

WEB-BASED SMART FARM VISUALIZATION GIS TOOL

**A Paper
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science**

By

Abhishek Agarwal

**In Partial Fulfillment of the Requirements
for the Degree of
MASTER OF SCIENCE**

**Major Department:
Computer Science**

November 2014

Fargo, North Dakota

North Dakota State University
Graduate School

Title

Web-based Smart Farm Visualization GIS Tool

By

Abhishek Agarwal

The Supervisory Committee certifies that this *disquisition* complies with North Dakota State University's regulations and meets the accepted standards for the degree of

MASTER OF SCIENCE

SUPERVISORY COMMITTEE:

Dr. Anne M. Denton

Chair

Dr. Kendall E. Nygard

Dr. Fred Riggins

Approved:

10/15/2015

Date

Dr. Brian M. Slator

Department Chair

ABSTRACT

Visualization of data is much more vibrant and informative than a written description could ever be. Continued research on information visualization has resulted in discovery of a number of designs and variety of techniques. Number of people who are engaged in the research of natural products who often either conduct field collections themselves or collaborate with partners. The information gleaned from such collecting trips (e.g. longitude/latitude coordinates, geography, elevation and other field observations) has provided valuable data to the scientific community. Geographic Information Systems (GIS) have been used to display, and analyze geographic data. The aim of the project is to provide a web interface to visualize data with geographic coordinates giving the ability to the users to select an area on the map, which will serve as an input for generating meaningful chart representation. Tools used in this project are JQuery, Google Maps/Charts, Spring MVC and MySQL database.

Keywords: *Visualization, Geographic Information Systems, Geo Spatial*

ACKNOWLEDGEMENTS

I would like to express my gratitude by thanking everyone who has supported me during the completion of this research paper. I would like to thank Dr. Anne M. Denton for guiding me all these semesters. Her support and vital encouragement contributed a lot in completing my research. I would also like to thank the Computer Science department for offering courses that were very useful in building this application. I would like to also thank Dr. Kendall E. Nygard for this constant support and direction. Last but not least, I would like to thank my parents, Sandeep Agarwal and Charu Agarwal and my brother Udit Agarwal, for guidance, love, much needed support, and making it possible to study at North Dakota State University

TABLE OF CONTENTS

ABSTRACT.....	iii
ACKNOWLEDGEMENTS.....	iv
LIST OF FIGURES	vii
INTRODUCTION	1
ORGANIZATION OF THE PAPER.....	5
Why do we use Google Maps?	5
Concept	7
Axis Organization	7
Interactive Markers	8
Evaluation	8
A prototype of web-based Smart Farm Application for Data Analysis.....	10
Google Maps API	12
Google Map’s KeyDragZoom	12
Google Charts	12
Geo-Spatial Functions.....	13
R-tree And Spatial Indexing	13
Spatial Functions.....	13
Architectural Taxonomy	14
Model-View-Controller	14
Dispatcher Servlet.....	14
RESTful Architecture	15
View Layer.....	16
Data Layer.....	17
Demonstration of Functionality	18

Area of Interest	18
Marking a Selection	19
Selecting a Marker	20
Analyzing Charts	21
Summary and Future Work.....	22
RELATED WORK.....	23
REFERENCES	25

LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
1: The interface role of maps in spatial data infrastructure	2
2: Visualization in GIScience environment using Google Maps API	6
3: Screen-shot showing the Map initialized ready for user-input	7
4: Application generated parallel coordinate graph	9
5: Generated scatter-chart	11
6: Generated parallel-coordinate graph that gets populated	11
7: Smart farm application architecture.....	14
8: Smart farm application dispatcher servlet	15
9: Looking at the Area of Interest	18
10: Marking a Selection	19
11: Selecting a Marker	20
12: Analyzing Charts	21

INTRODUCTION

In recent years, information visualization and computer-assisted use of visual processing to gain understanding, has become a topic of significant development and research. As people try to cope with massive amounts of data for electronic information, browsing through very large archives can frustrate and eventually impede the retrieval of information. It has become very important to search catalogs of very large electronic information that combines computer visualization with the descriptive and analytical power of geography. Information spaces come in many forms. Each is characterized by a certain structure and content. Consider for example, newspapers present information in chunks in the forms of articles, which are rarely related with each other [1]. One can easily jump back and forth between newspaper columns, but their commonalities are however important in determining the positioning of articles within categorical sections, e.g. “national”, “entertainment”, and “technology”. The information space for this project presented its own challenges, from deciding to use of layers on a map or representing the data in some other form. We decided to use charts because of the visual benefits to use charts for visualization rather than clutter the Map space with multi-colored layers on the map area. This way the map representation can remain clean and provide the user with a clean slate to provide inputs required for population of the charts. In addition to providing the user way to clearly visualize data, we have also made it easy for the users to select the data correctly on the map. By providing hot-key enabled area selection on the map, it becomes easy for the user to visually select the area of user’s choice and see the results populate dynamically for every selection they make.

Maps are used to visualize geospatial data, that is data that refer to the location or the attributes of objects or phenomena located on Earth. Geospatial information is different from

other information in a way that the data has a special characteristic that refers to objects or phenomena with a specific location in space and therefore has a spatial address [2]. Maps help their users to better understand geospatial relationship and from maps, information on distances, directions and area sizes can be retrieved, patterns are revealed and relations understood and quantified. They provide the functionality to query the database technology and computer graphics images that resulted in new and alternative presentation options such as animated maps. Access to interactive maps is no longer limited to professionals. In this project, the Google Maps has been used as an interface for the user to interact with the underlying data. Using maps as an interface allows the possibility for the users to link their application-based models to the data contained in the systems to find out how the underlying data can be visualized. New infrastructures for accessing geospatial data are being developed all over the world in order to monitor the resources of our modern societies. Access to the data needed, requires complex querying procedures that are simplified when using maps to pinpoint the areas and themes for which the data are needed (Figure 1.1).



Figure 1: The interface role of maps in spatial data infrastructure

Using Google Maps/ Earth has provided the flexibility to add my own data to the maps and even share it with others. My effort has been to develop an infrastructure for accessing the

Smart farm data to better visualize and understand the data/data-relationships hidden in the data, because this is a web-tool it also makes it easy for its users to access it from anywhere. While developing the tool we were very conscious to make it modular so that it is easily extendable, RESTful architecture is used to make-sure that the concerns are separated adequately and the backend data processing is completely independent. The map in this project is not a final product; rather it is used for storage and selection of geospatial data. There is an introduction of on-screen maps and according to the area selected on map, scatter plots are generated. The visualization that is generated can be applied in analysis, for instance in order to manipulate known data.

