

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Semi-Automatic Semantic Data Classification Expert System to Produce Thematic Maps

Luciene Stamato Delazari,
André Luiz Alencar de Mendonça,
João Vitor Meza Bravo, Mônica Cristina de Castro,
Pâmela Andressa Lunelli,
Marcio Augusto Reolon Schmidt and
Maria Engracinda dos Santos Ferreira

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/51848>

1. Introduction

This chapter presents an expert system, designed to classify semantic information in a geographic database, aiming to assist non-expert map-makers. Despite the fact that GIS science has been discussing how to deal with ordinary users and their relationship with map production, especially due to the popularization of GIS (*Geographic Information Systems*) software and webmapping technologies, there are still issues concerning map production and its quality. Some of these issues are related to data classification methods, knowledge about levels of measurement and, thus, to map symbolization itself. In Brazil, this subject can be of special interest to municipality and state government departments, NGOs and institutions which use maps for planning and for decision-making support. At least in part, problems seem to occur because of the ease of GIS use, together with employees' lack of education in cartography. In this context, an expert system seems to be a proper choice to ensure that ordinary users can take correct decisions in the map-making process.

Specifically in thematic map production, there are potential ways to ensure that correct choices will be suggested for users, and these range from long-term training to artificial intelligence techniques. In this context, Schmidt & Delazari [1] developed an expert system to classify data, comparing text information on class names with a file that contains a word

classification, called the “system dictionary”. Originally, this software was built to assist Social Assistance Department users, from Parana state in Brazil, in their activities of social data insertion and classification. Currently, this system has evolved to a web environment and is publicly available, which reinforces the need for automatic assistance in order to avoid mistakes that could impair the data analysis and, consequently, the decision-making based on this analytical process.

This chapter is divided into three main sections. The first addresses the motivations for creation of the expert system, considering thematic maps generation and issues related to non-specialist users and uses. Topics discussed in this section are: ways of providing user assistance in data classification and map symbolization according to map design rules; how users can achieve reasonable understanding of GIS and spatial data as trained users, for correct use of geospatial tools; and, since there is no unique and permanent set of rules for thematic map-making, what are the main aspects to consider when developing user assistance for building good maps.

The second section describes the expert system's theoretical proposition, regarding users and the data to be mapped, and examples of rules to establish and implement in order to achieve proper data classification results. The use of IF-THEN rules for this case study is a noteworthy project element, being initially defined as a stationary software code to support recognition of database entries. The algorithm proceeds with an evaluation of the level of measurement that best suits the data representation process. This data is then classified and stored in the knowledge base. When the total amount of classes are stored, rules indicate the most suitable color ramps, among those available, in order to match the data characteristics. To insert new information, the expert system automatically examines and tests the data type; numerical data are classified to a numerical level of measurement; nominal and ordinal data are classified according to the knowledge base and the system dictionary. When using non-numerical data – semantic classification – the level of measurement choice is more complex due to its subjective nature and requires greater attention.

Lastly, the third part will enclose the project overview demonstrating the code development for the current web environment paradigm and what could be the potential new improved functionality of this system, developed to assist users in building social maps. The results are presented together with a discussion about general aspects of system architecture, interface design and the expert system itself, with a functional point of view, in particular, on how a system can guide users in such activities.

2. Background

Originally, the objective of Social Atlas was to support the *Secretaria de Estado do Trabalho, Emprego e Promoção Social* (SETP). In Parana state, in Brazil, this bureau defines social assistance policies and their execution, besides acting as a government manager, defining the allocation of financial resources. SETP technicians, in 2000, needed to know how counties were organized in terms of Municipal Councils and Public Funds, in order to implant social

laws recently approved at that time. For this reason, Delazari [2] started to work on an electronic Atlas prototype, called Social Atlas. The objective of Social Atlas was to be a tool for carrying out spatial analysis and generating maps, by means of user interaction. This was based on the needs of this bureau, specifically under the context of LOAS (Portuguese acronym for Organic law for social assistance) (Schmidt & Delazari [1]).

The original system users were social scientists who had little or no knowledge of cartography or of any other methods to manipulate or represent spatial data. The proposed system must lead users by map generation tasks, avoiding any mistakes which might impair analysis. Since research data includes nominal and ordinal information collected for each county in Paraná, it is mandatory to implement functions that make it possible to choose between different options for data representation and also give users the possibility of using different data classification methods.

In the visual analysis of geographic information, acquiring knowledge is possible if graphic solutions defined for each map provide efficient visualization of the characteristics of geographic phenomena. Graphic solutions may represent the spatial phenomenon behavior, and emphasize important characteristics for each analysis moment. As stated by Fairbairn [3], “maybe the most important change in mapping, in the past ten years, was the appearance of a user who is also a map producer”. Regardless of the fact that producing map knowledge still seems to be a cartographer's responsibility, it is impossible to expect that every map built will have a cartographer as part of its genesis.

The main issue in this context is how to enable ordinary users to produce good quality maps which respect cartographic design principles. In other words, map software should offer a set of tools to guide the choice of all the steps in the map production process. There are two possible options: first, to use tutorials which guide users through the stages of map creation (Yufen [4]), or second, by means of expert systems, which automate basic decisions about the mapping process (Wang & Ormeling; Artimo; Su; Zhan & Buttenfield [5-8]).

The choice of an expert system was influenced by factors related to facility of development, knowledge about software, diversity of functions to be implemented and availability of *software* resources. Among other minor differences between expert systems (ES) and conventional systems, it is possible to rely on the ability of an ES to simulate human reasoning, inference and judgment, and derive conclusions and heuristics based on a specific domain of knowledge. This means an ES is a computer software that operates with symbolic objects (symbols), and relationships between objects (Chee & Power [9]), while conventional software generate results through algorithms, which manipulate numbers and character. According to Hemingway et. al. [10], the structure of an expert system has significant advantages over the traditional software, because once the information is correctly inserted into the knowledge base, this may be updated, modified and supplemented. In a general way, an expert system can be conceived as a four-module system that acts as an information manager. These modules include a user interface, a set of rules, a knowledge base and the inference motor (Figure 1)

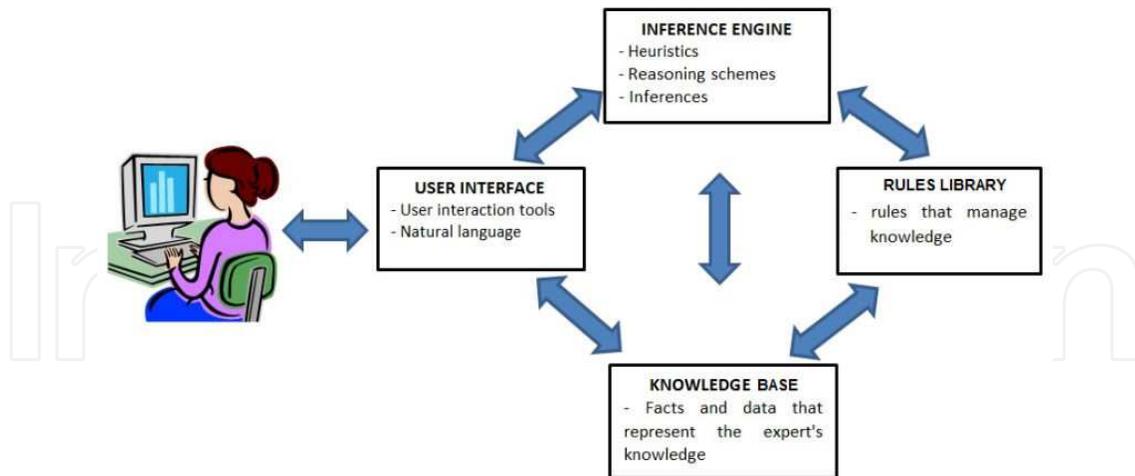


Figure 1. Basic structure of an expert system; Source: Adapted from Mendes [11].

