Architecture of Distributed Multiplatform GIS for Meteorological Data Analysis and Visualization

Marko Kovačević[1], Aleksandar Milosavljević[1], Vladan Mihajlović[1] and Dejan Rančić[1]

Abstract – **This paper presents a distributed multiplatform architecture of geographic information system (GIS) capable of providing flexible and efficient geospatial and meteorological data analysis and visualization to a broad spectrum of user profiles. Traditionally, geographic information systems were realized as monolithic and platform-dependent applications and used by rather small group of GIS professionals. With the development of computer hardware, Internet and availability of geospatial data in recent years, GIS evolved and adapted to new environments that emerged. Grater availability has lead to greater importance and use of GIS in many different domains. In this paper we will focus on architecture specification that allows distribution of geospatial data and functions to different hardware and software platforms. The proposed architecture consists of four subsystems: Desktop GIS, GIS Server, Web GIS and Mobile GIS. Each of the identified subsystems has its different role and intended user groups in the system as a whole. Great diversity of meteorological data applications (weather forecast, hail suppression, climate change monitoring, aviation, agriculture, military etc.) implies variety of different GIS user needs and profiles. This paper will focus on identifying subsystem roles and possible use case scenarios that can generally be applied for all meteorological data applications.**

Keywords – **GIS, Multiplatform Architecture, Meteorology.**

I. INTRODUCTION

A geographic information system (GIS) is a special type of computer-based information system, tailored to store, process, and manipulate geospatial data [1]. The ability of GIS to handle and process both location and attribute data, distinguishes GIS from other information systems. It also establishes GIS as a technology that is important for a wide variety of applications [2]. Traditionally, geographic information systems were built as monolithic and platform-dependent applications [3], but, with the development of computer hardware and availability of geospatial data in recent years, GIS evolved and adapted to new environments that emerged [4]. Greater availability has lead to greater importance and use of GIS in many different areas. What was reserved for a small group of professionals in the past now became available worldwide, using Internet, on personal computers and mobile devices.

Modern distributed GIS architectures rely on a client/server model where clients provide user access to geospatial data, while one or more servers provide their sharing. The task of sharing geospatial data and functions can be done by extending existing traditional desktop GISs with a set of "web interfaces". This kind of extension forms a GIS Server component. Besides GIS Server, proposed architecture consists of Desktop GIS, Web GIS and Mobile GIS subsystems. Each of the identified subsystems has its different role and intended user groups in the system as a whole. Great diversity of meteorological data applications (weather forecast, hail suppression, climate change monitoring, aviation, agriculture, military etc.) implies variety of different GIS user needs and profiles. This paper will focus on identifying subsystem roles and possible use case scenarios that can generally be applied for all meteorological data applications.

In order to build a distributed GIS that is open for connecting to a variety of different geospatial data sources, commonly acceptable standards for implementation of these web interfaces are needed [5]. Currently, Open Geospatial Consortium, as an international industrial consortium with an aim of developing publicly available standards in the field of GIS, has several implementation specifications that standardize this field. Our functional needs for web interfaces were met by three of these specifications. The first is Web Map Service (WMS) Implementation Specification [6] that describes a web service interface for custom maps retrieval. The second is Web Feature Service (WFS) Implementation Specification [7] which describes a web service interface for manipulating, querying, and retrieval of geospatial entities using both spatial and non-spatial criteria. The third is Web Coverage Service (WCS) Implementation Specification [8] which describes a web interface for retrieval of geospatial data as "coverages" – digital geospatial information representing space-varying phenomena. This paper focuses on a global GIS architecture specification that is based on previously mentioned specifications and that allows distribution of geospatial data and functions to different hardware and software platforms.

The paper is organized as follows: in the second part, we define subsystem-level architecture with emphasis on communication between them. The third part is dedicated to Desktop GIS and GIS Server subsystems. In the fourth part client side subsystems are discussed in more detail. The fifth part presents a case study concerning subsystems' use in meteorological data applications. Finally, in the conclusion, the achieved results are summarized.

