2011

Locating Mineral Exploration Targets using a Geographical Information System

Finnian O'Connor

Technological University Dublin, finnian.oconnor@gmail.com

Recommended Citation

O'Connor, F. (2011). Locating Mineral Exploration Targets using a Geographical Information System. *IEEE Conference on Communication, Science & Information Engineering (CCSIE 2011)* https://doi.org/10.21427/8zfs-5524

Locating Mineral Exploration Targets using a Geographical Information System

Finnian O’Connor
School of Computing
Dublin Institute of Technology
Kevin Street, Dublin 8, Ireland
finnian.oconnor@gmail.com

***Abstract*—This paper outlines the research and development of a complete open source geographic information system (GIS) that offers real-time geoprocessing over the Internet. The premise of the geoprocessing is to locate mineral exploration targets that have high potential for success based on parameters chosen by the end-user of the system.**

Components integrated in the system include a spatial database PostGIS, a GIS processing engine GRASS, a GIS server GeoServer, a web server Apache, and front-end technologies OpenLayers and GeoExt. Appropriate data was sourced from the Geological Survey of Ireland to be used for the geoprocessing.

With all the components of the GIS integrated, an individual not specialised in the use of a GIS can interact with and interrogate the data through a web browser. The GIS then provides a vital role as a decision support system for locating mineral exploration targets.

Keywords-Open source; GIS; GRASS; GeoServer; Geoprocessing; Mineral exploration

I. Introduction

This paper outlines the research and development of an open source Geographical Information System (GIS) that offers real-time raster processing through a web-based interface. The focus of the raster processing is based on locating mineral exploration targets in Ireland. Key to achieving this aim is the successful integration of database technology, a GIS processing engine, a GIS server, a web server and front-end web technology to display information to the end user through a web browser.

A GIS is a system that encompasses a network, hardware, software, data, people and procedures. This project researches and integrates individual components to form a complete GIS. In the world today geographic systems are becoming more widely available, bringing geographic data to a wide audience.

The fundamental data inside any GIS is either vector or raster based. Vector data represent objects geometrically while raster data use arrays of cells. Each type of data has advantages and disadvantages but for processing multiple data sets rasters offer greater simplicity with regard to algorithms for analysis.

Processing geographic data was once only available through desktop applications but more recently is becoming available over the Internet either via client-server interaction or using a web processing service [1]. This project uses the client-server approach.

II. Geologic Conditions

This research focuses on Irish mineral exploration targets, and more specifically zinc-lead exploration targets. The reason for this is the large size of the zinc-lead industry in Ireland [2]. According to Hitzman (1996) [3], Irish type deposits display a number of common characteristics. It is these characteristics that help mineral exploration companies search for and locate such deposits. Common characteristics also mean that this is an ideal GIS project as a computer can process large quantities of information faster and more robustly than a human. This is not to say that a GIS can replace the trained human eye for discovering an economic deposit, but rather a GIS can locate general target regions that a human can then examine in greater detail.

An economic deposit is a geological anomaly, so there is no exact science in pinpointing their exact location, but there are subtle indicator signs that can be used to identify favourable areas where there is a higher than normal chance of success. This project incorporates a number of indicator data layers and produces a final output map that combines all the inputs. More data layers could be added to the input set to fine-tune the favourable areas.

III. System Setup

There is no definitive set of components to use when creating a GIS. At each step, especially with regards choice of software, there are many options from which to choose. The software chosen for this project was chosen following a period of research into each component. As new versions of software are released, each bringing ever increasing functionality, the choices made may change. The only thing that can be said with any certainty is that a GIS will always contain a network, hardware, software, data people and procedures.

Table 1 lists all the applications used in the creation of the system.

