

**GeoVisual Interface – A Visual Query Interface for Geographic Information Systems**

Damires Ylusk de Souza Fernandes 1 2
Ana Carolina Salgado 1

1UFPE - Universidade Federal de Pernambuco, CP7851, 50732-970, Recife, PE, Brasil
{dysf,acs}@di.ufpe.br

2 CEFET-PB Centro Federal de Educação Tecnológica da Paraíba, 58015-430, João Pessoa, PB, Brasil
damires@gi.cefetpb.br

**Abstract**

This work presents GeoVisual Interface - an interface for visual queries in Geographic Information Systems (GIS). GeoVisual Interface is the upper level module of a complete visual query system architecture. It allows users to formulate queries using visual elements (i.e. spatial objects from a visualization area, from the database schemas or from a geographic entity metaphor) and spatial operators via direct manipulation. Cognitive, ergonomic and technical aspects are also considered to improve the spatial query formulation capability and enhance the human-computer interaction, making the system more attractive to users.

1. Introduction

It is known that the initial impression causes a very strong feeling, not just from person to person, but also between people and objects [16]. This is also the case for computational system interfaces because the interaction between users and these systems encompasses both technical aspects and those related to the human behaviour.

Both the design and implementation of natural and intuitive interfaces are also important in the context associated with the use of Geographic Information Systems (GIS). GIS are systems that perform data management and retrieval operations for georeferenced data. A GIS interface design should deal with the characteristics and difficulties faced in the elaboration of a DBMS interface and provide the specific GIS application requirements as well. These requirements include multiple representations for objects and spatial query formulation.

Spatial query formulation is a very important activity in the process of geographic data exploration. Nowadays, there are several ways users can build a spatial query including: item(operators, values) selection, composition of elements visualized on the screen or composition of objects presented in the database schema.

Considerable attention in recent years has been given to the relationship between spatial query language capabilities and users’ perception [3,5,12]. It has been suggested that the interface design for querying geographic databases should take into account the following requirements [4,8,10,11,12]: at which level is taken the user's initial interaction with GIS; if it is possible to think and formulate queries in a spatial way; what kind of spatial operators are available; if the database schema can be manipulated at the interface level; which help and error prevention
mechanisms are provided and, finally, what kind of ergonomic approaches can be used in the interface model design. These issues relating to the design of more friendly GIS user interface are the subject of the work presented in this paper.

We have been working on a visual query interface design as an effort to allow formulating queries on geographical databases by means of an easy to use visual form. This interface is the upper level module of a Visual Query System [22] and provides results derived from a multi-disciplinary study consisting of the following research subjects: query interface requirements, platform independence, spatial operators, GIS and ergonomic and human-computer interaction factors. In this paper, we present an overview of these related topics. Moreover, our visual query interface specification and prototype system will be presented.

This paper is organized as follows. Section 2 approaches the general requirements for spatial query interfaces. Section 3 defines a group of operators that were used in our interface. Section 4 presents our GeoVisual Interface outlining the main issues concerning its conceptual specification, implementation details, prototype system interface and experimental result derived from a user interface evaluation task. Finally, section 5 draws the conclusions and future work.

2. Query Interface Requirements

In a database environment, one of the main functional requirements is the use of a mechanism that allows the user to formulate queries. Each query generates results that can be visualized in several ways and that can be used in new queries.

A query formulation mechanism in Geographic Database Systems is also important [6]. However, a higher level of complexity is observed in geographic data manipulation because of their special characteristics (spatial location, size, continuity). Thus, there is a need for special visualization tools and exploration mechanisms in GIS to make provision for the spatial presentation of geographic data.

GIS interfaces which are rather closer to the way users work manually should be preferred. In other words, users would be more familiar working directly with maps, photos and geographic objects in general than dealing with non-visual query language interfaces. These objects should be manipulated and explored graphically to provide users with GIS interfaces that do indeed capture and reflect the way they perceive entities in geographical space. Examples of geographic data that compose such objects are: rivers, lakes, forests, land, countries.

