OWGIS 2.0: Open source Java application that builds Web GIS interfaces for desktop and mobile devices

Olmo Zavala-Romero
Department of Scientific Computing
Florida State University
Tallahassee, United States
osz09@fsu.edu

Eric P. Chassignet
Center of Ocean-Atmospheric Prediction Studies
Florida State University
Tallahassee FL, United States
echassignet@coaps.fsu.edu

Jorge Zavala-Hidalgo
Centro de Ciencias de la Atmósfera
Universidad Nacional Autónoma de México
DF, Mexico
jzavala@atmosfera.unam.mx

Panagiotis Velissariou
Center of Ocean-Atmospheric Prediction Studies
Florida State University
Tallahassee FL, United States
pvelissariou@coaps.fsu.edu

Harshul Pandav
Department of Computer Science
Florida State University
Tallahassee FL, United States
hnp13@my.fsu.edu

Anke Meyer-Baese
Department of Scientific Computing
Florida State University
Tallahassee FL, United States
ameyerbaese@fsu.edu

ABSTRACT

OWGIS version 2.0 is an open source Java and JavaScript application that builds easily configurable Web GIS sites for desktop and mobile devices. This version of OWGIS generates mobile interfaces based on HTML5 technology and can be used to create mobile applications. The style of the generated websites is modified using COMPASS, a well known CSS Authoring Framework. In addition, OWGIS uses several Open Geospatial Consortium standards to request data from the most common map servers, such as GeoServer. It is also able to request data from ncWMS servers allowing the display of 4D data from NetCDF files. This application is configured by XML files that define which layers, geographic datasets, are displayed on the Web GIS sites. Among other features, OWGIS allows for animations; vertical profiles and vertical transects; different color palettes; dynamic maps; the ability to download data, and display text in multiple languages. OWGIS users are mainly scientists in the oceanography, meteorology and climate fields.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Spatial databases and GIS*; J.2 [**Computer Applications**]: Physical Sciences and engineering— *Earth and atmospheric sciences*

General Terms

Algorithms, Design, Management

SIGSPATIAL '14, November 04-07 2014 Dallas/Fort Worth, TX, USA

http://dx.doi.org/10.1145/2666310.2666381

Keywords

OWGIS, Web GIS, OpenLayers, GeoServer, ncWMS

1. INTRODUCTION

In this work the new features included in OWGIS version 2.0 are presented. A description of OWGIS Version 1.0 is in Zavala-Romero et al. [16] and technical information is available at `http://owgis.org`. In order to have a complete view of the application, and not only the new features in version 2.0, some of the description in Zavala-Romero et al. (2014) [16] is followed here.

During recent decades geospatial data availability had a tremendous grew as a result of availability of satellite information, the increased use of numerical models and the use of Global Positioning Systems (GPS). In addition, the development of several geographic information systems (GISs) and Web GIS sites that assist in the analysis and visualization of georeferenced data raised [10]. Some of the Web GIS sites allow to access raw data and are able to overlap layers. OWGIS is used to generate Web GIS sites with a minimum set of features, allowing an efficient publication of 4D geospatial data without programming.

A common procedure for publishing georeferenced data on a Web GIS site includes the following steps. First, obtain the data. Then, upload the data into a map server such as GeoServer, MapServer, ArcMap Server, ncWMS, etc. These map servers provide several options to access data using standard protocols and can generate images from this data, and sometimes animations. Finally, with a group of programmers and web designers, build the web interface (the Web GIS site) that displays the data to the user. This interface gives the user diverse ways to visualize and serve data and is in charge of making the proper requests to the map servers. The software described in this paper greatly simplifies the final step, building Web GIS sites, thus reducing the time and costs for publishing geospatial data on the web.

OWGIS (`http://owgis.org`), originally *Open Web GIS*, is an open source software that creates self-contained Web GIS sites. OWGIS is a Java web application that generates websites by automatically writing HTML and JavaScript code. The websites built with OWGIS are configured by XML files and can accommodate maps with layers served through any map server, such as GeoServer [6], that complies with the standards: Web Mapping Services (WMS) [5], Web Feature Services (WFS) [14], and Web Coverage Services (WCS) [1]. OWGIS is also capable of constructing Web GIS sites that display 4D data served by the THREDDS servers [7] or the ncWMS servers, using the WMS extension proposed by Bowler et al. [2].

The features that version 2.0 of OWGIS provides on the generated Web GIS sites are: multiple languages; animations; mobile interface; identify features; the ability to download data as KML, GeoTIFF or shape files; Contextual Query Language (CQL) filtering `http://en.wikipedia.org/wiki/Contextual_Query_Language`; dynamic display of maps using GeoJSON data; custom styling with COMPASS; multiple background layers; and the capacity to request time series, vertical profiles, and vertical transects at different locations from ncWMS servers. OWGIS allows building and maintaining new sites with all these features by simply editing XML files.

Figure 1, adapted from *The 2012 free and open source GIS software map*[11], shows the Free and Open Source Software for GIS (FOSS4G) that has similar capabilities or are related to OWGIS. The two categories, *Web GIS libraries* and *Web Map Development Frameworks*, encompass GIS software that can assist in the development of Web GIS sites. The web GIS libraries are application programming interfaces (API's) for the visualization and manipulation of spatial data that implement the WMS, WFS, etc. standards. OpenLayers [9] is one of the most extensive libraries in this category, is implemented in JavaScript and the current version of the library (version 3.0) has been redesigned from the ground up to use modern design patterns (`http://openlayers.org`). OWGIS uses OpenLayers extensively on the Web GIS sites that it generates. The main upgrade in OWGIS 2.0 was switching from OpenLayers 2.4 to version 3.0. This major revision improves the performance and stability of the generated maps and takes advantage of new features in OpenLayers 3, for instance, client-side vector rendering and GeoJSON parsing.
The software in the *Web Map Development Frameworks* category, contributes with tools for creating advanced web applications. These frameworks use web GIS libraries, like OpenLayers or Leaflet, and contain tools for the management of layers, menus, and themes of the Web GIS sites. Some well known programs in this category are Mapbender3 (`http://mapbender3.org/`), GeoMajas (`http://www.geomajas.org/`), MapFish (`http://www.mapfish.org/`) and GeoMOOSE (`http://www.geomoose.org/`). OWGIS belongs to this category as it gives an easy way to develop Web GIS sites through the configuration of XML files. The main feature that differentiates OWGIS from the rest of the development frameworks is the set of tools that it provides to visualize 4D data.