The inference engine is an essential element for an expert system, since it works as the engine control that evaluates and applies rules. In the process of problem-solving, these rules must be in accordance with the information existing in the working memory (Araki[12]). According to Russell & Norvig [13], automated inference engines can be grouped into four categories: theorem proofs and logic programming languages; production systems; "frame" systems or semantic networks; and descriptive logic systems. The inference engine uses forward chaining, a method which seeks to validate the assumptions in the rules and to complete the actions (consequences), not only as a logical conclusion. The intermediate results are validated as assumptions and deduced conclusions are stored in a working memory (Russell & Norvig [13]).

The rules library and the working memory form the so-called 'knowledge base', representing the knowledge captured from a human expert on the problem domain. When an issue is submitted to the system evaluation, this rules library interacts with the user and the inference engine, allowing identification of the problem, possible solutions for it and the whole process that leads to conclusions. Much of the effort to develop an expert system relies on the elicitation of knowledge, i.e., how to capture and use the human knowledge in a computer application.

Rule-based systems are feasible for problems in which the solution process can be written in the form of 'IF-THEN' rules and for which the problem has no easy solution. According to Araki [12], when a system based on rules is created, it is necessary to consider the following:

- A set of facts to represent the memory of the initial work. This can be any relevant information related to the system's initial state;
- A set of rules, a library built to deal with the set of facts. This should include any action that should be within the scope of the problem;
- A condition stipulating that a solution was found or that no solution exists.

The set of facts describes the relevant characteristics of the phenomena, in the expert's point of view. Sometimes, even the expert does not realize all the features that he/she uses to make a decision. At this step, qualitative research tools such as questionnaires and interviews help to identify nuances involved in the particular decision-making process by the human expert, and, from them, it is possible to select key facts in order to build the initial working memory.

To define a set of rules, structures can be designed using the IF-THEN <condition> <action>, where:

<condition>, calls a conditional proposition. This condition provides a test the outcome of which depends on the current state of the knowledge basis. Typically, it is possible to test the presence or absence of certain information;

<action> performs some action, defined in a rule, and may even change the current state of the knowledge base, adding, modifying or removing units which are present in the knowledge basis.

Using an IF-THEN structure will cause the system to examine the condition of all ES rules and determine a subset of rules whose conditions are satisfied by the analysis of the working memory. The choice of the rule to be triggered is based on a strategy of conflict resolution. When the rule is triggered, actions specified in the THEN clause are carried out. These actions can modify the working memory, the rules library, or another specification included by the system programmer. The loop of rules is then triggered and actions will continue until there are no more conditions to be met or there is an action that terminates the program flow.

In cartography, the use of expert systems can have a wide field of applications. Understanding the basic concepts of data classification, level of measurement and visual variables in the process of map design is a major problem for casual GIS users. It is not unusual to find maps with continuous color ramps representing discrete data, map projection problems, complex symbols with no important information, and "noisy" visualization, facts that make interpretation almost impossible (Schmidt & Delazari[1]).

According to Schmidt & Delazari [1] there is a need to develop research on how to assist users in designing maps with GIS tools. The scientific literature presents some study cases on software and specific use and user issues. For automatic visualization, other researchers (Casner; Roth et al.; Senay & Ignatus [14 - 16]) investigated how to eliminate the need to specify, design and develop different visualizations for GIS software outputs, allowing users to focus their attention on determining and describing the information to be represented. Other initiatives are CommonGIS (Fraunhofer [17]) and GeodaTM (Anselin et al[18]). Those systems focus on HCI – Human Computer Interaction - through an interactive training assistance and EDA (Exploratory Data Analysis) assistance tool. CommonGIS adapts the interface as users explore the tool and acquire knowledge about the system. The exploration is guided by an expert system giving users hints and options. GeodaTM emphasizes data mining and spatial statistics, and includes functionality ranging from simple mapping to explor-

atory data analysis, using an interactive environment and combining maps with statistical graphics (Anselin et al. [18]).

Yet according to Schmidt & Delazari [1], if casual users do not understand map design concepts properly, it is important to determine how to help them to classify attributes and symbolize maps according to the principles of cartography. At the same time, it seems to be essential for cartographers to ensure that these users will achieve a minimum level of understanding of the correct use of GIS. If there are no map design principles for the digital environment, mainly because cartographers do not know exactly what can be adopted from traditional map design theory, what should be taken into account to make these principles feasible for ordinary users? In this context, the expert system application seems to be a plausible solution, regarding the characteristics discussed above.

2.1. The Social Atlas expert system

The expert system was built as an automated information manager placed between the database and the representation device, implemented with MapObjects (ESRI) (Figure 2). The ES controls the data flow in two situations: insertion of new data and carrying out SQL queries. When new information is added, the system goes to the knowledge base in order to try to find a similar configuration, concerning class names. If not all can be found, the rules library breaks it down into isolated words to try to find any kind of order in the dictionary. The same procedure occurs when an SQL is inserted in the system but, in this case, the information existing in the knowledge base is filtered by the rules and presented on a thematic map.

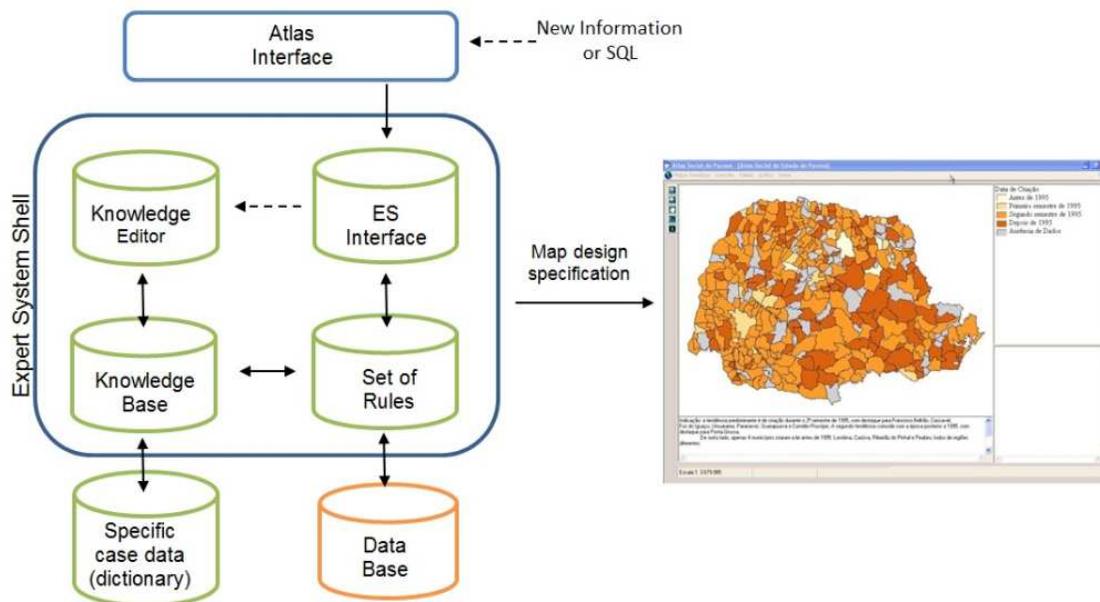


Figure 2. Expert system information flow

Initial memory, or initial facts, is the name given to information located in the database. This is collected by technicians of the SETP bureau for implementation of LOAS. Data is organ-

ized into three themes, or major groups, which separate information in terms of its characteristics. There are 26 different types of information in these three themes. Each one has its own classification and number of classes defined by the original map design from Delazari [2] and its condition is related to defining the data's level of measurement and knowledge base rank.

The rules library, unlike other expert systems, is embedded in the software, and the parameters of the rules are updated by use, and can be accessed by the user. The set of rules tries to identify which kind of data has been inserted. Through application of production rules, described as IF-THEN, the expert system tests the type of data to insert new information. Numerical data is classified by its numerical level. Social Atlas does not distinguish between the interval or ratio level of the measurement, because, in this case, the map design does not consider it (Schmidt & Delazari [1]).