- Reality: real world objects like houses, roads and fields
- Data landscape model: Contains abstracted information of world
- Data cartographic model: science behind drawing maps
- Map: Representation with dots, dashes and patches
- Cognitive map: Generating mental map from visual information

The trend toward web-based cropland management tool that is integrated with GIS to manipulate spatial information for crop analysis can be viewed as a natural trend due to the potential benefits of the Internet. The spread of broadband Internet has stimulated operation of models via the Internet communication environment, and several hydrologic models including WWW- enabled L-THIA [3] and WWW National Agricultural Pesticide Risk Analysis (NAPRA) [4] have been developed and operated in Internet environments.

Geographic information system: For most disciplines working with geospatial data, one of the first uses of computer was to create an inventory of discipline-dependent data.

Geographical Information Systems (GIS) has the functionality of working with geospatial data. It has introduced the integration of geospatial data from different kinds of sources.

GIS integrated with hydrologic models can supply land information including slope, aspect, stream lines, soil attributes, land use and numerous other data in grid and vector formats. Grid format data are relatively easy to manipulate compared to vector-based data because grids store “implicit topology”, whereas vector data requires “explicit topology”. The operators are more efficient for grid data and the grid format is often well matched with the spatially distributed simulation model conceptualization and data needs [5]. Thus, grid-based GIS tools have been widely integrated with distributed models [6]. Several distributed models have been integrated with GIS data preparation and visualization capabilities [7].

Think of it like this, the main aim for the user was to populate the database from which they could produce maps that were previously created manually. When this was achieved the next phase was spatial analysis of collected data, which led to the need of GIS. GIS offers the possibility of integration of geospatial data sets from different kinds of sources, such as surveys, remote sensing and statistical database.

ORGANIZATION OF THE PAPER

The paper is structured as follows. First we provide details about the need to use Google Maps, which is extensively used in this project. Then we introduce the concept used in the project i.e. the Axis organization and the Interactive Markers that motivated our architecture to manipulate the underlying data we have. Next we focus on the Evaluation and then talk more about the architectural framework that is specifically centered on the technical details of this project. We will describe this framework in more detail, in the Architectural Taxonomy section. We then present a Demonstration of the Application and finally we have our Results and Discussion section where we have talked about the outcome of the project and what are the possibilities of future work on this project.

Why do we use Google Maps?

Google Maps is one of the best free tools available for visualizing Geo-spatial data. The flexibility provided by Google Maps to combine non-geospatial data with geospatial data that is mainly used for identification and location of a specific object is what makes the use of Google Maps important in our application. Google Map also provides us with the ability to customize the Maps and present information, which can be very interactive and useful for the user.

Visualization can be used to explore, for instance in order to play with unknown data. For example “Which of those data sets reveals patters related to the current problem studied?”

Visualization is applied in analysis, for instance in order to manipulate known data. (Figure 1.2)

In a planning environment the nature of two separate data sets can be fully understood. These are not directly related but an overlay can combine both datasets to determine their possible geospatial relationship. Maps are used to synthesize the results of the analysis.

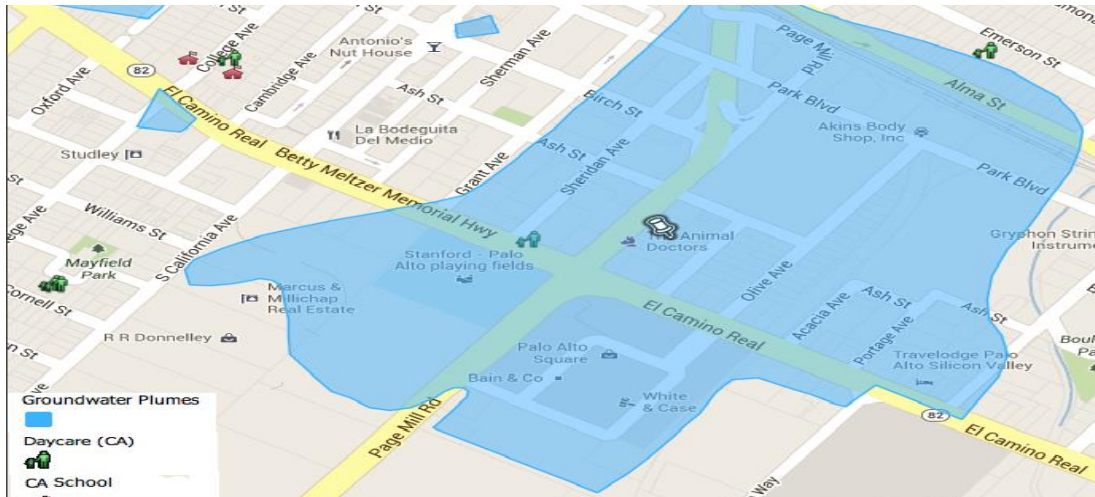


Figure 2: Visualization in GIScience environment using Google Maps API

The database from which they are generated can be queried and some basic analytical functionality can now be accessed through menus or legends. A major difficulty in the analysis of geospatial data [8] is visualization and the background against which the provision and visualization of geospatial information takes place. Web has become the source of spatial data and also a means for distributing the results of visualization. Research on information visualization has reached the point where a number of successful point designs have been proposed and a variety of techniques have been discovered.

Crop management queries typically include “how” and “what-if” questions. For example, if a cropland has experienced land use changes, or will be altered in the future, then the question may be how the land use alterations are affected. The problems in operating cropland models are typically not only the information technology and computational requirements, but also the effort required for data acquisition and preparation to run the model to obtain appropriate results that can be used in identifying a solution for the problem, including model validation. Thus, advances in information technology, including databases, GIS, graphical user interfaces and the Internet, have significant roles in the analysis of cropland.