Visual interfaces for databases, in particular, the so-called Visual Query Systems – VQS [20] have been proposed as an alternative approach to traditional query languages, such as SQL. VQS are characterized by some features, such as[20]:

- The use of icons and visual metaphors, which attracts the user attention and stimulates her/his curiosity, instead of text;
- The fact that the user does not need to have previous knowledge about the database schema and does not have to be familiar with a query language syntax;
- The availability of interactive mechanisms to support the typical process of query formulation.

The strategies for formulating a query are classified as follows[20]: by schema navigation, which allows concentrating on a group of concepts of interest, on which further conditions may be specified; by sub-queries, where the query is formulated by composing partial results; by
matching, which is based on the idea of presenting the structure of a possible answer that is matched against the stored data; and range selection, allowing a search conditioned by a given range on multi-key data sets. These options are not mutually exclusive. Thus, it is necessary to analyse the final objective of the interface and what kind of user it will be designed for.

VQS are also extensively used in geographic databases. As geographic data are usually presented graphically, they should also be optionally queried through the direct manipulation of visual elements. Visual elements would represent geographic data and other objects (e.g. spatial operators) needed for query formulation.

A spatial query provides a way of manipulating spatial relationships (e.g. intersection, adjacency, union, etc) between entities that are geometrically defined and located in the geographic space. Any query formulation strategy may be chosen in spatial query formulation. In a spatial query interface, the goal is to provide a query mechanism that allows users to think spatially using the provided visual elements to compose the query expression without having any previous information about the query language syntax.

A visual query interface is generally more natural and intuitive, in the sense of friendly and didactically guiding inexperienced users in the execution of their tasks. An interface that allows pictorial, interactive browsing and offers a simple and intuitive visual query mechanism provides a lot of advantages to users. Examples of these advantages include the fact that tasks previously seen as tedious and difficult will become easy and stimulating, and the human-computer interaction will become a natural extension of users' abilities.

Thus, to develop a query interface for GIS applications that is coherent with geographic reality, the following aspects should be considered [3,11,12,13,16]:

- to be independent of a specific GIS or platform;
- to provide powerful tools for expert users and, at the same time, to reduce the learning time of new user communities, providing a quite intuitive and stimulating initial interaction;
- to improve selection process and exploration of geographic data;
- to support browsing in two levels: maps and objects that should be directly manipulated; and database schema that should be graphically accessible, facilitating query formulation;
- to allow multiple representations of the same geographic object and multiple visualizations of a single representation, so that the user may customize data presentation form and layout;
- to provide help and error prevention mechanisms, through well elaborated messages that will not make users afraid of trying again;
- to organize interface layout, according to ergonomic principles of design: contrast, balance, color, symmetry, repetition and movement; and
- to support a set of spatial operators.

3.Spatial Operators

Spatial query formulation needs both the specification of one or more geographic objects and the application of one or more spatial operators. A query result may be a set of geographic objects or a set of scalar/boolean values. Optionally the query result may become part of another query. Thus, designing a spatial query interface implies in thinking about spatial operators.
Current GIS have so far included some spatial operators in their interface. However, nomenclature and semantic differences are observed in these systems, so that it is difficult to know which is the best name for a specific function or which is the most coherent meaning for a given operator description. Several authors [1,5,15,19,23] have worked on spatial operator definitions and usability, mainly in query languages.

This work has defined a group of spatial functions with two particular aims: (i) this group of operators is provided by our interface prototype system, (ii) it has been used as an important parameter when comparing existing interfaces with ours (ArcView [2], SPRING[21], MapInfo[17] and CIGALES[5]). Thus, our definition has selected some of the most primitive spatial operators that should be available in a GIS interface. The specification diagram that provides the semantics for the group of spatial operators, using the UML notation [14], is presented in figure 1.

For the sake of simplicity, the operators were subdivided into two classes:

- **Unary**: operator that receives a geographic object as input and returns a geographic object or a value (scalar or boolean). The set of unary operators is shown in table 1.