TABLE 1: APPLICATIONS USED IN SYSTEM

Function	Application	Version
Database	PostgreSQL [4]	8.4.6
Database	PostgreGIS [5]	1.5.2
GIS Server	GeoServer [6]	2.0.2
GIS Processing Engine	GRASS [7]	6.4.0
Web Server	Apache [8]	2.2.14
Web Server	PHP [9]	5.3.2
Web Server	cURL [10]	7.19.7
Front-end	OpenLayers [11]	2.8
Front-end	GeoExt [12]	1.0
Database data loader	QGIS [13]	1.5.0

IV. GEOPROCESSING

The main focus of this project is creating an integrated system that processes user specified geographic datasets to indicate the potential of a given area for mineral exploration. To this end a GIS processing application is required. The Geographic Resources Analysis Support System, GRASS, was chosen as it is the most mature Open Source GIS available. Originally developed by the US Army Construction Engineering Research Laboratory in 1982 [7], it is now maintained by a team of developers worldwide.

GRASS provides spatial modelling, visualization of raster and vector data, management and analysis of geospatial data. It is these features that make GRASS a perfect fit for this project.

Many variables and data sets can be incorporated into the geoprocessing and this project uses three, which are spatial extent, bedrock geology and historic mines. Extra data can be brought in and this would fine-tune the output.

While the focus of the processing in this project is identifying areas of mineral potential, the same processing routine could be used with different data for an entirely different purpose. For example, in place of mineral potential the focus could be locating the site of a new school. Historic mines data could be replaced by existing school locations and bedrock geology (which creates an exclusion area) could be replaced by industrial sites not suitable for school buildings. Running the processing with these data sets would create an output map ranking areas with distance to current schools while excluding areas unsuitable for a new school. This could form the first pass for locating a new school building.

A. *GRASS functions*

This section describes the GRASS functions that are used in the processing of the data layers.

1) *g.region*

Before processing takes place it is necessary to set the extent of the region within which the processing will occur. This is done to limit the area to that which is defined by the end-user of the system. The extent defines the maximum values in the north, south, east and west directions. Also to be set is the resolution of the region. The resolution will be used for the cell-size of rasters that are created. As this project is looking at data with a regional perspective (small-scale) it is sensible to set this variable to one where regional trends would be visible while smoothing out local variations. The resolution is therefore subjective. Another consideration for setting resolution is that a higher resolution generates larger raster files and takes longer to process. For example, reducing the cell-size by a half increases the number of cells four-fold. By trial and error it was found that a cell-size of 250 meters was appropriate for showing both enough detail and allowing the overall processing time be acceptable.

The north, south, east and west variables are taken from user input, with the resolution being hard-coded into the system.

2) *g.copy*

To move data from GRASS's PERMANENT mapset to the mapset used for processing, the g.copy function is used. The use of a mapset other than PERMANENT for processing is one of the business rules, or procedures, that is designed to separate data storage from data processing. The function g.copy is used to copy both raster and vector data.

3) *v.dissolve*

The vector dataset, historic mines, is dissolved based on the 'commodity' field. This creates a vector where there is one row in the database that corresponds to all the instances of the feature.

4) *v.extract*

Based on the end-users parameter selection the commodity of interest is extracted from the historic mines dissolved dataset. This function uses a 'where' clause to determine what features of the dataset to extract. In this case a variable is used to store the contents of the 'where' clause as it is dynamically created in PHP based on user input.

5) *v.db.addcol*

A column of type INT is necessary to convert a vector to a raster. For the historic mines dataset, it is the commodity field that the conversion is based on, but as commodity is of type string, it is necessary to create a column of type INT that will hold a numeric representation of the string field.

6) *v.db.update*

The newly added column is populated to allow the conversion to a raster. A value of 1 is assigned to the vector where it meets the criteria. For historic mines dataset, as data of interest has already been extracted, the value of 1 is the only value to be assigned. For the bedrock geology dataset a value of 1 is assigned to formations of interest, while a value of 0 is assigned to formations to be excluded from the final prospectivity map

7) *r.mask*

The GRASS module r.mask limits processing to only those cells within the specified mask. A mask is created using an outline of Ireland. This limits geoprocessing to within this area and thus not having data appear outside the coastal boundary of Ireland.

