Ginis MvcGen – a Prototype of WebGIS Application Source Code Generator

Miloš Bogdanović[1], Aleksandar Stanimirović[2], Leonid Stoimenov[3]

Abstract –**In this paper we will present our prototype WebGIS application source code generator called Ginis MvcGen. It can be used to create a WebGIS applications based on the usage of Model-View-Controller design pattern. This tool can contribute to faster development of high quality WebGIS software solutions. Ginis MvcGen can be easily used for creating WebGIS applications that integrate the framework for the development of WebGIS client with information obtained from heterogeneous information sources. It allows coupling of information retrieved from heterogeneous sources with map display of objects connected to obtained information using the Ginis MvcGen built-in JavaScript editor.**

Keywords – **Web, GIS, source code generator**

I. Introduction

Development and usage of Web applications is not a novelty in the field of software engineering. Since the appearance of Web technology, a substantial growth has been registered worldwide in the use of this technology. Methodology of Web application development has changed in parallel with the development of Internet technologies. From application based on static content and design, followed by the usage of languages that enable dynamic page content and layout creation, Web applications have reached the level of collaborate systems in which the role of user is similar to application designer role. Due to its operating environment, Web applications have the ability to access data from heterogeneous sources of information. These applications can combine (integrate) obtained information. Therefore, they can be observed as access points to the different content that is displayed at user request.

The influence of Web-based solutions is also present in the field of Geo-Information Systems (GIS). Web-based GIS solutions are often referred to as Web Geo-Information Systems (WebGIS) [1]. WebGIS solutions represent the evolution of standard GIS system as a result of changing the basic operating and development environment. These solutions implement a standard set of GIS functionality, and make a step forward in terms of opportunities offered by the new operating environment. One of the basic advantages of using WebGIS solutions is the possibility of coupling various types of information with spatial context. Therefore, WebGIS solutions can be treated as access point for integration of information from distributed and heterogeneous sources.

The time devoted to the implementation of GIS systems has led to the emergence of a series of framework used for the development of different types of GIS systems (desktop, Web, mobile) [2][3][4][5]. The advantage that these solutions offer GIS system developers is effective implementation of the future solution’s basic structure. The development of specific functionalities, usually information integration components, requires further implementation. Possible decrease of time spent in application development could be achieved through usage of frameworks that support information integration [6]. In order to be able to integrate retrieved data, GIS application frameworks use descriptions of information sources. Information source description is usually in the form of XML schema that provides information source access methods and characteristics of obtained information. Another possibility is the usage of semantic description of information sources in the form of intermediate semantic description layer created using special intermediate XML languages.

Although the existence of a framework significantly reduces the time required for system development, in most cases, usage of frameworks requires expert domain knowledge for development of basic application structure. The time that GIS system development experts initially spend could be further reduced if developers were using GIS system source code generator [7]. WebGIS source code generators produce most of the source code needed for application proper functioning.

This paper presents Ginis MvcGen prototype application. This tool is intended to contribute to faster development of high quality WebGIS software solutions. Generated solutions are often used as portals to spatial and geoinformations [8][9]. They can be also used as front-ends of spatial data infrastructures (SDI) [10].

II. Source code generator architecture

Ginis MvcGen is our prototype WebGIS application source code generator (Fig. 1.). It can be used to create a WebGIS applications based on the usage of MVC (Model-View-Controller) design pattern. In addition to the basic GIS functionality, Ginis MvcGen generates source code required for the acquisition and visualization of information from heterogeneous information sources. Obtained information can