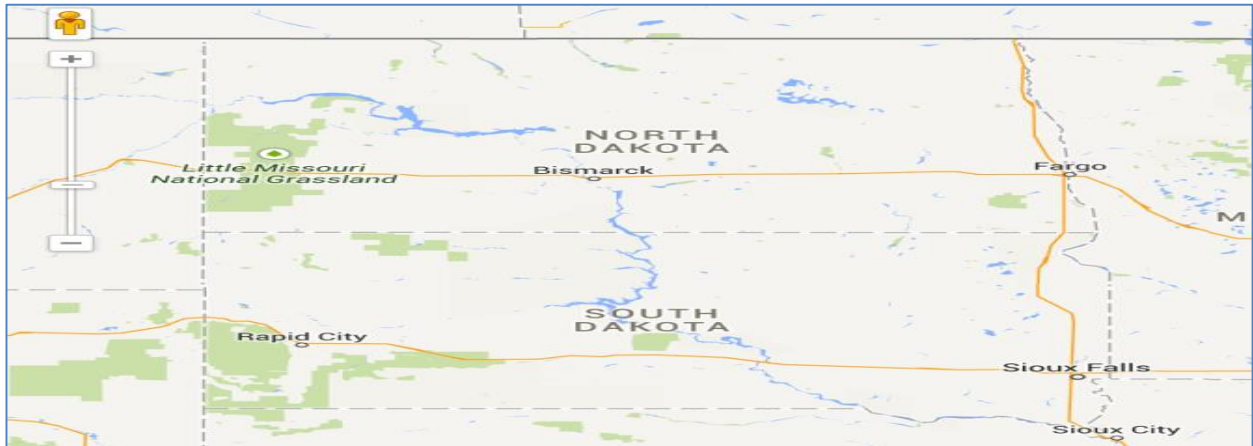


Figure 3: Screen-shot showing the Map initialized ready for user-input

Concept

Axis Organization

Axis organization in parallel coordinate is a common way of visualizing high-dimensional geometry and analyzing multivariate data. This visualization is closely related to time series visualization, except that it is applied to data where the axes do not correspond to points in time, and therefore do not have a natural order. Therefore, different axis arrangements may be of interest. Because of the possibilities derived from different axis arrangement we can potentially uncover some interesting patterns. When designing the axes for the graph, we tried meaningful combinations to come up with pattern, which can provide us some insight. We used parallel coordinate graphs to visualize the underlying data in different settings with different set of axis organization. Naturally different combination of axis sequence will produce graphs that look completely different and may display different sets of patterns or no patterns at-all depending on the axis configuration. After trying multiple different combinations we realized that the following axis configuration produce the best result with information that can be easily understood to represent a pattern.

- Axis 1: Observation Time
- Axis 2: cover-type
- Axis 3: cover height
- Axis 4: run-off

When defining the sequence of axis it is important to keep in mind that the parallel coordinate graphs resemble closely to the time series visualization so a logical construct to axis often works best.

Interactive Markers

Interactive Markers is another very useful feature provided by the Google Maps API, which enables the users to interact with the Map in a way that provides possibility for the user to take actions on the Markers on the Maps. When designing our application we needed a way for the users to analyze individual Marker points instead of putting/visualizing the whole underlying dataset. The tool that we developed, drops individual markers on the map which represent underlying information corresponding to the point where the Marker is dropped, this not only provides more information to the user about what data is present in the underlying dataset but it also helps them visually see the area they are analyzing. We attached events to the Markers that trigger an event to the backend service requesting underlying data for that Marker. This way of individual Marker analysis provides granular information about areas on the Maps.

Evaluation

Because we are trying to compare multivariate data and using visualization that is closely related to time series visualization, we thought comparing visualization for Water and Snow can be an interesting way to observe and compare the two visualization sets. Here's a sample chart that is generated for a point:

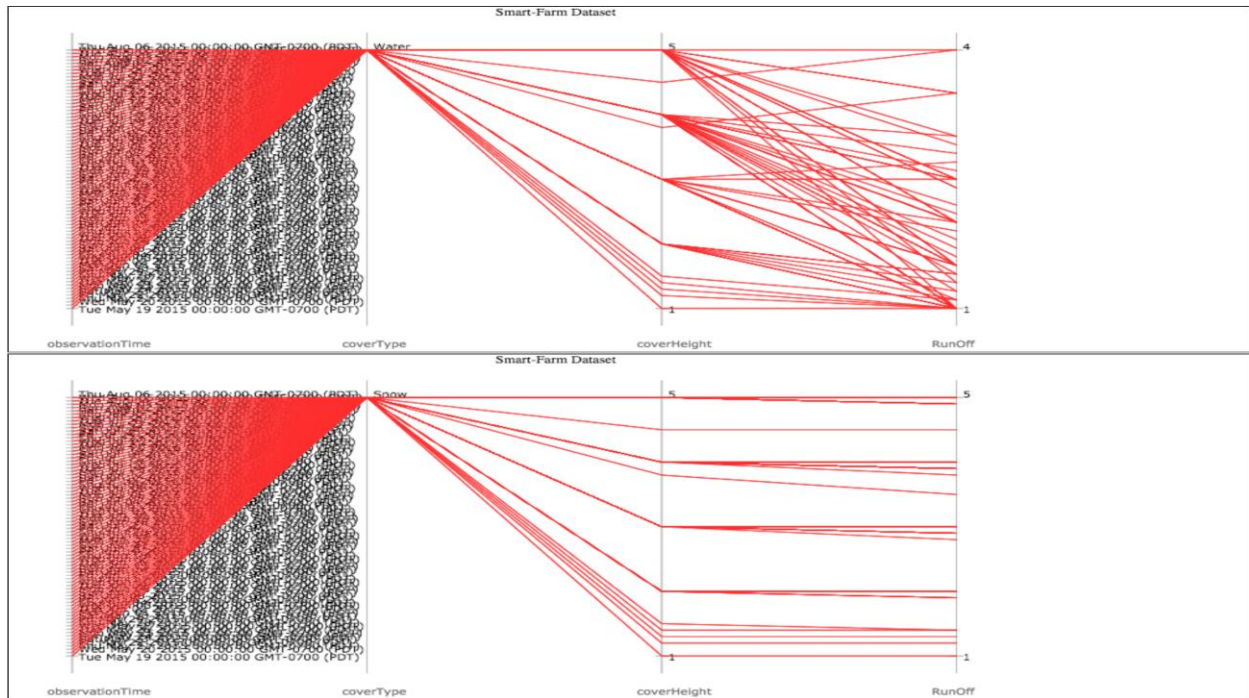


Figure 4: Application generated parallel coordinate graph

In the first case of "Water", we can observe that the run-off decreases more for a slight decrease in the cover height, which is evident from the parallel coordinate lines that get scattered towards the decrease in run-off. In the second case of Snow we observe that the decrease in run-off is proportional to the decrease in cover-height.