OWGIS is being developed at the Center for Ocean and Atmospheric Prediction Studies (COAPS), Florida State University (FSU), in collaboration with the Universidad Nacional Autónoma de México (UNAM). OWGIS 2.0 is used by the Deep-C Consortium (`http://deep-c.org`) in two different projects, the first one displays oceanographic data from the Gulf of Mexico (`http://viewer.coaps.fsu.edu/DeepCProject/mapviewer`), the second one visualizes forecast data for the coupled ocean-atmosphere model (`http://viewer.coaps.fsu.edu/GoM-FS/mapviewer`). A previous version of OWGIS is used as the interactive visualization map of the Digital Climatic Atlas of Mexico [15], which has been available since 2009 and provides access to more than 2000 layers of oceanic climate, bioclimatic variables, and socioeconomic indicators.

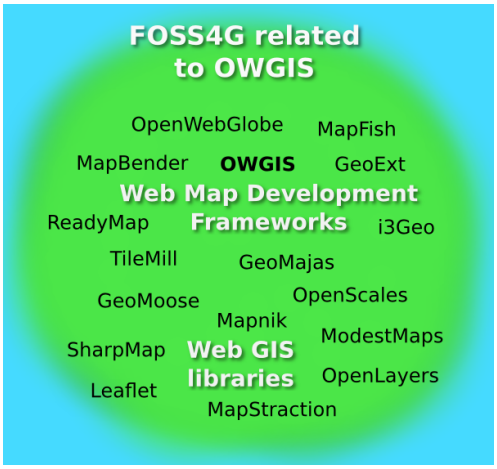

Figure 1: Map of the free and open source geographic information software for the web. Adapted from [11].

The current paper describes OWGIS 2.0, which is a major revision since version 1.0. This document recapitulates the OWGIS architecture and design principles, from Zavala-Romero et al. [16], in section 2. Section 3 describes the configuration of OWIGS 2.0. Section 4 illustrates the features provided by OWGIS with special emphasis in the new capabilities added on version 2.0. A case study, showing the new capabilities, is presented in section 5. Section 6 ends with a discussion on the presented software and outlines some future functionality of OWGIS.

2. ARCHITECTURE

OWGIS is contained in a web archive (WAR) file that can be deployed in a servlet container, such as Apache Tomcat. OWGIS generates web content (HTML and JavaScript) where the final websites are self-contained Web GIS sites. The JavaScript code generated by OWGIS contain customized requests made to map servers as well as a personalized configuration of the OpenLayers 3 library to display maps.

OWGIS-created websites display layers that are served over one or more map servers. The information about each layer, such as the name of the layer, is configured using XML files. These files are used to modify the look and content of the OWGIS sites and its configuration is described in Section 3.

OWGIS monitors any changes made to these files, and is able to update websites without reloading the web application on the server. Figure 2 shows a simple diagram of the process used by OWGIS to generate web content for the user.

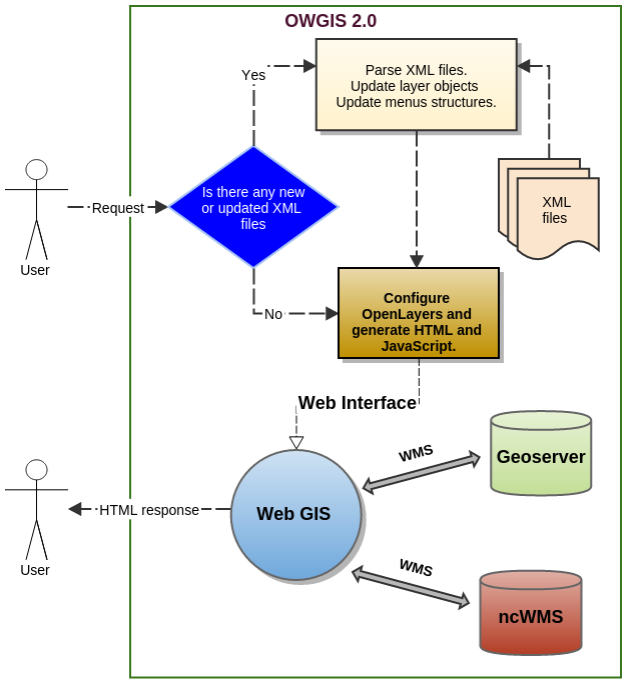

Figure 2: Simplified process of OWGIS to generate Web GIS sites from XML files.

When a user makes a new HTML request, OWGIS verifies that have been no modifications to the XML configuration files. If there have been any changes, OWGIS parses all the information in the XML files and rebuilds internal data structures for the sites. Once the information is stored internally, OWGIS uses it to build the desired interface, taking into account the specific user request. Finally, the Web GIS sites communicates with the map servers through the WMS and WFS standards.

The source code in OWGIS version 2.0 is grouped in five general modules: **model**, **business**, **servlets**, **messages**, and **tools**. The **model** module stores all the layers obtained from the XML files as well as the menus in a set of data structures. The **servlets** module contains all the servlets, which are in charge of receiving new requests from the user and redirecting each request to the proper method in the **business** module. The servlets module also sends customized JavaScript code to the user. The **business** module is in charge of reading the current user's configuration and the data of the model module to build the proper JavaScript code for the user. The **messages** module contains texts, in multiple languages, that are displayed in the web sites. These texts can be modified as explained in section 3.3. The **tools** module contains a number of static functions used to create web content.