- **Binary**: operator that receives two geographic objects as operands and returns a geographic object or a value (scalar or boolean). Table 2 shows the binary operators.

![Figure 1 – Spatial Operators](image)

<table>
<thead>
<tr>
<th>Group</th>
<th>Operator</th>
<th>Operand</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>MetricU</td>
<td>Length(O)</td>
<td>Line</td>
<td>R+</td>
</tr>
<tr>
<td>MetricU</td>
<td>Area(O)</td>
<td>Region</td>
<td>R+</td>
</tr>
<tr>
<td>MetricU</td>
<td>Perimeter(O)</td>
<td>Region</td>
<td>R+</td>
</tr>
</tbody>
</table>

**Table 1: Unary Operators**
4. GeoVisual Interface

The design of an interface for querying geographic databases is not a trivial task [8]. Each current GIS interface has a specific goal, a group of particular users and a different GIS application environment. Before starting the development of an interface, the following initial definitions must be taken into account: what is the most important aim, what kind of users will interact with the system, what kind of environment will be available to the interface and provided by it.

<table>
<thead>
<tr>
<th>Group</th>
<th>Operator</th>
<th>Operand 1</th>
<th>Operand 2</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set-oriented</td>
<td>Union(O1,O2)</td>
<td>Region or line</td>
<td>Region or Line</td>
<td>Set of spatial objects</td>
</tr>
<tr>
<td></td>
<td>Difference (O1,O2)</td>
<td>Region or line</td>
<td>Region or Line or Point</td>
<td>Set of spatial objects</td>
</tr>
<tr>
<td></td>
<td>Belongs(O1,O2)</td>
<td>Point</td>
<td>Region or Line</td>
<td>Boolean</td>
</tr>
<tr>
<td></td>
<td>Intersection(O1, O2)</td>
<td>Region or Line</td>
<td>Region or Line</td>
<td>Set of spatial objects</td>
</tr>
<tr>
<td>Topologic</td>
<td>Disjunction(O1,O2)</td>
<td>Region or Line</td>
<td>Region or Line</td>
<td>Boolean</td>
</tr>
<tr>
<td></td>
<td>Adjacency(O1,O2)</td>
<td>Region</td>
<td>Region</td>
<td>Boolean</td>
</tr>
<tr>
<td></td>
<td>Equality(O1,O2)</td>
<td>Point, Line or Region</td>
<td>Point, Line or Region</td>
<td>Boolean</td>
</tr>
<tr>
<td></td>
<td>Inside(O1,O2)</td>
<td>Point, Line or Region</td>
<td>Region</td>
<td>Boolean</td>
</tr>
<tr>
<td></td>
<td>Overlapping(O1,O2)</td>
<td>Region</td>
<td>Region</td>
<td>Boolean</td>
</tr>
<tr>
<td>Positional</td>
<td>At_north_of(O1,O2)</td>
<td>Region</td>
<td>Region</td>
<td>Boolean</td>
</tr>
<tr>
<td></td>
<td>At_south_of(O1,O2)</td>
<td>Region</td>
<td>Region</td>
<td>Boolean</td>
</tr>
<tr>
<td></td>
<td>At_west_of(O1,O2)</td>
<td>Region</td>
<td>Region</td>
<td>Boolean</td>
</tr>
<tr>
<td></td>
<td>At_east_of(O1,O2)</td>
<td>Region</td>
<td>Region</td>
<td>Boolean</td>
</tr>
<tr>
<td>MetricB</td>
<td>Distance(O1,O2)</td>
<td>Point</td>
<td>Point</td>
<td>R+</td>
</tr>
</tbody>
</table>

Table 2 – Binary Operators
In this work, the interface design has the following goals: (i) users can be novices or experts, but our main purpose is to design an easy-to-use interface for the less experienced users, (ii) the interface should be capable of providing geographic data exploration, and (iii) the environment where it will be developed should be platform independent and be able to be extended for use in the WEB.

