Up in the air: Adventures in serving geospatial data using open source software and the cloud

Ken Bunzel

GIS Applications Developer

Kingbird Software, LLC

208.882.0410

kbunzel@kingbirdsoftware.com

Alan Ager

Operations Research Analyst

USDA Forest Service

Western Wildland Environmental Threat Center

541.416.6603

aager@fs.fed.us

Charlie Schrader-Patton

GIS/Remote Sensing Analyst

USDA Forest Service

Remote Sensing Applications Center

541.312.4291

cschrader@fs.fed.us

ABSTRACT

In recent years GIS and web technologies have advanced to allow the sharing of geospatial data on the web. Land management agencies sometimes have a need to simply post GIS datasets on the web and don't always require advanced GIS modeling capabilities on the server. There are many products available to serve geospatial datasets on the web, but many of these have a complicated setup, costly license fees, and provide more capabilities than are needed in some use cases. There is also a need to simplify web server setup. In this paper we describe an inexpensive and simple solution for agencies to post and share GIS data on the web using open source map servers and open source GIS clients deployed on a virtual cloud server.

Keywords

GIS, Geospatial, land management agencies, Western Wildland Environmental Threat Assessment Center, GeoServer, MapServer, OpenLayers, GeoExt, Ext JS, WMS, Amazon Web Services, Cloud Computing, Virtual Server, EC2, JavaScript.

1. INTRODUCTION

The USDA Forest Service and other public land management agencies often compile and generate geospatial data as a result of their research and activities. Making this data available to the public is often hindered by a lack of web and map servers inside the agency. When the servers are available, they often require GIS expertise to setup and maintain. Some of the more popular map servers, while providing advanced geoprocessing capabilities, have high annual license fees. The need to post geospatial data continues to grow as Federal land management agencies grapple with wide ranging social and ecological issues with stakeholders and other interested external entities.

One potential solution to streamline the sharing of geospatial data under the current information policy is through the use of cloud computing and public domain geoservers. The federal government is encouraging the use of cloud computing by its agencies by launching the General Services Administration's cloud storefront Apps.gov. This online storefront enhances how the government leverages technology by enabling federal agencies to acquire and purchase cloud computing services in an efficient, effective manner.

This paper describes our efforts to build a cloud server based map server solution for the USDA Forest Service Western Wildland Environmental Threat Assessment Center (WWETAC). The center was established in 2005 with the mission of developing tools and conducting research on the interaction of multiple wildland threats including wildfire, insects, disease, invasive plants, and climate change. These disturbances can have profound effects on the sociological values and ecological functions of forests and wildlands. WWETAC is to "carry out a program using geospatial and information management technologies (including remote sensing imaging and decision support systems) to inventory, monitor, characterize, assess, and identify forest stands and potential forest stands…" [WWETAC, 2003]. One of our goals is build a system for clients to view existing geospatial data layers of wildland threats to examine spatial intersections of multiple threats with high value resources such as critical wildlife habitat, wildland urban interfaces, and other human and ecological values. Additionally, the system allows users to load local data as well as connect to internet map services.

2. METHODS

Our system has the following characteristics:

- Utilizes open source GIS software for both the map server and map client. Open source software is distributed for free, does not depend on proprietary vendor-specific files or formats, and is developed and maintained by peers thru the open availability of source code.

- Map client requires no installation and is embedded inside a web browser as pure JavaScript®. All major browsers are supported.

- Deployed on a virtual cloud server.

In particular we chose GeoServer as the map server, OpenLayers as the map client, and Amazon Elastic Compute Cloud®, also known as "EC2®", as the cloud server.
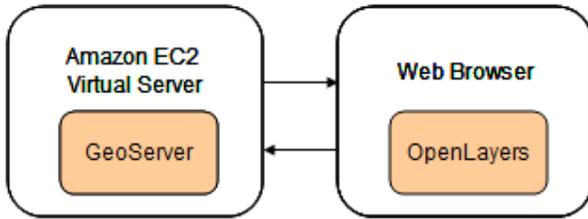
Figure 1. This system has GeoServer deployed on an Amazon EC2 server and OpenLayers in a browser.

2.1 Map Servers

Two popular open source map servers are MapServer and GeoServer. MapServer was originally developed by the University of Minnesota in the mid-1990s and is now a project of OSGeo. GeoServer is built with the Java programming language and is maintained by a diverse group of individuals and organizations from around the world. GeoServer is the reference implementation of the Open Geospatial Consortium (OGC) Web Feature Service (WFS) and Web Coverage Service (WCS) standards, as well as a high performance certified compliant Web Map Service (WMS). We have chosen GeoServer (http://geoserver.org), for our system.

