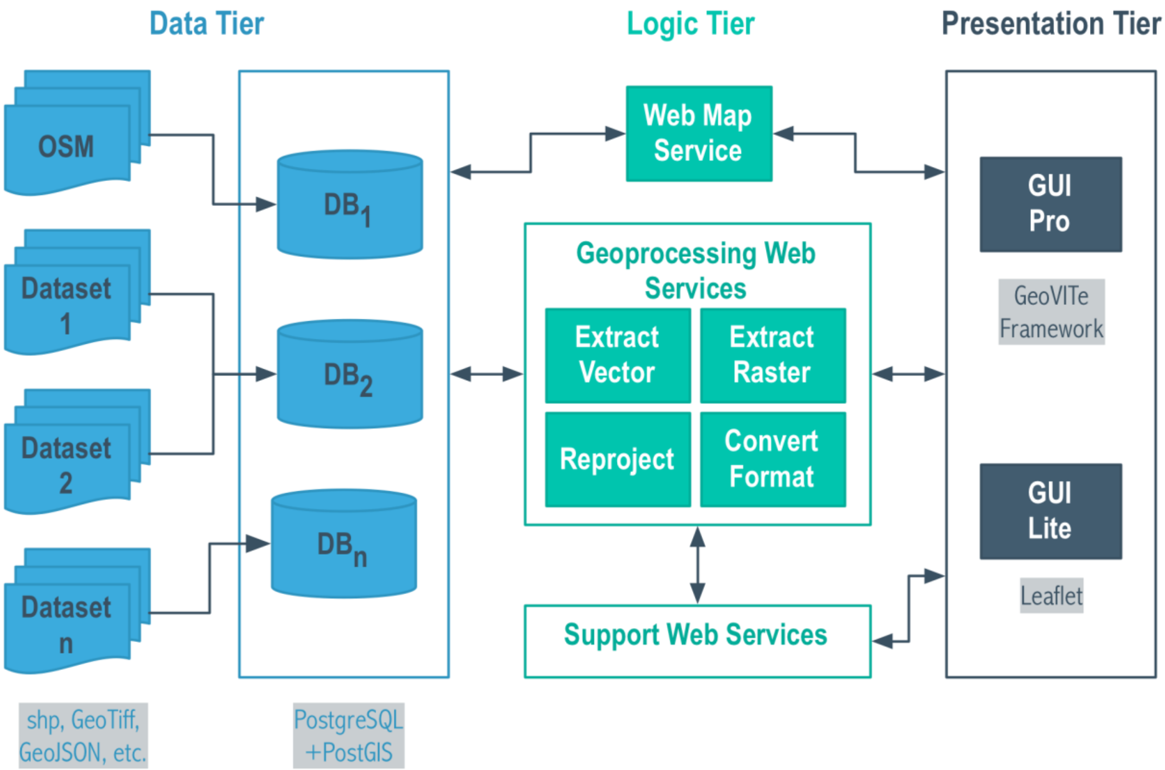

Figure 7: Implementation schematic for the geoportal

5.3.1 *Data Tier*

Several spatial databases store and manage the geodata and hold them at the disposition of the logic tier for visualization and processing services. Thus, they support spatial functions and indexes, which provide better performance for retrieval and processing spatial data. The different datasets must not necessarily be on one central database, because the system is modular and the Logic Tier can deal with different data sources. The Data Tier is managed using a PostgreSQL database with the PostGIS spatial extensions because this solution is the most mature. This database management system supports the import from and export to all major spatial formats such as Shapefile, GeoTiff, GeoJSON, Well-Know-Text (WKT), etc.

5.3.2 *Logic Tier*

This tier is the core of the platform and where the different functionalities are being executed. It is responsible for aggregating the data from the different sources and for sending them together to the Presentation Tier.

The WMS is compliant with the OGC standards and is responsible for the visualization of the data. The client (Presentation Tier) sends an HTTP request to the WMS specifying the extent, projection system, layers and styles, and format of the desired map and then the WMS gets the different data from the databases, symbolizes them and send the image back to the client.

For the geoprocessing web services and support services, the WPS standards will be used only for demonstration purposes if the performance is not adequate (this issue has arisen in other projects at IKG). In addition to the OGC standards, the core implementation of the processing services will have the freedom to use custom solutions based on a rapid service composition framework. This choice for a custom solution could allow an easier development of flexible and specific geoprocessing tools. Because the GDAL/OGR library is used for the input and output format, these services would be compatible with the other standardized part of the architecture.

Furthermore, it should also be possible to integrate third-party services. For instance, a third party service, Geonames.org, can be used for the geoname search as it already allows finding place names and their actual location in the form of coordinates.

5.3.3 *Presentation Tier*

The Presentation Tier is represented by the user interfaces through which the user can interact and make use of the geodata and services. This tier allows to enter inputs and to send requests to the different web services of the Logic Tier, as well as to receive and visualize the output responses of those web services. When the user requests to see a dataset, the GUI translates the parameters that the user chooses into a request for the required WMS and sends that request. The WMS will reply with the corresponding map as an image of the dataset and spatial region requested that will then be displayed in the user interface. Similarly, when downloading a dataset, the Presentation Tier will parameterize the spatial extent of the selected area and the name of the dataset; then it will form a request to the corresponding geoprocessing service, and finally it will offer the results to the user through the GUI.