Our graphic user interface has been called GeoVisual Interface. It has a Dialog Box composed by a Query Editor and the Query Results Box to interact with users, as shown in Figure 2. The Query Editor box is used to allow visual queries formulation. It is composed by an edition area and a spatial operators area. Query results will be graphically presented in the Query Results Box. GeoVisual Interface also presents a GeoVisual Diagram to help users formulate their queries. This Visual Diagram presents the conceptual schema of the geographic application which is in use in the system.

![Figure 2 – GeoVisual Interface Architecture](image)

*GeoVisual Interface* is the highest level module of a visual query system (VQS), whose complete architecture is defined in [22] and is composed of three other modules: Query Manager, Metadata Model and GIS Components.

In general, the query execution process includes the following tasks: the involved geographic entities are identified, the spatial operators are validated and the complete query is syntactically and semantically analysed. After the analysis, the query formulation is translated from a visual form to a textual query clause to be executed in a GIS. Finally, the query result is graphically presented by the GeoVisual Interface.

We shall now further describe more specifically our GeoVisual Interface system prototype. Next subsections present its general description (section 4.1), the current leg of our GeoVisual development journey (section 4.2), and finally the GeoVisual Interface experimental results obtained from testing some of the system usability aspects (section 4.3).

### 4.1 GeoVisual Interface Description and Specification

*GeoVisual Interface* has been designed in the context of a Visual Query System [7]. Thus, users are provided with two ways of representing their spatial information: (i) a diagrammatic
model of the database schema, and (ii) the iconic representation of the spatial elements (geographic entities and spatial operators). The first option is mostly used by database experts, but may help novice users in understanding the application context, if they have any database modelling skills. The second option is appreciated by novice users, because it provides more familiar and intuitive elements which makes the interface more attractive and friendly.

*GeoVisual Interface* adopts a hybrid strategy for formulating queries. It allows schema navigation and querying, selection of both geographic entities and spatial operators for the query construction and reuse of formulated queries.

We are specially interested in queries that cover the possible combinations of geographic objects and spatial operators (section 3), and in the reuse of the previously executed queries, always in a visual form. The main idea is to provide the user with a simple visual query environment where query formulation is performed in a homogeneous manner, through visual elements. Thus, non-expert users do not need to understand neither the underlying data model nor the syntax of the textual query.

Thus, the *GeoVisual Interface* must provide users with capabilities to: (i) open a working session, (ii) activate the application database schema, (iii) choose the geographic layers, (iv) explore geographic data (in vector format), (v) access the conceptual application model, (vi) choose a geographic entity and interrogate its descriptive attributes, (vii) select a geographic entity and spatial operators to formulate a query, (viii) save the query results in a historical list for later use, and (ix) create a thematic map and print it.

*GeoVisual Interface* has been specified using the UML diagram notation [14]. This conceptual modeling intended to expose real requirements and to facilitate the necessary understanding of *GeoVisual Interface* functionality. Thus, the *GeoVisual Interface* functional requirements were specified by a use-case diagram. Its class structure (class, attributes and methods definitions) and relationships have also been specified. The most important use-cases have been refined by using collaboration diagrams, as well as the most important classes have been refined by using state machine diagrams. The complete conceptual modeling is presented in [7].

### 4.2 *GeoVisual Interface* Development

A prototype of our *GeoVisual Interface* has been developed [7]. Due to portability and flexibility issues, the prototype has been implemented in the JAVA language (JDK 1.1.6 / JFC Swing). The overall system (the complete VQS) is actually under implementation. The prototype presents our visual query mechanism and the basic functionality of our spatial query interface.

When the user starts working, a database application is selected and its schema is activated. This schema is related to a particular application and refers to a single geographic region. Following this, the user can browse geographic data, navigate at schema level or formulate a visual query.

Figure 3 presents the main window of the *GeoVisual Interface*. It contains three basic areas (i.e. the visualization, query and message areas), the pictorial representations of spatial operators and metaphors and principal functions used to explore geographic data, represented as icons too.