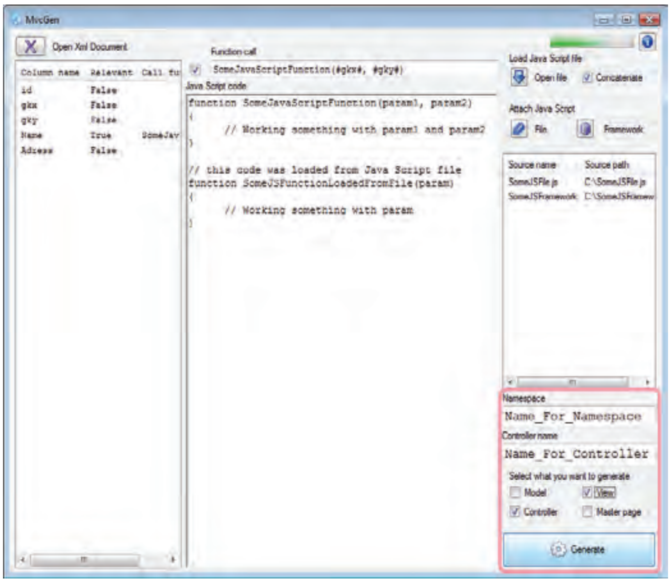
[1]Miloš D. Bogdanović is with the Faculty of Electronic Engineering, Aleksandra Medvedeva 14, 18000 Niš, Serbia, E-mail: milos.bogdanovic@elfak.ni.ac.rs

[2]Aleksandar S. Stanimirović is with the Faculty of Electronic Engineering, Aleksandra Medvedeva 14, 18000 Niš, Serbia, E-mail:aleksandar.stanimirovic@elfak.ni.ac.rs

[3]Leonid V. Stoimenov is with the Faculty of Electronic Engineering, Aleksandra Medvedeva 14, 18000 Niš, Serbia, E-mail: leonid.stoimenov@elfak.ni.ac.rs

be integrated and linked to specific functions of the generated application using the Ginis MvcGen built-in JavaScript editor.

Ginis MvcGen generates source code files needed for the development of WebGIS applications using ASP.NET MVC framework [11]. Generated files include source code files needed for application to be able to communicate with different information sources (RDBMS, SOAP services, REST services) and source code files for visualization of retrieved information. Each of the information sources is described through corresponding XML schema. XML schemas that describe the individual information sources contain data necessary for communication between generated application and described information sources. Source code generator performs parsing of XML schema that describes information source.



Column name	Relevant	Call fu
id	False	
gkx	False	
gky	False	
Name	True	SomeJav
Adress	False	

☑ SomeJavaScriptFunction(#gkx#, #gky#)

Java Script code

```
function SomeJavaScriptFunction(param1, param2)
{
      // Working something with param1 and param2
}

// this code was loaded from Java Script file
function SomeJSFunctionLoadedFromFile(param)
{
      // Working something with param
}
```

☑ Concatenate

Source name	Source path
SomeJSFile.js	C:\SomeJSFile.js
SomeJSFramework	C:\SomeJSFramew

Namespace: Name_For_Namespace

Controller name: Name_For_Controller

Select what you want to generate

□ Model ☑ View
☑ Controller □ Master page

Fig. 1. Ginis MvcGen – WebGIS application source code generator interface

XML schema parsing provides generator with the description of communication methods used for retrieval of data from information source. Furthermore, it provides generator with the descriptions of data sets that will be retrieved from the information source. Ginis MvcGen provides users with opportunity to create custom JavaScript code and/or use existing JavaScript files. These files will be integrated into the generated application.

Descriptions of information sources in the form of XML schemas represent the starting point in the process of WebGIS application source code generation. This can be observed in Fig. 2. which represents the Ginis MvcGen source code generator architecture. Ginis MvcGen contains IO module which is responsible for obtaining the XML schemas of information sources. Information source descriptions can be obtained via the Internet or they may be obtained locally. It is necessary to transform obtained description into classes that represented the object representation of information sources characteristics. For this purpose, Ginis MvcGen uses a collection of utility classes. Utility classes are divided into two major groups: the classes that describe the types of data groups that can be obtained from the information source and classes that describe methods of access to particular information source. Instantiation of utility classes is performed by information source schema parser. This module receives information source description from IO module and performs its parsing. For each of the information sources, information source schema parser instantiates one object of utility class that describes the access method for a particular information source. Also, this module instantiates appropriate number of utility classes that describe data group types according to a number of different data groups obtained from the information sources.