Nominal and ordinal data can be stored in the same knowledge base but the rules to deal with them are quite different and their functioning is based on ordering elements from the knowledge base. Therefore, any feature that indicates order, associated with class names, has to be searched for in the knowledge base. When dealing with semantic classification, the choice between one or another level of measurement demands attention, because correct order is a subjective concept. Data indicating temporal or any kind of order can be considered as ordinal data and needs to be evaluated in detail. For example, for LOAS implementation it is important to know County Council's creation data. Classes are "first semester of 1995", "before 1995", "second semester of 1995", "after 1995" and "no available data". The expert system searches the whole class, e.g. first semester of 1995, for possible ordering of categories (Schmidt & Delazari [1]).

If it is not possible to define them, category names are broken into a list of words, using a 'word-wrap' function. The position of each word in the sentence is stored as an index of words. Then the list of words is compared to the dictionary and the ES tries to classify the first word of each class, and then the second word of each class, and so on, trying to establish a hierarchical relationship between class names. This dictionary keeps words in a more generic sense, giving local and global ordering based on the index order of the words that compound the class names. Inside the dictionary there are words and prepositions like "until", "before", "between", "among" and "after", and they also work as a specific working memory, keeping all the words used for classification.

The dictionary functions as a full resource for carrying out efficient classifications. In this context, if a previous classification has been deleted and a new one with a similar name is inserted, the system is able to estimate a possible order for the new classification. The stopping condition, in this case, is defined if an order is or is not associated with the class names, or words associated with them. In this way, when ordering is found, for all categories, the rule library will choose the required visual schemes (color ramps, in the case of choropleth maps) to represent geographic phenomena. Also, the knowledge base must assemble the new classification.

However, as the system can be relatively weak in the early stages of professional use due to the uncertainty of the initial memory and dictionary, some cases of failure or partial success may occur. In these cases, the system asks the user if the order is correct. Then the system stores the user classification, feeds the dictionary and carries out the map symbolization. As long as specialists keep supplying the knowledge base with more information, the vocabulary becomes more extensive and the system can deal with complex situations. Also, user confirmation becomes unnecessary. Thus, the system becomes a more powerful tool, especially when experienced technicians build the knowledge base and dictionary, and distribute them along with the Social Atlas.

As a last point about the ES, there is a special level of access to allow users to edit the knowledge base and database in general. Different modifications can be made to the database and this will modify the final representation aspect, and also change the knowledge base or even the dictionary. The first is a common task performed in order to update available data in the database. In this case, any information deleted or inserted will pass through the Expert System. This step is necessary to update the knowledge base and the dictionary, and to keep the ES and the database synchronized.

The interaction with the ES occurs inside the Social Atlas interface (Figure 3). The dialog box is accessed from the Edition of Social Atlas menu. All other steps of the expert system run under this dialog box and users do not come into direct contact with the data or its classification. Themes are shown in the Themes Dialog box. Data, i.e., column name, is supplied by the Class Dialog box on the left. In this dialog box the class names are supplied and appear at the right hand side. Clicking on the 'confirm' button makes the system carry on with classification, as described previously. In the event of failure, the right hand side buttons are enabled and a message box pops up asking the user for intervention. Users sometimes require additional information storage as text. This action can be done in the 'Additional Information' field at the bottom of the dialog box.

3. New interface design and code development

For cartographers and professional mapmakers, it seems to be hard to think about code development and its relationship to the map production process itself. There are several varieties of GIS software which can help to store and process spatial data to produce high quality geographic representations. However, it cannot be denied that web environments are changing the way map use and users are understood and considered in cartographic activities. Since the web is the lair of interactivity, one mandatory issue is about the way in which casual and ordinary users rely on geographic data to produce thematic maps. Also, there is a major issue about how cartographers can act in this environment to ensure that these users are able to rely on these self-produced maps to take decisions and to analyze geographic phenomena efficiently.

Web applications, just like offline software, are dependent on programming languages and, besides the fact that websites are usually easier to design and to get working, be-

cause of common server specifications and widely known browser architecture (which includes client-side features that are constantly evolving), there are some critical issues in developing map applications for the web. Thus, if a cartographer wants to help any user on the world-wide web to make good maps by means of developing an automatic or expert system for it, there is a need to first understand and analyze how web architecture can be handled. This section presents a potential way to figure out this issue, by means of presenting the “*how it works*” on the current “Atlas Social do Paraná” (Delazari [2]) version, presented on <http://www.cartografia.ufpr.br/atlas/english>¹. Since the development process was carried out exclusively by cartographers and not by system analysts, perhaps the proposed solution is not as elegant as it could be, but discussions raised by it can be useful for interactive map designers and are currently defined as the main focus of this implementation.

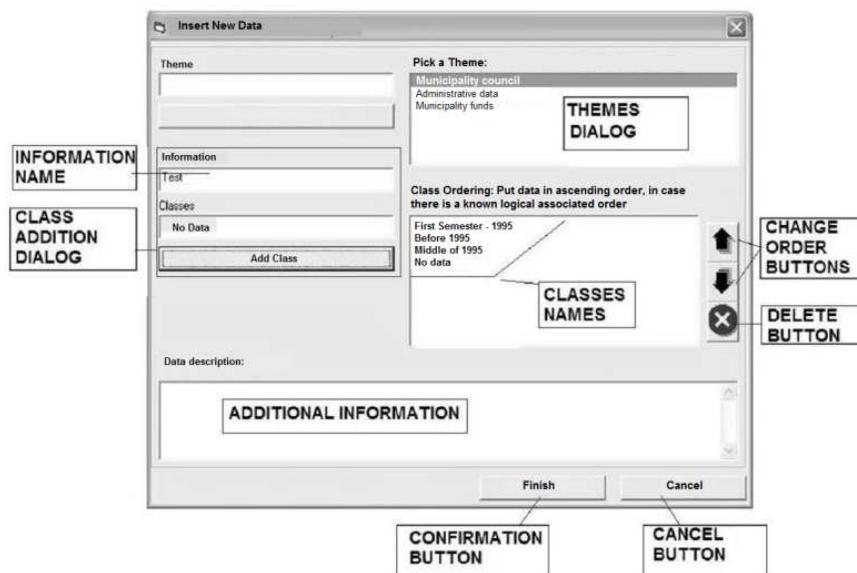


Figure 3. Original expert system interface; Source: Schmidt & Delazari [1]

Describing how an automatic system is developed can serve as the starting point for many related projects. The case study presented here is on the adoption of an automatic system inside the already existing “Atlas Social do Paraná”. As previously described, the Atlas comprises a huge amount of social data for the last two decades, which makes it a powerful tool, not only for government planners or public administrators, but also for ordinary citizens, all of them possibly using the atlas to gain a full understanding of social perspectives of Paraná state in Brazil.

Since the Atlas was developed initially as an offline product, the web version has to manage the following issues:

¹ All php codes are available for download in the same page.

- a. How to make the Atlas database easy to update and query by data producers who are not experts in either cartography or informatics;
- b. How to make the Atlas usable in the web, combining cartographic aspects and interface design in order to best suit the audience needs;
- c. How to make the Atlas structure serviceable to provide the user with the ability to produce maps on demand, using his own data and preserving the representation quality.

System functionality	Choosing thematic data (U,S)
	Choosing area units for mapping (U,S)
	Choosing level of measurement (S)
	Choosing method for data classification (S)
	Choosing number of classes (U,S)
	Choosing color ramp (U,S)
	Storing user's choices (S)
	Displaying thematic data table with statistics (U,S)
	Printing maps module (U, S)
	Searching location by text (U,S)
	Uploading own data to build maps (U,S)
	Finding address (U,S)
	Where am I (U,S)
	Webmap interface functionality
Next and back zoom buttons (U,S)	
Querying by click (U,S)	
Legend support (U,S)	
Scale (U,S)	
Latitude and longitude location (U,S)	
Measuring areas and distances (U,S)	

Table 1. Interface functionality - 'U' indicates its presence on User version and 'S' indicates its presence on Specialist version

The expert system described in the last section also had to be redesigned in order to deal with new database organization and its interface was rebuilt to consider the use of two groups of web users: specialists and general users. In the first group are the users who will build the expert database by choosing variables to construct their maps and evaluate them. The second group is those who will take advantage of this database to build their own maps, whether using data from the Atlas or using any other spatial data with attributes. Al-

so, the system interface needed to be redrawn in order to consider the browsers' actual style of navigation and the limitation of using only one window and less than 90% of the display area for map application in general.

