Knowledge Engineering for Automated Map Design in DESCARTES

Gennady Andrienko and Natalia Andrienko

GMD - German National Research Center for Information Technology
Schloss Birlinghoven, Sankt-Augustin, D-53754 Germany
Tel: +49-2241-142329 Fax: +49-2241-142072
E-mail: Gennady.Andrienko@gmd.de
URL http://allanon.gmd.de/and/

ABSTRACT

The paper deals with issues arising in connection to automated design of cartographic displays of spatially referenced data. A user adequately interprets a map only if it is built in compliance with the established principles of graphic and cartographic presentation. To achieve this, a mapping system should account for a number of characteristics of the data to present. Some of the characteristics concern data semantics, e.g. relationships among data components.

DESCARTES is a knowledge-based system capable of automated map design. To make the system work, a formal description of characteristics of data to present should be provided. Formalization of data semantics requires intelligent support to the user. For this purpose we have developed a special program called DESCARTES APPLICATION BUILDER. The program implements psychologically based interview techniques involving grouping and comparison of variables.

Keywords

Map visualization design, data characterization, knowledge engineering

1. DATA CHARACTERIZATION FOR AUTOMATED GRAPHICS DESIGN

Current GIS (Geographic Information Systems) do not support users in the process of visualization of thematic data. Typically a user can choose one of several proposed presentation techniques (e.g. choropleth map, graduated symbols, or pie charts) without any guidance or control from the system's side [14]. If the user has no sufficient cartographic background, it is highly probable that the resulting presentation will be inadequate to the characteristics of the data and therefore useless or misleading.

Last decades computer scientists tried to automate the process of creating correct and meaningful graphic presentations of data. Bertin [6] and McEachren [10] proposed theoretical basis for automated visualization design depending of data characteristics. Several implementations were made by Mackinlay [9], Senay and Ignatius [12], Roth and Mattis [11], and Jung [8] (see a detailed description and comparison of existing approaches in our paper [5]). All these systems do not account for certain very important semantic characteristics of data, such as relationships between variables (usually all variables are treated as unrelated, or only algebraic dependencies are considered). Even with this limitation, the existing software requires rather complex characterization of data to visualize. For this purpose certain formal knowledge representation language is used. Obviously, an end user cannot be proposed to describe a data set using a complicated formal syntax. This seems to be one of the reasons why the systems automating graphic design still remain research prototypes.

Our system, DESCARTES, was the first system on automated data mapping considering semantic relationships among data components. The functionality of DESCARTES and its knowledge base on map design is described elsewhere [3][1]. In the current paper we focus on knowledge engineering issues, i.e. on data characteristics accounted for in map design and on interactive acquisition of them. The next section gives some examples of data DESCARTES works with and of maps it produces demonstrating the necessity of knowing data semantics for adequate map design. Then we present the approach to data characterization adopted in DESCARTES (section 3). In section 4 we describe the software called DESCARTES APPLICATION BUILDER intended to assist users in the process of data characterization.

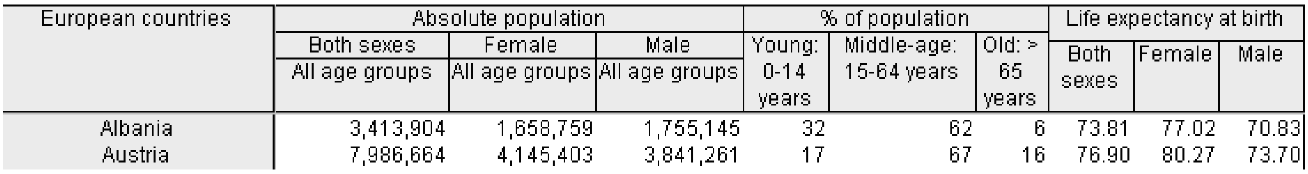
2. IMPORTANCE OF DATA SEMANTICS FOR INTELLIGENT MAP DESIGN

DESCARTES works with thematic data organized in tables. The data should refer to some geographical objects listed in one of the columns of a table. An example data set that can be visualized in DESCARTES is shown in Fig.1. Note that the hierarchical table caption indicating meanings of data components is not a part of the data set.