In addition to the visualization of retrieved information, Ginis MvcGen allows the creation of WebGIS applications, which will integrate retrieved information, using one of the WebGIS client development frameworks. In the current development stage, Ginis MvcGen uses OpenLayers framework [5] for the implementation of the basic GIS functionalities. In order to generate an application that effectively integrates retrieved information with OpenLayers framework, Ginis MvcGen generator users have the ability to create custom JavaScript functions using JavaScript Editor module. The usage of JavaScript Editor module provides users with opportunity to invoke functions that belong to OpenLayers framework and combine them with information obtained from particular information source. Data obtained from specific information source can be transmitted into the custom function in the form of custom function input parameters. Groups of data that can be used for integration with OpenLayers framework are specially marked inside the XML schema of information source. If the information integration demands the usage of existing function defined in separate files, Ginis MvcGen offers the possibility of including these files into the generated application.

Generation of files that contain source code of generated application is implemented within the Code Generator Engine module. Code Generator Engine module performs integration of Information Source Schema Parser services and WebGIS framework Loader module services with custom functions created by Ginis MvcGen users. Code Generator Engine module contains static functions that can be divided into 4 groups according to their user interface capabilities and MVC component that they belong to:

1. Model generation methods
2. View generation methods
3. Master page generation methods
4. Controller generation methods

Code Generator Engine can use an arbitrary subset of static functions depending on the components that are expected as output (red frame in Fig. 1). Components expected as output are defined by Ginis MvcGen users through tools’ interface. They can be combined with previously generated components simply by adding them into existing solution.

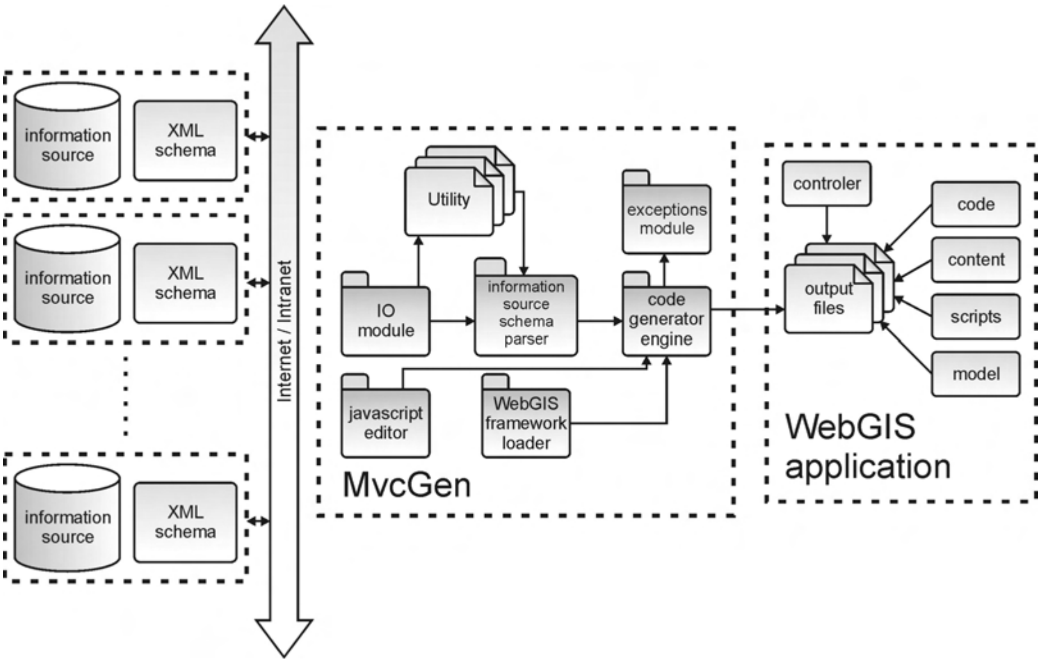


Fig. 2. Ginis MvcGen architecture

III. SOURCE CODE GENERATION PROCESS

WebGIS application source code generation process is initiated through loading of the description of one or more information sources as shown in Fig. 3.

