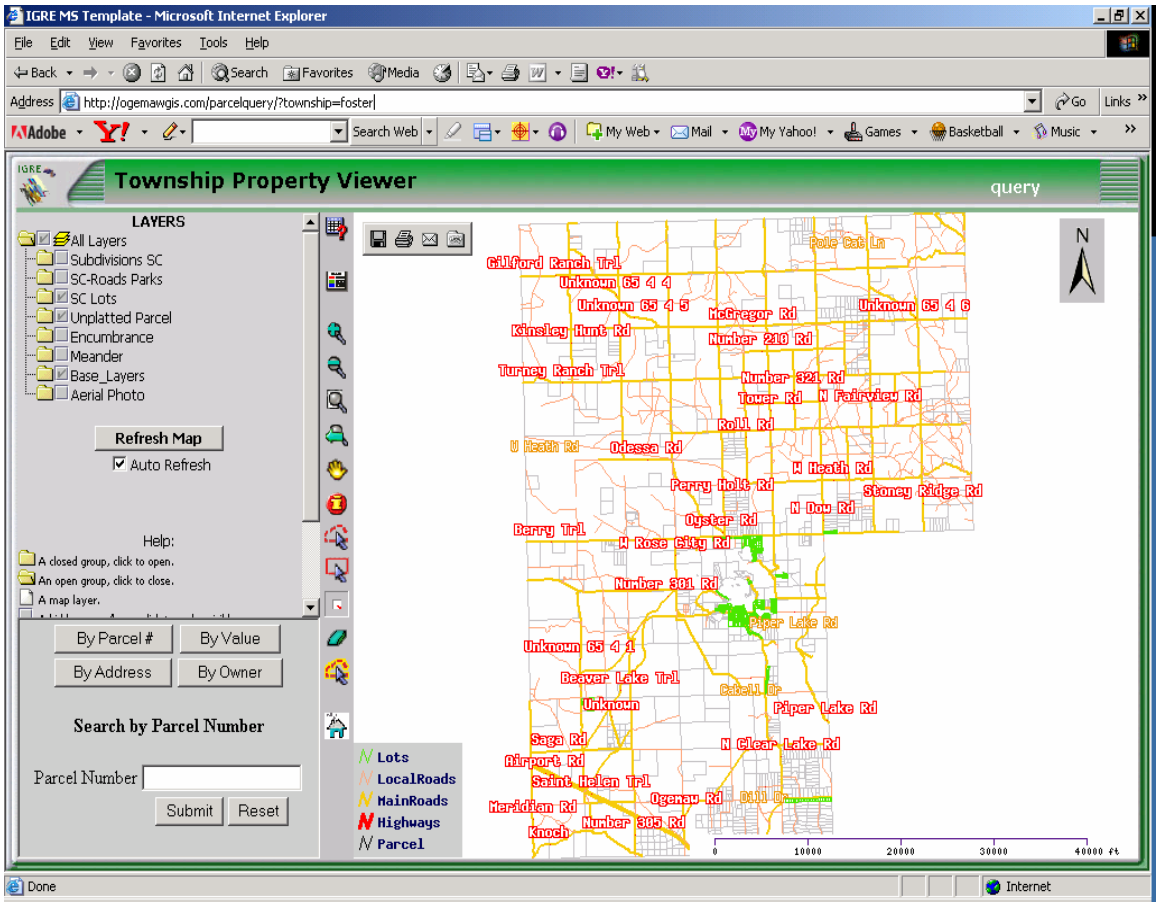
Opensource Web Mapping Applications With Mapserver and Related Technologies



Introduction: GIS and geospatial data are growing in importance for the community, because they allow users to examine new types of information in new ways (GIS and Land Records, Nancy von Meyer, 2004). More and more we are finding that land records and cadastral information represent one of the most strategic layers in GIS Today .Traditionally, mapping and land records work has involved compiling the data for focused use, such as for assessing taxes. Parcel mapping is important to see the information about growth in terms of land values and to compare that information with data showing where development is already occurring, GIS could generate another map or series of maps showing where permits have been issued, when, and for what kinds of development. For several years, some key individuals must have taken on the task of building models, layers for parcel mapping. Web-mapping is high on everyone's agenda and continue to receive massive attention and focus due to the speed of the computer, networks and the speed of the internet (bandwidth).