8) *v.to.rast*

The vector layer is converted to a raster based on the newly updated column of type INT.

9) *r.grow.distance*

This function generates a raster map showing distance to the nearest non-null cell from the input raster map. As this function generates a map showing distance to the nearest non-null cell, this is why it is necessary to extract from the historic mines dataset only the commodity specified by the user.

10) r.recode

In order to produce a map ranking areas into general categories of prospectivity a classification scheme has to be chosen. It was decided to use a range of values, one to five, where one has the highest potential and five the lowest potential. To do this the raster created in the previous step has to be reclassified. This is done using the r.recode function. The recoding parameters used here are based on Tobler's First Law of Geography [14] that says everything is related to everything else, but near things are more related than distant things. So using this law, areas near the commodity selected by the end-user are ranked higher in potential than those areas far away.

A predesigned rule file is used which is composed of three colon separated values - old value low, old value high and new value. Values less than 5000 are given a new value of 1, between 5001 and 10000 has a new value of 2 and so on to the final recoded value of 5 being assigned to original values of greater than 50001. The old values in this rules file represent distance, in meters, to the nearest historic mine of interest.

11) r.mapcalc

The GRASS module r.mapcalc performs arithmetic on raster map layers. New raster map layers can be created from existing raster map layers, numeric constants, and functions. There are currently over thirty built-in functions that are supported by r.mapcalc. The function that is used in this project takes the form of 'if(A,B)', meaning the output raster will take the value of B if A is not zero, otherwise the value assigned to the output raster is zero.

12) r.out.gdal

This function exports a GRASS raster to any raster format supported by GDAL. At this point the exported raster can be used in other applications. In this case it is necessary to export the raster to allow it be used by GeoServer. Initially the raster was exported as a GTIFF, but encountered projection issues when bringing the GTIFF into GeoServer. The raster's projection, Irish National Grid with datum TM65, was not being recognised by GeoServer automatically. The projection had to be manually set through GeoServer's graphical user interface. Having to manually set any parameter does not allow complete system automation to happen. This problem was solved by exporting the GRASS raster as an Arc/Info ASCII Grid. This format was recognised by GeoServer automatically which allows the raster to be displayed without the need for manual intervention.

13) g.remove

Upon completion of all the processing steps, the GRASS mapset used for the processing is cleaned by removing all the layers that were copied and manipulated. This prepares the mapset for the next time processing takes place. This function is used to remove both vector and raster data.

V. SYSTEM INTEGRATION

Creating a GIS involves integrating multiple technologies. Each of these has to be configured to communicate with each other to allow the automated flow of data throughout the system.

The end-user of this web based GIS interacts via a web browser with Apache, which in turn requests and receives data from a GIS server GeoServer, a GIS processing engine GRASS and a spatial database PostGIS. All of the server-side applications integrate to provide the end-user with spatially referenced maps displaying mineral potential in Ireland.

A. Automating the Components

Figure 1 shows the overall architecture of the system. The interaction between each component will be further described.

1) PostGIS to GRASS

This is the only manual part of the system and it is necessary to load data from PostGIS into a GRASS mapset only once at system creation and when additional datasets are to be added. All available vector datasets must be loaded using the GRASS command v.in.ogr.

2) PostGIS to GeoServer

GeoServer can publish data from a variety of sources, including spatial databases like PostGIS. In this instance GeoServer is being used to display the original datasets to the end-user of the system before processing parameters are chosen.

3) GRASS to GeoServer

When the processing steps are complete in GRASS an output raster is sent to GeoServer's data directory. It is this output raster that will be displayed to the end-user.

4) GeoServer to Apache

GeoServer has the ability to publish simple web pages such as HTML, CSS and JavaScript but as this system requires server-side processing a designated web server is required that can run the server-side processing. For this project Apache running PHP is used. GeoServer's WMS capability is used for displaying geographic data.