To further clarify the observation we see above, here's an example with hypothetical values to better understand the observation:

Case 1: Water

Cover height: 10 cm

Run-off: 50 [m³/s]

After few days the drop in cover height and run-off will look like the following:

Cover height: 5cm

Run-off: ~25 [m³/s]

Case 2: Snow

Cover height: 10 cm

Run-off: 40 [m³/s]

After few days the drop in cover height and run-off will look like the following:

Cover height: 5cm

Run-off: ~15 [m³/s]

A prototype of web-based Smart Farm Application for Data Analysis

A prototype application that utilizes the conceptual system described in the previous section is presented in the following section. The run-off and cover-type observation corresponding to a point (latitude/longitude) for an observation-time is based on the underlying data-model on which we apply geo-spatial functions to generate the correct JSON structure that serves as an input used for population of charts on the web-page. The HTML-based user interface including the GIS functionality ties itself to the backend data-model to make sense out of the underlying data-model.

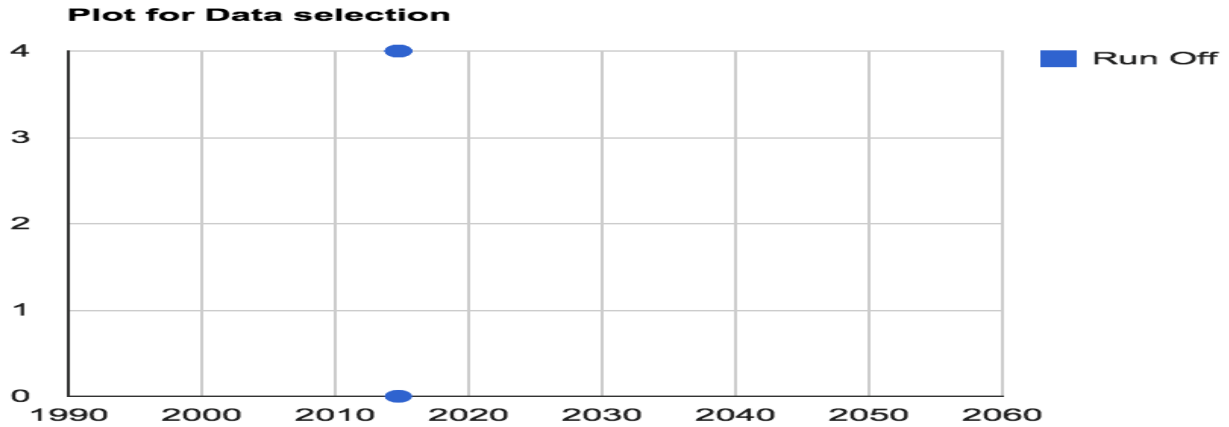


Figure 5: Generated scatter-chart

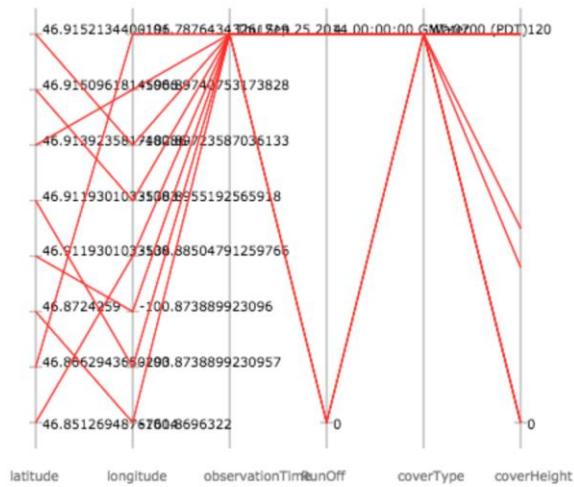


Figure 6: Generated parallel-coordinate graph that gets populated

[Figure 5 and 6] above represent the two charts that were selected for visualization to be included as part of the Smart Farm application.

In an effort to make the project easily extensible for future work we made use of the best software engineering principles. Following resources were used in development of the tool:

- Google Maps API
 - Google Map - KeyDragZoom
- Google Charts
- Spring MVC / Spring JDBC-Template

- MySQL Spatial Analysis Functions
- Maven

Google Maps API

Google Maps API is a collection of APIs that enable us to overlay our own data on a customized Google Map which helps us create engaging web and mobile applications with Google's powerful mapping platform. Our decision to use Google Maps API was primarily made because of its industry leading standard. The landing page on the tool has a Map section on the left side of the screen (Figure 1.3), which serves as the scratchpad for the user to play and make selections for data processing.

Google Map's KeyDragZoom

This utility class helps user provide the ability to define selection on a Map area, which serves as an input for the Google's chart population.

Google Charts

Provides us with the ability to create powerful, simple and easy to use charts that are easily extendable and visually appealing.

Geo-Spatial Functions

Smart Farm application needs to access data, which is different in nature than the tabular data represented in tables. When drawing rectangle on a Map representation, we are actually drawing on a flat surface that shows a projected image of the earth. Minimum bounding rectangle (MBR) also known as bounding box or envelope, is an expression of the maximum extents of a 2-dimensional object i.e. polygon in our case, in other words min (x), max (x), min (y), max (y). MBR is a 2-dimensional case of the minimum-bounding box. MBR's are also an essential prerequisite for the R-tree method of spatial indexing.

R-tree And Spatial Indexing

R-trees are tree data structures used for spatial access methods i.e., for indexing multi-dimensional information such as geographical coordinates, polygons. A common real world usage for an R-tree might be to store spatial objects such as restaurant locations or polygons that can be used to represent all latitude/longitude points that fall in that polygon.

Spatial Functions

Spatial functions are used to query data in a geometric dataset. In the dataset that we are querying we are mostly concerned with the geometric object of polygon, we for this project decided to use MySQL Geo-Spatial extension that made it easy for us to query complex relationships on the Geo-Spatial dataset. Just like database systems use indexes to quickly look up values and the way that most databases index data is not optimal for spatial queries we instead decided to use a special index called "spatial index" that makes it easy to speed up database operations on Geo-Spatial data. Spatial indices are used by spatial databases to optimize spatial queries. Conventional index types do not efficiently handle spatial queries such as how far two points differ or whether points fall within a spatial area of interest. Second scenario is a valid

scenario for our project and we wanted to make the data retrieval from this spatial data as fast as we could so we created a spatial index using the following:

```
UPDATE mytable SET coords = GeomFromText(CONCAT('POINT (' , lon, ' ', lat, ''))
```

Architectural Taxonomy

Model-View-Controller

Model-View-Controller architecture has been followed in developing the Smart Farm tool. Spring web MVC framework provides model-view-controller architecture support and other ready components that can be used to develop flexible and loosely coupled web applications. MVC pattern results in separating the different aspects of the application (input logic, business logic and UI logic), while providing a loose coupling between these elements.

