12-2012

Integrating Expert System and Geographic Information System for Spatial Decision Making

Sriharsha Shesham
Western Kentucky University, sriharsha.shesham120@topper.wku.edu

INTEGRATING EXPERT SYSTEM AND GEOGRAPHIC INFORMATION SYSTEM FOR SPATIAL DECISION MAKING

A Thesis
Presented to
The Faculty of the Department of Computer Science
Western Kentucky University
Bowling Green, Kentucky

In Partial Fulfillment
Of the Requirements for the Degree
Master of Science

By
Sriharsha Shesham

December 2012

INTEGRATING EXPERT SYSTEM AND GEOGRAPHIC INFORMATION SYSTEM
FOR SPATIAL DECISION MAKING

Date Recommended 12/7/2012

Dr. Guangming Xing, Director of Thesis

Dr. Andrew Ernest

Dr. Huanjing Wang

Dean, Graduate Studies and Research 12/12/12
Date

ACKNOWLEDGMENTS

I thank my advisor Dr. Guangming Xing for helping me throughout the preparation and completion of the thesis. You have taught me many technical approaches to various problems. I thank you for your long lasting support. You are always my favorite professor.

I wish to thank Dr. Andrew Ernest for providing me a job, which gave me an opportunity to do this thesis. Thank you for helping me in gathering the information related to my thesis and guiding me in the right direction.

I would also like to thank Dr. Huanjing Wang for being a part of my advisor committee. Thank you for pointing me in the right direction at the final stage.

I thank my parents Mr. S. NarasimhaCharya and Mrs. S. Sridevi who gave me mental support for completing this thesis. Mom and Dad you’re the reason behind my success. I also wanted to thank all my well-wishers.

TABLE OF CONTENTS

LIST OF FIGURES

INTEGRATING EXPERT SYSTEM AND GEOGRAPHIC INFORMATION SYSTEM FOR SPATIAL DECISION MAKING

Sriharsha Shesham December 2012 55 Pages

Directed by: Dr. Guangming Xing, Dr. Andrew Ernest, and Huanjing Wang

Department of Computer Science Western Kentucky University

Spatial decision making is a process of providing an effective solution for a problem that encompasses semi-structured spatial data. It is a challenging task which involves various factors to consider. For example, in order to build a new industry, an appropriate site must be selected for which several factors have to be taken into consideration. Some of the factors, which can affect the decision in this particular case, are air pollution, noise pollution, and distance from living areas, which makes the decision difficult. The geographic information systems (GIS) and the expert systems (ES) have many advantages in solving problems in their prospective areas. Integrating these two systems will benefit in solving spatial decision making problems. In the past, many researchers have proposed integrating systems which extracts the data from the GIS and saves it in the database for decision making. Most of the frameworks which have been developed were system dependent and are not properly structured. So it is difficult to search the data. This thesis proposes a framework which extracts the GIS data and processes it with the help of ES decision making capabilities to solve the spatial decision making problem. This framework is named GeoFilter. This research classifies various types of mechanisms that can be used to integrate these two systems.

Chapter 1: Introduction

The choice to be made among multiple alternatives is a decision. Spatial decisions are made in our day to day life such as figuring out the shortest path to travel from one place to another, finding the nearest restaurants, finding a site for a new restaurant, etc. Spatial data, which is also called geo-spatial data or geographic information, represents the data that is mined over the years. Geographic information plays a major role in spatial decision making [**1**]. Organizations use the geographic information for various purposes which aid the experts to make spatial related decisions. Sometimes a decision is to be made based on the previous results, based on the information gathered from various sources, and based on decisions made by various experts. To process the information technically a system is to be developed. It must have the capability to hold and process the geographic information. And it must be able to make efficient decisions.

Problem Statement

The assessment of surface water availability is measuring the amount of water which can be used from surface water sources like rivers, lakes, etc. For this assessment pinpointing the locations of surface water availability is important. This thesis involves developing an integrated system which filters the GIS data from maps using ES rules and then displays the locations.

Limitations in GIS:

GIS can provide the functionalities like handling and querying the geographic data, but it has the limitation to support decision making in the problem solving process.

The data is semi-structured i.e., the order for the attributes is not a major factor, not all the attributes are required, and the type of attributes of the same group may be different, etc. Therefore handling the insertion of GIS (semi-structured) data into the database is a cumbersome process. A detailed description of the types of data (structured, semi-structured and unstructured) is clearly explained in Chapter 2.

Limitations in ES:

Although it has the capability of problem solving using the heuristic approach, it also has the limitation to collect and visualize the spatial data. (The heuristic approach is a powerful and general approach, but there is no guarantee that it provides the best solutions).

To minimize these limitations in case of GIS and ES in the context of spatial decision making, a web based framework integrating GIS and ES is proposed based on the following objectives: The framework to be developed must provide us with the cross platform support i.e.; it should world perfectly fine on operating systems. An approach is to be followed which helps in retrieving the data faster.

Different mechanisms and the various factors to be considered while developing an integrated framework will be focused in our research. A different mechanism is also proposed in which the GIS data is extracted and used for processing with reduced search time.

The framework that is developed has been built by making use of the following tools: Mapserver - a geographic information system, JESS - an expert system and Ogr2ogr tool - to handle the shape file conversion.

The Mapserver uses different types of data sources. The default format used is ‘Esri shape file’, a popular vector data format to display the various layers on the map. A detailed description of shape file is discussed in chapter 2.

Thesis structure

The remainder of the thesis is organized as follows. Chapter 2 presents an introduction to GIS and ES. It reviews the types of data (structured, semi-structured and unstructured), the shape and GML files, and the process of forward and backward chaining in expert systems. Chapter 3 gives a literature review on different systems that were previously developed and their drawbacks. Some of the different mechanisms in integrating an expert system and geographic information system are defined in Chapter 4. The architecture and the implementation process of target framework are discussed in Chapter 5. Chapter 6 discusses the results obtained by the framework developed. Finally, conclusions of this thesis are given in Chapter 7 and possible future work has been presented in Chapter 8.

Chapter 2: Background

This chapter is divided into 5 sections. Section 2.1 is an introduction to GIS. Section 2.2 discusses types of data (structured, semi-structured and unstructured). Section 2.3 introduces the expert systems. Section 2.4 focuses on the shape and GML files and Section 2.5 explains the process of forward and backward chaining in expert systems.

2.1 Geographical Information System (GIS)

Any kind of information that we experience in this real world can be represented and put together in the form of a map as it would be definitely assigned to some kind of place [**2**]. This data can be used to find places of interest, cities and other geographic information. GIS can hold large amounts of organizational data which is useful to communicate with the organization members by visualizing the different situations of that area [**3**]. With the improvements made in the digital maps and their representations in the form of mathematical calculations of the coordinates, these evolved into the most popular way to view the world of resources. These digital maps are stored in the data set of a GIS in various formats to process simple queries.

The language of GIS can be stated as the language that contains the alphabets: lines, shapes, shades and words [**2**]. A map contains symbols which are represented with a distinct set of shapes such as squares, circles, polygons, etc. Each of these symbols mean something related to the information in the map. The following figure (Figure 1: A simple Geographic Information System) is an example presenting how the data is represented on a Geographic Information System in the form of a map.

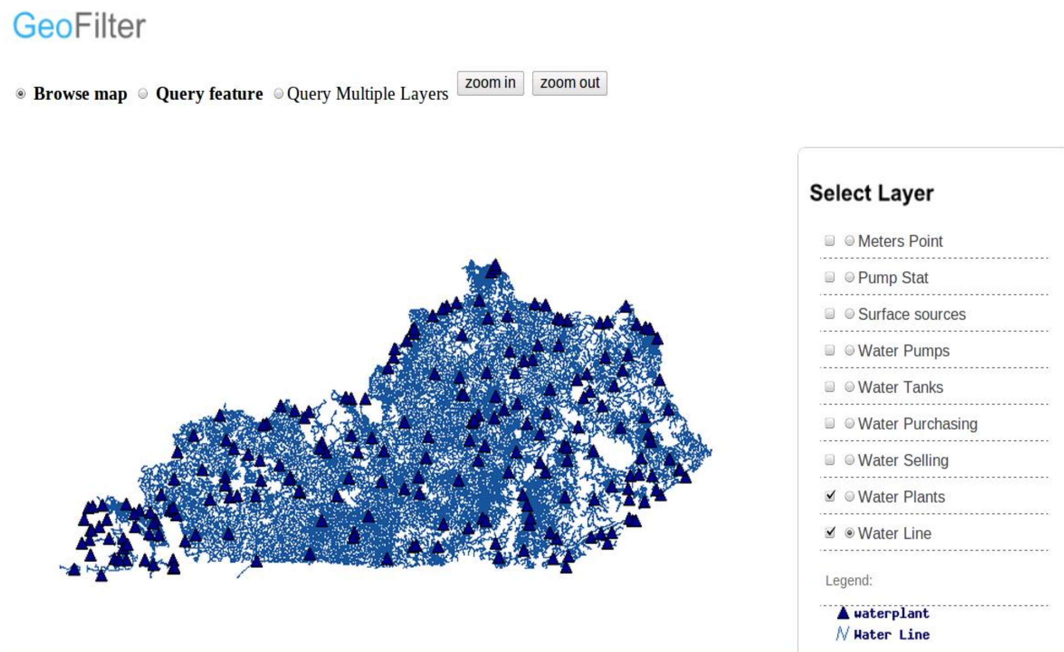

Figure 1: A simple Geographic Information System

Each point on the map represents some kind of real world data. For example, in Figure 1 each blue polygon represents a waterplant, which holds the data of the coordinates, name of the waterplant, location, etc. The architecture of a GIS is shown in Figure 2. A typical GIS has the ability to capture, store, query, analyze and display the spatial data.