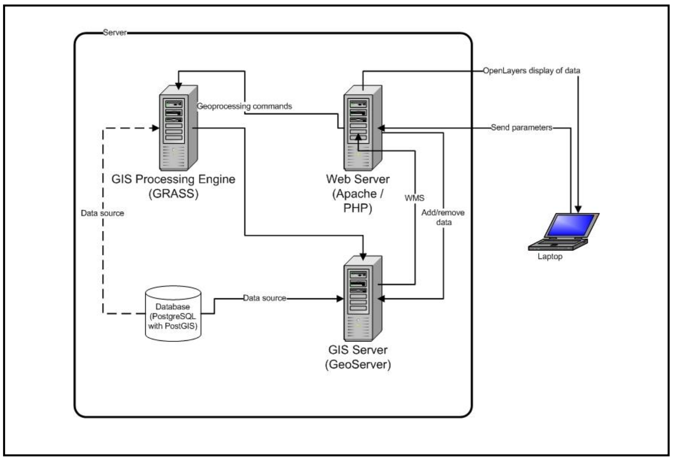

Server
Geoprocessing commands
OpenLayers display of data
Send parameters
GIS Processing Engine (GRASS)
Web Server (Apache / PHP)
WMS
Add/remove data
Data source
Laptop
Database (PostgreSQL with PostGIS)
Data source
GIS Server (GeoServer)

Figure 1: Overview of system architecture

5) *Apache to GeoServer*

Three distinct phases of tasks are carried out. These phases are pre GRASS processing, GRASS processing and post GRASS processing.

6) *Apache to GRASS*

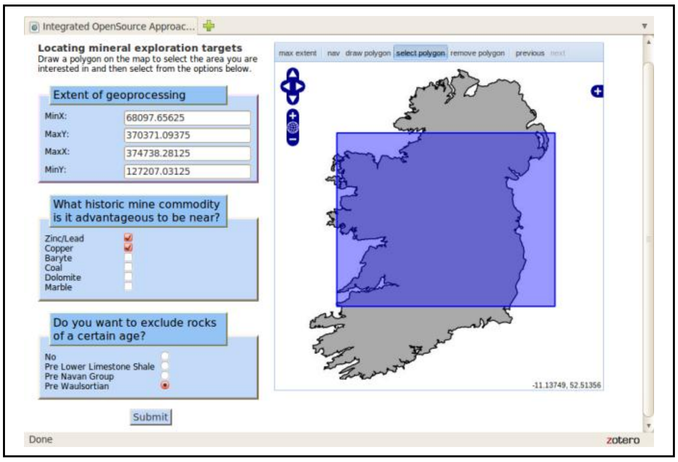
To run GRASS without using a GUI, it is first necessary to set the GRASS environment variables. This is done through PHP.

7) *User to Apache*

A user's interaction with the system is handled through the web server Apache. When the user makes a request for the home page to be loaded Apache requests the data from GeoServer before rendering the page. GeoServer in turn requests data from PostGIS if necessary. In fig. 2 the user is then presented with a page where they can select various processing parameters. The parameters that they select are taken by PHP and used in the processing by GRASS. Figure 2 shows the application home page with a number of parameters chosen. The user has the option of selecting the area within which processing will take place in addition to the selection of datasets. When selecting a data layer to include in the processing the user also selects certain parameters which are specific to that layer. The idea is to give the user as much control as possible over the processing that takes place.

8) *Apache to User*

Apache renders the HTML and PHP pages for the end-user. OpenLayers and GeoExt being JavaScript libraries mean the client machine is also involved in rendering some pages.



Locating mineral exploration targets
Draw a polygon on the map to select the area you are interested in and then select from the options below.

Extent of geoprocessing

MinX:	68097.65625
MaxY:	370371.09375
MaxX:	374738.28125
MinY:	127207.03125

What historic mine commodity is it advantageous to be near?