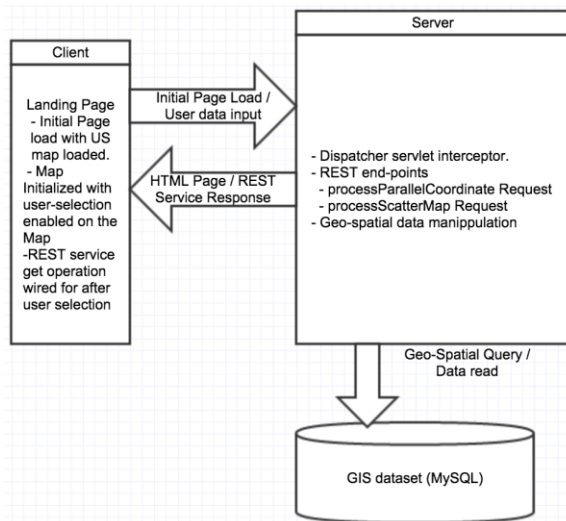


Figure 7: Smart farm application architecture

Dispatcher Servlet

Dispatcher-Servlet serves as the basis for the Spring Web model-view-controller framework that handles all the HTTP requests and responses (Figure 6).

When an HTTP request is received, the Dispatcher Servlet consults the *HandlerMapping* to call the appropriate Controller.

The Controller takes the request and calls the appropriate service method i.e. GET or POST method. Controller takes care of applying the business logic and then returns the view name to the *DispatcherServlet*.

The DispatcherServlet then asks *ViewResolver* to pickup the defined view for the Request.

As the last step the DispatcherServlet passes the model data to the view, which is finally rendered, on the browser.

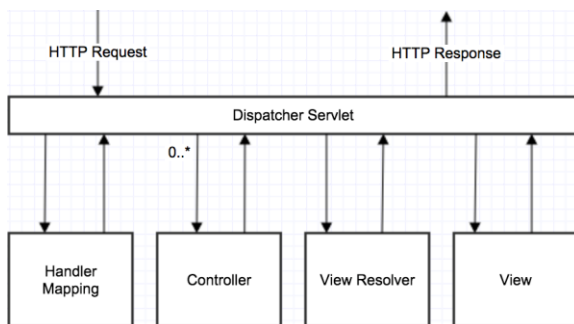


Figure 8: Smart farm application dispatcher servlet

RESTful Architecture

RESTful architecture requests and responses are built around the transfer of representations of resources. The representation of a resource is typically a document (XML or JSON) that captures the current or inherited state of a resource. REST recognizes everything as a resource (map object, user etc.) and each resource implements a standard uniform interface, resources have name and addresses (URIs), each resource has one or more representation (like JSON or XML) and resource representations move across the network over HTTP. RESTful API separates user interface concerns from data storage concerns. RESTful services are lightweight and invoking them just needs a browser or a simple HTTP stack. Another major benefit of RESTful API is the flexibility for data representation i.e. you could serialize the data as either XML or JSON format. With REST, I was able to dynamically build unique URLs to represent

remote Smart-Farm records objects as needed. The web application sends HTTP to obtain a JavaScript Object Notation (JSON) representation of data appropriate for populating the map types. JSON was chosen because, as a compact data format, it offers better performance compared to the complexity of an XML representation. With REST, the identification of the resources is relatively straightforward, in our case the JSON structure and URI requests are dictated by the kind of map that needs populated. There is no need for the client to create complex request envelopes to query the server. The initial features of the web client are to access the data in a read-only mode; no data is created or modified. As a result, all the operations use GET HTTP methods, which requests data from a specific resource. All request and response headers have a content type “application/json,” which means that complex queries and responses are in the form of JSON arrays or JSON objects.

View Layer

In the implementation of the Smart Farm tool, open source JavaScript library jQuery and other tools like Google Maps and Google Charts are used to develop a clean user-interface which can be accessed through most common browsers, for example, Internet Explorer, Firefox, Safari, Opera and Chrome, without installing any additional software or plug-in. The users do not need to have any specialized computer skills or GIS experiences. We have used JQuery over existing code browsing techniques due to the following three advantages. First, because it is a rich logic programming language performing a wide variety of searches as it is the basis of all queries. Second, because JQuery can inherently support multiple types of browsing, decreasing the complexity included to switch between tools to complete a single task as all the tasks can be performed within the same tool. Lastly, it is extensible as one can edit a configuration file to add own queries to the JQuery menus. Google Maps API is based on a set of classes within a

JavaScript container and these classes are loaded each time we open a Google Map web page. The interface of the API hides some powerful classes, objects and interfaces that enable us to manipulate Google Map. We have combined the interface with the data that we want to display and allowing user to leverage the interactive elements on the web without reloading the page or redisplaying the portions of the map, the entire process is handled by JavaScript library and Google Maps Application. Google Maps applications usually start with a simple map. To this map a number of different elements designed have been added to provide application functionality. In this work the following elements have been used: Controls – basic interface controls that enable user to zoom in and out the map, move the map effectively, and ability to change between map types. The KeyDragZoom library has been implemented which adds a drag zoom capability to the Google Map. I have also used the Google Chart API that allowed me to create the scatter chart dynamically when the user selects an area on the map.

Data Layer

The dataset for the Smart Farm application is relatively straightforward with just one table that contains information about the latitude, longitude, observation time and runoff data. The Spring JDBC Framework has been used to interface with the MySQL database. It provided the framework to all low-level details, like open/close a connection, prepare and execute SQL statements, process exceptions and handle transactions. Thus, the only thing we defined is the connection parameters and specifies the SQL statement to be executed. Spring JDBC provides several approaches and different classes to form the basis for a JDBC database access.