Many of the answers to the issues presented were discussed during the process of implementing the Atlas. However, the interface design and website functionality were the first set of decisions to be taken, and guided further design on the database, server architecture and web services, used to make spatial data representations available on the internet. The list of interface functionalities can be divided into two steps: the system functionality itself and the webmap interface (Table 1). The initial effort on this new version was designed to build only choropleth maps, but its structure would deal with other mapping techniques, on demand.

3.1. Database and server structure

The first step in designing the current database consisted of defining the former entities and relationships, implemented by DBMS PostgreSQL. The existing data structure – considering spatial data and attributes data to be represented in maps - was first considered from the point of view of common GIS software architecture: producing feature data by surveying, transforming it on spatial files and joining relational tables with area units to produce map symbology on specific themes. Problems arise when there are changes to spatial data, such as when a new municipality or new administration area is created, or when data producers need to load new data or to rectify existing data, since using a form specially built for updating data is required, apart from performance and software compiling issues.

Thus, the next step was to define the database structure for spatial data. The new version of the atlas comprises the introduction of a spatial database paradigm, with spatial data organized as tables with associated geometry information and foreign keys, in order to be related to any thematic data to be added to the database. The use of the *Postgis* spatial extension, *PostgreSQL* support for spatial data, was mandatory in the design of the database structure. Spatial data were built separately from attribute data and divided into types of area units. Since official data from the Atlas has to be from Paraná state only, there are only two spatial subsets: one for municipalities and associated data, and the other for census sectors, official area units from the Brazilian official census. One advantage of the use of this kind of database organization is that associated data like regions, micro-regions and macro-regions can be used as area units (Figure 4), using dissolve operations by means of SQL queries² and using names as attribute fields on joining attributes with spatial area units. By means of using the same Brazilian official geocode for census sectors and for municipalities, it is possible to guarantee that joining the attribute information will be an effective choice for updating the database and maintaining consistency between spatial and attribute data.

The system architecture's first criterion was the use of open source and free software where possible. It was also decided that the spatial database must be available remotely, together with the instance for the spatial data server and the web server itself, in the same physical server. Based on this, the architecture (Figure 5) was defined with an *Apache* and *Tomcat*

² Select ST_UNION(the_geom) ... group by 'mesoregion'

webserver, along with instances of *Geoserver* as the mapserver, *PostgreSQL* as the DBMS, PHP to process server-side data, to produce XML (SLD) symbology and to query database and Javascript libraries (*jquery*, *Ext* and *GeoExt*, *Openlayers*) to deal with client-side functionality, like div display, map zoom and map legends. Last, *Curl* was used to establish the communication between PHP and *Geoserver* Rest API, which facilitates the transaction between web servers, making possible to perform map server administration tasks remotely.

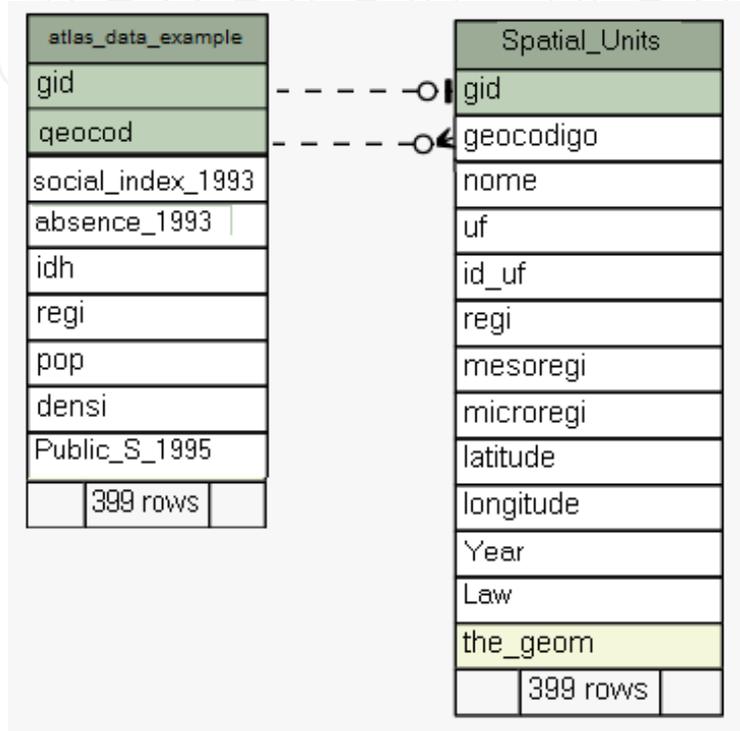


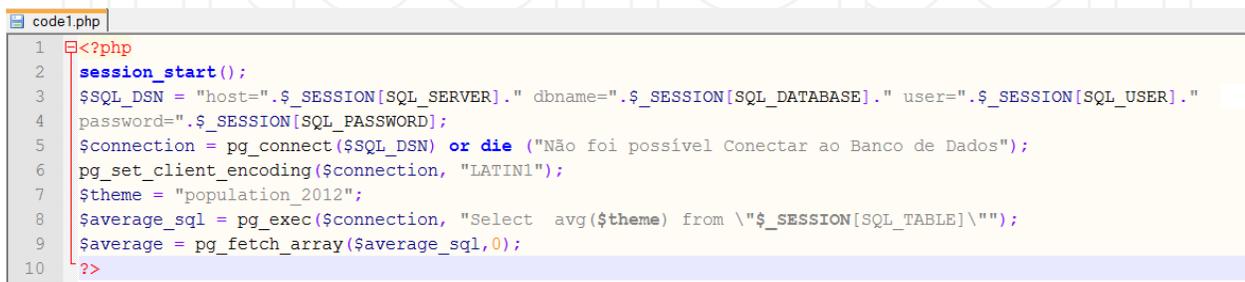
Figure 4. Example of database table's relationship for spatial features and attributes

The PHP language is used to process server-side data, being necessary to the expert system code, since it deals with the database access, together with the creation of rules to create symbols based on users' input. The PHP code is part of the web page code itself, since a ".php" file can also handle *html* and *javascript* code and is a well-documented and easy-to-use language, which offsets its lack of consistency and predictability. PHP connects the database using the *pgsql* extension, and a piece of code (Figure 6) can connect the database, set the default encoding to prevent from incorrectly displaying accents and make a simple query to return the average for some column, being the resultant row stored in a vector PHP array. For security reasons, it seems important to maintain the connection information (host, name of database, user and password) in a separate file, which must be included³ in the main PHP file.

Another server-side task that must be taken into account involves publishing spatial data on the web. To accomplish this, the server must be an OGC (Open Geospatial Consortium)

³ Include `names_and_passwords_file.php`; //line included in the beginning of PHP code

specs compliant, capable of putting data on the internet throughout web services. The chosen one was the *Geoserver* mapserver, a *Java*-based tool to publish spatial data, which was installed using a war file, and configured to optimize performance in a production environment [19]. Since there is no need for vector analysis or satellite imagery, the choice was to publish data using WMS requisitions in the map server. The use of this web service displays a set of image tiles to frame the spatial data and its symbology in the user's client. Also, it uses a simple set of *xml* rules to construct symbols that can be used to get on-demand data classification and symbols choice.



```
code1.php
1  <?php
2  session_start();
3  $SQL_DSN = "host=".$_SESSION[SQL_SERVER]. " dbname=".$_SESSION[SQL_DATABASE]. " user=".$_SESSION[SQL_USER]. "
4  password=".$_SESSION[SQL_PASSWORD];
5  $connection = pg_connect($SQL_DSN or die ("Não foi possível Conectar ao Banco de Dados");
6  pg_set_client_encoding($connection, "LATIN1");
7  $theme = "population_2012";
8  $average_sql = pg_exec($connection, "Select avg($theme) from \"$_SESSION[SQL_TABLE]\"");
9  $average = pg_fetch_array($average_sql,0);
10 ?>
```