More specifically, each area has a special purpose: (i) *visualization area* is used for exploration of geographic data objects or selection of geographic data for query formulation, (ii)
query area is used for query composition, and (iii) message area is used to show tips and error messages. Another important area (that is optionally visualized) is called conceptual schema area which allows users to have access to the application database schema.

Another important concept presented in the screen is the use of metaphors which represent entity categories and are manipulated by the GeoVisual Interface. In other words, three geographic entity types are provided: point, line and polygon. Each metaphor provides access to the list of point, line and polygon instances found in the active geographic application. Then, users can select a geographic object through this list for query composition.

From the functional point of view, our GeoVisual Interface current release provides:

a) Exploration of Geographic Objects: Geographic objects belong to data coverages (or overlays), and consequently, to a database schema that is activated by using the interface. The exploration of geographic objects means that objects are shown in the visualization area and can be selected for getting information about their descriptive attributes, for visualization operations (zoom, pan) or for spatial queries. GeoVisual Interface also allows users to activate or disable the overlays (layers).

b) Access to the Application Conceptual Schema, using the diagrammatic visual formalism Mgeo+/UML[9,14].
c) Spatial Query Formulation: Query editor is composed of two areas called the query area and the spatial operators area. According to the iconic representation approach, a set of icons representing all the available spatial operators is displayed. Thus, the process of building a query is dynamically visualized on the screen query area and involves the following steps: (i) the geographic objects of interest are selected and dragged to the query area (ii) spatial operators are selected and (iii) the query is validated and executed.

d) Hints and help messages during the execution of each user task.

e) GeoVisual Interface layout designed according to VQS approach and to ergonomic design principles: use of representative icons and metaphors; use of balance in determining the proper sizes for both the basic areas and tool bars; asymmetric composition and colour contrast; position, size and colour repetition; use of grey colour as it has already been a pattern; and a careful design of the icons drawing.

4.2.1 Example

We show our GeoVisual Interface “look and feel” by means of an example of a geographic application about the State of Paraiba, which was called PBGeo. This application aims to model the municipal limits of the State of Paraiba through its map (scale: 1:1,000,000).

First, PBGeo conceptual modelling was done, using MGeo+ [9] and UML notations [14]. This conceptual schema may be visualized in GeoVisual Interface as presented in figure 4.

![Figure 4 – Access to application conceptual model](image)

When formulating a spatial query, users may choose a geographic object to be a query operand from three different areas:
- From the Visualization Area: when a geographic object of an active layer is selected, it is represented as an icon and dragged to the query area as a query operand. For example, if the user selects a city (e.g. João Pessoa) from the PBGeo Map, it is highlighted and may be dragged to the query area.
- From the Visual Schema Area: through this window, the users have a global view of the classes and their relationships which are present in the geographic application. When selecting a class, users may be provided with its list of instances. Thus they can choose the complete class or one of its instances and drag it to the query area as a query operand.
- From the geographic category metaphor area: after the user has selected a metaphor, a list of all instances of this category is displayed. Then, he/she may select one of them and drag it to the query area as a query operand. For example, when choosing the polygon metaphor, the user is provided with all cities found in the PBGeo application. Then, he/she may select one to take part in the query formulation.

Once the user has selected the first query operand and it has been dragged to the query area, he/she selects the spatial operator. If it is an unary operation, the query may be validated. However, if it is a binary operation, another geographic object will be selected. Figure 5 shows a query composition where the user has selected a geographic object from the visualization area, together with the adjacency spatial operator and another operand by using the polygon metaphor.

**Figure 5 – Visual Query Formulation**

In addition to formulating queries, users are able to explore geographic data and execute some GIS basic functions, such as: zoom in, zoom out and pan. The GeoVisual Interface allows users to
activate or to disable the application layers and to obtain any information about the descriptive
attributes of a geographic entity. An illustration of these operations is presented in figure 6.

![Image of GeoVisual Interface](image)

**Figure 6 – An illustration of zoom in, zoom out, layers activation and descriptive attributes interrogation**

While users are interacting with the system prototype, tips and help messages are shown in
the message area. This area is always displayed on the screen and it aims to help novice users by
providing them with friendly messages. These aims to report possible problems, to prevent errors
from happening or to guide users in each step of a possible solution, without letting them afraid of
using the interface. An example of message displayed by the system prototype is given in figure 5,
at the bottom of the picture.