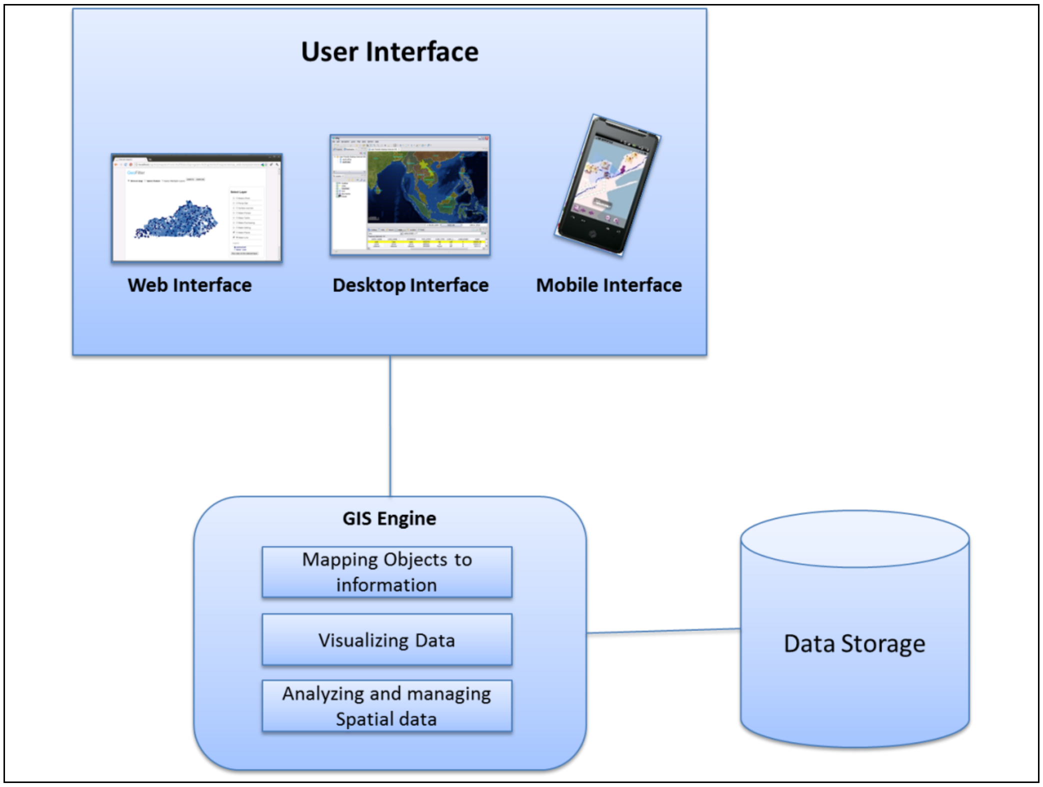

Figure 2: Architecture of Geographic Information System

Simple GIS components are the User interface, the Data storage and the GIS Engine

The user interface of a GIS acts as a communication medium between the user and the GIS engine, which functions based on the user request and response related to the GIS server. The user interface of the GIS can be designed for three different mediums: the web, the desktop and the mobile, based on the requirement.

GIS stores large volumes of spatial data that is changing from time to time. Data storing, retrieving and visualizing are the critical tasks of a GIS. A digital map can store most amount of information. To use a map it is important to understand the type of information a map can hold. The following are the types of information that is stored on the digital maps [3]:

Geographic Information constitutes data related to the position and shape of the map file. Here the information is stored in the format of coordinates. Most of the GIS have the following main geometric objects:

Point: A dot represented with the coordinate axis. Examples: coffee shop, book store, school.

Line: Collection of points that are connected. Examples: road, river, boundary.

Area: Collection of lines that are connected in a circuit. Examples: lake, country, state.

Attribute Information represents the information of the features that are included in the map content. Each of these features is associated with a map object. This is dependent on the GISs that are being used. The updatability of mapping the objects and the data varies from GIS to GIS. The data has a feature member ID which acts as a key. This feature member ID is used to retrieve all the attribute information bounded to it. This is explained in detail in chapter 5.

Display Information is the graphical information which holds the details of the colors and ways of representing an object on the map. The quality of the display depends on the user's requirements in using the GIS. If the user uses the GIS to display the map contents only, then the display information plays a crucial role.

A GIS engine manipulates, analyses the spatial data and provides the visualization capabilities to the user [**3**] [**4**]. The querying process of GIS is highly based on the structure of the data that is stored. The GIS engine has the capability of analyzing the data and retrieving the requested details and presenting them to the user. The GIS Engine handles the mapping of the objects with the corresponding data.

The two basic data models:

Vector data model: In this model the data is categorized as well defined elements, also called features. These features are bounded to the objects with lines, areas or points [**5**]. This model is useful to represent the features such as roads, pipelines, and points of interest.

Raster data model: In this model the data is categorized as equally divided cells or grids of a surface. Each cell contains a value bounded to a category, an interpreted value or a measurement. This model is useful to represent continuous data across a surface [**5**] [**6**].

2.2 System independent open source GIS systems

This section elaborates on the system independent open source GIS and the features that are available.

PostGIS is an open source GIS system that provides support to PostgreSQL to process geographic information [**7**]. It acts as a back-end spatial database for geographic application. It was developed by Refractions Research [**7**] and has several functionalities such as coordinate transformation, data validation and programming API. As per the survey made on geospatial technologies for web mapping [**8**], developers are more comfortable with PostGIS than MySQL.

GeoServer is a popular open source GIS project [**8**]that is used as an administration tool to configure spatial and non-spatial databases. It is built on Java technology and has a built in support for OpenLayers and Google Earth. It reads multiple data formats such as PostGIS, DB2, MySQL and Shapefiles.

This is one of the widely used open source GIS application which is applied for developing interactive web mapping application [**9**]. Mapserver is a common gateway interface (CGI) application which can be run on both Apache and Microsoft web servers. It supports various scripting languages such as PHP, Java and .Net. It also supports direct database connection to Oracle, DB2, PostGIS, etc [**8**]. Mapserver requires a map file to configure, add vector and raster layers in order to provide the main functionality. One of the main advantages of Mapserver is its map file simplicity and the map file ability to connect directly for many databases [**7**]. Considering the advantages of Mapserver and its support for PHP, in this contention Mapserver is used as the GIS.

2.3 Types of data (structured, semi-structured, unstructured)

Structured data is systematically arranged in the form of entities. The groups are formed based on the type of entities. In this the groups are formed based on the type of entity. Each group of an entity has the same format. The length and order of the entities in a group are same in the structured data. Database can be considered as an example for structured data.

Semi-structured data is arranged systematically to some extent. In semi-structured data the groups are formed based on the type of entities. In this the attributes of entities belonging to the same group may vary. In this type of data the existence of all the attributes is not a mandatory requirement. The order, size and type are not necessarily considered in semi-structured data.

Example: XML document

```
<bibliography>
        <book>    <title> Information… </title>
                  <author> Peter </author>
        </book>
        <book>    <title> Security in… </title>
                  <author> John </author>
                  <author> Evan </author>
        </book>
</bibliography>
```

The example above demonstrates the semi-structured data, where the primary <book> entity has two attributes <title> and <author>, but the secondary <book> entity has three attributes <title>, <author> and <author>.

Unstructured data is inconsistent. This type of data does not follow any particular format or structure. It does not follow any rubrics and hence the data is asymmetric. Examples: text, video, image, sound, etc.

2.4 Shape and GML files

A shape file is a popular vector data format in GIS software. A shape file stores geospatial data types such as points, lines and areas. It is a semi-structured file i.e., it stores data of entity attributes which may vary in their type of representation, information and structure. A shape file is associated with two other files: shape index formatted file

(.shx) and attribute formatted file (.dbf) [**10**]. GML stands for “geography markup language”. It is a text based markup language which stores the vector and attribute data. It is an xml representation of a shape file and its associates. The majority of the GIS provide the ability to display a GML file.

Advantages of GML:

GML consumes less system memory compared to a shape file. GML files are easy to parse, query and transform. The query process is similar to that of XML files. The data in GML is associated with content of the system such as attributes, geometry and where features are located. Any information related to how a map is to be displayed is not provided which makes the system customizable with user defined styles. GML is easily updatable and data can easily be modified or added for example a URL address can easily be associated to a point [**11**].

2.5 Expert system

Expert system is one of the branches of artificial intelligence which serves as a decision making system with human expertise. These particularly help in solving the complex problems without the help of a human brain [**12**]. In the past 3 decades, there has been rapid development in the field of expert systems. Expert systems are used in many fields such as healthcare, government sectors, educational services and survey tools. Expert systems are special computer programs which are different from the routine programming and have the capability of human reasoning, logical thinking and rules [**13**]

[**14**]. The knowledge base acts as the main source of an expert system. The information in an ES may be in the form of procedures, data related to a domain, or problem solving rules. The knowledge of an expert system can be represented in three main types: semantic networks, frames and if-then rules.

2.6 Types of expert systems

In this section, the types of expert systems available and their features are discussed.

A frame-based expert system follow object oriented programming structure for analysis and implementation [**16**] [**17**]. A frame is a data structure which is associated with knowledge about a particular object. Each frame has attributes and name associated with it [**15**] [**16**]. The main drawback of this system is that it cannot identify the difference between the essential properties and the accidental properties [**17**].

A model based expert system is a system which uses structure and function of a system as knowledge. In this type of expert system the system model and its predicted outcome are defined with which the actual output is compared and accuracy is measured [**15**]. If the problem is more general and knowledge is more explicit the system creates longer inference chains. Hence, the systems developed are slower and complicated [**17**].

Case based expert systems store previous results of a similar or same process (old process). When a case matching the previous process is detected the solution of that process is assigned to the new process [**15**]. This type of system has complex symbolic and numerical representations [**18**]. Also, these systems are used where the problem conditions frequently changes and the problem is not clearly understood [**18**].

Rule based expert systems are mostly used industrial artificial intelligent systems with simple rules [**19**] [**20**]. The rules implemented in this system can be easily understood by an expert or a programmer [**15**]. The rules can be easily updated. Once the expert decisions are captured in the system in the form of rules they are saved and can be used for a long time [**19**]. The main difference between conventional programming and rule based expert system is that, the order of execution of conditions or rules is not important.

Considering all the advantages and disadvantages of the above discussed expert systems, in this proposal, a rule based expert system is used to process the information.

A simple demonstration of a rule based expert system can be given considering an example of a robot while driving a car.

The rules for this process would be as shown in Figure 3:

- ✓ **If** the engine is off **then** turn on the engine.
- ✓ **If** the mode is parking **then** switch to Drive mode.
- ✓ **If** the traffic signal is Red **then** stop the car.
- ✓ **If** the traffic signal is Green **then** start driving.

Figure 3: Rules for Robot driving a car

The facts for this process would be:

- ✓ (engine is-on)
- ✓ (mode is-parking)
- ✓ (signal is-Green)

Figure 4: Facts for Robot driving a car

An inference engine is associated with the set of rules and facts as defined above. And when the engine is set to run, it analyzes the set of facts and based on the set of facts, the inference engine tries to apply them on the corresponding rules and only then the rules get triggered. A simple architecture of a rule based expert system is shown in the following figure (Figure 5: Simple architecture of a Rule based Expert System):

The four important parts of a rule based expert system are the knowledge base, the inference engine, explanation facilities and the user interface.