Figure 5. Piece of PHP code showing the connection and simple query to the database

3.2. On-demand map symbology

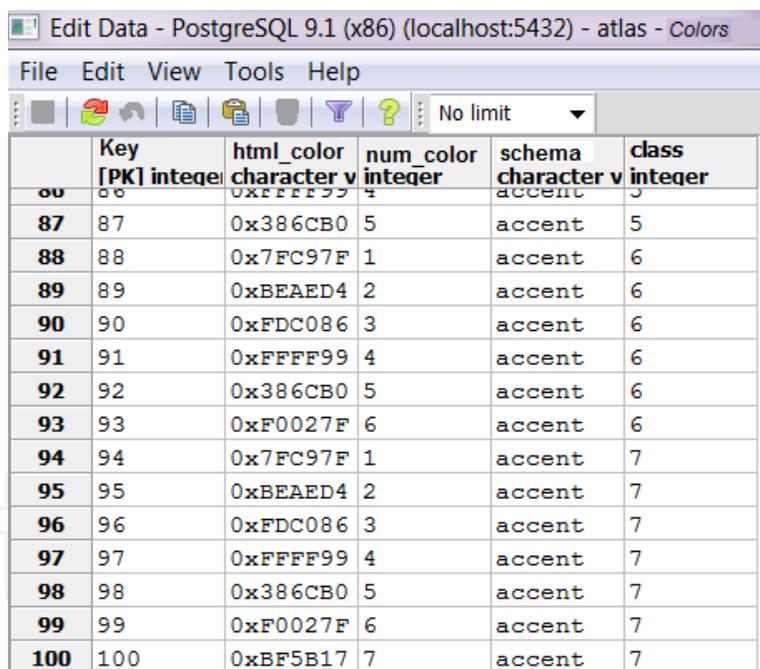
The SLD (Styled Layer Descriptor) is an XML-based file format used to transmit symbols, according to OGC specification on WMS symbology. Traditionally, this file contains parameters which are used together with features stored in a geospatial server, comprising symbolization information, such as color to be used in a point symbol or the thickness value for lines. These parameters are applied to one or more layers stored in the map server and displayed in a static environment, i.e., once generated there is no possibility to change these parameters except by generating a new SLD file for each different map. The SLD construction is often accomplished through direct insertion of the algorithm on the server, usually by means of writing it from a GIS software. The concept of on-demand map symbology is presented here as an algorithm which already possesses variables to build the SLD file to be applied to a map according to users' input, being the file stored in a server folder and read by WMS requisition.

3.2.1. SLD file structure

The file structure for SLD specification is based on rules. These mold the set of styles which symbolize a feature. In choropleth maps, for example, each rule makes it possible to generate a different color value applied to an area feature, resulting from a data classification algorithm. To apply color values to spatial data features, the system's database provides a color table (Figure 6) based on what is suggested by the "Colorbrewer" software (ColorBrewer [20]). Several possible color ramps are stored in the database and their names are then accessed by PHP code, as a list of different options for the user. The fields for the colors table are also designed to make the

process of data classification easier. For each class, there is a set of 'x' different *html* colors, 'x' being the number of classes. The field "num_color", which varies together with the "html_color" value, since the color changes as the number of class elements change, is defined from '1' – to be applied to features contained in first class – to 'x'.

To build the map and construct the SLD file, users have to choose the number of classes, the level of measurement and, if the level of measurement is numeric, the data classification method to be used. These variables are merged to *xml* declarations along the PHP file with a string concatenation operator, and are used in the process of data classification (Figure 7), which defines the final SLD file, written in the server and made a default for the WMS layer, by means of setting the parameter `<IsDefault>1</IsDefault>`. Thus, the map server just serves the WMS layer. The symbology is prepared on the server by PHP code and is accessed directly by the *openlayers* map library to build the map in the client side. It is overwritten automatically when the server executes a new classification method. The iterative process is based on users' choice built in PHP variables, accessing database views. The database can be fed to create as many classes or symbol properties as necessary for users' data needs.



	Key [PK] integer	html_color character varying	num_color integer	schema character varying	class integer
87	87	0x386CB0	5	accent	5
88	88	0x7FC97F	1	accent	6
89	89	0xBEAED4	2	accent	6
90	90	0xFDC086	3	accent	6
91	91	0xFFFF99	4	accent	6
92	92	0x386CB0	5	accent	6
93	93	0xF0027F	6	accent	6
94	94	0x7FC97F	1	accent	7
95	95	0xBEAED4	2	accent	7
96	96	0xFDC086	3	accent	7
97	97	0xFFFF99	4	accent	7
98	98	0x386CB0	5	accent	7
99	99	0xF0027F	6	accent	7
100	100	0xBF5B17	7	accent	7

Figure 6. Part of color table in the database

3.3. Functionality and interface design

User interaction with a computer system always occurs through the use of an interface. There are several issues about map-related systems and their interface, especially for web

use (Nivala [21]), since these interfaces deal with issues related both to computational interfaces and map users and use. Besides that, there are new technologies which allow data organization to be done dynamically, in a way that reduces the amount of decisions and interactions by users, when generating a cartographic representation. This can make interface use easier and also raise performance for functional approaches (de Mendonça & Delazari [22]).

```

1 <?php
2 if ($level_measurement=='numeric') //for numeric data only
3 {
4     if ($method== 'equal_int') //Equal Interval data classification method
5     {
6         $namemethod = 'Equal Interval'; //for legend purposes
7         $max_sql = pg_query($connection, "SELECT max($theme) as maximo FROM $SQL_TABLE");
8         $min_sql = pg_query($connection, "SELECT min($theme) as minimo FROM $SQL_TABLE");
9         $max = pg_fetch_array($max, 0); //maximum theme value
10        $min = pg_fetch_array($min, 0); //maximum theme value
11        $interval = (($maximo[0]-$minimo[0])/$classes);
12        $color_sql = pg_query($connection, "SELECT html_color FROM colors WHERE schema = '$schema'
13        and classes = '$classes' and num_color = '1'");
14        $color = pg_fetch_array($color, 0); //color to be used for the first class
15        $linferior = $minimo[0];
16        $lsuperior = $minimo[0] + $interval;
17        //here the Sld is formed. all variables are defined before as strings, cointaning the mandatory
18        //elements for a SLD file
19        $SLD_EI = $cabecalho . $pre . $rule_beggining . $linferior . "-" . $lsuperior . $rule_quant_1 .
20        $theme . $rule_quant_2 . $linferior . $rule_quant_3 . $theme .
21        $rule_quant_4 . $lsuperior . $rule_quant_5 .
22        $color[0] . $rule_end;
23        //for iteration, for next classes
24        for ($i=2; $i<=$classes ;$i++)
25        {
26            $j = $i-1;
27            $linferior = $min[0] + ($j*$interval);
28            $lsuperior = $min[0] + (($i)*$interval);
29            $color_sql = pg_query($connection, "SELECT html_color as cor FROM colors WHERE schema =
30            '$schema' and classes = '$classes' and num_color= '$i'");
31            $color = pg_fetch_array($color_sql, 0);
32            //using the previously sld, which has the symbol definition for the first element
33            //the rest of elements are filled in this iteration, according to the number of classes
34            $SLD_EI = $SLD_EI . $rule_beggining . $linferior . "-" . $lsuperior . $rule_quant_1 .
35            $theme . $rule_quant_2 . $linferior . $rule_quant_3 . $theme . $rule_quant_4 . $lsuperior .
36            $rule_quant_5 . $color[0] . $rule_end;
37        }
38        $SLD_EI = $SLD_EI . $closure; //finish the sld file
39        //from here, sld file is written on server
40        $mysld = "testFile.sld";
41        $fh = fopen($meusld, 'w') or die ("Cannot creat file, contact webadmin");
42        fwrite($fh, $SLD_EI);
43        fclose($fh);
44    }
45 }
46 ?>

```

Figure 7. Example for PHP code on Equal Interval Data Classification method

The interface for specialists and atlas' users was planned to be clean and to manage only essential functionality, but since there was a desire to implement an expert system, with a knowledge base fed by specialists, the decision was to make slight changes on functionality, in order to avail specialist knowledge. Thus, specialist users can identify themselves with a

login role, in order to contribute to the knowledge database on data classification and symbolization decisions. Users, then, can take advantage of specialists' choices on similar data to make their own decisions. The interface functionalities for this two group of distinct users can be divided by map-interface functions and web interface itself, accessed by login roles: “specialist” and “user”.