II. SUBSYSTEM-LEVEL ARCHITECTURE

The architecture of proposed GIS solution consists of four subsystems: Desktop GIS, GIS Server, Web GIS, and Mobile

[1]Marko Kovačević, Aleksandar Milosavljević, Vladan Mihajlović and Dejan Rančić are with the Faculty of Electronic Engineering at University of Niš, Aleksandra Medvedeva 14, Niš 18000, Republic of Serbia, E-mail: marko.kovacevic@elfak.ni.ac.rs, alexm@elfak.ni.ac.rs, wlada@elfak.ni.ac.rs, ranca@elfak.ni.ac.rs.

GIS. Like all modern distributed GIS architectures, proposed solution architecture relies on a client/server model.

Communication between server side subsystems is depicted by the component UML diagram shown in Fig. 1. The responsibility of Desktop GIS component, in the server side of the system, is creation and preparation of a content that can be distributed via GIS Server. GIS Server implements OGC WMS, WFS and WCS services in order to distribute geospatial data and functions to its clients. For preparation of geospatial content, Desktop GIS relies on use of a project file. The project file is an XML document that contains specification of all coordinate systems, geodata services, coverage data, layers, and styles that are used for geospatial and coverage data organizing and presentation. While Desktop GIS allows creation and manipulation of GIS projects, GIS Server only use previously prepared projects for geospatial content distribution.

Component UML diagram in Fig. 2 shows client side subsystems of proposed GIS solution. Mobile GIS and Web GIS rely on web interfaces provided by the GIS Server (WMS, WFS and WCS) in order to provide their own functionalities. Desktop GIS also has the ability of utilizing web interfaces provided by the GIS Server. GIS Server is intentionally left out in Fig. 2 because, from the client standpoint, it can be any GIS server (or servers) that implements OGC standardized WMS, WFS and WCS.

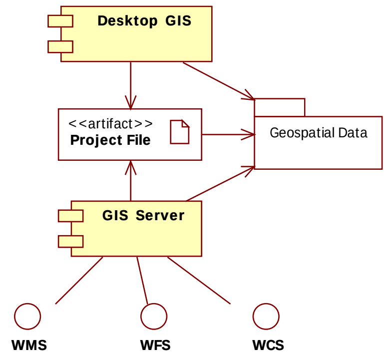

Fig. 1. Server side subsystems

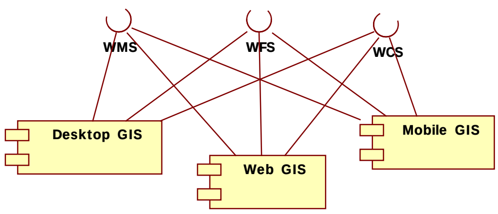

Fig. 2. Client side subsystems

III. Desktop GIS and GIS Server

The described relation between Desktop GIS and GIS Server implies tightly coupled implementation of these two components. Core of Desktop GIS application and GIS Server system is GIS framework – a set of C++ class libraries developed in Laboratory for Computer Graphics and GIS (CG&GIS Lab), on Faculty of Electronic Engineering in Niš. GIS framework implements common data model and most of the functionalities. Desktop GIS application and GIS Server system service can be seen as a kind of an interface toward them. The main advantage of sharing the same core between these two subsystems is in maintainability. When some new feature (or change) is added to the framework (e.g. support for new data format) it automatically becomes present in both Desktop GIS application and GIS Server. A simplified overview of the implementation architecture of GIS framework along with Desktop GIS and GIS Server is shown in Fig. 3. There are two packages (Visual C++ projects) responsible for production of executable components of Desktop GIS application and GIS Server system service, while the other packages (i.e. projects) represents previous introduced GIS framework.