2.1 Design principles

The websites that are created by OWGIS follow a set of design principles that we believe can be applied in a large number of circumstances. These design principles were defined by analyzing a few of the currently available Web GIS sites. For example, sites that display scientific data have, in many cases, a large set of layers that can be overlayed all at the same time. Having many layers available at the same time makes it difficult, for a normal user, to find the specific data to visualize. Moreover, there are several Web GIS sites where the majority of the data are raster (pixel based) layers, which occlude themselves when superimposed with each other, and overlaying raster layers makes a map incomprehensible. With this in mind OWGIS separates layers into three different categories:

- **Background.** Layers in this category are used as the spatial reference of the map and normally cover the whole domain. This category is designed to have just a few layers that are visible bellow layers in other categories. The Blue Marble dataset [12] is widely used as a *background* layer on scientific Web GIS sites.
- **Main.** This category should contain the majority of the data being displayed by OWGIS sites. There can be thousands of *main* layers on each OWGIS site. Main layers are hierarchically organized and can be grouped easily. Layers in the main category are displayed one at the time and are positioned above background layers and below optional layers. This category is expected to contain raster data but is not restricted to do so.
- **Optional.** Layers in this category can be overlayed with any other optional layer and are located above the main layers. This category is designed to contain just a couple of tens of layers, to avoid overwhelming the user, but there is no limit on the total number of optional layers. Optional layers are hierarchically organized and, even though they are not restricted to do so, optional layers are designed to accommodate vector data.

3. CONFIGURATION

OWGIS sites are configured by two types of files: Java properties file and XML files. There is only one Java properties file for each OWGIS site and it is used to define general display options of the Web GIS interface, such as setting the default center of the map. The other type of files, the XML files, are used to configure layers that will be displayed on the map as well as texts of menus used to select layers. In most OWGIS's sites, these two types of files will be the only ones ever modified. If it is necessary to adjust other texts displayed in the websites, which are not configured through XML files, the *Java resource bundles* needs to be modified, this type of text configuration is explained in section 3.3.

3.1 Default view

The default view of the maps created by OWGIS is configured using one Java properties file. This property file contains a set of parameters which are simply determined using the syntax *var=value*. For example, the parameter *mapcenter* defines the default center of the map and the *defaultLanguage* indicates which language is the default to be used in the Web GIS site. Table 1 contains all the parameters configured using the Java properties file with a small

description of each variable. For a detailed explanation of these parameters we refer the reader to the online documentation at `http://owgis.org`.

Table 1: OWGIS Default view parameters.

Variable	Description
mapcenter	Defines the initial center of the map.
zoomLevels	Defines the number of levels available on the sites.
zoomFactor	Defines the factor that the maps are zoomed in or out when the user adjusts the zoom. This parameter has to be greater than 1.
zoom	Initial zoom level to use on the website. Ranges from 0 to *zoomLevels*.
maxResolution	Defines the maximum resolution of the map. Larger numbers allow the map to be seen from farther away.
backgroundLayers	Indicates the background layer to be used. Some of the available options are OpenStreetMaps, Bing, Map Quest, and custom maps.
restrictedExtent	Restricts the domain where the user can move on the map. The default value is [-180, -90, 180, 90]
defaultLanguage	Defines the default language to be used on the website. The default language is EN.
availableLanguages	Defines which languages are available on the interface. Current options are English, Spanish, Hindi and Chinese.

In OWGIS 2.0 all changes made in the Java configuration file are reflected in the web sites without the need of reloading the web application on the server. This new feature accelerates the configuration of new instances of OWGIS. Additionally, since OWGIS 2.0, the Java properties file can be used to select background layers from a set of widely used maps, and at the same time it preserves the original option of custom background layers made through the XML configuration files. Figure 3 shows four different maps that can be displayed by OWGIS 2.0: OpenStreet maps [8], Bing maps `http://www.bing.com/maps/`, and two MapQuest maps `http://www.mapquest.com/`.

3.2 Map Layers

The XML configuration files are used to designate which layers are going to be displayed on the Web GIS sites as well as the menus that correspond to each of those layers. The XML files are mainly divided into four element tags: **Menus**, **BackgroundLayers**, **MainLayers**, and **OptionalLayers**. The purpose of the **Menus** tag is to allow the creation of independent menus to which multiple layers can point to. For example, layers that hold temperature data from the month of January for different years can all point to the same menu *January*. Each menu is defined using the **Menu** tag which has two required attributes: *ID*, to identify the menu, and *EN*, to specify the menu text in English. OWGIS allows additional menu languages, by using web browser language identification codes, such as *ES* for Spanish.

Figure 3: Background layers that can be selected on OWGIS 2.0. From left to right and top to bottom: Bing maps, OpenStreet Maps, MapQuest Hybrid, and MapQuestroads

As mentioned in Section 2, layers are grouped in three categories: background, main and optional. The **BackgroundLayers, MainLayers** and **OptionalLayers** XML elements relate directly to this division. There are many configurable attributes for each layer type but for most Web GIS sites, the default values will generate the expected behavior. Nevertheless, each layer type requires some attributes to be specified. For **BackgroundLayers** the required attributes are: *server*, which contains the URL of the WMS server that stores the layer, and *name*, which denotes the name of the layer on the server. **MainLayers** have *server* and *name* as required attributes plus *EN*, which defines the title of the main layer in English and *MenuIds*, which is a set of menu ids separated by colons that define the menu hierarchy of the layer. The title of the main layers can be defined for other languages but only English is mandatory. **OptionalLayers** have *server*, *name* and *MenuIds* as required attributes. The complete set of attributes that can be configured with **BackgroundLayers**, **MainLayers** and **OptionalLayers** element tags are described in OWGIS website.

The code in Listing 1 is an example of an XML configuration file with three layers: the BlueMarble as a background layer; the average precipitation of the month of January for the U.S. in 2004 as a main layer; and the U.S. States boundary as an optional layer.