Inspired by the ‘openstreetmap’ project interface, the main interface for “Atlas Social do Paraná” (Figure 8) incorporated on its server a javascript map library – *openlayers* – integrated to a window and *css* management *javascript* library – *Jquery*. Both of them make possible to call predetermined functions to display maps, their symbology and hide or show functional windows on the interface, in which users can input data or make choices about the map production process in terms of forms. Making forms (only one allowed per tab) and the table associated with the chosen theme, available on the same page as the map is then considered a major decision in interface design. Based on this, a small and simple flow that expresses the common use for the website was devised to be followed by both casual users (Figure 9a) and specialists (Figure 9b).

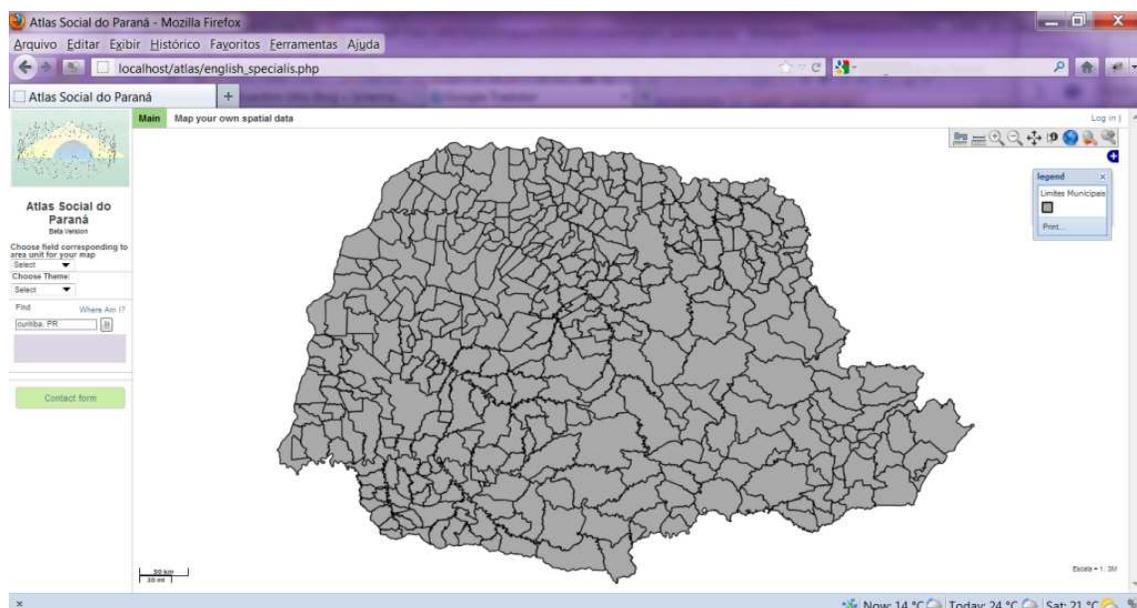


Figure 8. General Initial Atlas interface

3.4. Storing user's own data

One of the main functions of the new Atlas is the ability of reading and storing user's data in the database. Every Atlas page has the option of upload user's spatial data, by using a *html div*, powered by *jquery* (Figure 10). In order to make this possible, it's important to configure the web server to accept uploaded data. In the current Atlas architecture, it is also important to ensure that the PHP installation is able to save temporary uploaded files in a suitable server folder and to allow *Curl* to call *Geoserver Rest API* in order to write the new layer in the map server. All upload and store steps are called by PHP files, included in the main up-

load form action page. The algorithm to allow user data upload (Figure 11) considers that user's data will be available for 4 hours in the database, when a trigger is activated to clean all new inserted data. After displaying user's data, the system now will use this as default table, to be used for every subsequent query.

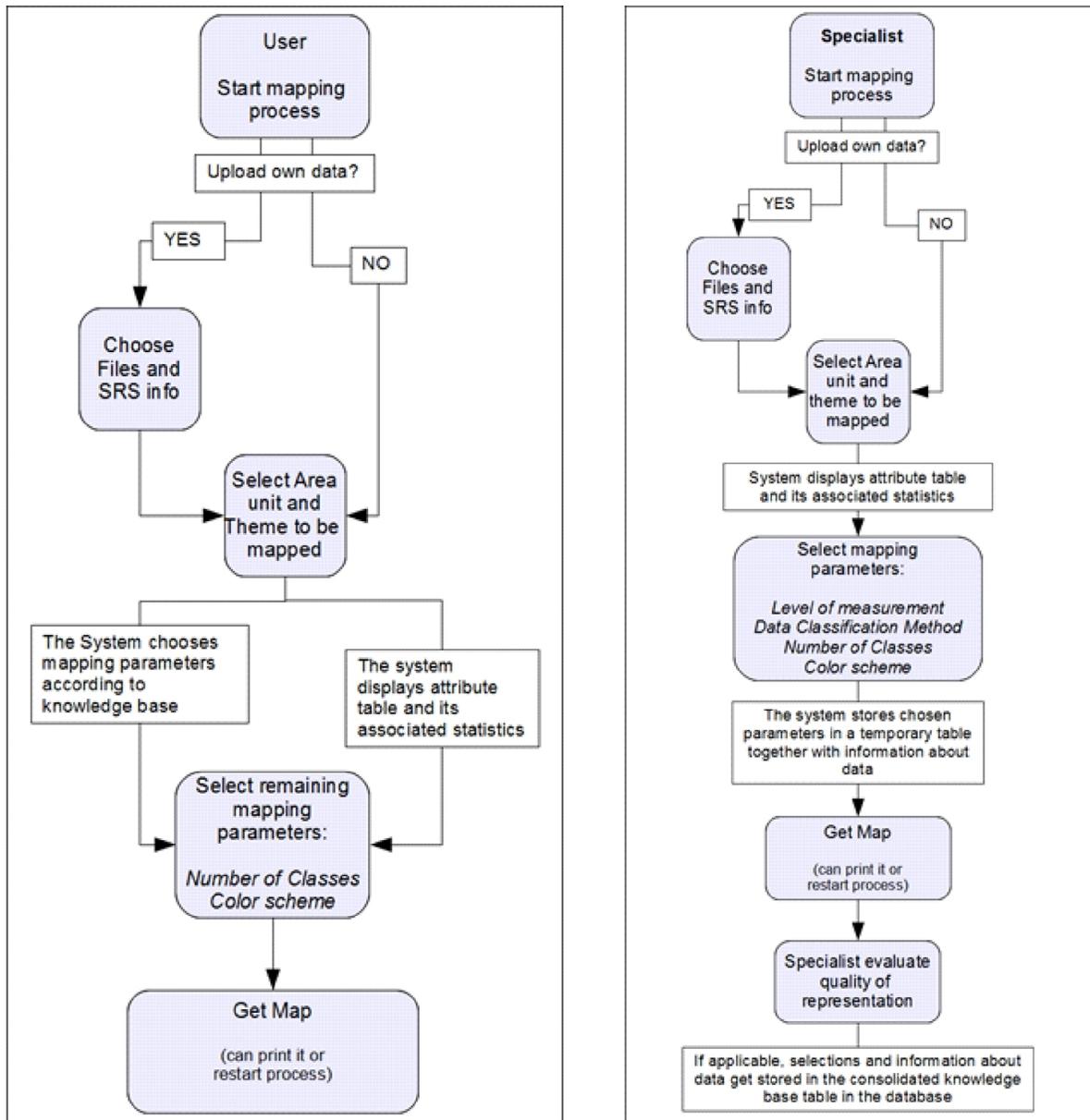


Figure 9. (a - left): Expected interface use flow; (b - right): Expected flow for specialists

3.5. Expert system functionality

The expert system comprises a PHP set of pages which access the database in order to compare specialists' decisions with measurements about the data itself. There is a table in the database

that stores every level of measurement decision, data classification method, number of classes and color ramp made by a specialist logged into the website. According to the comparison between what is chosen by them and the data characteristics, it is possible for the system to learn the occurrence of patterns. To learn, in this case, is to store these patterns in the consolidated table of specialists' decisions and using this knowledge base to ordinary users. So, it will often occur when common users upload data that is similar to the ones used by specialists' before.