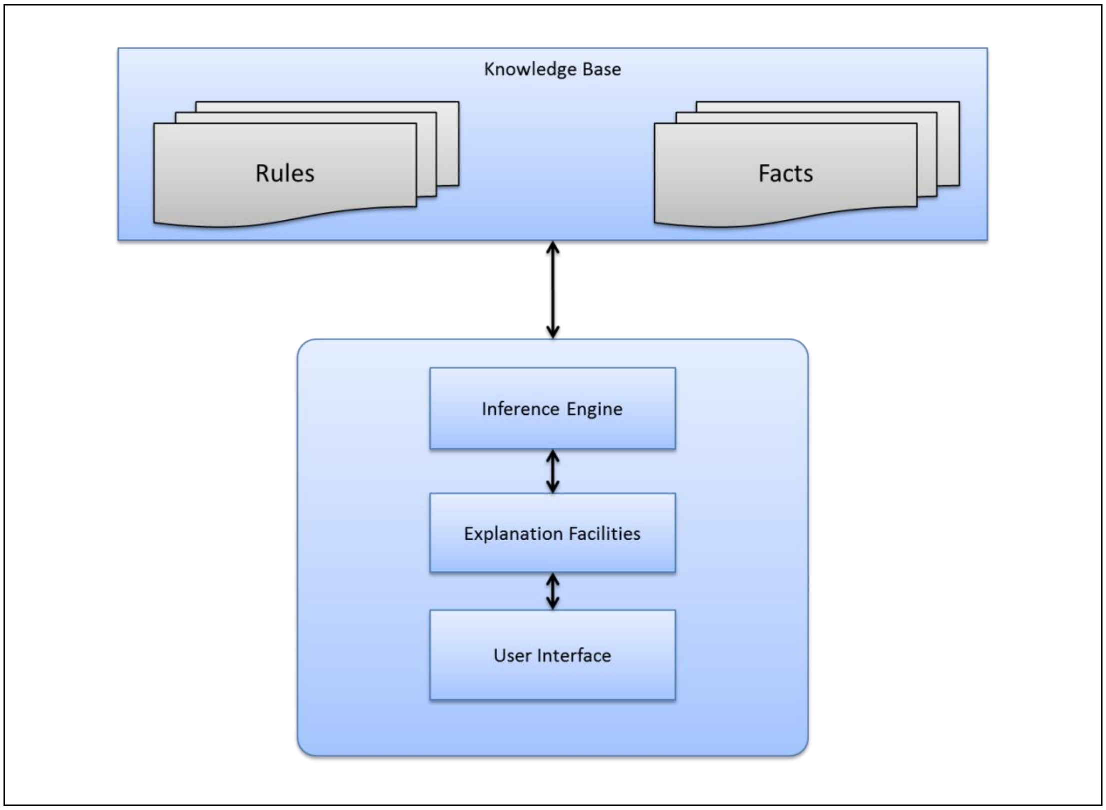

Figure 5: Simple architecture of a Rule based Expert System

The knowledge base of the system is comprised of facts and rules. The rules are defined by a relation, recommendation, strategy, etc. The structure is of the form IF (condition) - THEN (action). Multiple conditions can be joined to a rule using keywords AND and OR. The knowledge to ES is provided by the experts in the domain area [**21**]. The knowledge base of the ES is independent of all the components and hence it is easily updatable and new data can easily be added. The main reason that expert system is different from the conventional programming is its ability to determine the action based on the previous results disregarding the order of execution. Hence the knowledge base acts as the major role.

The inference engine provides the reasoning capability to the ES. It is almost similar to a search engine; the search engine allows the user to enter a keyword and retrieves the content matching the keyword from different sources, whereas the inference engine checks the knowledge base for the data that matches the query provided by the user. The user is provided with the results matching the query. There are two ways of defining the rules: forward and backward chaining. This is discussed in detail in the next section (section 2.4 forward and backward chaining).

Explanation facilities explain the approach followed in the process of making a decision. It provides answers to the user on why a certain conclusion is made and what are the facts selected for reasoning.

User interface of an expert system acts as the communication layer between the user and the ES. In most of the expert systems user interface is kept as simple as possible for easy adaptability [**21**]. Though the user interface plays a vital role, they are optional in case of an expert system, as the main functionalities are reasoning and decision making.

2.7 Java Expert System Shell (JESS) rule engine

JESS is a rule engine and scripting language which adds rules technology to java-based applications. It is a small and lightweight system and it is also one of the fastest rule engines [**22**]. It can easily be integrated with the programs written in java which makes the JESS a system independent expert system resembling the platform independence of java [**23**]. It supports both forward and backward chaining methods. It provides direct interface to java components [**24**]. JESS uses Rete algorithm for pattern matching i.e., when to fire which rules [**22**]. A Rete based expert system uses a set of nodes and each node indicates the pattern occurring based on the facts. When the facts or set of all facts are matched the pattern corresponding rule at the leaf node is triggered. Rete algorithm increases the speed of matching the pattern and triggering the rules. Hence, JESS is one of the fastest available rule engines.

As JESS is one of the fastest platform independent rule based expert systems available, in this framework JESS is used as an ES in the proposed framework. One of the main reasons for considering Mapserver and JESS is that they provide a better documentation and also provide many resources for a good start.

2.8 Forward and backward chaining in an Expert System

In this section the concepts involved in decision support feature of a rule based expert system are discussed. A rule based expert system has two types of reasoning capabilities: forward and backward chaining.

Forward chaining:

In the process of forward chaining the data is analyzed and the rules are fired to retrieve more data and the process is stopped once the final result is obtained [**25**]. In the process of forward chaining the process continues until the goal is reached. For example a spatial data of connected water pipelines is considered. If the goal is to find whether there is flow of water from Junction A to junction F, given that junction A and junction D, junction D and junction E, junction C and junction B, and junction B and junction F are connected. The water pipeline connections are shown in Figure 6. Given that the rule base contains the following rules:

1. **If** A and D, and D and E are connected **then** A and E are connected.
2. **If** A and C, and C and B are connected **then** A and B are connected.
3. **If** A and E, and E and F are connected **then** A and F are connected, and water flows from junction A to junction F.
4. **If** A and B, and B and F are connected **then** A and F are connected, and water flows from junction A to junction F.

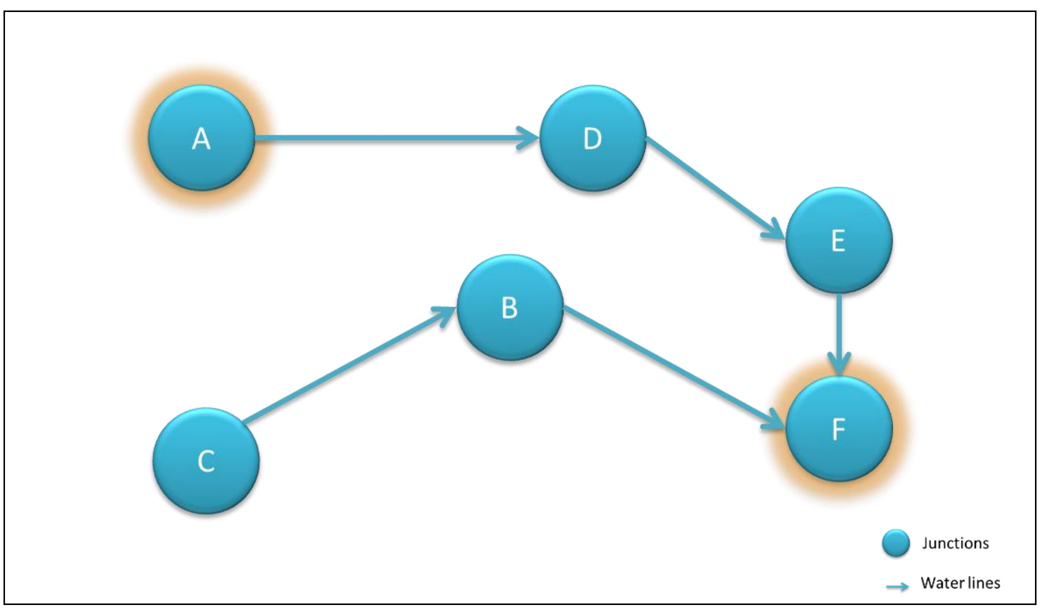

Figure 6: Connected water pipelines

From the above set of rules provided the approach of a rule engine is as follows:

It starts with the first rule; if the first rule is satisfied then the consequents are added to a temporary dataset. In the example above the first rule would be selected, as the facts state that junction A and junction D, and junction D and junction E are connected. Now the rule base starts searching and when it matches with the third condition and the consequents are added to the dataset. The consequents added to the dataset determine what rules are to be fired next. This approach is called data-driven approach [**25**].

Backward Chaining:

In the process of backward chaining the inference engine first does the processing of all the rules and checks the knowledge base for the facts matching these rules. Once the facts matching these rules are found then the rule engine goes up to the next step. This is a bottom up approach. For example if the above example of finding whether there is flow of water from Junction A to junction F is considered, the approach of the rule engine is as follows:

The rule base goes through all the available rules and adds the consequents 3 and 4 to the temporary dataset (as the goal AF is connected matches). Both the antecedents are added to the goals list and next the rule base searches again and the rules 1 and 2 are selected as the consequents matching new goals. But it is known that junction A and junction D are connected. It is concluded that the junctions A and F are connected and there is a flow of water from junction A to junction F.

Chapter 3: Literature Review

The complexity of spatial decision making problems depends on several factors such as data, accuracy, the number of conditions, priorities, etc. The different approaches on implementing a decision support system by integrating GIS and ES are summarized in this chapter. The process of integrating GIS and ES to solve the spatial decision making problems requires a broad literature review and hence this chapter is subdivided into different aspects.

GIS-Expert systems that are proposed for various purposes were analyzed during the study. While reviewing the formulated fertilization technology [**26**], it has been stated that most of the expert systems use the secondary development model of a GIS as the component of integration which helps in solving the decision making problems. A platform + plugin based framework was proposed, where a decision support system is integrated into a GIS as a plugin, which makes the system able to be used for multiple purposes. The framework solves the problem of scientifically managing and using the data that holds the valuable information about the soil fertility to service the agricultural fertilizer [**26**]. On the other hand while discussing the critical aspects involved in Highway Geometric Design (HGDesign), it has been stated that it is difficult to select an economical path by manual design, as a route with minimum cost involves careful design, planning and construction. A good proposed route must consider various factors such as detailed assessments of geographic features, the land used for construction and the evaluation of environment [**27**]. Hence ES-HGDesign is proposed where the GIS engine gets the relevant database data of the contour map and ES uses this data to select the optimal route for highway geometric design [**27**].

Various problems are involved in matching the address typed in by the user and plotting it on the map. While discussing the process of batch Geocoding it has been stated that for the purpose of address matching the recent GIS systems are bundled with an expert system [**28**]. Geocoding is a process of determining the location based on the textual address information provided. Correcting the semi-formatted location information manually is a difficult task when the data that needs to be parsed is in large amounts. A solution to the spatial problem of finding the addresses that are semi-formatted is proposed in which GIS and ES are integrated, where the expert system handles the unmatched data and formats the information based on the rules [**28**]. An example of how a user may enter the address to search in different formats is given to describe the problem.

It has been discussed how Ecotones are determined with the help of decision support ES and GIS. Ecotones are regions of landscape where elements, patterns and prevailing and functional processes exist [**29**]. The spatial decision problem encountered was to perform the analysis of stability in integrating options, choices and decisions by solving the problem of location or allocation as the GIS alone cannot provide the support for multiple decision making problems [**29**]. It has been stated that this evaluation of landscape from several perspectives can be achieved by integrating the expert system and GIS [**29**]. In this research the fuzzy theory was implemented, this approach is discussed in the later section of this chapter. The fuzzy approach of the Expert system is different from the classical set theory approach where the elements are sorted based on the conditions such as belonging to or not belonging to a set [**29**]. The fuzzy theory follows the membership logic the value of which lies in between zero and one [**29**]. The value one indicates the element fully included in the set and any value in between zero and one is a fuzzy member [**29**]. The fuzzy approach determines the ecotones, and GIS uses this information to plot on the map [**29**].