This paper aims to discuss the web mapping applications, available resources, and processes involved. However decision over finding a best application requires detailed needs assessment. The brief overview will cover, why open source web-mapping is popular than commercial customized applications.

Keywords: Open-Source Software (OSS), Mapserver (University of Minnesota Applications), Structured Query Language (SQL), DBMS (database management systems), Interactive Map.

1. Introduction to the Research Questions

The internet provides the information about the world by sitting on the desktop machine at one place. An internet is being used more and more to view the route applications and ESRI, Google Maps; Yahoo Maps the backbone of these applications. As part of that process a specific research question were asked.

A. Web Map should contain information for the entire city, neighborhoods, land use zones, specific street addresses and mainly parcel information. How we can build a Web-Map application based on the available resources and the partners needs.

The main purpose of the Web-map is to deliver dynamic GIS and image processing content via the World-Wide Web (WWW). The method should be effective from multiple perspectives and multiple levels to design the interface and tools for Parcel Web-Mapping. The application should have the ability to integrate data from multiple sources and the customization of products and services. The application need to be cheaper than commercial Software including cost of the development. Commercial softwares distributed by ESRI (ARCIMS) and Geomedia for Webmapping are relatively expensive and Client would still needs to hire a developer for customization based on the organization needs.

Open Source Web-Mapping Applications Methodology:

Several software manufacturers provide the market products for creating Online GIS systems. Depending on the needs and usage, the budget can go upto $5,000 and upto $30,000 to purchase the software. Most enterprises require the ability to mix and match open source and commercial software and applications in order to create a robust and flexible IT environment. Open source softwares (OSS) are popular due to following reasons,

- Easy Access
- Easy Installation
- Excellent support networks
- Cost (Cheaper than Commercial Software including cost of the development)
- Easy storage
- Easy dissemination
- The facilitation of data exchange/sharing
- Good user group and discussion threads support
- Faster and easier updating and correcting information
- The ability to integrate data from multiple sources and the customization of products and services.

Today Major Private Companies and County Governments have started adopting opensource softwares as a part of their technology due to the durability of the products. Most popular among the available products are 1) ESRI (ARCIMS) 2) Opensource Software modules. Commercial softwares ESRI are known for their standard customized applications. Webmapping with the opensource softwares needs a developer and can be done at the initial procurement cost. Open source softwares are distributed under GNU public license.

Web mapping application normally requires following components:-

A. Most important: THE MAP(s) layers

B. Map and/or Layer Control

C. Overview/Navigation Map

D. Functions (buttons): such as Navigation, Buffer, Query etc. and helpful tooltips

E. Search Options (By Owner Name, By Parcel Number, By Street Address, By Land Value)

F. Query Information

G. Online Help for the browsers

Hiring a developer would cost money, but for commercial software (customized applications) like ARCIMS, Organization needs a developer too. MapServer is an Open Source development environment for building spatially-enabled internet applications. It is originally developed by University of Minnesota (ForNet), NASA & Minnesota Dept. of Natural Resources.

The Mapserver is advantageous for the following reasons:

- Access to source code and ability and right to modify it
- Right to redistribute modifications to benefit wider community
- Free
- Excellent support networks
- Large and enthusiastic user base
- Support for many other languages
- Enabling 'spatial analysis' on-demand
- Development and support with the programming languages includes PHP (**Hypertext Preprocessor**), Python, Perl, Ruby, Java, and C#.
- Compatibility and Use of ESRI shapefiles, PostGIS, ESRI ArcSDE, Oracle Spatial, MySQL, PostgreSQL and many others via Open GIS Consortium (OGC) connection

General Structure: Traditionally GIS architecture follows a standard client-server schema, where the data processing takes place on the server side and the results are sent to clients for visualization.