The screenshot shows a web interface for uploading spatial data. At the top, there is a navigation bar with 'Main' and 'Map your own spatial data'. Below this is a 'Hide' button. The main heading is 'Upload your own shapefile', followed by a note: 'At this moment, it will work only for polygon data'. The instructions state: 'Select 3 files of the same name with 3 different extensions: SHP, DBF and SHX'. Below this, it says: 'Before sending, choose the EPSG number associated to the shapefile'. A dropdown menu is set to '4874 - Sirgas2000 lat/long'. Under 'Select files', there are three rows: 'Select one SHP file', 'Select the associated SHX file', and 'Select the associated DBF file'. Each row has a text input field and a 'Seleccionar archivo...' button. At the bottom of the form is an 'Upload' button. The background of the page shows a map of a region with a grid overlay.

Figure 10. Detail on the form used to upload *shapefiles*

In order to make this knowledge functional, some metrics are defined for any kind of data uploaded by a user or the Atlas original data. These metrics can be divided among the four possible decisions made by a specialist, plus the final evaluation of the produced map. At least three different specialists must take the same decision, based on relevant data characteristics matching exactly, and transmit the most positive feedback allowed by the system for the built map, in order for a decision to be stored into a consolidated decisions table. At first experimental build, the expert system can learn behavior related to two decisions, as follows.

3.5.1. Level of measurement

For this system, the level of measurement of data is simplified into numeric, ordinal and nominal types. To classify data in one of these three, the expert system needs first to identify the name of the column that stores what has been mapped and the distinct values found in this column. Second, it is important to identify what type of data is being classified, in terms of databases' data types. The system makes the following assumptions:

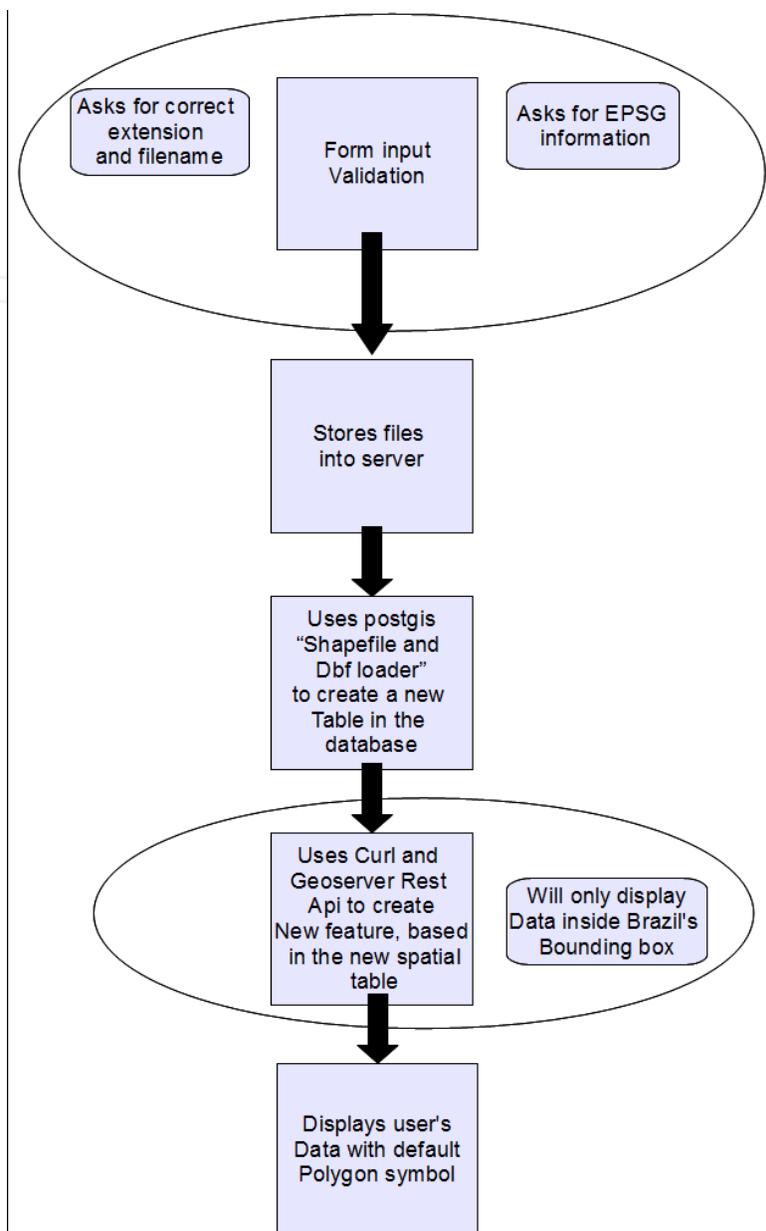


Figure 11. Schema for the algorithm that provide upload and store user's spatial data

- Numerical data are always composed of numeric, integer, float or similar formats;
- Nominal and ordinal data are always text, character varying or other similar formats;
- Numerical data can be stored as text or in a similar format, since when the data is exploded, at least 2 or more characters in the sequence match. When this case occurs, the system should be asked to convert text into numbers, in the user's own database;
- Ordinal data must be compared to the database field corresponding to the knowledge base for the ordinal text group.
- All other cases must be considered as nominal data.

3.5.2. Data classification method

For numerical data only, the expert system measures statistics for the theme data: standard deviation, variance, mean, deviance from mean (Figure 12). These are all taken in order to measure kurtosis (Dent [23]), which is a metric to measure the flatness of a distribution. The following assumptions are considered:

- When kurtosis is between 2.5 and 3.5, the distribution is normal. Equal interval and quintile methods are then suggested to be used;
- When kurtosis is below 2.5, the distribution is considered flat and a standard deviation method is recommended;
- When kurtosis is above 3.5, the distribution is full of peaks. In this case, maximum breaks or Jenks methods fit the data better.



Avg : 67,9767| Std. Deviation: 105,5881| Variance: 11148,8390

State	Pop_Density (hab/km ²)	Deviance
Acre	4.8	-63,1767
Alagoas	112.39	44,4133
Amazonas	4.68	-63,2967
Amapa	2.21	-65,7667
Bahia	24.83	-43,1467
Ceara	56.76	-11,2167
Distrito Federal	441.74	373,7633
Espirito Santo	76.23	8,2533
Goiias	17.65	-50,3267
Maranhao	19.78	-48,1967
Minas Gerais	3.36	-64,6167
Mato Grosso do Sul	6.85	-61,1267
Mato Grosso	33.4	-34,5767
Para	52.37	-15,6067
Paraiba	66.74	-1,2367
Pernambuco	6.08	-61,8967
Piaui	89.47	21,4933
Parana	12.4	-55,5767
Rio de Janeiro	366.01	298,0333
Rio Grande do Norte	60	-7,9767
Rondonia	37.96	-30,0167
Roraima	6.56	-61,4167
Rio Grande do Sul	2.01	-65,9667
Santa Catarina	65.54	-2,4367
Sergipe	94.38	26,4033
Sao Paulo	166.19	98,2133
Tocantins	4.98	-62,9967

Kurtosis : 8.2466865045365

Figure 12. Example of generated theme table and associated statistics

3.5.3. Feedback on map quality

This is simply asking the specialist: “How would you classify the quality of the representation generated?” Answers can vary from 0 to 10, and only higher grades should be assumed

as maps that should have their decisions against data to be considered as knowledge base. Using this metric will ensure that the consolidated table has only the best representations, according to specialists.