Listing 1: Example of an XML configuration file for one background layer, one base layer and one optional layer.

```
<MenuConf xmlns:xsi="..." xsi:noNamespace="...">
  <Menus>
      <Menu ID="Id1" EN="Menu ENG txt" />
      <Menu ID="Id2" EN="Precipitation"/>
      <Menu ID="Id3" EN="US States" />
  </Menus>
  <BackgroundLayers server="http...">
      <layer name="bluemarble" />
  </BackgroundLayers>
  <MainLayers server="http..." >
      <layer MenuIds="Id1,Id2" name="prec" EN="Title: Prec"/>
  </MainLayers>
  <OptionalLayers server="http..." >
      <layer MenuIds="Id3" name="states" selected="true" />
  </OptionalLayers>
</MenuConf>
```

Figure 4 shows the self contained Web GIS site built by OWGIS using the example XML file shown in Listing 1. OWGIS creates a black window that contains the optional layers and a transparent window with drop-down menus for the main layers. The example in Figure 4 also shows a set of features built by OWGIS (Google Earth, Transparency, Download Data, and Help). Section 4 describes all the features that can be generated by OWGIS. If the user wants

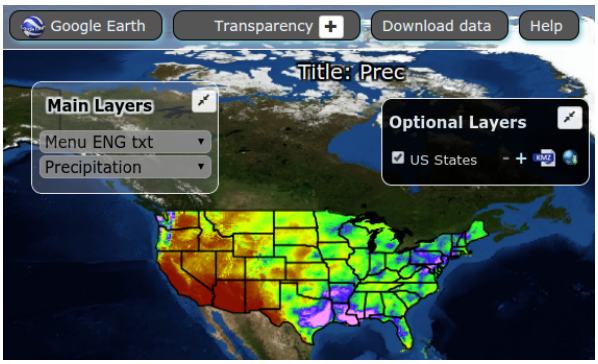

Figure 4: Example interface built by OWGIS 2.0 with three layers: the Blue Marble as a background image, precipitation data as a main layer and states boundaries as an optional layer.

to add new layers into the Web GIS site, it is necessary to modify the XML file by adding new menus and new layers. The configuration made through the XML files can be divided into multiple files or it can be automatically created using a scripting language. Using scripting languages it is possible to build Web GIS sites with hundreds of layers quickly. OWGIS will monitor any change in the XML files and update the generated Web GIS site automatically.

3.3 Languages

The texts used for the menus can be modified for different languages as described in the previous Section, but the Web GIS sites built by OWGIS contain many more texts, for example texts on buttons, headers, help texts, etc. These texts can also be modified for different languages using the standard internationalization approach for JSP/Servlets web applications. This approach uses a *resource bundle* (properties file) for each language. The resource bundles in OWGIS are stored in the *messages* package. Current version contains English, Spanish, Hindi, German, Chinese and Italian resource bundles

To add another language, it is necessary to create a new bundle, which can be done easily by copying one of the available bundles, translating each text inside it to the desired language, and saving it with the rest of the bundles. Because this software is open source, any new translation made by an individual can be shared with the rest of the world, expanding in this way the available languages of OWGIS. Figure 5 shows the obtained interface when the selected language is Hindi.

4. FEATURES

The features available through OWGIS depend on the type of layer. In this case, the division of layers is different from

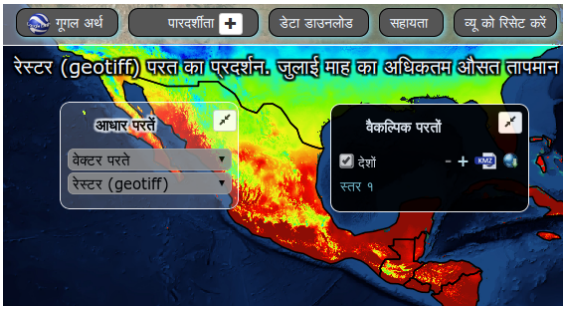

Figure 5: Example interface built by OWGIS when the selected language is Hindi.

the categories explained in Section 2. Here, the types of layers are divided into: *vector layers*; raster layers from GeoTIFF files (*GeoTIFF layers*); and raster layers served through ncWMS servers (*ncWMS layers*). This division is necessary because each type of layer has its own characteristics. For example, it is not possible to create vertical profiles from vector layers only from ncWMS layers. *Vector layers* and *GeoTIFF layers* can be used in any of the categories explained in Section 2, but *ncWMS layers* can only be used as **main layers**. This restriction is to avoid overwhelming the users with too many options. The features that are available independently of the layer type are: *mobile interface*; *multiple languages*; *identify feature*; *download as KML*; and *transparency*.

The *mobile interface* is a specially designed interface that is displayed when OWGIS sites are accessed through a mobile phone or tablet. The mobile interface was redesigned for version 2.0 of OWGIS and it has been developed taking into consideration the limited space available on the screens of mobile devices. In this new design, shown in figure 6, the options and menus are hidden by default, improving the visibility of the maps, and can be accessed with a single touch.

Figure 6: Example layout of the mobile interface generated by OWGIS 2.0.

The mobile interface contains two top menus, the first one

accommodates the map tools and the second one encompass the main and optional layers. Secondary menus, like the calendars and the color palettes, are accessed using sticker panels. Finally, the animation controls are displayed on a sliding bottom drawer by touching its handle. Thanks to OpenLayers 3, the mobile interface is controlled through touch events that allow map dragging, pinch to zoom, and rotate maps with two fingers.

The *identify feature* feature uses the WMS and the WFS standards to request data at specific locations in the map. OWGIS uses AJAX to make the data requests asynchronously, allowing multiple requests to run simultaneously without having to queue each request. Figure 7 shows an example of the *identify feature* capability used on a map with three visible layers: topography; forecast temperature of the Gulf of Mexico for the 6th of July 2014; and state boundaries for the United States.