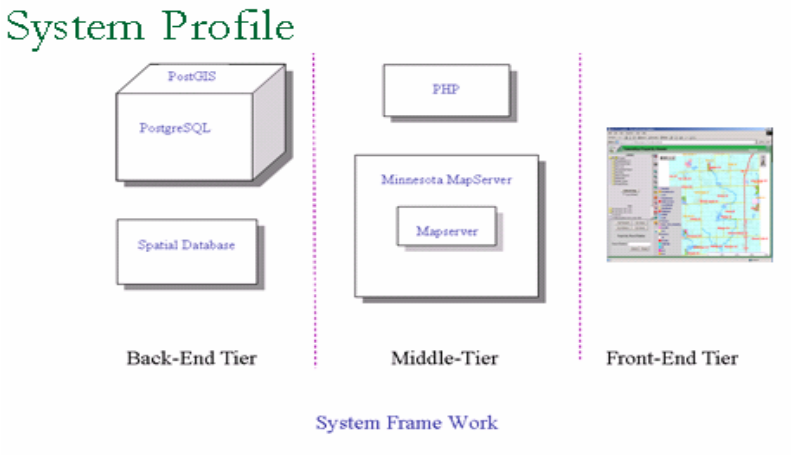


Figure 1: A System profile for Mapserver front end and back end applications. A front end is always a client side. The back-end and middle end is known as server side.

Implementation Plans:

Implementation can be done in four different phases. It is important to make a schema of the implementation. The schema may include the flow chart or data flow diagram of the processes and relevant applications. The initial product of the application should be necessarily done on the Development server and then on the Production Server. This may take a longer time, but it would save the data from being corrupted and the data will have a backup as a mirror image of the systems.

Phase 1

Server Configuration: Configuration of the Server

Operating System – Windows 2003 (recommended)

Configuration as an Application server (IIS6) and Terminal server for remote access

At the beginning Mapserver installation can be done. Following steps are for the Mapserver installation:-

- Download MapServer 4.6 with PHP 4.4.0 and up (IIS or Apache)
    - Create d:\downloads
    - From the Mapserver website, select Software Tools, Servers. Download the Minnesota MapServer 4.6 (Win32 Binaries) from the website http://www.maptools.org/php_mapscript/
    - Save zip to d:\downloads
- Open and Uncompress ZIP file
    - Double click the mapserver-4.6.1-win32-php4.4.0.zip file and uncompress into d:\downloads
- Copy MapServer executable to Web Server
    - Copy mapserv.exe to appropriate web server script directory as shown below
        - IIS d:\Inetpub\scripts\
        - Apache C:\Program Files\Apache Group\Apache\cgi-bin\
- Download the accompanying set of libwww dll's
- Open and Uncompress the libwww dlls
    - Double click the libwww_dll.zip file and uncompress into your system directory
        - (i.e. c:\Winnt\System32 or c:\Windows\System)
- Install Projection Information
    - Create a directory called c:\Proj\nad
    - Copy the file from d:\downloads\proj\nad\epsg to c:\Proj\nad

Phase 2

Mapserver is compatible with the PHP. It is also compatible with the programming languages includes Perl, C, VB 6.0 (visual basic). It is recommended to use PHP due to its ability as a web application and easiness in terms of developing the codes. PHP 4.4.0 is available free on the website (URL http://www.php.net) to download.

Phase 3

Any GIS system belongs to the class of computer systems that require the building of large databases before they become useful. The databases are able to save a spatial data due to the advancement in a technology called R-tree. The opensource database such as MySQL, PostgreSQL is capable to store the non-spatial and a spatial data upto giga-byte size. PostgreSQL is a powerful, open source relational database system. PostGIS is a project which adds support for geographic objects in PostgreSQL, allowing it to be used as a spatial database for geographic information systems (GIS), much like ESRI's SDE or Oracle's Spatial extension. PostgreSQL 8.0 is available free on the website (URL http://www.postgresql.org/) to download.

Phase 4

Maintenance of the application is often considered as a "critical" success "factor" in successful GIS usage. The Maintenance of the application on the server side is a vital step. The Database systems play an important role in a decision while doing needs assessment. The data flow diagram can be developed based on the system configuration. Sometimes schema can be developed with MS-Visio 2000 and convert it into UML or case tools. The above mentioned method is very popular for developing the Database and converts the schema into ESRI shapefile format. The database can only contain non-spatial data, which may includes assessor and legal description.