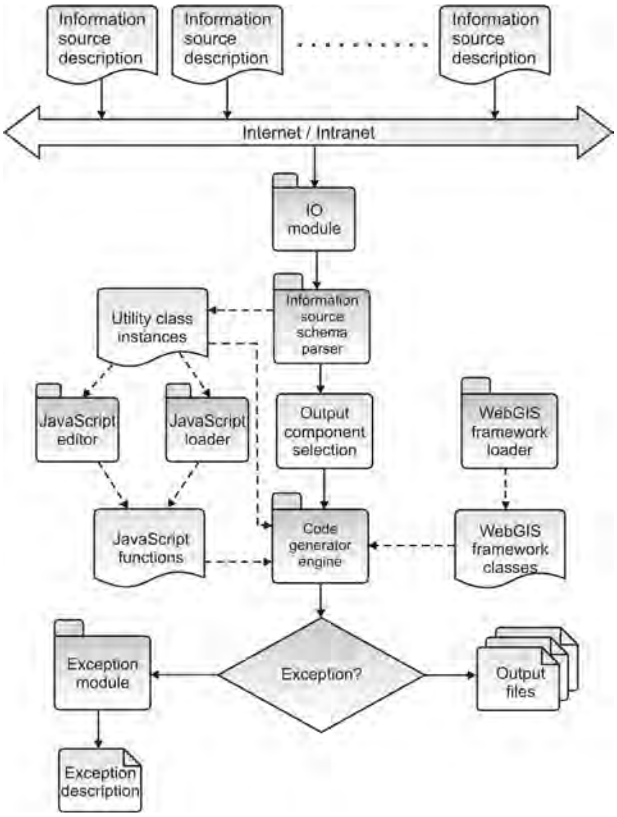


Fig. 3. Source code generation process

Information obtained from individual information sources will be integrated with the WebGIS client development framework. After selecting appropriate information source, Ginis MvcGen IO module obtains its XML schema description. IO module performs internal transformation of XML schema into two groups of utility classes: classes that describe method used for communication with the information source and classes that describe the data that can be retrieved from information sources. Generated utility class represents the input to the Information Source Schema Parser module. On the basis of information source description, an Information Source Schema Parser instantiates utility class e.g. creates an object representation of information source XML schema description. This fulfils all the prerequisites necessary for Ginis MvcGen user to determine how the integration of obtained information with WebGIS client development framework will be performed.

After parsing the description of information source, the integration of acquired data and WebGIS client development framework can be carried out. Ginis MvcGen gives users the ability to create custom functions using JavaScript language. As previously stated, inside the information source description it is possible to determine the group of data that can be used as input parameters of custom JavaScript functions. These functions integrate retrieved data with WebGIS client development framework. Based on the input parameters of custom JavaScript functions, it is possible to generate source code that invokes procedures from WebGIS client development framework.

Thus, the behavior of application that will be generated is conditioned by the information obtained, e.g. the integration of information from a geographic and non-geographic domain is performed. The development of custom functions is performed using the JavaScript editor that is embedded in

Ginis MvcGen. If Ginis MvcGen users have a pre-prepared functions, these functions can be integrated into generated application by JavaScript Loader module. Both functions created in the built-in JavaScript editor and functions loaded by the JavaScript Loader module are passed to Code Generator Engine module.

Code Generator Engine of Ginis MvcGen can generate a complete application or some parts of the application. Components selection is based on user preferences. As input, Code Generator Engine module receives the following components: WebGIS client development framework libraries, custom JavaScript functions (from built-in JavaScript editor and/or JavaScript Loaded module), object representation of information sources description and a list of components to be generated. Code Generator Engine module incorporates information retrieval procedure calls into the appropriate components of generated WebGIS application source code. Also, it mounts WebGIS development framework and binds it with retrieved informations by incorporating custom JavaScript functions.

IV. CONCLUSION AND FUTURE WORK

Ginis MvcGen belongs to the group of source code generators. It represents a good starting point when developing WebGIS software solutions. With the advantages and disadvantages that it brings, this tool can contribute to faster development of high quality WebGIS software solutions. Ginis MvcGen prototype source code generator can be easily used for making WebGIS applications that integrate the framework for the development of WebGIS client with information obtained from heterogeneous information sources. For the purposes of information integration Ginis MvcGen uses specially developed XML schemes for the description of information sources. It allows coupling of information retrieved from heterogeneous sources with map display of objects connected to obtained information. Ginis MvcGen requires user input in the form of functions that perform the integration of information e.g. connecting information obtained from a single information source with the framework for the development of WebGIS clients.