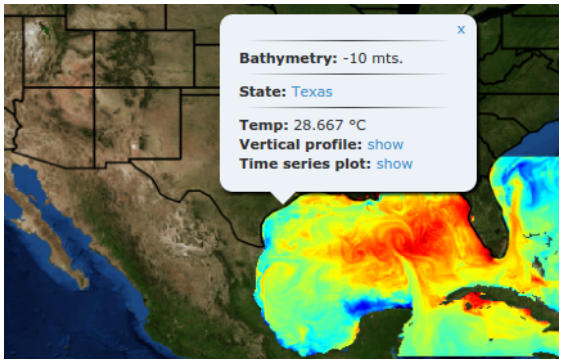

Figure 7: Data obtained with OWGIS by the *identify feature* capability for three layers: topography, forecast temperature of the Gulf of Mexico for the 6th of July fo 2014, and states boundaries for the U.S.

OWGIS builds the proper requests to generate KML files from map servers such as GeoServer or ncWMS. This feature, *download data as KML*, allows visualizing layers in clients that display KML data, such as Google Earth. People have being using this feature to compare data from OWGIS's servers with different data available in the KML clients. Figure 8 shows sea surface temperature of the Gulf of Mexico displayed using Google Earth.

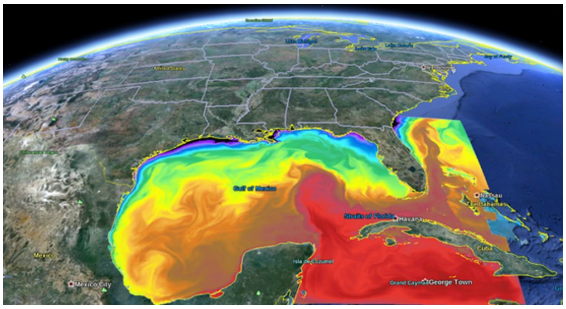

Figure 8: Example of the *download as KML* feature of OWGIS. Sea surface temperature of the Gulf of Mexico visualized using Google Earth.

The *transparency* feature allows users to change the opacity of any main layer or optional layer. Being able to modify the opacity of the layers has become a core functionality in many Web GIS sites and is used for comparison between layers, mainly when the layers contain raster data.

The *CQL filter* feature is exclusively available for vector layers and allows, as the name implies, filtering vector data using CQL. The scientific community uses this feature regularly to analyze data by filtering vector data. CQL filters modify data being displayed by the main layer and update the KML and *download data* URL requests by applying the filter into the requests. Figure 9 shows an example of the generated interface for filtering vector data using CQL.

Figure 9: CQL filtering example. In this example, the layer that displays the state boundaries for the U.S. is filtered to show only the state of Texas.

Another feature that is only available for vector layers is the *dynamic vector layers* feature. This feature, recently added in OWGIS 2.0, requests vector layers using the JSONP communication framework, in which the vector layers are served by the map server as JSON objects `http://en.wikipedia.org/wiki/JSONP` rather than images. The visualization of the layers, using this feature, is generated directly on the client machine using OpenLayers 3, and the style of the layers can be modified in real time. This feature improves considerably the navigation experience of the user. In the future, the *dynamic vectors* component will allow to modify the original content of vector layers as well as generate plots from vector data directly in the client machine. Figure 10 shows an example of this feature applied to the United States vector layer.

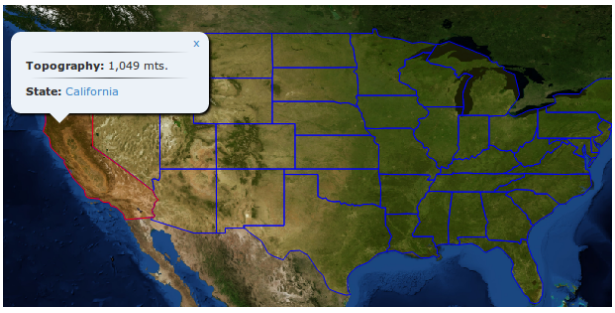

Figure 10: Dynamic vector layer example. The style of the California state is highlighted when the mouse hovers above the state boundaries.

Some features are only available for ncWMS layers. Those

features are *animations, vertical profile, vertical transect, time series, color palettes,* and *depth selection.* Animations and time series are available for temporal data. These two features are very useful for scientists, especially oceanographers and climatologists, because these features allow them to visualize and analyze changes in data across time.

In version 2.0 of OWGIS the *animations* feature was rebuilt from scratch. Earlier versions of OWGIS used animated gif files, retrieved from ncWMS servers, as animations. This approach is efficient because the animations are stored as one small file, but it has several disadvantages. Some of these disadvantages are: the speed of the animation can not be modified, is not possible to pause or stop the animation, and it is difficult to animate gif files using the canvas of HTML 5. In the animations created with OWGIS 2.0, the users can increase or decrease the speed of the animation, pause the animation at any time, traverse the animation frame by frame, and download the animation to the user's computers. OWGIS 2.0 has reduced the smallest time step of the animations from one day to one hour allowing several environmental variables, like clouds, to animate smoothly. Figure 11 shows an example of the controls provided by OWGIS to manipulate animations.

Figure 11: Animation controls available on OWGIS version 2.

Depth selection is available for ncWMS layers that contain a z-axis coordinate, normally depth, but the z-axis can refer to a different variable, such as pressure. OWGIS builds a nicely paginated window where users can select from the available z-axis values. The *depth selection* feature is one of the tools OWGIS provides for visualizing 3D data. Figure 12 shows an example of the interface built by OWGIS that shows sea temperature from the Gulf of Mexico at different depths.

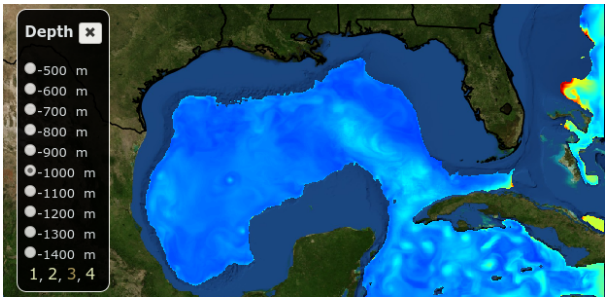

Figure 12: Multiple depth selection example. Paginated window displays different elevations available for layers served by ncWMS servers.