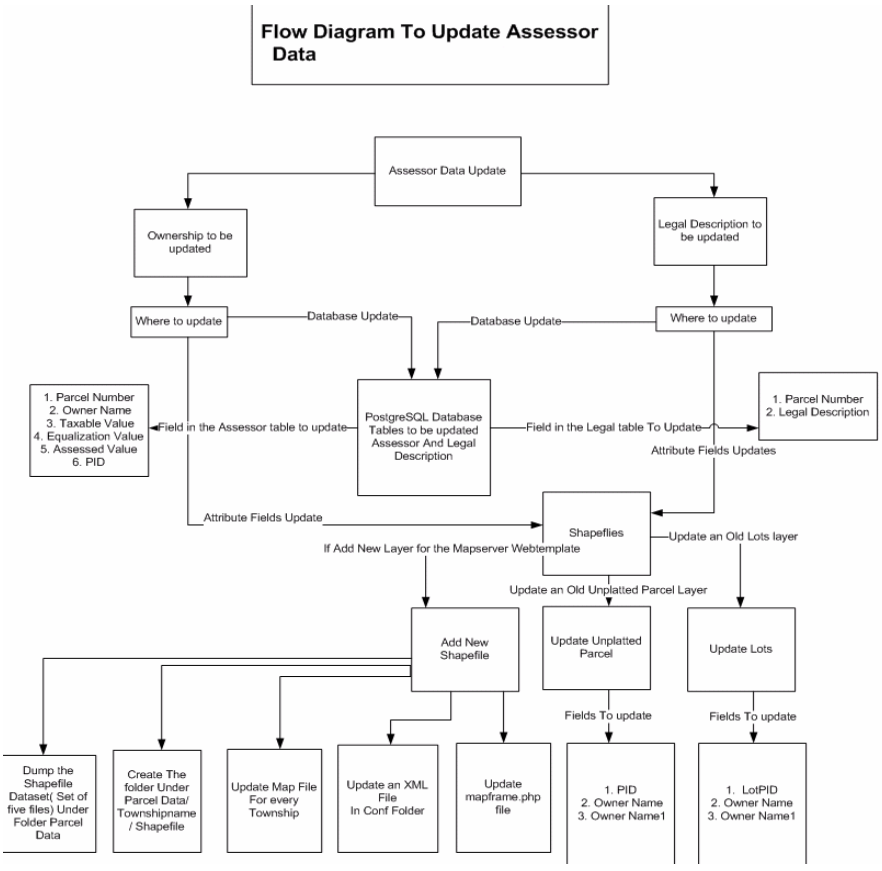


Figure 2: Data Flow diagram to develop a GUI (Graphical User Interface) for the Mapserver Application.

Testing the Configuration:

Mapserver installation and version can be tested by running the command prompt shown in figure 3