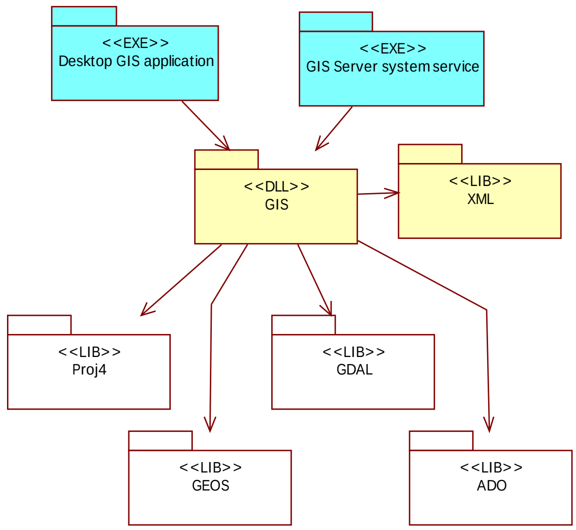

Fig. 3. Simplified implementation architecture of Desktop GIS and GIS Server

The GIS framework is implemented with support of four open source libraries: Proj4, GDAL, GEOS, and ADO. Proj4 [9] is a library that implements mechanism for definition of various coordinate reference systems (i.e. projections) and reprojection of coordinates between them. GDAL [10] is a library for raster geospatial data formats management, while GEOS [11] library enables support for spatial querying and geoprocessing. Finally, ADO [12] library contains a set of classes that simplifies database accessing and management.

The majority of the framework implementation (e.g. data model) is contained within GIS dynamic library classes, while XML is only helper library for accessing, managing, and creating XML documents. Since Desktop GIS and GIS Server share the same implementation framework, both of them are tied to the same platform. In our development we relied on Microsoft Visual C++ 2008 and Microsoft Foundation Classes (MFC). As a result, current Desktop GIS and GIS Server platform is limited to Microsoft Windows and Microsoft Windows Server series of operating systems.

IV. CLIENT SIDE SUBSYSTEMS

As we already stated, client side of the proposed GIS solution includes Web GIS, Mobile GIS and Desktop GIS. Web GIS and Mobile GIS completely depend on services and geospatial data supplied by GIS Server. Desktop GIS has the capability of utilizing web interfaces provided by the GIS Server, but also has other responsibilities and functionalities, as described in section 3.

Web GIS's primarily use is geospatial data visualization and querying. GIS Server's WMS service is used for retrieval of custom maps in form of raster images. WFS service enables query execution and results retrieval in form of GML [13] encoded geographic features. WCS service enables coverage data retrieval for defined spatial and temporal constraints. The implementation of Web GIS heavily relies on OpenLayers [14], open source JavaScript library developed by the Open Source Geospatial Foundation. The implementation of Web GIS is based on several web technologies such as: PHP, JavaScript, AJAX, and .NET.

Mobile GIS has similar background architecture, i.e. it also use WMS, WFS and WCS services of GIS Server, but the implementation and intended use is quite different. As it is intended to run on hand-held devices, as "on field" tool for data acquisition and situation awareness, the platform of choice was Microsoft Windows Mobile. Mobile GIS subsystem is implemented using C# programming language and .NET Compact Framework. Unlike Web GIS that drives huge support form OpenLayers library, Mobile GIS's communication with WMS, WFS and WCS services is implemented from scratch. Since Mobile GIS, besides map visualization and coverage data querying, also deals with geospatial data acquisition, GIS Server's WFS service has to be extended with transactional part of the specification. Transactional WFS, in addition to querying, allows manipulation of geographic features using insert, update and delete operations.

As previously stated, Desktop GIS is able to use web interfaces provided by GIS Server: WMS, Transactional WFS and WCS. Classes that implement communication between Desktop GIS and above listed web interfaces are contained within GIS library (Fig. 3). Since Desktop GIS and GIS Server share the same GIS framework, GIS Server can also act as a client of some other GIS server, thus enabling the creation of cascade connection between servers. Being the most powerful platform of the three, Desktop GIS provides additional functionalities which are not implemented in Web GIS and Mobile GIS: various tools for WCS coverage data analysis and flexible style management for visual representation of WCS coverage data, WMS raster images and WFS geographic features.

V. CASE STUDY

Using GIS in meteorological domain involves variety of different user needs and profiles. Meteorological data applications are vastly diverse: weather forecast, hail suppression, climate change monitoring, aviation, agriculture, military etc. From the GIS user perspective, each meteorological data application requires different user profiles that differ in:

- Education level,
- Responsibility for decision making,
- Working environments,
- Experience with GIS tools, etc.