6. CONCLUSION AND OUTLOOK

In the field of web mapping and geoportal technologies, there are many available and equivalent solutions. Thus, in the area of free open source software, it is important to choose technologies either with a wide community support or for which in-house know-how is available and, therefore, that are familiar and respond to specific needs.

The choices made for the system architecture of the geoportal are based on state-of-the-art solutions, standard technologies and well-known tools. Building upon the experience existing at the IKG and CCIAS as well as upon international standard and open technologies, the system architecture of the geoportal for the GEOIDEA.RO project is based on a modular three-tier architecture. Using standardized technologies enables an easier integration of the different tiers and the optimal leverage of the strengths of those different technologies while allowing for an interoperable solution. The modular aspect additionally allows to use already developed modules that have proven their strengths in other projects and to integrate them in the architecture.

The future tasks regarding the prototype of geoportal will focus first and foremost on the transition to the first official version of the geoportal. Then, in a second step, tasks regarding the data visualization and the GUI interface, and the further development of geoportal technologies will be continued.

Regarding the GUIs in a medium-term perspective, the implementation of the **GUI Lite** must be realized, as well as the accompanying tutorials enabling users to make use of the web map services already in place. An additional goal is to optimize the **GUI pro** to facilitate the manipulation of the geoportal and the access the open data available. Furthermore, in the matter of the **visualization of the datasets** in the geoportal should be improved. For instance, scale-dependent symbology should be better defined to offer a smoother experience for the exploration of data.

For the long term planning, the special function **User Map** should be further developed and better integrated within the **GUI pro**. This also involves the development of a cartographic smart wizard to guide the users when combining datasets from the geoportal. This aims at adapting semi-automatically the symbolization of the different datasets based on the chosen combination of datasets. Such a wizard requires the formalization of cartographic principles into rules that can be implemented into a generic data model in the logic tier. New geoportal tools and technologies should not only be developed to support the above-mentioned tasks, but also in the form of new functionality, especially regarding smart cartographic symbolization. These new tools should also take into account the feedback received from the other parts of the project and integrate them in the implementation process.

7. REFERENCES

Alonso G., Casati F., Kuno H. and Machiraju V. (2004): Web Services: Concepts, Architectures and Applications. Berlin: Springer Berlin-Heidelberg.

Craciunescu V., Constantinescu S., Ovejanu I., Rus I. (2011): Project eHarta: a collaborative initiative to digitally preserve and freely share old cartographic documents in Romania. e-Perimetron, Vol. 6, No. 4, 2011 p. 261-269.

European Commission (2013): Network Services.Infrastructure for Spatial Information in the European Community. http://inspire.jrc.ec.europa.eu/index.cfm/pageid/5/list/2 (retrieved on 24.10.2013).

Garcia-Molina H., Ullman J. D. and Widom J. (2002): Database Systems: The Complete Book. International Edition. New Jersey: Prenctice Hall.

Ilie C. M. (2013): Report on Existing Open Data Strategies. Deliverable for the *GEodata Openness Initiative for Development and Economic Advancement in Romania.*

Newbery D., Bently L. and Pollock R. (2008): Models of Public Sector Information Provision via Trading Funds. Cambridge University, commissioned jointly by the Department for Business, Enterprise and Regulatory Reform (BERR) and HM Treasury.

Obe R., Hsu L. (2011): PostGIS in Action. Stamford. Manning Publications.

Office of Fair Trading (2006): The commercial use of public information (CUPI). http://tinyurl.com/7pu672v (retrieved on 5.12.2013).

Open Geospatial Consortium (2006): OpenGIS® Web Map Service Implementation Specification. Version 1.3.0. http://www.opengeospatial.org/standards/wms (retrieved on 23.10.2013).

Open Geospatial Consortium (2007): OpenGIS® Web Processing Service. Version 1.0.0. http://www.opengeospatial.org/standards/wps (retrieved on 23.10.2013).

Open Geospatial Consortium (2008): OGC® KML. Version 2.2.0. http://www.opengeospatial.org/standards/kml (retrieved on 23.10.2013).

Open Geospatial Consortium (2010a): OpenGIS® Web Map Tile Service Implementation Standard. Version 1.0.0. http://www.opengeospatial.org/standards/wmts (retrieved on 23.10.2013).

Open Geospatial Consortium (2010b): OpenGIS Web Feature Service 2.0 Interface Standard. Version 2.0.0. http://www.opengeospatial.org/standards/wfs (retrieved on 23.10.2013).

Open Geospatial Consortium (2012a): OGC® Geography Markup Language (GML) — Extended schemas and encoding rules. Version 3.3.0. http://www.opengeospatial.org/standards/gml (retrieved on 23.10.2013).

Open Geospatial Consortium (2012b): OGC® WCS 2.0 Interface Standard- Core: Corrigendum. Version 2.0.1. http://www.opengeospatial.org/standards/wcs (retrieved on 23.10.2013).

Open Geospatial Consortium (2013): About OGC. http://www.opengeospatial.org/ogc (retrieved on 5.12.2013).

Turner (2006): Introduction to Neogeography. O'Reilly Media.