While discussing the solution to provide a viable development plan for local community it has been stated that it is difficult to make a decision in selecting the land to use after the mine closure as the decision involves various factors to consider [**30**]. It has been also stated that environment based decisions are difficult, multi-faceted and depends on the decisions made by different stakeholders with various priorities. An integrated system was developed to analyze the system by combining a geographic information system and a multi-criteria decision-making system [**30**]. The integrated system is used in selecting the land use in various parts of a post-mining area by considering the environmental, technical, safety and economic criteria [**30**]. This system was used to select the optimal landscape reclamation strategy of the mine of Amynteon lignite which is located at West Macedonia Lignite Centre, Northern Greece [**30**].

GeoExpert – a framework which assesses and cleans the data quality has been developed which focuses on correcting the erroneous spatial data by integrating analysis capabilities of a GIS engine and inference capability of an ES [**31**]. As the multi-dimensional spatial databases hold crucial and hidden information and as the data needs to be accurate and error free, GeoExpert is used as a cleansing tool to assess the data quality. The main problems which were found during the study were calibration errors, threshold errors, missing data, superfluous data, omitted records, garbling errors, replicas, etc [**31**]. The system was implemented using MVC architecture, to separate the application and design logic which allows the GeoExpert to work with any kind of spatial data irrespective of domain [**31**]. Two types of systems- standalone and web based were developed for the user convenience [**31**].

A rule based ES framework was developed to help small water system operators in solving the problems of managerial, technical and financial tasks [**32**]. This system helps the operators in getting informed about the various EPA rules that may be applied based on the decisions [**32**]. The author describes that it is difficult for small water systems to afford a knowledgeable expert for solving the managerial challenges and hence a system with an ES can help solve the problem [**32**].

A plugin based application framework was previously proposed based on .net platform which can be used to solve the conflicting problem between changing business requirements and the projects complexity [**33**]. An integrated system for live hazard monitoring is developed by integrating GIS and ES which is used in making decisions using sensor web technology [**34**]. As per Sugumaran and DeGroote [**35**], there are three types of decision making problems which are decisions related to resource allocation, decisions related to resource status and decisions related to policies.

The development of the GeoFilter framework has been initiated by using the information collected from the previous work that has been implemented for the Department of Water Resources, Western Kentucky University [**32**] [**31**]. The previous work, which has been implemented for the small water systems [**32**], was an expert system which can solve the problem of small water system operators by presenting the decisions based on the various Environment Protection Agency (EPA) rules that may apply to them. The framework was developed with the help of two expert systems

MANDRAX and JESS. Though the system was able to help in making expert decisions, it lacks in visualizing the important geographic data which may affect the decisions. On the other hand, following this work a GIS expert system was implemented by integrating the ArcGis server and JESS expert system solving the problem of quality inconsistency of data [**31**]. Both web based and standalone versions of the framework were implemented. The web framework was developed using the MVC architecture (this system falls under the section 4.4 Model View Controller (MVC) based mechanism from chapter 4). The main drawback of this system is that it is not an operating system independent architecture; it has been developed for a specific operating system. The data extraction process of this system involves the database operation on the semi-structured.

Chapter 4: Integrating Mechanisms

In the previous chapter, it's been shown in what way the integration helps in solving the spatial decision making problems. Based on the research, the integration of GIS and ES are categorized into several mechanisms. In this chapter some of the different mechanisms in integrating an expert system and geographic information system are defined.

Before planning to integrate the systems and before choosing a mechanism for the target system, the following factors should be collected. The purpose for integrating the systems, the type of framework is to be determined. Example: A framework with multiuser support or a single user support, a desktop application or a web based application and answers for the questions such as- is there a need for the program logic to be updated? Will the program be applicable in different decision making scenarios? are to be prepared. And the various rules that must be triggered for the process of decision making must be identified and the details of when to trigger which rule/rules must be noted.

Based on the collected information from the above factors the type of mechanism can be determined. The following are some of the types of mechanisms to integrate the GIS and ES.

4.1 A simple mechanism (Standalone)

A simple mechanism (standalone) can be followed if the purpose of integrating the system is to solve simple spatial decision making problems. This mechanism can be followed if the system is to be accessed within an organization. The program logic in this mechanism cannot easily be updated. The system developed by following this approach can be used only for a single decision making problem.

A simple mechanism integrates the standalone GIS system and an expert system to solve simple spatial decision making problems. A framework developed following this mechanism can be used in a small organization.

Advantages: Framework developed following this mechanism is simple to implement. It has limited features which reduces the user memory load.

Drawbacks: In this framework, updating the program logic is difficult. The framework can be used only for a specific functionality i.e., as a single purpose decision-making system. It cannot be used from outside the organization. The data cannot be shared across the branches of organization.

4.2 A simple mechanism (Web-based)

A simple mechanism (web-based) can be followed if the purpose of developing the system is to solve simple spatial decision making problems. A system developed by following this mechanism can be used throughout the organization networks. If an organization do not want to update the program logic frequently and if the system is used for a single purpose this mechanism can be followed.

A simple mechanism that is web-based is similar to that of the standalone framework which integrates the web-based GIS system and an expert system to solve simple spatial decision making problems. A framework developed following this mechanism can be used in a small organization with diverse branches.

Advantages: It can be used from outside the organization. The data can be shared between organization branches.

Drawbacks: Updating the program logic is difficult in this mechanism. The framework developed following this mechanism can be used only for a specific functionality i.e., as a single purpose decision-making system.

4.3 A Plugin based mechanism

A plugin based mechanism with “Platform + plugin” is a standalone technology [**26**] [**33**]. This mechanism can be followed if the system is to be accessed within an organization and if the program logic is not updatable and if the program is to be reused for different decision making processes.

A “Platform + plugin” mechanism integrates the standalone GIS system and an expert system to solve simple spatial decision making problems. A framework developed following this mechanism can be used in a small organization.

Advantages: The framework can be used for multiple purposes. As the system is plugin based, any number of or any type of decision support system can be added to the framework. The plugin based mechanism allows user to easily update the rules that the system has to be followed.

Drawbacks: Updating the program logic is difficult in the framework developed by following this mechanism. It cannot be used from outside the organization.

4.4 Model View Controller (MVC) based mechanism

MVC based mechanism is a mechanism with separate model, view and control [**31**]. A model is included with business logic that tells the system what to do. A view is for the purpose of visualization and a controller acts as the communication medium between the model and view. This mechanism can be followed if the system is to be accessed anywhere, if the program logic must be updatable, and if the program is to be reused for different decision making processes.

A "MVC" mechanism integrates the web-based GIS system and an expert system to solve simple spatial decision making problems. A framework developed following this mechanism can be used in any type of organization.

Advantages: The MVC approach allows separate updating of the two main parts of the system. The GIS can be considered as the view part and the ES can be considered as the model part. The framework can be used for multiple purposes. As the system is web based and as it follows the MVC architecture, the system is flexible in many aspects. The MVC based mechanism allows the user to easily update the rules of the system that have to be followed. As the program logic is independent of the other aspects it can easily be updated. As it is a web-based application the data can be accessed from anywhere irrespective of location.

Chapter 5: Architecture and Implementation

This chapter emphasizes the various steps followed while developing a framework named GeoFilter by integrating the ES and the GIS (the framework is implemented in a simple web based mechanism which is elaborated in the previous chapter). A different approach of extracting the semi-structured GIS data and storing it in a different format for a comparatively fast searching process is explained.

5.1 Architecture

The architecture of the system is shown in figure 7. This system demonstrates the web based mechanism of integrating the GIS and ES. The GeoFilter framework is implemented by following the forward chaining process of an expert system.

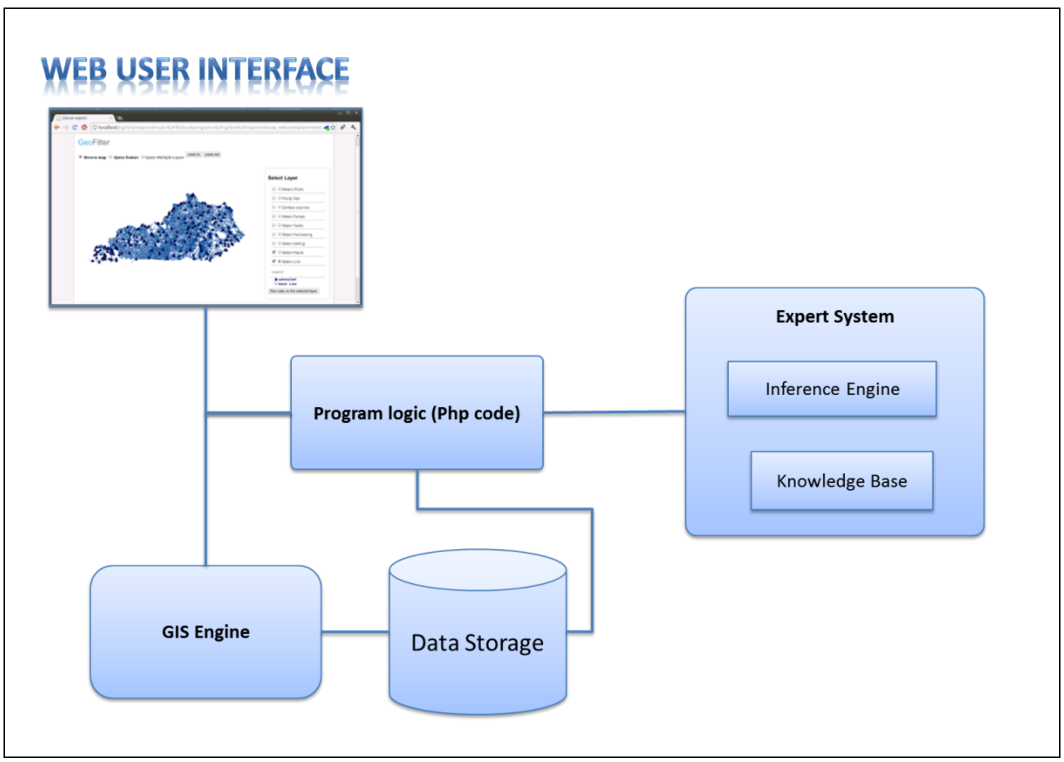

Figure 7: Architecture of GeoFilter Framework

When the user selects the layer in the web interface the GIS server retrieves the requested data from the data storage and presents it to the user. The data storage has the geographic information related to the domain. The knowledge base of the ES stores rules and facts of the system. The rules are defined based on the organization needs. When the user selects an active layer and clicks on the run rules button the program logic retrieves the selected layer details and generates facts. Based on the facts generated the corresponding rules are executed by the inference engine of the Expert system. The decision results of the system are posted back to the program logic which converts the data to new layer and adds to the GIS engine data.