For all numbered reasons, using single platform solution (one GIS application) can be inappropriate for wider user audience. In order to analyze the use of proposed GIS subsystems, we should identify several typical GIS user profiles of meteorological data application in general:

- Meteorological data analyzers
- Decision makers
- Simple viewers
- Users involved in "on field" operational tasks, and
- Applications administrators.

Meteorological data analyzers are the most complex group of users that includes experts in particular meteorological (or some related) domain which are also specialized for GIS tools use. For that reason, powerful Desktop GIS applications are what this spectrum of users needs. Furthermore, these users often need specialized tools for geospatial and meteorological data analysis. Our solution includes development and delivery of such specialized tools within Desktop GIS. These tools are developed as plugins – DLLs that conform to a predefined interface used for interaction with the application and underlying geospatial data. When present, a plugin is automatically loaded into Desktop GIS application and can be accessed from appropriate menu item. A sample screen shoot of Desktop GIS application is shown in Fig. 4.

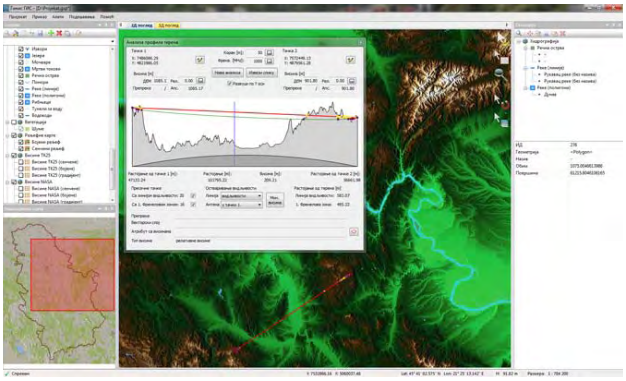

Fig. 4. Screen shot of Desktop GIS application

Decision makers are users that have authority to make decisions (e.g. hail suppression activity initiation, aviation flight plan regulation, taking appropriate actions in agriculture, military etc.). Although usually experts in particular domain, decision makers are characterized as casual GIS users. Their primarily tasks are not GIS related, i.e. they (can) only use GIS as another information source for decision making. More often, decision making must by fast, so this category of users must be quickly presented with the exact as possible information they need. By our opinion, simplified applications with easy-to-use user interfaces, along with adequate geospatial and meteorological data content are what decision makers need. Client side subsystems, primarily Web

GIS and Mobile GIS applications, are what we see as the most appropriate solution for this category. A sample screen shots of Web GIS and Mobile GIS are shown in Fig. 5.

The third group, simple viewers, includes broader audience of generally non-specialized people interested in some aspect of particular meteorological domain (e.g. weather forecast). Web GIS is the most suitable tool for this user group.

Many meteorological applications include “on field” operational tasks. In terms of GIS functions we recognized two major categories: simple meteorological data collecting and support for enhanced situation awareness. On field operations, regarding the environment, require specialized computing hardware and the running software. Mobile GIS, as the proposed tools of choice, basically satisfies listed requirements for this user group.

Finally, the last identified category of GIS users is involved in applications administration. The personnel appointed to these tasks must be well educated and especially trained to support maintenance of the proposed GIS system. Although they directly do not use any of the presented subsystems they must have good knowledge of all of them.

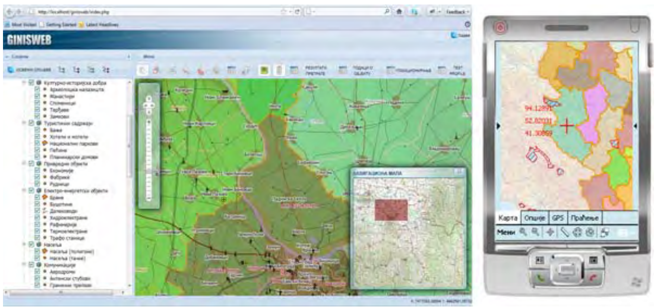

Fig. 5. Screen shot of Web GIS (left) and Mobile GIS (right)

VI. Conclusion

Great diversity of meteorological data applications implies large spectrum of potential GIS users. In this paper we focused on two main issues. Firstly, we presented distributed multiplatform architecture of GIS applicable in meteorological applications that consists of four subsystems: Desktop GIS, GIS Server, Web GIS and Mobile GIS. Second issue was identifying subsystem roles and possible use case scenarios that can generally be applied for all meteorological data applications.