### 4.2.2 Further implementation

Although we have made some progress towards an implementation of a visual query interface
for GIS applications, there are a number of functionalities that need further research in the future.
These functionalities complement the results presented in this paper. The GeoVisual Interface will
provide a thematic mapping creation resource, with the objective of generating thematic maps
according to ranges, conditions or values related to objects attributes. This facility can be found in
some current GIS interfaces and is a very important capability.
The GeoVisual Interface will also provide a query reusability tool. Each query result should be stored in a historical list and represented as an icon. When a previously executed query is needed, it should be dragged from this list and then, used as an element of a new query composition. When this query is no longer in use, it may be removed from the historical list by the user. At the end of each session, the user should be able to save these kind of queries in the database for later use.

Furthermore, our prototype implementation has not addressed the presentation of query results, as it still depends on complete VQS development and on definitions from future work [22].

### 4.3 GeoVisual Interface Usability Testing

Designers can become so entranced with their creations that they may fail to evaluate those objects adequately [4]. Motivated by this sentence, we have done a usability testing to check user satisfaction and the GeoVisual Interface friendliness and capabilities.

Our evaluation process took into account the following steps:

1. Our expert reviewers were GIS professionals. Some of them work with databases and query languages, but most of them are novices in computer terminology (e.g. Engineers and Geographers).
2. The considered development level was the initial, thus it was very important to get suggestions.
3. Each expert reviewer received a letter providing the GeoVisual Interface guidelines and a basic understanding of the task domain. It was also considered the importance of the interface friendliness, initial interaction, learning time and query capability in our evaluation process.
4. Each expert reviewer had a time to learn and to use the interface. Then, they answered a specific questionnaire developed for this evaluation, and based on [18].

The answer, comments and suggestions collected in our experiment can be summarised as follows: “GeoVisual Interface is very friendly, presenting a good initial impression and layout, with a reduced learning time. Metaphors representation are good, but must be improved, through the use of actual GIS patterns. Messages are very important and useful. GeoVisual Interface has been considered adequate for spatial queries, with a very simple visual query mechanism. It must include: more exploratory functionality (such total zoom, window zoom and map scale), the use of “polygon” instead of “region” terminology, the schema area as an optional visualization.”

More details about GeoVisual Interface and its evaluation (including the designed questionnaire and evaluation results) are presented in [7].

### 5. Conclusion

This work is an attempt to put an easy-to-use visual form in the task of formulating geographic queries on GIS. Aspects related to Geographic Databases query interface design have been analysed, in order to provide the basic concepts for the development of the GeoVisual Interface prototype.
GeoVisual Interface intends to provide a simple, intuitive and didactic environment to formulate visual queries in Geographic Information Systems. It is actually the front end module of a visual query system which is being developed [22].

It is worthwhile to note that during the GeoVisual Interface design, we did a comparative analysis on some existing GIS interfaces, with the purpose of evaluating their characteristics, behavior, easiness of use, visualization power of geographic data and query language. This analysis took into account some of the presently available commercial interfaces, such as ARCVIEW [2] and MAPINFO [17], as well as interfaces developed by the academic community, such as SPRING [21] and CIGALES[5]. In particular, it also compared these interfaces with our GeoVisual Interface presented in this paper.

GeoVisual Interface has the following advantages over other GIS interfaces:

- It does not require that users have previous knowledge about the underlying database schema or query language;
- It gives the user a closer view of spatial reality where he is able to work;
- It supports a predefined set of spatial operators that improve GIS query capability;
- It allows users to pose queries by a visual paradigm, helped by message tips that cover all necessary tasks.

Future work aims to extend the GeoVisual Interface functionality by providing: a thematic mapping capability, a query reusability tool, a complete help and message mechanism, multiple visual representations and a complete integration with the other modules of the VQS.

References