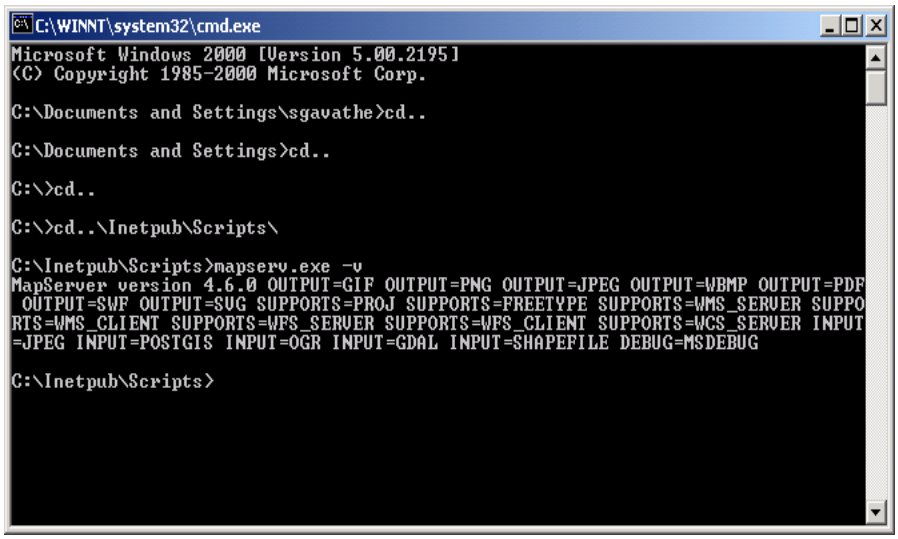


Figure 3: Mapserver version and supported applications to view are important, when we build a mapserver.

Role of the Mapserver:

Mapserver is comprised of the many different components. Once the testing of the installation is done. Next step is to create a mapfile.

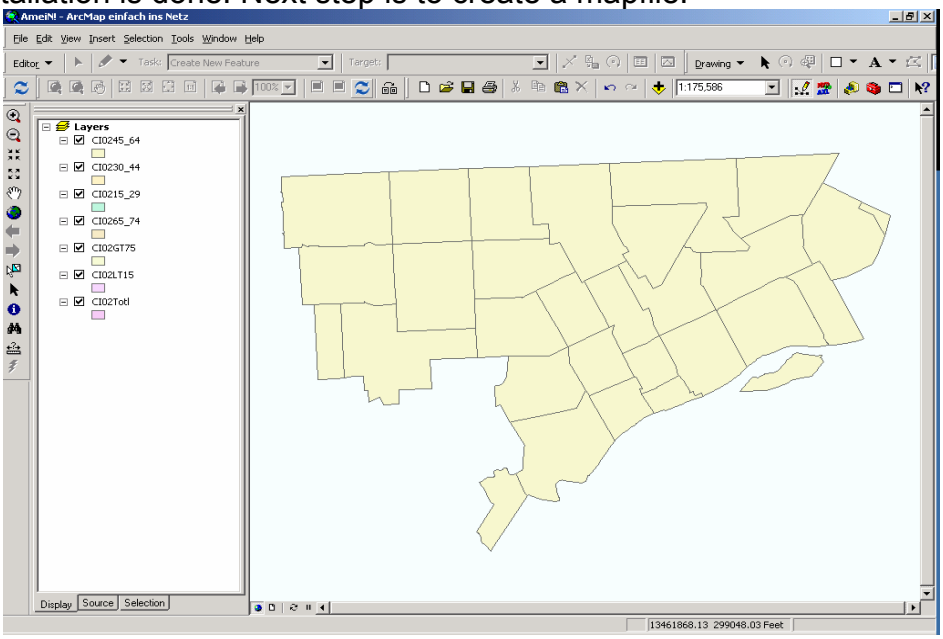


Figure 4: ArcMap showing Layers with the symbology

The amein_101 an opensource ArcMap extension can be used to convert

ArcMap MXD to the map file. The second option available is to open the notepad and create the Map File from scratch. The map file is an important component of the Mapserver to communicate with the Web Servers and parse the ESRI shapefile geometry, annotation and symbology.

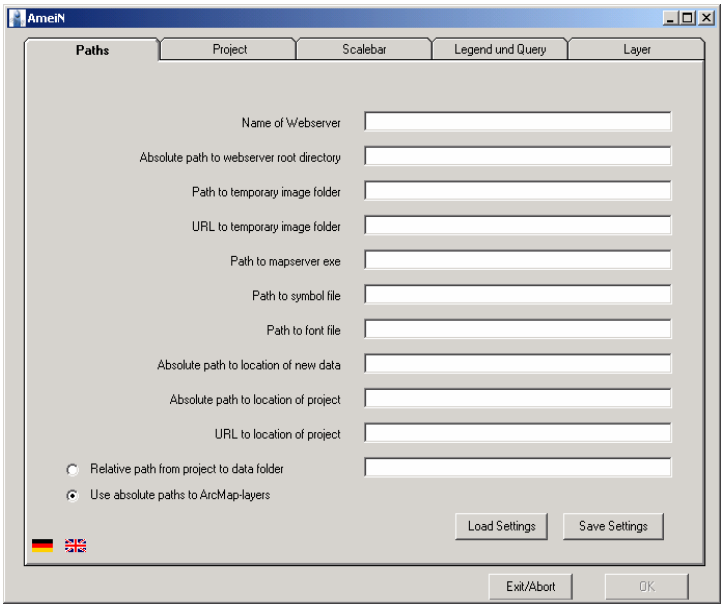


Figure 5: A GUI (Graphical User Interface) of Amien ArcMap to Mapserver Extension.