5.2 Implementation

This framework is developed to demonstrate a spatial problem of filtering the geographic data, based on organizational rules (decisions). This framework implements the process of extracting and filtering a GIS layer based on fulfilling certain criteria that is processed by the ES and exhibiting the filtered output to the web interface. The framework uses Mapserver as the GIS, JESS as the ES and ogr2ogr tools for data extraction process. The user interface of the system is an html page with map. The framework also has options to zoom-in and zoom-out and query the map features and run the rules. The geospatial data of "Water Resource Information System of Kentucky State" is used, which has been gathered since 1988 [**36**].

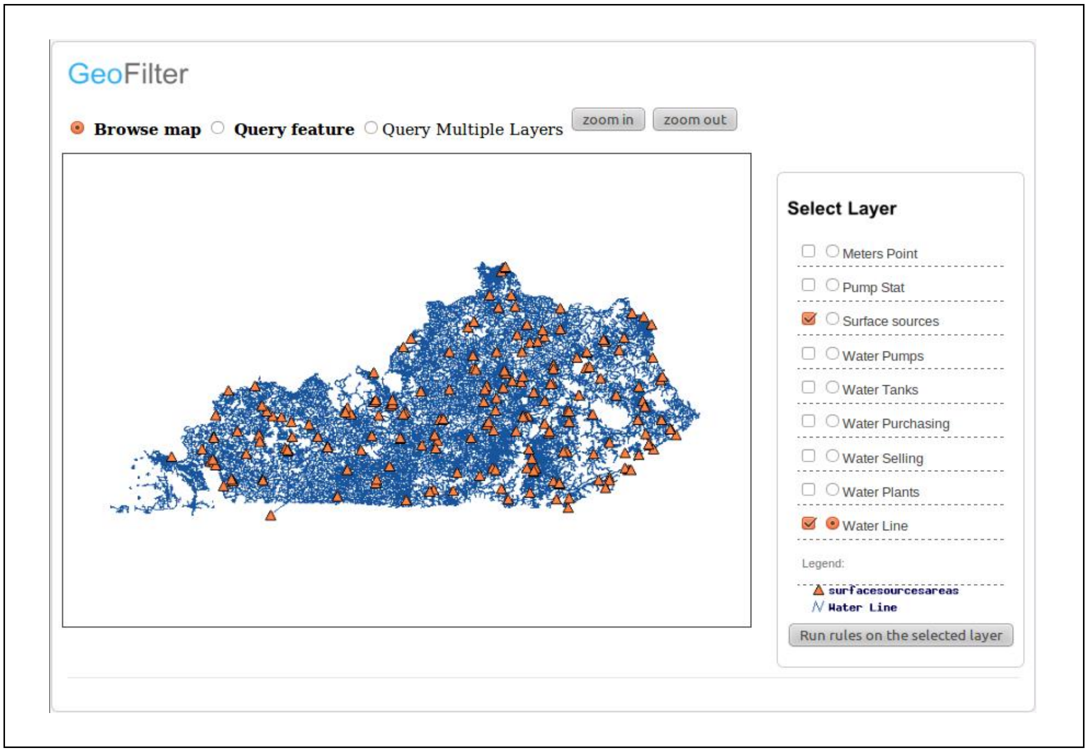

Figure 8: GeoFilter framework

Figure 8 shows the interface of a GeoFilter framework. The framework is provided with different layers with a checkbox and a radio button adjacent to them. The check boxes allow the user to view the layers by selecting them. Whenever a checkbox is selected a new request is sent to the Mapserver cgi. Every request sent to the Mapserver is treated as a new request which carries all the data that has to be displayed. Any number of layers can be selected at a particular instance. The radio buttons allow the user to select a layer. This layer is set as the active layer. Only one active layer can be selected at a time which acts as the layer on which the rules are applied and the query is processed. The waterlines that are selected by default are represented by blue lines.

5.2.1 Various features in the framework

Various features are included in the GeoFilter which are used off the shelf from the Mapserver api. The features that are used are as follows:

Browse map: Browse map allows user to move to various locations on the map. When user clicks on a particular point on the map the map is moved to that point.

Select layers: The layer selection can be done by selecting the checkbox beside the layer names.

Select active layer: To select an active layer the radio button beside the layer should be selected. This allows the user to run queries and rules on that particular layer.

Query single layer: This feature acquires the details of the point selected from the active layer.

Query multiple layers: When multiple layers are selected the features of the point selected and the features of the layers selected are displayed.

Zoom in & Zoom out: This allows the user to zoom in and zoom out the map to make the area clear.

Run rules on the selected layer: This feature allows the user to run the rules on the active layer.

5.2.2 Process description

The layers are displayed when the checkbox is selected. From this point the implementation is described considering the “Surface Sources” layer as the active layer.

Figure 9 shows the surface source layer with the points of the surface sources which are indicated with the orange triangle.

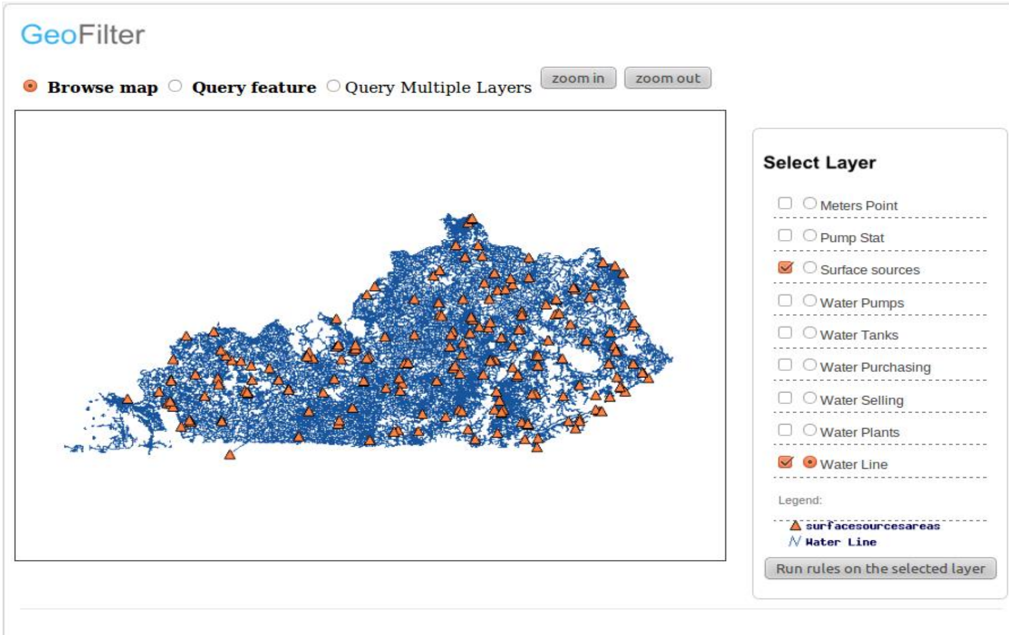

Figure 9: Surface source layer

To check the details of a particular point, the user has to select the active layer. To query the map for details the user selects the feature "query single layer" or "query multiple layers". In order to query the details of a layer, the layer should be associated with a template file. Hence each layer on a map is associated with three html template files: _header, _footer and _query. All the three files are combined while displaying the results of a query. The _query file has the information of the attribute names, based on which the details of the attributes in the selected layer are displayed. A query map is associated to the user query selection. In Figure 10, considering the active layer as the surface sources, the details of the selected point are displayed highlighting the selected point coloring in red as shown. In this way the framework allows the user to directly interact with the geospatial data.

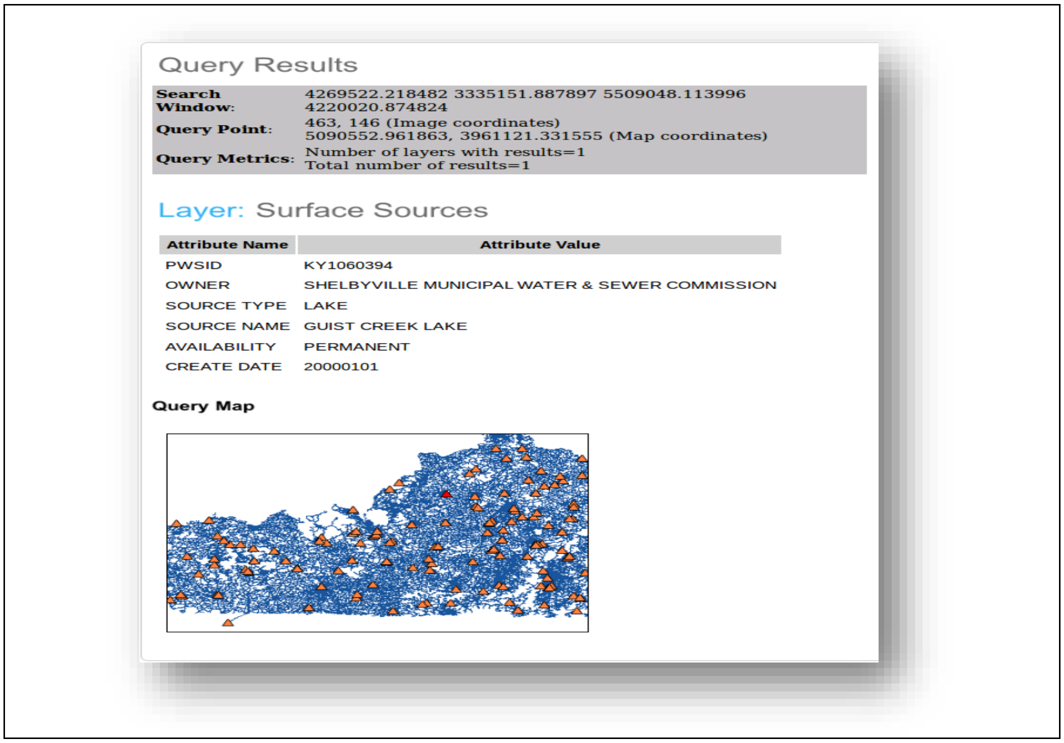

Figure 10: Query results

5.2.3 Running Rules

To run a rule on a layer, the layer must be selected as an active layer then the user has to click on the button “Run rules on the selected layer”. Now when the user clicks the button the selected layer is passed to the “runrules”, a PHP page. This page gets the selected layer and based on the layer selected the corresponding action is taken. Each of the layers from the framework is associated with a corresponding shapefile.