A feasibility study was conducted to determine the usefulness of GeoServer for deploying map data on the web with a cloud server. This study has indicated that GeoServer is capable of serving map data fast, efficient and inexpensively on a cloud server. GeoServer provides a web based administration tool that controls all aspects of map server and layer configuration (Figure 2).

Web map services serve georeferenced map images in jpeg, png, or gif formats. These images are downloaded in the web browser for each map layer and compiled into a single composite image by the map client software. GeoWebCache (http://geowebcache.org) is integrated with GeoServer and provides a mechanism to cache map tiles for future use. This avoids the need to generate the images anew for each map request. A tile cache significantly improves performance of a map server. GeoWebCache is preconfigured to cache tiles as they are created if the tile does not already exist in the cache. In addition, an interface is provided for seeding the tile cache in advance to improve performance.

GeoServer is an OGC compliant WMS server with support for multiple data sources and the style layer descriptor format.
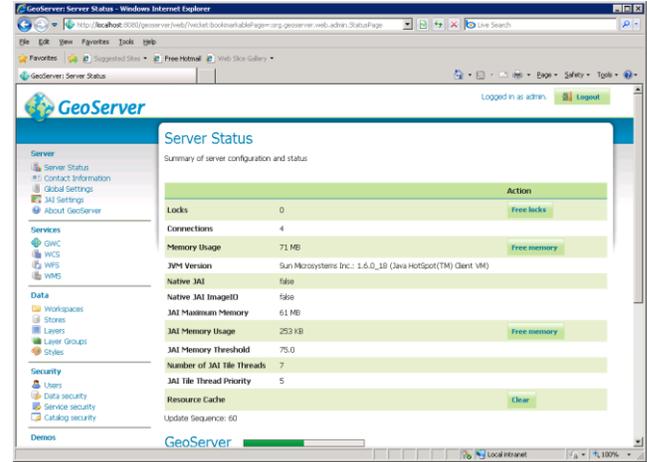

Figure 2. GeoServer Administrator interface

2.2 Client Applications

There are many open source options for map clients including OpenLayers GeoExt, ka-Map, MapBender, OpenScale, MapBuilder, MapFish, CartoWeb, etc. We have chosen to use OpenLayers (http://openlayers.org) because it has popular support and a growing developer community. OpenLayers is a pure JavaScript library for displaying map data in most modern web browsers, with no server-side dependencies. OpenLayers implements industry-standard methods for geographic data access, such as the OpenGIS Consortium's Web Mapping Service (WMS) and Web Feature Service (WFS) protocols.

In addition to OpenLayers another open source library called GeoExt (http://geoext.org) is used in our system. GeoExt combines the geospatial capabilities of OpenLayers with the rich user interface components of Ext JS (http://www.extjs.com). Ext JS is available under an open source license option.

By combining OpenLayers, GeoExt and Ext JS we have built a thin map client with a rich user interface supported by all major browsers (Figure 3).
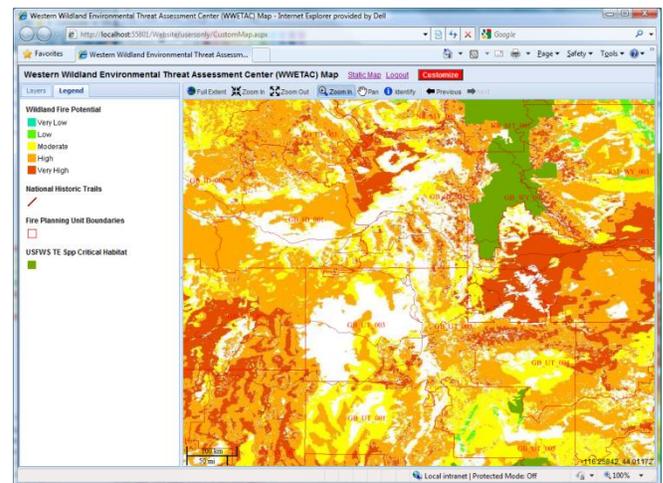
Figure 3. The map client requires no installation and is supported by all major browsers.

Our system also includes an option to customize the map in the following ways:

- Add layers from any OGC compliant web map service.

- Search for web map servers and layers by a user specified search term.

- Preview layers before adding them to the map.

- Remove layers from the map.

- Reorder the layers.

- Specify the default visibility of the layers.

- Save the map settings in an xml file on the server. The custom map is retrieved by user login.

Because the most commonly supported spatial reference by non commercial map servers is EPSG 4326 (latitude/longitude based on the WGS84 datum), we have chosen this as the spatial reference of the map. We also have a non customizable map that uses EPSG 900913 ("Spherical Mercator") which is commonly used by commercial providers such as Google Maps®, Microsoft Virtual Earth® (Bing Maps®), Yahoo Maps®, and other commercial API providers. This map includes both WWETAC GeoServer provided layers as well as Bing Maps® (Microsoft Virtual Earth®) as a base layer. Metadata links for each of the hosted layers will be available when the OpenLayers map client is integrated into the WWETAC web site.