The structure of the Map file is important to display the map including the order of the layer in the Map file.

The map file starts with the “MAP” as the first tag and it ends with the “END” Tag

```
#
# Start of map file
#
MAP
 NAME PARCELS
  IMAGETYPE png24
  STATUS ON
  SIZE 800 800

  EXTENT 649256.612337 412449.245459 659112.141770 422310.769293

  UNITS METERS
  SHAPEPATH "Shapefiles"
  IMAGECOLOR 255 255 255
  TRANSPARENT OFF
```

```
  FONTSET   "fonts\fonts.list"
  SYMBOLSET "symbols\symbols35.sym"
  DEBUG ON

  OUTPUTFORMAT
    NAME "png"
    MIMETYPE "image/png"
    DRIVER "GD/PNG"
    EXTENSION "png"
    IMAGEMODE PC256
    TRANSPARENT FALSE
  END

  WEB
    TEMPLATE  'maptemplate.htm'
    IMAGEPATH '/output/'
    IMAGEURL  '/output/'
  END
END
```

A definition of every object within Mapfile is described below:-

NAME: prefix attached to the map and legend GIF names created using this Map file. Should be kept short.

EXTENT: Spatial extent of the map to be served.

SIZE: Size in pixels of the output map

SHAPEPATH: Path to the directory holding the shapefiles or tiles. Can include sub-directories

LAYER: Singles the start of the Layer object.

NAME: Unique short name for the layer

DATA: Full filename of spatial data file

STATUS: Sets the current status of the layer. Defaults to be turned on.

TYPE: Defines how the data will be drawn (point, line, polygon, circle, annotation, rater, query)

CLASS: Signals the start of the Class object

COLOR: Color for drawing features

OUTLINECOLOR: Polygon outline color

The directory structure layout is an example of a single MapServer application. If we plan to provide multiple applications, we should have a lot of redundant data if we replicated this structure for each application

```
MapServer_App        <-- this is where the map and template files will reside
      |
      |-----images    <-- for graphics/images used in HTML teplates, or for
      |                    reference map images.
      |-----data
      |        |
      |        |----vector  <-- for your vector data, i.e. shapefiles
      |        |
      |        |----raster  <-- for raster data, i.e. GeoTIFF images
      |
      |-----fonts    <-- for your truetype fonts (and fonts.list)
      |
      |-----symbols <-- for symbolset files
```

Figure 6: Mapserver Application directory standard structure

The static web-map testing can be done by typing (http://localhost/Scripts/Mapsrv.exe?Map=<path of the Mapfile>&mode=Map) in the browser.