The downside of using Ginis MvcGen source code generator is reflected in the fact that it implies that users understand the semantics of information obtained from individual information sources. If this is not the case, Ginis MvcGen source code generator usage would be very difficult. For this reason, it is necessary to extend the architecture Ginis MvcGen. Instead of the previously used XML scheme for the description of information sources, a semantic description of information sources should be used. Such a description would be a priori known and easier to understand to Ginis MvcGen users. Namely, it would be enough to once get familiar with the characteristics of the semantics of information that can be obtained from heterogeneous information sources instead of having the obligation of knowing the characteristics of each of the information sources.

XML schema for the description of information sources that is currently being used would be replaced by the intermediate layer of mappings between groups of information that can be obtained from individual information sources and concepts that belong to semantic descriptions of information sources e.g. local ontology. For the purpose of creating intermediate mapping layer, it is possible to use existing techniques for semantic annotation of information sources. Since Ginis MvcGen uses RDBMS, SOAP and REST services as information sources, it is possible to use some of the techniques for deep Web annotation, Semantic Annotation for WSDL (SAWSDL) [12] and SA-REST [13]. In this way, users Ginis MvcGen would be given a unified view of heterogeneous information sources, and an opportunity to decide on information that needs to be integrated inside the application. The choice would be made at the concept level of local (domain) ontology.

ACKNOWLEDGEMENT

Research presented in this paper were partially funded by the Ministry of Science of the Republic of Serbia and PD Jugoistok Niš, within the project "Intelligent integration of geo-, business and technical information on the company level", ev. No 13003.

REFERENCES

[1] A. Shaig, *“An Overview of Web based Geographic Information Systems”*, SIRC, Dunedin, New Zealand, 2001

[2] MapTools, *“ka-Map”*, 2007, http://ka-map.maptools.org, last accessed April 2010

[3] MapTools, *“Chameleon”*, 2007, http://chameleon.maptools.org/index.phtml, last accessed April 2010

[4] A. Mitrović, D. Mitrović, S. Djordjević-Kajan, *“A Scalable, Object-Oriented GIS Framework”*, International Society for Photogrammetry and Remote Sensing, Workshop on New Developments in Geographic Information Systems, Milan, Italy, 1996

[5] Open Source Geospatial Foundation, *“OpenLayers”*, 2007, http://www.openlayers.org, last accessed April 2010

[6] L. Stoimenov, A. Stanimirović, S. Djordjević-Kajan, *“Realization of Component-Based GIS Application Framework”*, 7th AGILE Conference on Geographic Information Science, Heraklion, Greece, 2004

[7] N. Chalainanont, J. Sano, T. Minoura, *“Automatic Generation of Web-based GIS/Database Application”*, OSGeo Journal, Vol. 3, pp. 5-17, ISSN:1994-1897, 2007

[8] *“GEON project”*, 2002, http://www.geongrid.org/, last accessed April 2010

[9] *“NGDC Interactive Map Services”*, 2010, http://www.ngdc.noaa.gov/maps/interactivemaps.html, last accessed April 2010

[10] *“EIONET - European Environment Information and Observation Network”*, 1994, http://www.eionet.europa.eu/, last accessed April 2010

[11] S. Sanderson, *“Pro ASP.NET MVC Framework”*, ISBN: 978-1-4302-1008-5, Apress, 2009

[12] Semantic Annotation for WSDL Working Group, *“Semantic Annotations for WSDL and XML Schema”*, 2007, http://www.w3.org/2002/ws/sawsdl/, last accessed April 2010

[13] A. P. Sheth, K. Gomadam, J. Lathem, *“SA-REST: Semantically Interoperable and Easier-to-Use Services and Mashups"*, IEEE Internet Computing, vol. 11, no. 6, pp. 91-94, 2007