2.3 Virtual Web Server (Amazon Cloud)

In our system the map server is deployed on a virtual server (EC2 instance) in the Amazon Web Services cloud. This provides the following benefits:

- Scalable – Once you have installed software, copied data, and configured your virtual server, you can save this as a stored machine image. Amazon refers to this as an Amazon Machine Image (AMI®). With a few mouse clicks you can then easily launch new instances of the virtual server from the stored machine image with all the software and settings already installed. This avoids the need to install applications and data for each instance. The user chooses the size of the instance (RAM) when launching new instances. A feature called Auto Scaling allows you to automatically scale your Amazon EC2 capacity up or down according to conditions you define. Users are able to create, launch and terminate server instances on demand.

- Simple Setup – Amazon allows remote desktop connections to the server you launch. The online AWS Management Console provides full control of all your EC2® instances and AMIs.

- Cost-effective – You pay for your EC2® instance compute capacity by the hour and there are no long term commitments. This avoids the large up-front fixed costs of server setup and maintenance.

- Reliable – The Amazon Web Services cloud is designed to be resilient, distributed and secure.

- Elastic Load Balancing - Allows you to automatically scale your Amazon EC2® capacity up or down according to conditions you define.

3. Map Services Search Engine

Our client application will include an interface to a WMS search engine developed by MicroImages®, Inc. (http://www.microimages.com, Ager et al. this proceedings). This search engine allows users to enter search terms and retrieve a list of WMS servers and layers that match the terms. The user will then be able to view a preview map of the matched layers, and add the layers to their custom map. The map customizations are saved on the server and will be retrieved the next time the user logs in.

As published map services become more common and available to the general public, the ability to find map services with a particular subject focus is essential. The map search engine provides the means to locate these services. The map client interface we are developing will provide the means to add these services to a custom map. More information about the map services crawler is available in Ager et al. these proceedings.

4. Conclusions

The evaluation of GeoServer, Amazon cloud servers, and open source map clients has led to some important discoveries. We have found that when the needs are to simply serve map data on the web, this can be accomplished without high license fees and with a minimum setup effort. In addition, there is good performance on open source mapping software. While there are limitations with data support and symbolizations in GeoServer, the developer community is actively working on improving the software. Since the software is based on OGC standards, this helps ensure its future viability. Our efforts here are conducted in parallel with the development of map services and client applications using ESRI® software (Schrader-Patton et al. these proceedings); future work will focus on comparing these two approaches.

Amazon EC2® virtual servers have so far proven to be very reliable. The ability to store a server instance as a machine image and later launch new identical instances is valuable for both scalability and ease of maintenance.

On the client side, OpenLayers provides an efficient map interface that uses AJAX technology to load layers into the map panel without freezing the user interface. The GeoExt legend control is able to create a legend dynamically by downloading published legend images from the map servers. These technologies have been designed to work together to produce a complete web mapping system.

5. REFERENCES

[1] Ager, A.A., Schrader-Patton, C., Bunzel, K., Colombe, B. (this proceedings) Internet Map Servers: A new portal for global ecological monitoring, or geodata junkyard?

[2] OGC. 2010. Open Geospatial Consortium – Open GIS Standards. http://www.opengeospatial.org/standards. Accessed February 18, 2010.

[3] WWETAC. 2003. USDA Forest Service Western Wildland Environmental Threat Center Charter Statement. http://www.fs.fed.us/wwetac/wwetac_charter.html. Accessed February 18, 2010.

[4] Obama Administration Launches GSA Cloud Storefront Apps.gov, September 15, 2009. http://www.gsa.gov/Portal/gsa/ep/contentView.do?contentType=GSA_BASIC&contentId=28477. Accessed February 26, 2010.

[5] GeoServer. http://geoserver.org/display/GEOS/What+is+Geoserver. Accessed February 26, 2010.

[6] OpenLayers: Free Maps for the Web, http://openlayers.org. Accessed February 26, 2010.

[7] GeoExt - JavaScript Toolkit for Rich Web Mapping Applications. http://geoext.org. Accessed February 26, 2010.

[8] Spherical Mercator. http://docs.openlayers.org/library/spherical_mercator.html. Accessed February 26, 2010.

[9] Using WMS Servers, How to find the right URL for the configuration file. http://code.google.com/p/gpsvp/wiki/UsingWMSServers. Accessed February 26, 2010.

[10] What is AWS? http://aws.amazon.com/what-is-aws/. Accessed 26 February 2010.