Demonstration of Functionality

Area of Interest

Area of interest on Google Maps. We decided to mark an Area of interest on the Map that helps the user identify the section on the Map that has data underlying that section in SQL tables

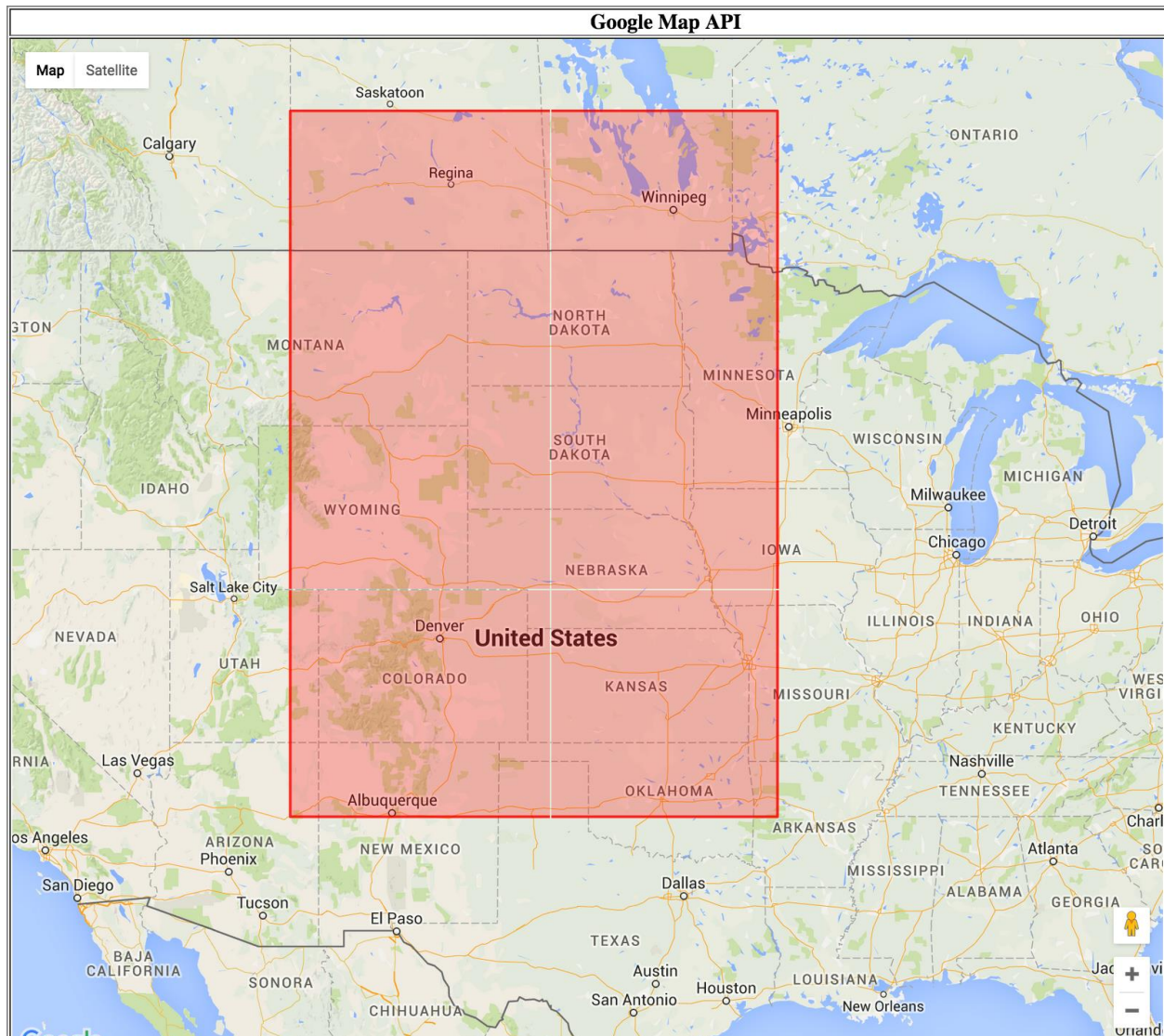


Figure 9: Looking at the Area of Interest

Marking a Selection

Marking a selection on the Map that drops Markers in the area under consideration.

Using Key drag zoom API we enable the user to make a selection on the Map by dragging mouse pointer to make a selection. If the area selected on the Map has corresponding data in underlying tables the system will drop Markers on the Map that further helps the users understand what points are available for further examination.