Vertical profiles and *vertical transects* are also available for ncWMS layers with z-axis information. Vertical profiles and vertical transects can be requested at any location on the map and are obtained asynchronously by the ncWMS servers. These two features are also very important for the scientific community and used regularly. An example of a vertical transect of Gulf of Mexico temperature's is shown in Figure 13.

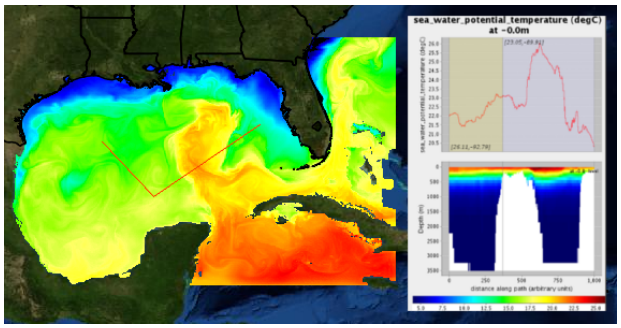

Figure 13: Vertical transects obtained through the OWGIS software using temperature data for the Gulf of Mexico.

Color palettes is another feature available for ncWMS layers. OWGIS queries ncWMS servers, asking for available color palettes, and generates the proper web interface that allows the user to switch between these palettes. Figure 14 shows an example of the web interface generated by OWGIS to select color palettes.

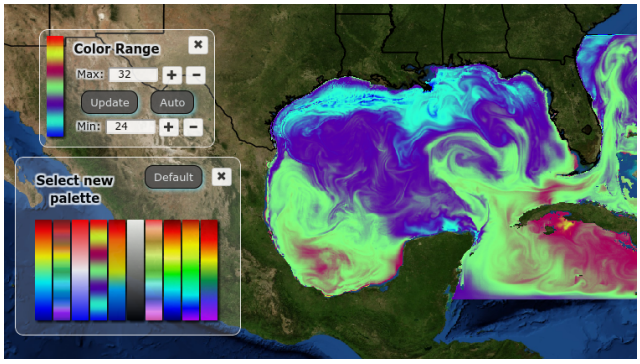

Figure 14: Example of the windows generated by OWGIS that allow users to select color palettes and to modify color ranges.

The look and formatting for OWGIS sites, including colors and fonts, is defined using SASS (`http://sass-lang.com/`) and COMPASS (`http://compass-style.org`). These two technologies allow enthusiastic users to modify the look of the generated Web GIS sites easily. Most of the styling of OWGIS's sites is made through three *scss* files which control the fonts, colors, and other global configurations respectively. Figure 15 shows two different interface styles obtained by modifying a couple of lines in these styling files.

Some features request data by using extensions of the OGC standards and have only been tested with specific map servers and may not work with others. The CQL filter feature has only been tested with GeoServer, and the color palette, verti-

Figure 15: Two different interfaces obtained with OWGIS 2.0 by modifying styling files.

cal profile, vertical transect, animations, and time series features are only feasible for layers served by ncWMS servers.

5. CASE STUDY

In this Section a successful example where OWGIS was used to build a self-contained Web GIS site that displays environmental data from an operational coupled ocean-atmosphere model system is described. The Web GIS site was developed for the The Deep-C Consortium at the Center for Ocean-Atmospheric Prediction Studies (COAPS), Florida State University (FSU) and is available at `http://viewer.coaps.fsu.edu/GoM-EFS/mapviewer`, The Deep-C Consortium is a long-term, interdisciplinary study of deep sea to coast connectivity in the northeastern Gulf of Mexico.

This case study, referred to as the *GoM-EFS MapViewer* (Gulf of Mexico Earth Forecasting System Map Viewer), displays forecast variables for the Gulf of Mexico. This instance of OWGIS in an integrated component of GoM-EFS. GoM-EFS [13], is a two-way atmosphere-ocean coupled system that provides forecasts for the atmosphere and the ocean using the Weather Research and Forecasting Model (WRF, `http://www.wrf-model.org`) and the Regional Ocean Modeling System (ROMS, `http://www.myroms.org`). The *GoM-EFS MapViewer* displays 4D (longitude, latitude, depth, time) ocean data (currents, water-temperature, salinity and surface heat fluxes) and 3D (longitude, latitude, time) atmospheric data (wind speed, air-temperature, pressure and incoming heat fluxes).

The main benefit that OWGIS provides to this project is the ability to display 4D data, generated by the ROMS and WRF models and served through a ncWMS server. Most environmental models, such as the HYbrid Coordinate Ocean Model (HYCOM) [4] or the WAVEWATCH III wave model, use the NetCDF output format and could benefit from OWGIS following a similar architecture than this project. A wide variety of variables, such as wind, sea ice, relative humidity, surface temperature, etc., could also be displayed by OWGIS.

The *GoM-EFS MapViewer* uses four modules to display contents on the Web GIS site: a configuration of the forecast system as an operational system; an instance of the ncWMS server to provide 4D data using WMS; an instance of GeoServer to provide 2D geospatial layers through the WMS and WFS standards; and an instance of OWGIS to generate a self-contained Web GIS user interface. These four modules interact with each other as shown in Figure 16.

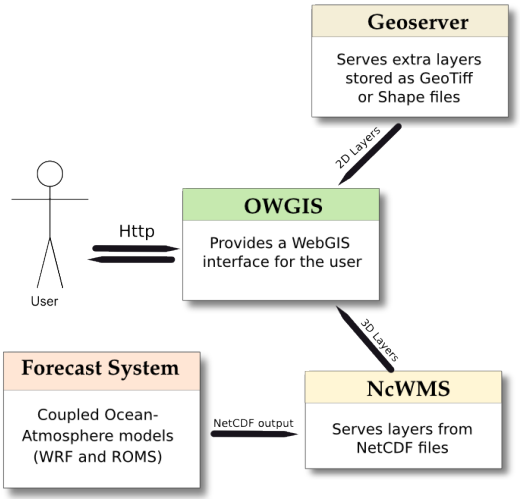

Figure 16: Architecture used to build the *GoM-FS MapViewer*. Its four modules are: an instance of *GeoServer* and *ncWMS* servers; a *Forecast System* using the WRF and ROM models; and *OWGIS* to generate the web interface for the user.