Zinc/Lead ☑
Copper ☑
Baryte □
Coal □
Dolomite □
Marble □

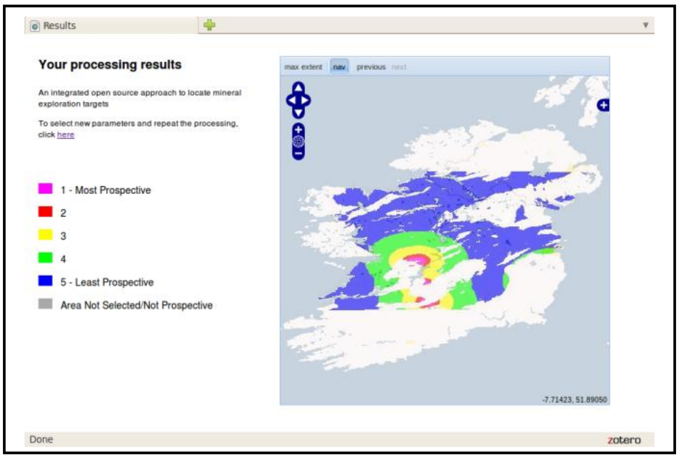
Do you want to exclude rocks of a certain age?

No ○
Pre Lower Limestone Shale ○
Pre Navan Group ○
Pre Waulsortian ●

Figure 2: Application homepage

The map on the application's home page is generated for the user with a coordinate system of TM65 Irish National Grid (ING); however the display projection is set to WGS84. ING is the underlying projection because the GRASS location was created with this coordinate system as the data available from the GSI is natively projected using ING.

To make the system usable worldwide it would be necessary to change the GRASS location to a global coordinate system such as WGS84 or Spherical Mercator. The output prospectivity map, shown in Figure 3, displayed to the user uses WGS84 as this allows overlaying the data on the global tiles of OSGEO.



Your processing results

An integrated open source approach to locate mineral exploration targets

To select new parameters and repeat the processing, click here

🟪 1 - Most Prospective
🟥 2
🟨 3
🟩 4
🟦 5 - Least Prospective
⬜ Area Not Selected/Not Prospective

Figure 3: Application results page showing final prospectivity map

VI. Conclusions

Using Open Source software the creation of a web based GIS that dynamically processes data to locate mineral exploration targets based on user input is possible. The parameters that identify the common characteristics of mineral deposits were researched and data was sourced accordingly. The geoprocessing is based on these parameters and datasets.

With all the components of the GIS integrated, an individual not specialised in the use of a GIS can interact with and interrogate the data. The GIS then provides a vital role as a decision support system for location based problems.

References

[1] P. Schut. Opengis web processing service. OGC Standard OGC 05-007r7, Open Geospatial Consortium Inc., 2001.

[2] Mining history in Ireland. http://www.mineralsireland.ie/Mining+in+Ireland/. [Accessed Apr. 22, 2011].

[3] M.W. Hitzman and D.W. Beaty. The Irish Zn-Pb-(Ba) Orefield. Economic Geology, 4:112-143, 1996.

[4] PostgreSQL Database. http://www.postgresql.org/. [Accessed Apr 22, 2011]

[5] PostGIS Spatial Database. http://postgis.refractions.net/. [Accessed Mar. 19, 2011].

[6] GeoServer Home Page. http://geoserver.org/display/GEOS/Welcome. [Accessed Apr. 21, 2011].

[7] GRASS GIS OSGeo Live v4.0 Documentation. http://live.osgeo.org/overview/grass_overview.html. [Accessed Mar. 19, 2011].

[8] Apache web server. http://httpd.apache.org/. [Accessed Apr. 20, 2011].

[9] PHP server scripting. http://www.php.net/. [Accessed Apr. 20, 2011].

[10] cURL. http://curl.haxx.se/. [Accessed Apr. 21, 2011].

[11] Open Layers. http://openlayers.org/. [Accessed Apr. 20, 2011].

[12] GeoExt Rich Mapping. http://www.geoext.org/. [Accessed Apr. 29, 2011].

[13] Quantum GIS. http://www.qgis.org/. [Accessed Apr. 18, 2011].

[14] W. Tobler. A Computer Movie Simulating Urban Growth in the Detroit Region. Economic Geography, 46:234-240, 1970.