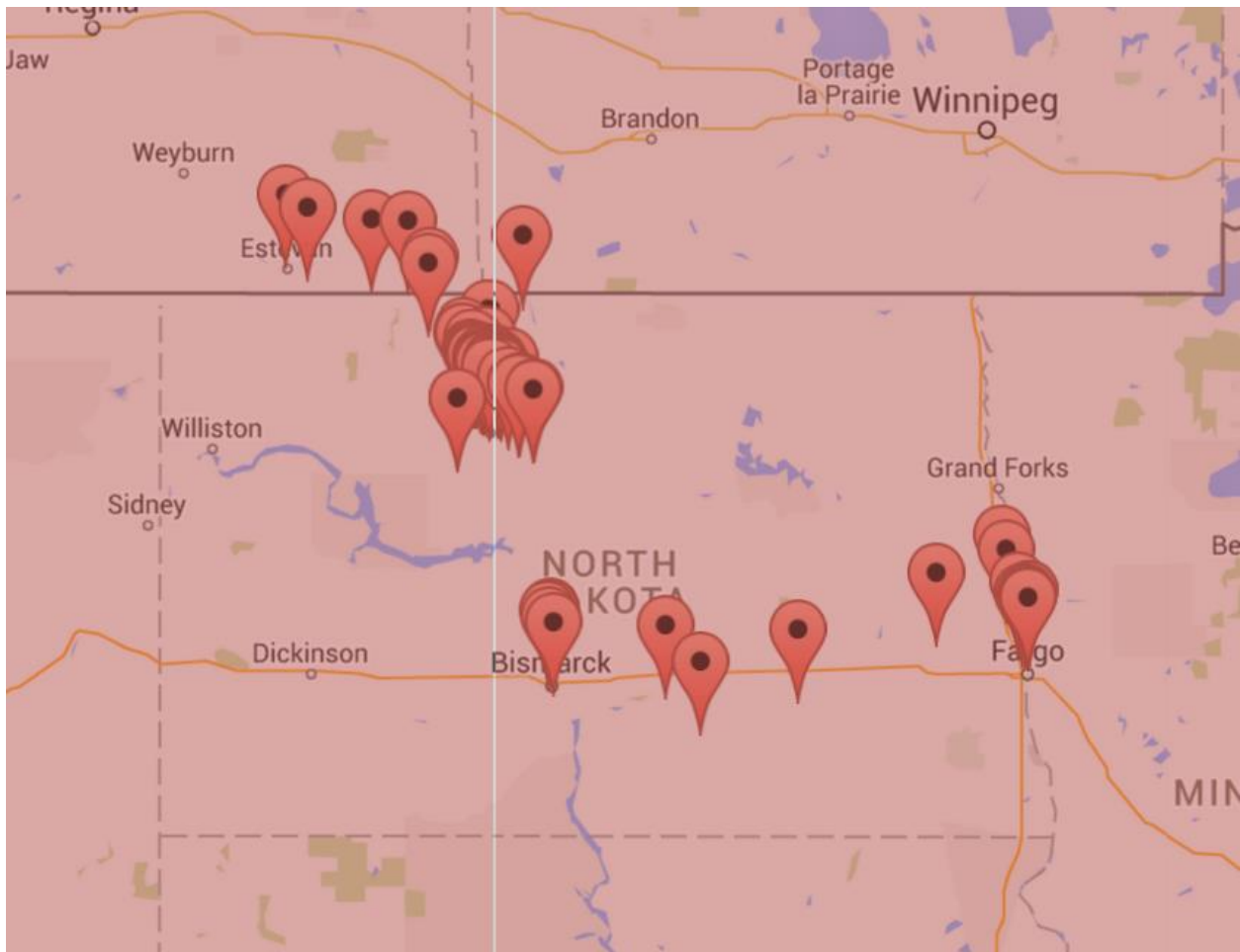


Figure 10: Marking a Selection

Selecting a Marker

Selecting a Marker of interest will trigger an event in the system and pass the selected coordinates to the backend RESTful service and in return the service will populate the graph for the selected Marker.

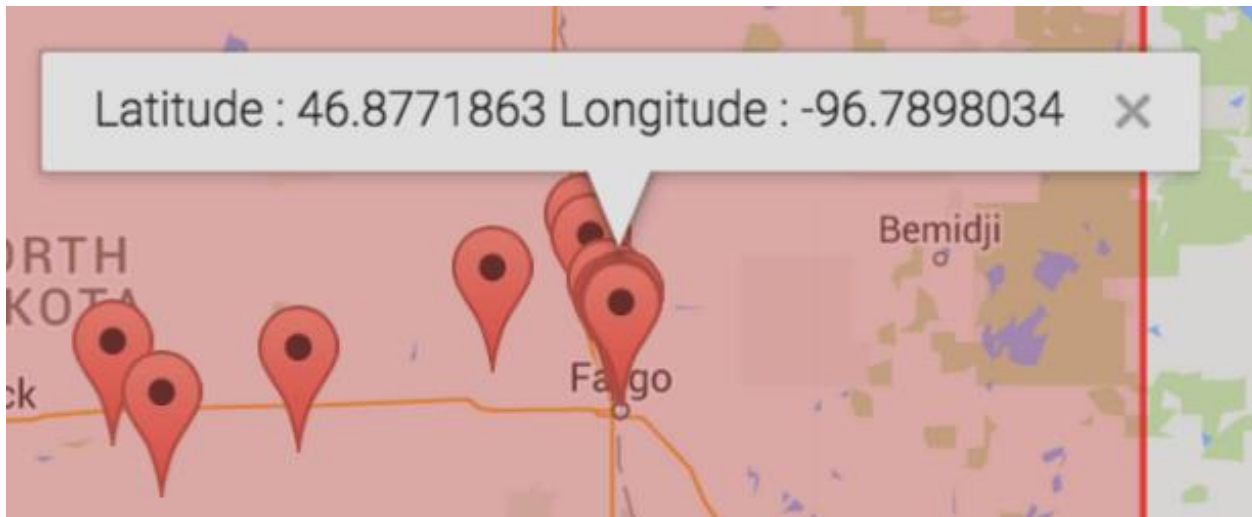


Figure 11: Selecting a Marker

Analyzing Charts

Analyzing charts after Marker selection can help us identify the underlying patterns which gets populated on the graphs that get created.