Whichever environmental model the project uses, the architecture shown in Figure 16 can be applied. If the environmental data is already available through an ncWMS server or other WMS compliant map server, the only module that is required to build a Web GIS site is the OWGIS module. The rest of the Section explains in detail how each of these four modules where implemented for the *GoM-EFS MapViewer*.

5.1 Forecasting System (WRF and ROMS)

WRF and ROMS domains encompass the Gulf of Mexico region (latitude: 18°00′ N to 32°00′ N, longitude: 98°00′ W to 74°24′ W) with model grid resolutions of 15 km x 15 km and 8 km x 8 km respectively. GoM-EFS runs routinely once a day at 00:00:00 UTC. Upon completion, the output data are automatically updated in the ncWMS server and also the visualization in the *GoM-EFS MapViewer*. It is anticipated that the resolution of these two modeling components of GoM-EFS will be increased to 5 km x 5 km (WRF) and 1 km x 1 km (ROMS) imposing an additional computational burden that will be accomondated through the parallel processing of the data. GoM-EFS will also be extended to incorporate calculations for the wave fields using the Simulating Waves Nearshore model (SWAN, `http://swanmodel.sourceforge.net/`) and transport of sediments using the Community Sediment Transport Modeling System (CSTMS).

5.2 ncWMS

ncWMS [3] is a web application used to visualize multidimensional environmental data stored as NetCDF files. The ncWMS can be easily deployed in an application server such as WebSphere, JBoss, Jetty, or, as in this project, Apache Tomcat. ncWMS is configured through a web interface where the administrator defines each of the datasets being published. For the *GoM-EFS MapViewer*, ncWMS is used to serve the output of the coupled model and has five datasets configured, one for the current forecast and four more for the previous four days. Each dataset points to a folder where NetCDF files with the oceanographic and atmospheric variables for one forecast day are stored; this datasets are updated daily.

5.3 GeoServer

GeoServer [6] is an open source web application that is used to view and edit geospatial data using standards from the Open Geospatial Consortium (OGC). GeoServer is better suited for serving geospatial data, that is not stored as NetCDF files, than ncWMS. For the *GoM-FS MapViewer* project, GeoServer is configured to serve the Blue Marble dataset (the background layer) and the vector layer that contains the shore lines. There is extensive documentation on the GeoServer website (http://geoserver.org/) on how to configure vector and raster layers on the server.

5.4 OWGIS

OWGIS is used to build the interface with the user. Once all the data is available through the map servers, OWGIS provides an easy way to build Web GIS sites to visualize, analyze and share that data. In this case study, OWGIS is configured with the Blue Marble layer as a background, 152 main layers that store oceanographic and atmospheric variables, and one optional layer that display shore lines of the continent.

The Java properties configuration file, described in section 3.1, has *lon=-87.34* and *lat=26.70* as the center of the map and uses *6* zoom levels. The rest of the parameters retain their default values from the OWGIS template. For the *GoM-FS MapViewer*, twelve XML files define the menu titles, background layers, main layers and optional layers. As an example, the XML code on listing 2 shows how to configure five atmospheric variables for the latest forecast.

Listing 2: Example of an XML configuration file for the GoM-FS MapVierwer project.

```
<MenuConf xmlns:xsi="..." xsi:noNamespaceSchemaLocation="...">
    <MenuEntries>
        <MenvEntry ID="d2" EN="2D" ES="2D"/>
        <MenuEntry ID="atm" EN="Atmosphere" ES="Atmosfera"/>
        <MenuEntry ID="latest" EN="Latest" ES="Actual"/>

        <MenvEntry ID="wind" EN="Wind Speed 10m" ES="Viento"/>
        <MenvEntry ID="t2" EN="Temp. at 2m" ES="Temp. a 2m"/>
        <MenvEntry ID="sst" EN="Sea Surface Temp" ES="Temperatura"/>
        <MenvEntry ID="lh" EN="Latent Heat Flux" ES="Flujo de calor"/>
    </MenuEntries>

    <MainLayers BBOX="-123.828622,-2.459646,-58.255699,48.7692"
        server=".../ncWMS/wms"
        ncWMS="true" >
        <layer Menu="latest,atm,d2,wind"
            name="atm_forecast_latest/wind" ES="..."
            EN="Wind speed at 10m" style="vector"/>
        <layer Menu="latest,atm,d2,t2"
            name="atm_forecast_latest/T2" ES="..."
            EN="Air Temperature at 2m" />
        <layer Menu="latest,atm,d2,sst"
            name="atm_forecast_latest/SST" ES="..."
            EN="Sea Surface Temperature" />
        <layer Menu="latest,atm,d2,lh"
            name="atm_forecast_latest/LH" ES="..."
            EN="Net Latent Heat Flux"/>
    </MainLayers>
</MenuConf>
```

By creating these configuration files, OWGIS builds a Web GIS site with all the required functionality to visualize the oceanographic and atmospheric variables: color palettes, depth selection, animations, multiple language menus, etc. Figure 17 shows the interface of the *GoM-FS MapViewer* project when ocean net heat flux for the 6th of July, 2014 is selected.

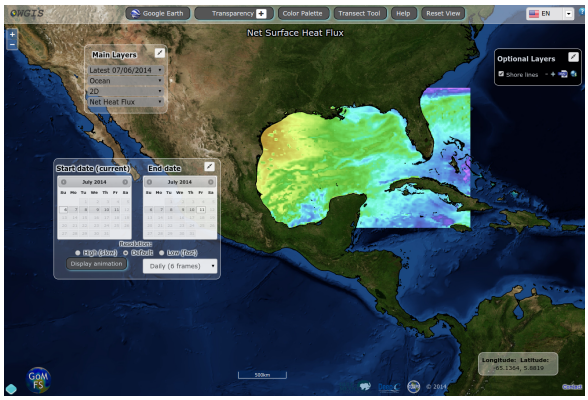

Figure 17: Net heat flux displayed from the *GoM-FS MapViewer* for the 6th of July, 2014.