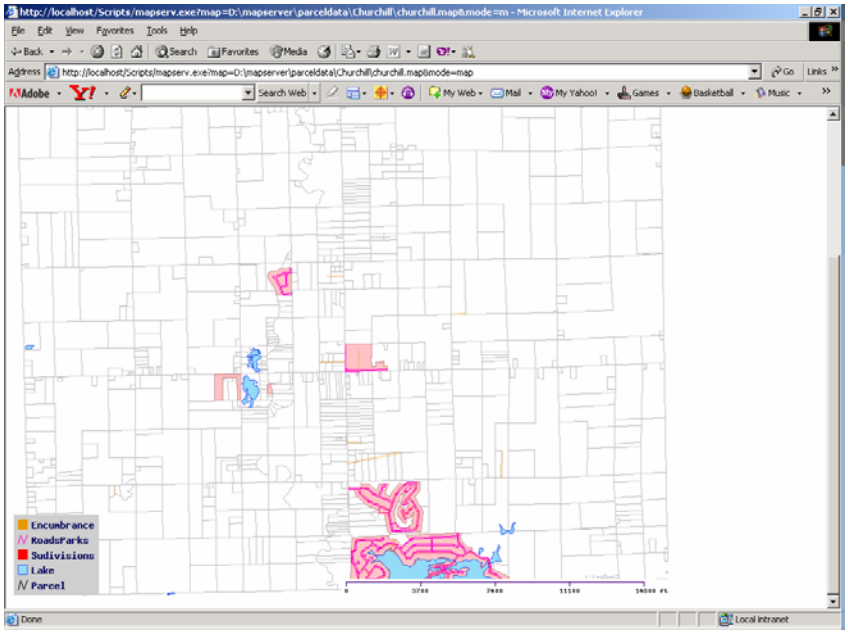


Figure 7: A static Map showing legends and a PNG/JPEG Image generated by Mapserver.

Data Collection & Limitations:

The data is usually collected from the partner's GIS storage. The partner would need to purchase a data or distribute. Maps display layers of geographic features representing the entities such as cities, rivers, property parcels, Lots etc. The "Success of the Web-GIS Application" also depends upon the speed of the internet. In an Online GIS, however, care must be taken to encourage user interaction by keeping the initial displays simple and general. In the initial display the layers which are in Mega-Bytes size such as aerial photography, satellite imageries should turned off by default, while developing a Map file for the Web-Mapping application.

Developing the codes:

The dynamic map (interactive map) need to develop a code based on the basic functions required for Web-Mapping application. The codes are to be developed on the following basic required functions of the Map:-

- Show Legend Or Layer List
- Zoom In
- Zoom Out
- Full Extent
- Previous Extent
- Panning Map Display Area
- Identifying Features
- Select Feature By Polygon
- Select Feature By Rectangle
- Measuring Distances
- Clear Selected Features

- Buffering Features

The query function for the Buffer needs to pay more attention while developing the codes. It should be based on the unique ID within the attribute table of the shapefile and a PostgreSQL database. The Search Engine application is mainly for the parcel ID query, owner address, owner name and the Land Value.

Figure 8: The Parcel Information Query Templates

Attribute	Description	DataType	Field Width
Subdivision boundary: Polygon Feature			
ConveyName	Subdivision Name	Text	85
ConveyID	Subdivision Number (e.g. 350A, 610F)	Text	10
CalAcreage	Calculated Acreage	Double	Precision 14 Scale 2
SC First Division (Block, ROW, Lake, Other): Polygon Feature			
FirstDivID	Block Number/ ROW Name/ Lake Name	Text	30
CalAcreage	Calculated Acreage	Double	Precision 14 Scale 2
ConveyName	Subdivision Name	Text	85
SC Second Division (lots): Polygon Feature			
SecDivID	Lot Number/Other	Text	15
LotPID	Lot ID (e.g. 001- 340 - 00). 00 is the split number	Text	15
CalAcreage	Calculated Acreage	Double	Precision 14 Scale 2
ConveyName	Subdivision Name	Text	85
FirstDivID	Block Number/ ROW Name/ Lake Name	Text	30
OwnerName	Owner Name	Text	50
OwnerName2	Owner Name 2	Text	50
Ow_Address	Owner Address (St Num, St Name, Apt)	Text	40
City	City of Owner	Text	25
State	State of Owner	Text	15
ZIPCode	ZIP Code of Owner	Text	15
Unplatted Parcel (Unplatted Parcel, Park, Lake, GovtLot, ROW, GAP): Polygon Feature			
PID	8 digit Short Parcel ID (e.g. 001- 340 - 00)	Text	15
ParcelName	E.g. 'OwnerParcel', 'Park', 'Lake', 'GovtLot', 'ROW', 'Unknown', ROW Name, Lake Name	Text	30
CalAcreage	Calculated Acreage	Double	Precision 14 Scale 2
OwnerName	Owner Name	Text	50
OwnerName2	Owner Name 2	Text	50
Ow_Address	Owner Address (St Num, St Name, Apt)	Text	40
City	City of Owner	Text	25
State	State of Owner	Text	15
ZIPCode	ZIP Code of Owner	Text	15

Figure 9: An attribute table data dictionary.