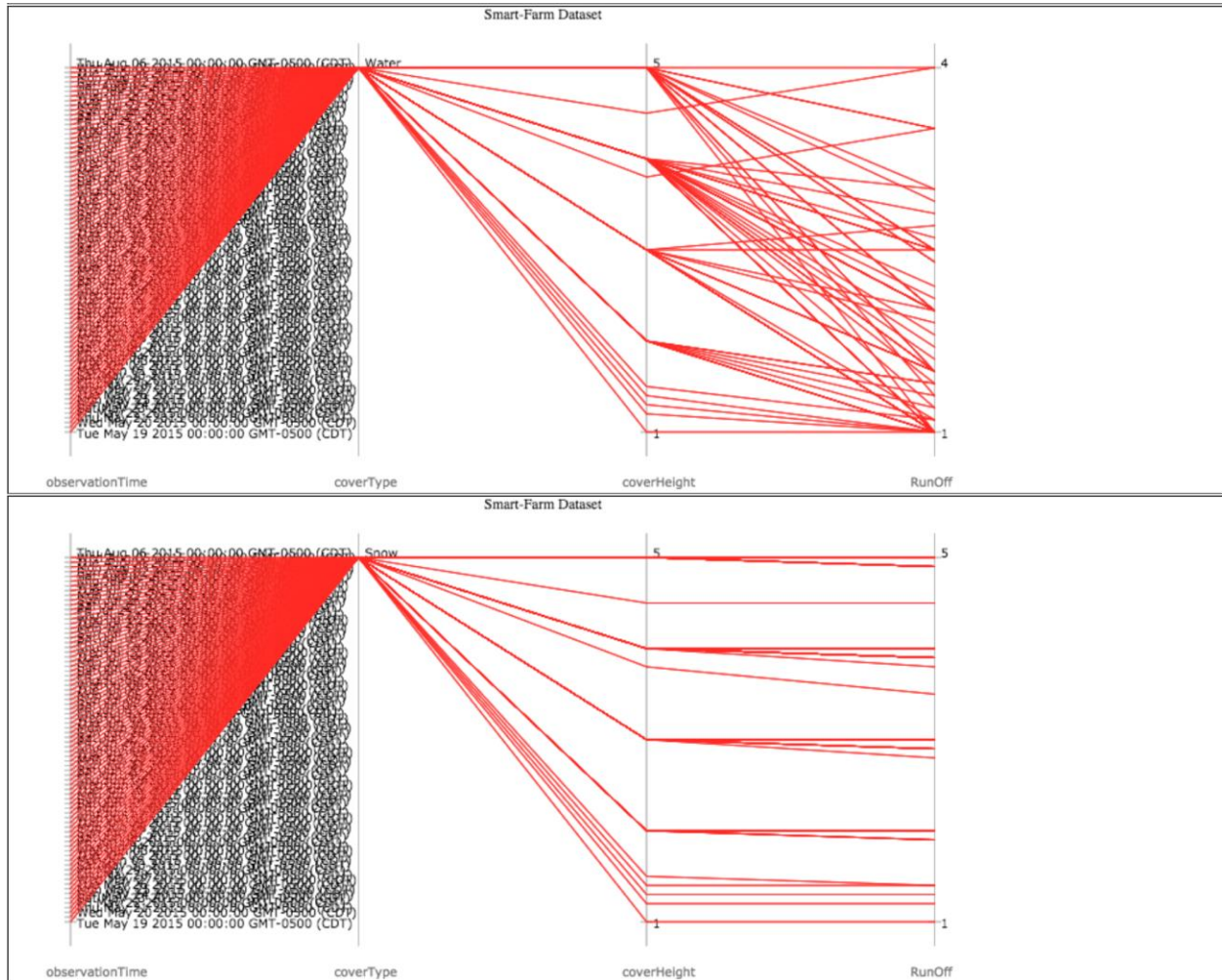


Figure 12: Analyzing Charts

Summary and Future Work

This paper explored creating the basic framework and prototyping a tool that can be used to visualize data in a more meaningful way. The tool architecture allows for it to be open for extension and easy integration of new maps for future work. As the dataset increases in size better and more useful data patterns can be identified and visualized on the UI. After exploring the Maps and visualization on them there are some recommendation for future work on this tool. First extension to the tool could be to integrate layers on maps that help user visually identify the areas on map corresponding to the location on the map. There are additional enhancements that can be made for improving the usability of the tool by giving user the ability to select and choose the map that he/she wants to be populated.

RELATED WORK

Some of the related work is also discussed in this section, where the basic functionality of the application is to visualize data with geographic coordinates with users providing the ability to select area on map. Although they are not visualizing the agricultural data but the approach for visualization is quite similar. In [9], the authors introduced the concept of geo-referencing, which means highlighting an area on the Google Map and the application would select pictures from database that correspond to that area. This is related to our application as it allows user to select area on the Google Map and then generates results according to the area selected. The VIEW-IT [10] project is a more sophisticated tool used to make decisions related to land use/land cover. The main aim of the project was to design a system that would integrate geospatial information to offer opportunities for remote sensing science. This application provides the user with an option to enter information related to the data to be visualized and then it queries the database and generates visualization for that. The [13] GLIMS project is a tool to acquire satellite multispectral images of the world's glaciers and analyze them for glacier extent and changes, and to understand these change data in terms of climatic and other forcing. This application is very big as they are estimation around 160000 glaciers and has a variety of tools for the user interface. The web interface has a tool to provide an overlay of glaciers and on the left there are number of options to select the data based on what the user needs. Our tool is different in a way that users can provide an input through map and results like scatter plots are generated dynamically. Another very interesting project is [11]. In this the authors talk about transforming information from any basic or application maps of a computer controlled system into two or three dimensional representation that is projected using an external gear known as heads-up display. In [12] the seamless nature of the map interfaces (i.e. Google Earth, Google

Maps and Ka-Map) fosters interactive exploration — something that is typically difficult with conventional paper surveys and especially useful in a teaching/learning environment. The graphical summaries in this project would be helpful to non-specialists (consultants or engineers) just as in [13], who need help with getting started with soil survey data.

REFERENCES

- [1] W. R. Garner and G. L. Felfoldy, "Integrality of stimulus dimensions in various types of information processing," *Cognitive Psychology*, (1970, March 1), pp. 225-241.
- [2] R. Groot. and J.D. McLaughlin, eds. *Geospatial data infrastructure: concepts, cases, and good practice*. Oxford: Oxford University Press, 2000.
- [3] C.L. Richard, and P. Sorensen. "Integrating normative location models into GIS: problems and prospects with thep-median model." *Spatial analysis: modelling in a GIS environment* (1996, December): 179-190.
- [4] Ahern, Shane, et al. "World explorer: visualizing aggregate data from unstructured text in geo-referenced collections." *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*. ACM, (2007).
- [5] M.L. Clark, and T. M. Aide. "Virtual interpretation of Earth Web-Interface Tool (VIEW-IT) for collecting land-use/land-cover reference data. "Remote Sensing 3.3 (2011): pp. 601-620.
- [6] R.J. Monson "Animated map display method for computer-controlled agricultural product application equipment." U.S. Patent No. 5,751,576. (1998, May 12).
- [7] D. E. Beaudette, and A. T. O'Geen. "Soil-Web: an online soil survey for California, Arizona, and Nevada." *Computers & Geosciences* (2009): pp. 2119-2128.
- [8] Raup, Bruce, et al. "The GLIMS geospatial glacier database: a new tool for studying glacier change." *Global and Planetary Change* (2007): pp. 101-110.
- [9] J.P.Wilson, H.Mitasova & D.J.Wright, 2000 Water resources application of geographic information systems. *URISA J.* 12 (2), pp. 61–79.
- [10] C. C. Rewerts, *ANSWERS on GRASS: Integration of a Watershed Simulation with a Geographic Information System (Nonpoint Source Pollution)* PhD Thesis, Purdue University (1992).
- [11] F. Olivera & D. Maidment *Geographic Information Systems (GIS)-based spatially distributed model for runoff routing.* (1999), pp. 1155–1164.
- [12] S. Pandey, R. Gunn, K.J. Lim, B. Engel & J. Harbor *Developing a web-enabled tool to assess long-term hydrologic impacts of land-use change: information technology issues and a case study.* (2000) *URISA J.* 12 (4), 5–17.
- [13] K. J. Lim & B. A. Engel *Evolution and evaluation of nutrient enabled NAPRA WWW decision support system.* (1999).