The amount of data displayed on the *GoM-FS MapViewer* website is growing rapidly thanks to the ease of configuration of OWGIS.

6. CONCLUSIONS AND FUTURE WORK

OWGIS 2.0 improved considerably the animations that are shown on the interfaces as well as the mobile interface. The configuration in OWGIS 2.0 has been simplified and optional background layers are now available. OWGIS is the first open source software that can build Web GIS sites that displays 2D, 3D, and 4D data served from distinct map servers that can be located anywhere. The main features OWGIS provides in the interfaces it builds are: multiple languages; animations; time series; vertical profiles and vertical transects; color palettes; and the ability to download data. All these features are created automatically depending on the type of data and without any additional web programming.

Configuring new instances of OWGIS is made easy through XML files. In these files, the layers and texts of the websites are defined, providing an easy way to add and edit new layers that expand the content of each project. At the same time it allows sharing content between institutions and eases the maintenance of the websites. OWGIS conveniently stores all the texts in separate files to allow multiple languages on the Web GIS sites, thus increasing the potential number of users who can access the data.

OWGIS has been used by scientists who store their data in

NetCDF files, but it has also being used to display other types of geospatial data. OWGIS is used to visualize ocean and atmospheric variables in the *GoM-FS MapViewer* project (`http://viewer.coaps.fsu.edu/GoM-EFS/mapviewer`); ocean variables of the Gulf of Mexico in the *Deep-C MapViewer* (`http://viewer.coaps.fsu.edu/DeepCProject/mapviewer`); is the main component of the Digital Climatic Atlas of Mexico, which displays more than 2000 layers of climate data (`http://uniatmos.atmosfera.unam.mx/ACDM/servmapas`); it displays public transportation data (`http://viajandodf.com`) and useful information for immigrants (`http://www.americas.datafest.net`).

In the future, we want to make OWGIS easier to configure and current efforts are directed in the development of an administration interface that makes the management of sites straightforward. This administration interface will be web based and will be used to configure the layers in the map, which are currently configured through XML files. Other variables associated to the map such as background layer selection, resolution, map center, and zooming step will also be modified with this interface.

We hope that OWGIS becomes a popular open source software and the standard tool for building Web GIS sites in scientific fields where netCDF files are the default file format for storing data.

Acknowledgments

This project was made possible by a grant from BP/The Gulf of Mexico Research Initiative to the Deep-C Consortium. The present work benefited from the input of Ossian Foley and Rosario Romero who provided valuable comments on the writing of this paper.

7. REFERENCES

[1] P. Baumann. The OGC web coverage processing service (WCPS) standard. *GeoInformatica*, 14(4):447–479, Oct. 2010.

[2] J. Blower, A. Gemmell, G. Griffiths, K. Haines, A. Santokhee, and X. Yang. A web map service implementation for the visualization of multidimensional gridded environmental data. *Environmental Modelling & Software*, 47(0):218 – 224, 2013.

[3] J. Blower, A. Gemmell, G. Griffiths, K. Haines, A. Santokhee, and X. Yang. A web map service implementation for the visualization of multidimensional gridded environmental data. *Environmental Modelling & Software*, 47(0):218 – 224, 2013.

[4] E. P. Chassignet, H. E. Hurlburt, E. J. Metzger, O. M. Smedstad, J. A. Cummings, G. R. Halliwell, R. Bleck, R. Baraille, A. J. Wallcraft, and C. Lozano. US GODAE: global ocean prediction with the HYbrid coordinate ocean model (HYCOM). Technical report, DTIC Document, 2009.

[5] J. de la Beaujardiere. OpenGIS web map server implementation specification. *Open Geospatial Consortium Inc*, 2006.

[6] J. Deoliveira. GeoServer: uniting the GeoWeb and spatial data infrastructures. In *Proceedings of the 10th International Conference for Spatial Data Infrastructure, St. Augustine, Trinidad*, 2008.

[7] B. Domenico, J. Caron, E. Davis, R. Kambic, and S. Nativi. Thematic real-time environmental distributed data services (thredds): Incorporating interactive analysis tools into nsdl. *Journal of Digital Information*, 2(4), 2006.

[8] M. Haklay and P. Weber. Openstreetmap: User-generated street maps. *Pervasive Computing, IEEE*, 7(4):12–18, 2008.

[9] E. Hazzard. *OpenLayers 2.10: Create, optimize, and deploy stunning crossbrowser web maps with OpenLayers JavaScript web-mapping library _ Beginner's Guide*. Packt, Birmingham, UK, 2011.

[10] S. Steiniger and A. J. Hunter. Free and open source GIS software for building a spatial data infrastructure. In *Geospatial Free and Open Source Software in the 21st Century*, pages 247–261. Springer, 2012.

[11] S. Steiniger and A. J. Hunter. The 2012 free and open source GIS software map. a guide to facilitate research, development, and adoption. *Computers, Environment and Urban Systems*, 39:136–150, May 2013.

[12] R. Stöckli, E. Vermote, N. Saleous, R. Simmon, and D. Herring. The blue marble next generation-a true color earth dataset including seasonal dynamics from MODIS. *Published by the NASA Earth Observatory*, 2005.

[13] P. Velissariou. Gulf of mexico earth forecasting system, 2014.

[14] P. A. Vretanos. OpenGIS web feature server implementation specification. *Open Geospatial Consortium Inc*, 2005.

[15] J. Zavala-Hidalgo, A. Fernandez-Eguiarte, R. Romero-Centeno, and O. Zavala-Romero. Digital atlas of Mexico provides accessible climate information. *Eos, Transactions American Geophysical Union*, 91(14):125–126, 2010.

[16] O. Zavala-Romero, A. Ahmed, E. P. Chassignet, J. Zavala-Hidalgo, A. F. Eguiarte, and A. Meyer-Baese. An open source java web application to build self-contained web {GIS} sites. *Environmental Modelling & Software*, 62(0):210 – 220, 2014.