Proposed GIS solution architecture relies on a client/server model. Server side subsystems are GIS Server and Desktop GIS. In this context, the main role of Desktop GIS application is content preparation in terms of data acquisition and processing. In order to distribute prepared data, we rely on well-known concept of project file. Once the project file and corresponding geospatial data are prepared, they can be shared using WMS, WFS and WCS services of GIS Server subsystem. Client side subsystems are Web GIS, Mobile GIS and Desktop GIS. Web GIS and Mobile GIS completely depend on services and geospatial data supplied by GIS Server, while utilization of web interfaces provided by GIS Server is only part of the responsibilities and functionalities of Desktop GIS. Primary use of Web GIS is geospatial data visualization and querying, and it is intended for broader spectrum of less specialized users. Mobile GIS, as it runs on hand-held devices, is basically intended for “on field” data acquisition and situation awareness. As the most powerful client side subsystem that provides various specialized analysis tools along with basic GIS functionalities, Desktop GIS is primarily indented for specialized users.

Acknowledgement

Research presented in this paper is funded by Ministry of Education and Science, Republic of Serbia as part of the project “Environmental Protection and Climate Change Monitoring and Adaptation”, Nr. III-43007.

References

[1] Worboys, M., Duckham, M., GIS: A Computing Perspective. Second Edition. CRC Press, Boca Raton, FL, 2004.
[2] Chang, K., Introduction to Geographic Information Systems, Third Edition. McGraw-Hill, New York, NY, 2005.
[3] Wong, S. H., Swartz, S. L., Sarkar, D., “A Middleware Architecture for Open and Interoperable GISs”, IEEE MultiMedia, 9(2)(2002) pp. 62-76.
[4] Shekhar, S., Vatsavai, R. R., Sahay, N., Lime, S., “WMS and GML based Interoperable Web Mapping System”, Proceedings of the 9th ACM international symposium on Advances in geographic information systems, Atlanta, Georgia, USA, 2001, pp. 106-111.
[5] Milosavljević, A., Đorđević-Kajan, S., Stoimenov, L., “An Application Framework for Rapid Development of Web based GIS: GinisWeb”, in: J.T. Sample, K. Shaw, S. Tu, M. Abdelguerfim eds. Geospatial Services and Applications for the Internet, Springer, New York, 2008, pp. 49–72.
[6] La Beaujardière, J. ed., Web Map Service Implementation Specification (Ver. 1.1.1), document 01-068r3, Open Geospatial Consortium Inc., 2002, http://www.opengeospatial.org/standards/wms.
[7] Vretanos, P. A., ed., Web Feature Service Implementation Specification (Ver. 1.0.0), document 02-058, Open Geospatial Consortium Inc., 2002, http://www.opengeospatial.org/standards/wfs.
[8] Whiteside, A., Evans, J. D., ed., Web Coverage Service (WCS), Implementation Specification (Ver. 1.1.0.), document 06-083r8, Open Geospatial Consortium Inc., 2006, http://www.opengeospatial.org/standards/wcs.
[9] PROJ.4 – Cartographic Projections Library, http://proj.osgeo.org.
[10] GDAL – Geospatial Data Abstraction Library, http://www.gdal.org.
[11] GEOS – Geometry Engine, Open Source, http://trac.osgeo.org/geos.
[12] Antollini, C., "A set of ADO classes - version 2.20", The Code Project, 6 Sep 2005, http://www.codeproject.com/KB/database/caaadoclass1.aspx.
[13] Cox, S., Cuthbert, A., Lake, R., Martell, R. eds., OpenGIS Geography Markup Language Implementation Specification (Ver. 2.1.2), document 02-069, Open Geospatial Consortium, 2002, http://www.opengeospatial.org/standards/gml.
[14] OpenLayers – Free Maps for the Web, http://www.openlayers.org.