Esri shapefile stores the information associated with the geometry which is non-topological. It is faster to extract and edit the data as the shape files do not have the processing overhead [**37**]. Ogr2ogr is a tool used to convert the shapefile to a GML file. GML is geographic markup language, a flavor of XML. The command in the Figure 11 is executed using the PHP code. The command is a unix shell command.

```
$run=shell_exec("ogr2ogr -f \"gml\" surfsrc.gml surfsrc.shp");
```

Figure 11: Command to convert a Shape file to GML

From the converted GML file it can be clearly observed that the data associated with a shape file is semi structured. The converted GML file has the details of points: if it is a layer of points, lines: if it is a layer of lines, etc. The semi structured data for the surface sources is shown in the Figure 12.

```
<ogr:FeatureCollection
     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xsi:schemaLocation="http://ogr.maptools.org/ surfsrc.xsd"
     xmlns:ogr="http://ogr.maptools.org/"
     xmlns:gml="http://www.opengis.net/gml">
  <gml:boundedBy>
    <gml:Box>
<gml:coord><gml:X>4087720.746940493</gml:X><gml:Y>3353675.031210368</gml:
Y></gml:coord>
<gml:coord><gml:X>5912070.477199151</gml:X><gml:Y>4283830.996003891</gml:
Y></gml:coord>
    </gml:Box>
  </gml:boundedBy>
  <gml:featureMember>
    <ogr:surfsrc fid="F0">
```

<ogr:geometryProperty><gml:Point><gml:coordinates>5218142.494258911348879,38 22886.250301834195852</gml:coordinates></gml:Point></ogr:geometryProperty>

<ogr:PWSID>KY0840180</ogr:PWSID>

<ogr:OWNER>HARRODSBURG MUNICIPAL WATER DEPARTMENT</ogr:OWNER>

<ogr:SRCNAME>KENTUCKY RIVER POOL #7</ogr:SRCNAME>

<ogr:INTK_LOC>TWO ADJACENT SURFACE WATER INTAKES LOCATED AT RM 117.85L OF THE KENTUCKY RI</ogr:INTK_LOC>

<ogr:SRCTYPE>RIVER</ogr:SRCTYPE>

<ogr:AVAIL>PERMANENT</ogr:AVAIL>

<ogr:INTK_EL>533.00</ogr:INTK_EL>

<ogr:PERM_ID>0264</ogr:PERM_ID>

<ogr:PWDRAW1>3.20</ogr:PWDRAW1>

<ogr:PWDRAW2>3.20</ogr:PWDRAW2>

<ogr:AVGWDRAW>2.78</ogr:AVGWDRAW>

<ogr:HWDRAW>3.80</ogr:HWDRAW>

<ogr:INTKTYPE>FREESTANDING CONCRETE</ogr:INTKTYPE>

<ogr:AGENCY>BGADD</ogr:AGENCY>

<ogr:CONTACT>SAMANTHA MYERS</ogr:CONTACT>

<ogr:XY_SOURCE>COUNTY BASE MAPS</ogr:XY_SOURCE>

<ogr:ATT_SOURCE>ROBERT NORMAN, SUPERINTENDENT</ogr:ATT_SOURCE>

<ogr:MODIFYDATE>2009/02/24</ogr:MODIFYDATE>

<ogr:CREATEDATE>2007/05/21</ogr:CREATEDATE>

</ogr:surfsrc>

```
</gml:featureMember>
```

Figure 12: Structure of a converted GML file

JESS triggers rules only when the corresponding facts are encountered by the JESS engine. So the details from the GML file are extracted using the following PHP code and facts are generated. PHP uses the XMLREADER function to parse through the GML file.

```
if(file_exists($filename))
	{
		$reader=new XMLReader();
		$fid=array();
		$srctyp=array();
		$avail=array();
		$reader->open("surfsrc.gml");
		$i=0;
		$j=0;
		$k=0;
		while($reader->read())
		{
			if($reader->nodeType==XMLReader::ELEMENT)
			{
				if($reader->localName=="surfsrc")
				{
					$fid[$i]= $reader->getAttribute("fid");
					$i++;
				}
				if($reader->localName=="SRCTYPE")
				{
					$srctyp[$j]= $reader->expand()-
>textContent;
					echo "srctypval=".$srctyp[$j];
					$j++;
				}
				if($reader->localName=="AVAIL")
				{
					$avail[$k]= $reader->expand()-
>textContent;
					$k++;
				}
```

```
                }
        }

        $ourFileName = "sources.clp";
        $ourFileHandle = fopen($ourFileName, 'w') or die("can't open
file");
        for($i=0;$i<sizeof($srctyp);$i++)
        {
                echo "(sources(fid-is ".$fid[$i].")(srctyp-is " .
$srctyp[$i] . ")(availablity-is ".$avail[$i]."))<br>";
                $stringData = "(sources(fid-is ".$fid[$i].")(srctyp-is " .
$srctyp[$i] . ")(availablity-is ".$avail[$i].")) \n";
                fwrite($ourFileHandle, $stringData);
        }
        fclose($ourFileHandle);
```

Figure 13: Php code to generate the facts

These facts are saved in a file named “sources.clp”. Figure 14 shows the structure of the facts file. The file consists of the parameters: feature id, the source type and the availability of the source.

```
(sources(fid-is F17)(srctyp-is CREEK)(availablity-is PERMANENT))

(sources(fid-is F18)(srctyp-is RESERVOIR)(availablity-is PERMANENT))

(sources(fid-is F19)(srctyp-is RESERVOIR)(availablity-is EMERGENCY))

(sources(fid-is F20)(srctyp-is LAKE)(availablity-is PERMANENT))

(sources(fid-is F21)(srctyp-is RESERVOIR)(availablity-is EMERGENCY))
```

Figure 14: Structure of a facts file (sources.clp)

A simple rule to filter the “Surface Sources” is written in the form of a query and extracts the feature member id’s of the attributes that has the availability-is = “EMERGENCY”. This rule is executed using a java program importing the JESS library.

A template is defined to identify the fact structure and a query for processing data is shown in the Figure 15.

```
//Template

(deftemplate sources (slot fid-is) (slot srctyp-is)(slot availablity-is))

//query

(batch facts.clp)
(load-facts sources.clp)
(defquery search-sources
   (declare
      (variables ?availability))
   (sources
      (fid-is ?fid)
      (srctyp-is ?src)
      (availablity-is ?availability)))
```

Figure 15: A simple template and query

The java program applies the facts on the rules and the filtered result with the feature member ids saved into a file. Referencing these feature member ids, the data is filtered from the GML file (from the shape file) and a new GML file is created with the attributes. The GML file generated is added to the map file which acts as a new layer.

Chapter 6: Discussions and Results

In this thesis a web based framework is developed to prove that the integration of GIS and ES can solve the spatial decision making problems. This framework is operating system independent which facilitates a need to use less number of system resources and it can run on a system with minimal configuration. To demonstrate this process a simple spatial decision problem "filtering a layer of a GIS based on certain condition" is taken into consideration. This process involves: converting the shape file selected by a user into a GML file; extracting the contents of GML; generating a set of facts; applying these facts on a set of rules proposed by experts; generating a new GML file based on the filtered details; and adding this GML file which constitutes filtered results of the selected layer as a new layer to the map. As the geographic data is of a semi-structured format, the number of fields associated with a feature member may not be consistent. The database entities are designed to accept maximum data, i.e., the entities are designed with the number of fields that are equal to the maximum number of fields associated to any feature member. The semi-structured data sometimes does not have values to certain attributes which are replaced by empty values in the attribute fields; this increases the size of the database. In this thesis, first the process of extraction of GIS data takes place and then this extracted data is stored in a GML file and then this GML file, which contains the geographic data, is used in the process of visualization and data processing. The GML format which is similar to XML supports query and processing of semi-structured data easily. The space consumed for the new layer file is comparatively less, as the system uses GML as an alternative to shapefile. The process of converting a shapefile to GML is explained in detail in chapter 5. To implement the GeoFilter framework, an operating system independent GIS server: Mapserver and JESS expert system are integrated. The final framework is a web based integrated system (which is developed based on 4.2 A simple mechanism (Web-based)), which can be used irrespective of the operating systems and which can retrieve the user requested data from the graphical user interface of the GIS and the expert system makes a decision and then based on the decision made the associated data is visualized. The final output of the system looks as shown in the Figure 16

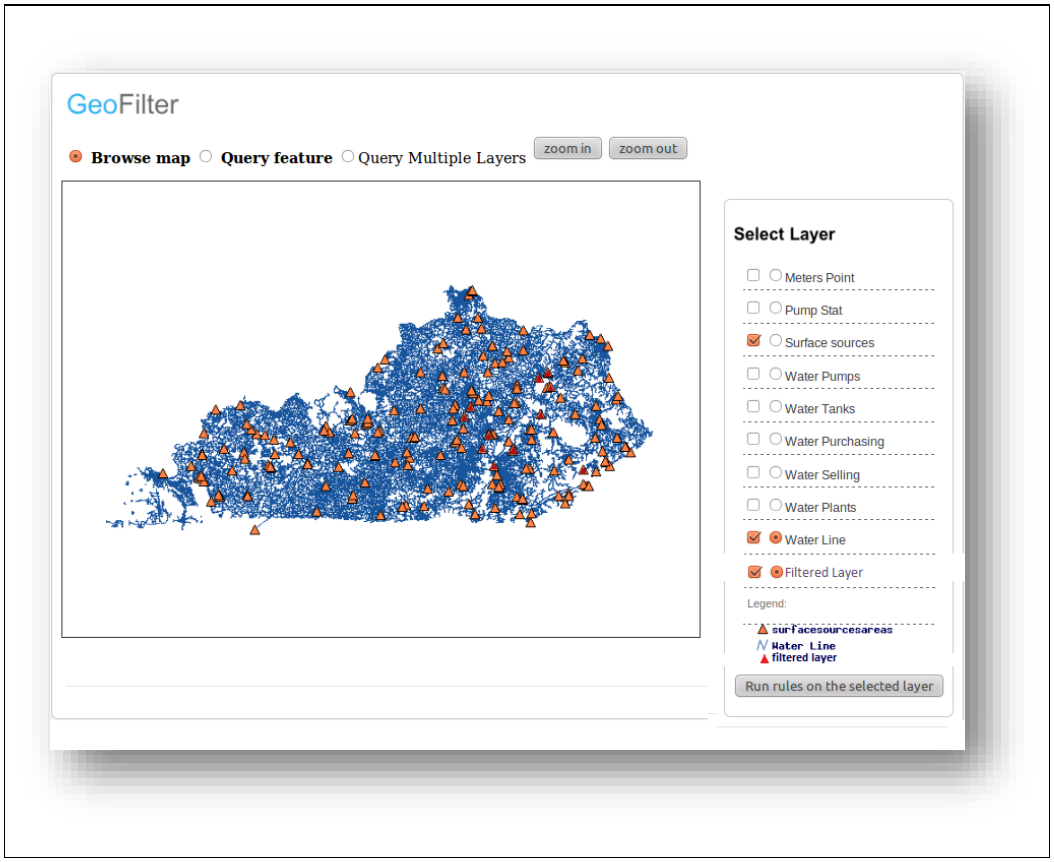

Figure 16: Final Result

Chapter 7: Conclusion

With the growth of spatial data usage in many areas such as organizations, companies, government sectors, etc., the necessity of using different framework architectures to solve spatial decision problems has also increased. This thesis explains how the integration of a Geographic Information System and Expert System can solve the problems of spatial decision making. Various mechanisms are proposed for achieving the decision making support based on the organizational needs. Using one of these mechanisms, a geographic expert system named “GeoFilter” to solve a spatial decision making problem is developed which uses the analytical knowledge of the expert system and geographic information availability and visualization capabilities of the Geographic Information System.