The final output after the development is tested on the development server.

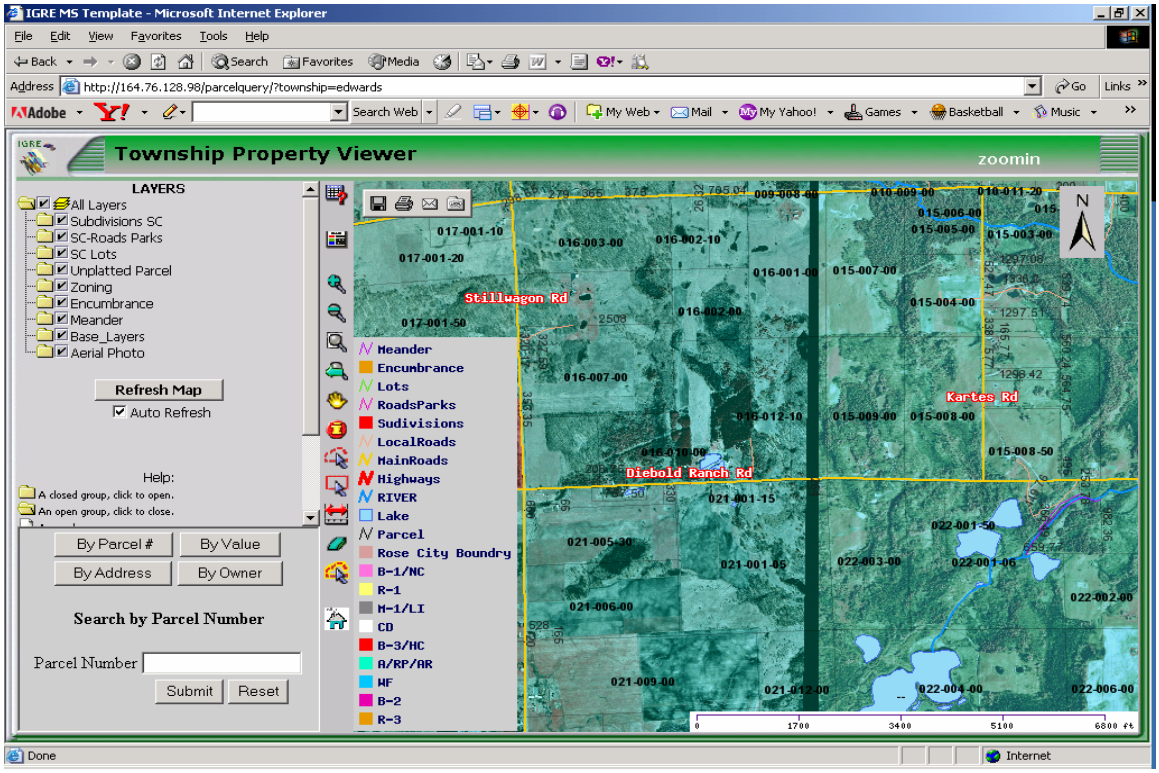


Figure 10: The final web-map showed the Map, which could have also builted with the commercial expensive software.

Conclusion:

This introductory guide is intended to help in the investigation of what is involved in developing a system that will allow the partner to take advantage of the tremendous benefits of the web-mapping application. The needs of every organization are unique and the range of possible applications so broad. Since GIS technology is taking its step more towards web-mapping applications, it is important to do the initial assessment of the available dataset and functional requirements. The tools to harness the power of web-mapping GIS are more use

and more affordable than ever before due to Open Source Softwares and its integration with the commercial softwares.

Reference:

The Design and implementation of Geographic Information Systems (John E. Harmon, Steven J. Anderson)

Internet-Based GIS for Local Government (Nicholas Colas, Cayuga County Planning Department Auburn, New York)

The GIS and Land Records, The ArcGIS Parcel Data Model (Nancy Von Meyer, ESRI press, 2004)

Beginning Mapserver, Open Source Development (Bill Kropla)