Based on results, there are important remarks about the proposed interface design and code implementation. First, user testing is part of the development process, and only doing it can ensure the interface acceptability and usability. However, the proposed framework provides clear step-by-step guidance in order to allow users to produce thematic maps. First informal pre-tests show that the interface has no usability gaps and that the expert system suggestions seems to be an desirable aid, especially for those unfamiliar with cartography and map production environments. Second, there are known limitations on this first web version. In production environments, this software must allow DBMS configuration for multiple database access. Also SLD specification needs to be improved to support more complex mapping techniques. Last, more research on which information about data could be analyzed in order to define more adequate criteria for the system's decisions.

4. Conclusion and further work

An expert system development should consider the subject particularities. In the case of maps, use and users are mandatory issues to be taken into account when designing data storage and analysis and also the way to interact with them. The presented ES can manage not only the LOAS data, users and framework, but is now designed to cover an unpredictable amount of uses, since users can upload and analyze their own data. The system's interface was also carefully discussed in order to present to users the most practical and simple way to interact with complex map design decisions. Usability pre-tests have been carried out, and current feedback is positive.

After testing this first version of the online Atlas, it is intended to develop additional functionalities in order to improve the expert system concept that has been started with this research, as well as the interface use experience. The main objective is to make this system a reference, not only for LOAS technicians but also for ordinary internet users who need to get their data symbolized according to map design expertise.

Currently, users have the option to upload their own data, both geometry and attributes. There is work in progress for the system to recognize if the mapping method is suitable for the data characteristics. One aspect that can be discussed, regarding the analysis of numeric data types, is a mapping technique chosen against relative or absolute data. Here, absolute data are considered those not related to any other data, e.g. people counting; relative data are related to any other data and can be related to area units, e.g. population density. Besides the importance of this classification, there is no formal way to discover if data are absolute or relative. A possible solution for this problem could be to ask the user a set of questions in order to verify this information. After this questionnaire, the system would suggest the most suitable method for data classification and, consequently, the choice of mapping technique.

Another issue that has to be considered is the number of classes. This first version allows the user (with no distinction between common users and specialists) to choose this parameter without restriction. However, this is an important decision that can affect map understanding and legibility, since it can mask the whole distribution, given the number of elements per class and the relationship among the theme elements. To prevent incorrect choices, it is necessary also that system suggests to users a suitable number of classes. This parameter is then decided considering the number of elements in the raw data, i.e., the number of elements in the sample and their metrics, as median or standard deviation.

Acknowledgments

This work was funded by CNPq (The National Council for Scientific and Technological Development, grant n. 306862/2011-5).

Author details

Luciene Stamato Delazari^{1*}, André Luiz Alencar de Mendonça¹, João Vitor Meza Bravo¹,
Mônica Cristina de Castro¹, Pâmela Andressa Lunelli¹,
Marcio Augusto Reolon Schmidt² and Maria Engracinda dos Santos Ferreira³

1 Federal University of Parana, Geodetic Science Program, Curitiba, Paraná, Brazil

2 Federal University of Uberlândia, Brazil

3 Federal Institute of Sergipe, Brazil

References

- [1] Expert system to classify semantic information to improve map design. In: *The World's Geo-Spatial Solutions: Proceedings of the 24th International Cartographic Conference, ICA/ICC2009, Santiago, Chile.*
- [2] Delazari, L. S. Modelagem e implementação de um Atlas Eletrônico Interativo utilizando métodos de visualização cartográfica. Ph D Thesis. Escola Politécnica da Universidade de São Paulo- Departamento de Engenharia de Transportes. São Paulo: (2004).
- [3] Fairbain, D. J. The Frontier of cartography: mapping a changing discipline. *Photogrammetric Record* (1994). , 14(84), 903-915.
- [4] Yufen, C. Visual cognition experiments on electronic maps. In: *Proceedings of the 19th International Cartographic Conference, ICA, (1999). Ottawa, Canada.*

- [5] Wang, Z., & Ormeling, F. (1996). The representation of quantitative and ordinal information. *The Cartographic Journal*, 33(2), 87-91.
- [6] Artimo, K. The bridge between cartographic and geographic information systems. In: MacEachren, A.M.; Taylor, D.R.F (ed). *Visualization in modern cartography*. Grã-Bretanha: Pergamon, (1994). , 45-61.
- [7] Su, B. A. Generalized frame for cartographic knowledge representation. In: *Proceedings of the 17th International Cartographic Conference, ICA, (1995)*. Barcelona, Spain.
- [8] Zhan, F. R., & Battenfield, B. P. Object-oriented knowledge-based symbol selection for visualizing statistical information. *International Journal of Geographic Information Systems (1995)*. , 9(3), 293-315.
- [9] Chee, W. J., Power, M. A., *Expert Systems, & Maintainability, . Reliability and Maintainability Symposium*. In: *Proceedings of IEEE. (1990)*.
- [10] Hemingway, D. E., Katzberg, J. D., & Vandenberghe, D. G. A. *Technology Management Methodology Implemented Using Expert Systems*. In: *Proceedings of Conference on Communications, Power and Computing- WESCANEX 97, (1997)*. Winnipeg.
- [11] *Ciência da Informação ; 26(1)*. Available from: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S &lng=en&nrm=iso accessed 24 October 2006)
- [12] Araki, H. *Fusão de informações espectrais, altimétrica e de dados auxiliares na classificação de imagens de alta resolução espectral* PhD Thesis. Federal University of Paraná. Curitiba; (2005).
- [13] Russell, S. J.; Norvig, P. *Artificial intelligence: a modern approach*. New Jersey: Prentice-Hall, 1995.
- [14] Casner, S. M. A. *Task-Analytic Approach to the Automated Design of Graphic Presentations*. *ACM Transactions on Graphics (TOG) archive (1991)*. , 10(2), 111-151.
- [15] Roth, S. F., Kolojejchick, J., Mattis, J., & Goldstein, J. *Interactive Graphic Design Using Automatic Presentation Knowledge*. In: *Conference on Human Factors in Computing Systems, (1994)*. Boston, Massachusetts.
- [16] Senay, H., & Ignatius, E. *A Knowledge-Based System For Visualization Design Computer Graphics And Applications*. *IEEE(1994)*. , 14(6), 36-47.
- [17] *Fraunhofer Institute for Intelligent Analysis and Information Systems IAIS. (2012)*. <http://http://www.iais.fraunhofer.de/index.php?id=1863&L=1> (accessed 12 March 2012).
- [18] Anselin, L., Syabri, I., & Kho, Y. (2004). *GeoDa: An Introduction to Spatial Data Analysis*. GeoDa Center for Geospatial Analysis and Computation. College of Liberal Arts and Sciences. Available at: <https://geodacenter.asu.edu/research/publications> (accessed in 24 November 2007).

- [19] Official documentation of geoserver project.<http://docs.geoserver.org/>.accessed in 25 June (2012).
- [20] ColorBrewer 2.0. <http://colorbrewer2.org/> (accessed in 17May (2012)).
- [21] Nivala, A. M. Usability Perspectives for the Design of Interactive Maps. PhD Thesis, Department of Computer Science and Engineering, Helsinki University of Technology, Helsinki, Finland; (2007).
- [22] de Mendonça, A. L. A., & Delazari, L. S. Remote Evaluation of the Execution of Spatial Analysis Tasks with Interactive Web Maps: A Functional and Quantitative Approach. *Cartographic Journal*, The, February (2012). , 49(1), 7-20.
- [23] Dent, B. D. (1999). *Cartography: Thematic Map Design*. WCB McGraw-Hill. Nova York, EUA.

© 2012 The Author(s). Licensee IntechOpen. This is an open access article distributed under the terms of the [Creative Commons Attribution 3.0 License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

IntechOpen

IntechOpen