The mechanism that is implemented in the thesis is a simple web based mechanism among the various mechanisms proposed in Chapter 4 (Section 4.2 A simple mechanism (Web-based)). The process of retrieving data in comparatively less time is also proposed. Based on the organization requirements the spatial decision making problem can be solved by following the other mechanisms which include a standalone mechanism, a plugin based mechanism and MVC based mechanism.

Chapter 8: Future work

The system implemented in this thesis which is operating system independent can be developed further by following the MVC mechanism as stated in the Chapter 4 (Section 4.4 Model View Controller (MVC) based mechanism). As the user interface of the system is mostly the GIS engine and as the processing part is mostly handled by the ES, the MVC architecture helps in separating the design logic and business logic such that the user can easily update the GIS engine preferences and ES preferences without disturbing the whole system functionality.

APPENDIX

GeoFilter – simple mechanism (web based)

Kentuckysection.html (Some parts of the code of the user interface of GeoFilter is copied from the demo application of mapserver website which can be found http://mapserver.org/download.html).

In the following code a simple html version of mapserver is used to display the interface of the GIS. PHP code is used to communicate as the layer between mapserver the GIS and JESS the Expert system.

```
<!-- MapServer Template -->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Decon-expert</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link href="http://localhost/eclipse-php/ky_rules/htdocs/style.css" rel="stylesheet"
type="text/css" />
<script type="text/javascript" src="http://localhost/eclipse-php/ky_rules/htdocs/js/cufon-
yui.js"></script>
<script type="text/javascript" src="http://localhost/eclipse-
php/ky_rules/htdocs/js/arial.js"></script>
<script type="text/javascript" src="http://localhost/eclipse-
php/ky_rules/htdocs/js/cuf_run.js"></script>
<script src="http://localhost/eclipse-php/ky_rules/htdocs/js/jquery.js"
type="text/javascript"></script>

<script src="http://localhost/eclipse-php/ky_rules/htdocs/js/custom.js"
type="text/javascript"></script>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/jquery.min.js"
type="text/javascript" charset="utf-8"></script>
<style type="text/css">

</style>
<script type="text/javascript">
    //window.alert(getRadioValue("qvalue"));
    var selected_layer;
    function getRadioValue(RadioName) {
        var colRadio = document.getElementsByName(RadioName);
        for (var i = 0; i < colRadio.length; i++) {
            if (colRadio[i].checked == true) {
                selected_layer = (colRadio[i].value);
                document.getElementById("selVal").value = selected_layer;
            }
            //return null;

        }
        //alert(selected_layer.toString());
        document.forms["rule_form"].submit();
    }
    //vValue = getRadioValue("qvalue");
```

```
    function zoomin() {
        document.getElementById("zoomid").selectedIndex = 0;
        document.mapserv.submit();
    }
    function zoomout() {
        document.getElementById("zoomid").selectedIndex = 2;
        document.mapserv.submit();
    }
    function onQuery() {
        document.getElementById('layerdiv').style.display = 'none';
        document.getElementById('zoomdiv').style.display = 'none';
        document.getElementById('tableid').border = "0";
        document.getElementById('selectlayers').innerHTML='Please switch to Browse Mode to View
Other Maps';
        document.mapserv.mode.value = "browse";
        document.mapserv.zoom.value = "1";
    }
    function onNQuery() {
        document.getElementById('layerdiv').style.display = 'none';
        document.getElementById('zoomdiv').style.display = 'none';
        document.getElementById('tableid').border = "0";
        document.getElementById('selectlayers').innerHTML='Please switch to Browse Mode to View
Other Maps';
        document.mapserv.mode.value = "nquery";
        document.mapserv.zoom.value = "1";
    }
    function onBrowse() {
        document.getElementById('layerdiv').style.display = 'block';
        document.getElementById('zoomdiv').style.display = 'block';
        document.getElementById('tableid').border = "1";
        document.getElementById('selectlayers').innerHTML = 'Select Layers :';
    }
    function showImage(nodeId) {
        document.getElementById(nodeId).style.visibility = 'visible';
    }
    function jump2J(ext) {
        alert('im in');
        if (ext != "") {
            alert('im again in');
            document.mapserv.imgext.value = ext;
            document.mapserv.mode.value = 'browse';
            document.mapserv.zoom.value = '1';
            document.mapserv.submit();
        }
        alert('im out');
    }
</script>



</head>
<body>
<div class="main">

</div>
<form name="mapserv" method="GET" action="[program]">
<input type="hidden" name="root" value="[root]">
<input type="hidden" name="program" value="[program]">
<input type="hidden" name="map_web" value="[map_web]">
<input type="hidden" name="map" value="[map]">
<input type="hidden" id ="zoom" name="imgext" value="[mapext]">
<input type="hidden" name="imgxy" value="[center]">
<input type="hidden" name="savequery" value="true">
<input type="hidden" name="layer" value="county_label">
<div class="body_bg">
<div class="body">
<div class="left_resize block">
<div class="left ">
```

```
<!--<h2><span>Process</span> Flow</h2>-->
<div>
<h2><span>Geo</span>Filter</h2>
<table>
<tr>
<table>
<tr>
<td>    <input type="radio" name="mode" value="browse" checked onclick="onBrowse()"> <b>Browse
map</b></td>
<td><input type="radio" name="mode" value="query" [query_select] onclick="onQuery()"> <b>Query
feature</b></td>
<td><input type="radio" name="mode" value="nquery" [nquery_select] onclick="onNQuery()">Query
Multiple Layers</td>

<td><select id="zoomid" name="zoom" style="display:none;">
<option value="2" [zoom_2_select]>Zoom In 2x</option>
<option value="1" [zoom_1_select] selected>Recenter</option>
<option value="-2" [zoom_-2_select]>Zoom Out 2x</option>
</select></td>

<td><div id="zoomdiv">
<input type="button" id="zoominbutton" value="zoom in" onclick="zoomin()"> </input>
<input type="button" id="zoomoutbutton" value="zoom out" onclick="zoomout()"> </input>

</div>
<br/>
</td>
</tr>
</table>

</tr>

<tr>
<td>
<input type="image" name="img" src="[img]"
width="[mapwidth]" height="[mapheight]" border="1" /></td>
<div class="right block">
<td>
<h3><span>Select </span>Layer</h3>

<ul>

<li><input type="checkbox" value="meters_point" name="layer" [meters_point_check]
onClick="document.mapserv.mode.value='browse';document.mapserv.zoom.value='1';submit()"><input
id="qlayer" type="radio" name="qlayer" value="meters_point" [meters_point_select]>Meters
Point</li>

<li><input type="checkbox" value="pumpstat" name="layer" [pumpstat_check]
onClick="document.mapserv.mode.value='browse';document.mapserv.zoom.value='1';submit()"><input
id="qlayer" type="radio" name="qlayer" value="pumpstat" [pumpstat_select]>Pump Stat</li>

<li><input type="checkbox" value="surfacesources" name="layer" [surfacesources_check]
onClick="document.mapserv.mode.value='browse';document.mapserv.zoom.value='1';submit()"><input
id="qlayer" type="radio" name="qlayer" value="surfacesources" [surfacesources_select]>Surface
sources</li>

<li><input type="checkbox" value="water_pumps" [water_pumps_check] name="layer"
onClick="document.mapserv.mode.value='browse';document.mapserv.zoom.value='1';submit()"><input
id="qlayer" type="radio" name="qlayer" vale="water_pumps" [water_pumps_select]>Water Pumps</li>

<li><input type="checkbox" value="water_tanks" [water_tanks_check] name="layer"
onClick="document.mapserv.mode.value='browse';document.mapserv.zoom.value='1';submit()"><input
id="qlayer" type="radio" name="qlayer" value="water_tanks" [water_tanks_select]>Water Tanks</li>

<li><input type="checkbox" value="water_purchasing" [water_purchasing_check] name="layer"
onClick="document.mapserv.mode.value='browse';document.mapserv.zoom.value='1';submit()"><input
id="qlayer" type="radio" name="qlayer" value="water_purchasing" [water_purchasing_select]>Water
Purchasing</li>
```

```
<li><input type="checkbox" value="water_selling" [water_selling_check] name="layer"
onClick="document.mapserv.mode.value='browse';document.mapserv.zoom.value='1';submit()"><input
id="qlayer" type="radio" name="qlayer" value="water_selling" [water_selling_check]>Water
Selling</li>

<li><input type="checkbox" value="wtp" [wtp_check] name="layer"
onClick="document.mapserv.mode.value='browse';document.mapserv.zoom.value='1';submit()"><input
type="radio" id="qlayer" name="qlayer" value="wtp" [wtp_select]>Water Plants</li>

<li><input type="checkbox" name="layer" value="water_line" [water_line_check]
onClick="document.mapserv.mode.value='browse';document.mapserv.zoom.value='1';submit()"><input
id="qlayer" type="radio" checked="true" value="water_lines" name="qlayer"[water_line_select]>Water
Line</li>

</ul>
<ul>
<li><p>Legend:<br>
<img src="[legend]"></li>
</ul>
</form>
<form name="rule_form" action="/eclipse-php/ky_rules/htdocs/surfsrc_parts/runrules.php"
method="post">
<input type="button" value="Run rules on the selected layer" name="run_rules"
onClick="getRadioValue('qlayer')"/>
<input type="hidden" name="selVal" Id="selVal" value="hi"/>
</form>
</td>

</tr>
</table>
</p>
</div>
</div>

 </div>
<div class="bg"></div>
</div>

<div class="clr"></div>
</div>
</div>
<div id="outputtext"></div>
<div class="FBG_bg">
<div class="FBG">
<div class="clr"></div>
</div>
</div>
<div class="clr"></div>
<div class="footer">
<div class="footer_resize">
<p class="lf">&copy; Copyright <a href="#">Water-expert</a>.</p>
<div class="clr"></div>
</div>
<div class="clr"></div>
</div>
</div>
</body>
</html>
```

findingsurfsrc.php **(This code checks the GML file and creates facts for that layer. Here the code is shown for surface sources layer)**

```
<html>

<head>
  <title></title>
</head>

<body>

<?php
$filename="srufsrc.gml";
$chkshpFile="surfsrc.shp";
$chkshxFile="surfsrc.shx";
$chkdbfFile="surfsrc.dbf";
if (file_exists($chkshpFile)&&file_exists($chkshxFile)&&file_exists($chkdbfFile))
{
        //$run=shell_exec("./createGml.sh");
        $run=shell_exec("ogr2ogr -f \"gml\" surfsrc.gml surfsrc.shp");
        if(file_exists($filename))
        {
                $reader=new XMLReader();
                $fid=array();
                $srctyp=array();
                $avail=array();
                $reader->open("surfsrc.gml");
                $i=0;
                $j=0;
                $k=0;
                while($reader->read())
                {
                        if($reader->nodeType==XMLReader::ELEMENT)
                        {
                                if($reader->localName=="surfsrc")
                                {
                                        $fid[$i]= $reader->getAttribute("fid");
                                        $i++;
                                }
                                if($reader->localName=="SRCTYPE")
                                {
                                        $srctyp[$j]= $reader->expand()->textContent;
                                        $j++;
                                }
                                if($reader->localName=="AVAIL")
                                {
                                        $avail[$k]= $reader->expand()->textContent;
                                                                                $k++;
                                }
                        }
                }

                $ourFileName = "sources.clp";
                $ourFileHandle = fopen($ourFileName, 'w') or die("can't open file");
                for($i=0;$i<sizeof($srctyp);$i++)
                {
                echo "(sources(fid-is ".$fid[$i].")(srctyp-is " . $srctyp[$i].") (availablity-is
".$avail[$i]."))<br>";
                $stringData = "(sources(fid-is ".$fid[$i].")(srctyp-is ".$srctyp[$i] . ")
(availablity-is ".$avail[$i].")) \n";
                        fwrite($ourFileHandle, $stringData);
                }
                fclose($ourFileHandle);
        }
        else
        echo "The file 'surfsrc.gml' doesnt exist";
}
else
```

```
{
    echo "Please check if the files 'surfsrc.shp','surfsrc.shx' and 'surfsrc.dbf' exist";
}
?>
</body>
</html>
```

fetching.java

/* Based on the facts created by the php code the rules are executed. The Rete class is used which is used to communicate between the java code and the jess expert system. */

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.Scanner;
import javax.xml.parsers.SAXParser;
import javax.xml.parsers.SAXParserFactory;
import org.xml.sax.Attributes;
import org.xml.sax.SAXException;
import org.xml.sax.SAXParseException;
import org.xml.sax.helpers.DefaultHandler;
import jess.*;
public class fetching {
        public static void main(String[] argv) throws JessException, IOException
        {
                Rete r=new Rete();
                r.batch("query.clp");
                QueryResult res=r.runQueryStar("search-sources", new ValueVector().
                add("EMERGENCY"));
                int count=1;
                BufferedWriter differentMaterials=new BufferedWriter(new
                FileWriter("filtered.clp"));
                while(res.next())
                {

                        differentMaterials.write(res.getString("fid")+"\n");
                        System.out.println(res.getString("fid")+ "
                        "+res.getString("availability"));
                        count++;
                }
                differentMaterials.close();

        }

}
```

REFERENCES

[1] L Worrall, *Spatial analysis and spatial policy using geographic information systems*. Newyork: Belhaven Press, 1991.

[2] Michael N. DeMers, "GIS for Dummies," in *GIS for Dummies*., 2009.

[3] M.S., Maier, H.R, Dandy, G.C. Gibbs, "The use of GIS to derive distributed unit hydrographs for streamflow prediction," in *18th World IMACS / MODSIM Congress*, Cairns, Australia, 2009, pp. 3704-3710.

[4] Chuen-Jung Changa, Hsiao-Han Tua, Shiou-Wen Yehb Jia-Jiunn Loa, "Applying GIS to develop a web-based spatial-person-temporal history educational system," *Computers & Education*, vol. 53, no. 1, pp. 155–168, August 2009.

[5] Khalid Eldrandaly, "EXPERT SYSTEMS, GIS, AND SPATIAL DECISION MAKING: CURRENT PRACTICES AND NEW TRENDS," in *Expert Systems Research Trends*.: Nova Science Publishers, 2007, pp. 207- 228.

[6] Mark Coletti, Sean Luke Keith Sullivan, "GeoMason: GeoSpatial Support for MASON," George Mason University, Fairfax, Technical Report GMU-CS-TR-2010-16, 2010.

[7] Andrea Ballatore, Ali Tahir, Michela Bertolotto Gavin McArdle, "AN OPEN-SOURCE WEB ARCHITECTURE FOR ADAPTIVE LOCATION-BASED SERVICES," *Computer Science and Informatics*, vol. 38, no. 2, pp. 296–301, 2010.

[8] Ali Tahir, Gavin Mcardle, Michela Bertolotto Andrea Ballatore, "A comparison of open source geospatial technologies for web mapping," *Web Engineering and Technology*, vol. 6, no. 4, pp. 354-374, 2011.

[9] Bill Kropla, *Beginning MapServer: Open Source GIS Development*, Jason Gilmore, Ed. United States of America: Apress, 2005.

[10] Zhao Guo-Cheng, Kan Ying-Hong Xiao Qiang, "Transfer between ShapeFile data and geographic information exchange format data," *Science of Surveying and Mapping*, vol. 35, no. 6, pp. 82-83, Nov 2010.

[11] Shen Ye, "An approach to efficient compression transmission schema of GML," in *Information Computing and Telecommunications (YC-ICT)*, Beijing, 2010, pp. 41-44.

[12] M.B. Jain, "A novel Web based Expert System Architecture for on-line and off-line fault diagnosis and control (FDC) of transformers," in *TENCON 2008 - 2008 IEEE Region 10 Conference*, Hyderabad, 2008, pp. 1- 5.

[13] M Godse, "Using Rule Based Expert System to Wrap Procedural Code of Integrated Watershed Model," in *Emerging Trends in Engineering and Technology, 2008.*, Nagpur, Maharashtra, 2008, pp. 332- 336.

[14] Vassiliki Kollias, Nikos A. Lorentzos, Dionisios Kalivas and Alexander B. Sideridis Constantine P. Yialouris, "An Integrated Expert Geographical Information System for soil suitability and soil evaluation," *Journal of Geographic Information and Decision Analysis*, vol. 1, no. 2, pp. 89-99, 1997.

[15] Wu Le-Nan, Wang Shui-Hua Zhang Yu-Dong, "Survey on development of expert system ," *Computer Engineering and Applications*, vol. 46, no. 19, pp. 43-47, JAN 2010.

[16] Marvin Minsky, *A Framework for Representing Knowledge*.: MIT Press.

[17] Lai Nanjun, "Application of Rule Based Expert System to Sand Control in Oil Fields," in *Intelligent Computation Technology and Automation (ICICTA)*, Zhangjiajie, Hunan, 2012, pp. 110- 113.

[18] Petter Gottschalk, "Expert systems at stage IV of the knowledge management technology stage model: The case of police investigations," *Expert Systems with Applications*, vol. 31, no. 3, pp. 617-628, OCT 2006.

[19] Cees Bil, Xinghuo Yu, Adrian Smith Christian Van der Velden, "An Intelligent System for Rule Based Automated Routing," in *5th Australasian Congress on Applied Mechanics*, vol. 17, Brisbane, Australia, 2007, pp. 10-12.

[20] Ajith Abraham, *Handbook of Measuring System Design*, Peter H. Sydenham and Richard T horn, Ed. Stillwater, United States of America: John Wiley & Sons, 2005.

[21] Nwigbo Stella N and Agbo Okechuku Chuks, "EXPERT SYSTEM: A CATALYST IN EDUCATIONAL DEVELOPMENT IN NIGERIA".

[22] Ching-Wen Chen, Weissor Shiue Li-Yen Shue, "The development of an ontology-based expertsystem for corporate financial rating," *Expert Systems with Applications*, vol. 36, no. 2, pp. 2130–2142, MAR 2009.

[23] Charles Hunn, "1st International Workshop on Intelligent Support for Exploratory Environments," *Employing a Java Expert System Shell for Intelligent Support in Exploratory Activities*, vol. 381, September 2008.

[24] Ralph Grove, "Expert Systems," *Internet-based expert systems*, vol. 17, no. 3, July 2000.

[25] Andrew Colesand, Maria Foxand, Derek Long Amanda Colesand, "Forward-Chaining Partial-Order Planning," in *Twentieth International Conference on Automated Planning and Scheduling*, Toronto, Canada, 2010, pp. 42-49.

[26] Shiqi Yang Yongjin Xu, "Design of the Fertilization Expert System Based on Plug-in Technology," in *PIAGENG 2009: Intelligent Information, Control, and Communication Technology for Agricultural Engineering*, China, 2009, pp. Vol 74901G-(1-8).

[27] Amiruddin Ismail,Riza Atiq and Othman A. Karim Deprizon Syamsunur, "Knowledge-based Expert System for Route Selection of Road Alignment," *Australian Journal of Basic and Applied Sciences*, pp. 208-213, 2011.

[28] Peter Y Wu and Eugene Rathswohl, "Address Matching: An Expert System and Decision Support Application for GIS," in *Information Systems Educators Conference*, Nashville Tennessee, USA, 2010.

[29] Jan Brus, and Jan Caha Vilem Pechanec, "Determining Ecotones by Decision Support Systems," in *Dateso 2011*, vol. 706, Czech Republic, 2011.

[30] M. Galetakis and Ch. Roumpos F. Pavloudakis, "A spatial decision support system for the optimal environmental reclamation of open-pit coal mines in Greece," *International Journal of Mining, Reclamation and Environment*, vol. 23, no. 4, pp. 291-303, December, 2009.

[31] Aditya Kumar Tadakaluru, "GeoExpert - An Expert System Based Framework for Data Quality in Spatial Databases," Bowling Green, Kentucky, 2006.

[32] Suresh C Jayanty, "A Rule Based Expert System Framework for Small Water Systems," Bowling Green, Kentucky, 2005.

[33] Zhang Yi and Wu Li-li, "A Study on Plug-in-Based GIS Application Framework," in *ICCIS '10 Proceedings of the 2010 International Conference on Computational and Information Sciences*, Chengdu, 2010, pp. 85-88.

[34] J.D. McCarthy, P.A. Graniero, and S.M Rozic, "Sensors," *An Integrated GIS-Expert System Framework for Live Hazard Monitoring and Detection*, vol. 8, no. 2, pp. 830-846, 2008.

[35] Ramanathan Sugumaran and John DeGroote, *Spatial Decision Support Systems - Principles and Practices*. 6000 Broken Sound Parkway NW, Suite 300: Taylor and Francis Group, LLC, 2011.

[36] WRIS Geospatial Data. [Online]. http://kia.ky.gov/wris/data.htm

[37] Kaspars Krauklis, Irina Eglite, Viesturs Skibelis, Antons Macans, Janis Slangens, "Matching of the ESRI Shapefile Format with the GDI Software," *Boundary Field Problems and Computer Simulation*, vol. 45, pp. 53-59, 2010.

[38] Martin Strauss, "JESS-The Java Expert System Shell," *AI Tools*, April 2007.

[39] J T COPPOCK and D W RHIND, "Predicting cover types in a mountain range with long evolutionary grazing history: a GIS approach," *Journal of Biogeography*, vol. 35, no. 3, pp. 538-551, MAR 2008.