The user of DESCARTES needs only to select columns to visualize, and the system automatically finds allowable map presentations of the data from these columns. In so doing, the system takes into account, among other data characteristics, semantic relationships between the selected columns. For example, if the user selects the columns with male and female population, the system will build several variants of maps including those with pie charts, bar charts, and segmented bars. Bar charts are enabled by the fact of comparability of the data

ACM GIS '99 11/99 Kansas City, MO USA
© 1999 ACM 1-58113-235-2/99/0011 … $5.00

[1] See an online demo of DESCARTES in the Internet at the URL http://allanon.gmd.de/and/java/iris/. A commercial variant of the system is available from Dialogis GmbH, URL http://www.dialogis.com/



European countries	Absolute population			% of population			Life expectancy at birth		
	Both sexes	Female	Male	Young: 0-14 years	Middle-age: 15-64 years	Old: > 65 years	Both sexes	Female	Male
	All age groups	All age groups	All age groups						
Albania	3,413,904	1,658,759	1,755,145	32	62	6	73.81	77.02	70.83
Austria	7,986,664	4,145,403	3,841,261	17	67	16	76.90	80.27	73.70

Figure 1. An example table as it is shown in DESCARTES

items, and pie charts and segmented bars are possible to use since male and female populations together make a meaningful whole. When the columns with total population and female population are chosen, one of the presentations will be a map with nested squares allowed by the *part-of* relationship between the columns (see Fig. 2).

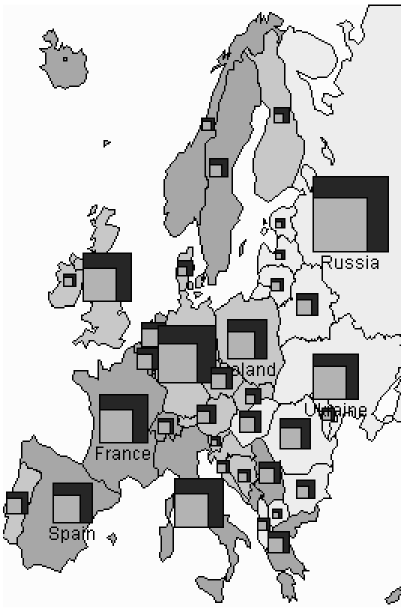


Figure 2. A map built by DESCARTES. The nested squares represent total population numbers and the parts of female population. The contours of the countries are painted according to values of the attribute 'life expectancy'

To provide such an intelligent behavior, we need to simulate in the system a certain degree of understanding of data semantics.

3. REPRESENTATION OF DATA SEMANTICS

3.1 The approach to data characterization

Understanding of data semantics by DESCARTES is achieved through so called **conceptual data characterization**. Essentially, this is a formal description of what is contained in table columns. It should indicate, for example, that the first column of the table shown in Fig.1 contains European countries, the next three columns contain absolute numbers of the whole population and of males and females, and so on. Such a description necessarily refers to certain ***domain notions*** like "countries", "population number", "gender", "male", and "female" in our example. Note that such notions and relationships among them are, in fact, data-independent: they capture semantics of the underlying problem domain.

The approach adopted in DESCARTES is schematically shown in Fig.3. Data characterization involves two components: a domain model and a conceptual data index based on the model. A domain model is a collection of notions necessary to describe the meaning of a spatially referenced data set. The notions are linked by relationships. Examples of domain notions are "Both sexes", "Female", and "Male" where "Both sexes" includes "Female" and "Male". A conceptual index represents meanings of data components by referring them to notions of the domain model. These links allow the system to interpret the data and to know relationships among data components. It is due to the conceptual indexing that the system can automatically generate table captions like the one shown in Fig.1. Such a caption not only presents meanings of data components but also indicates relatedness of columns through its hierarchical organization.

Currently in DESCARTES table (thematic) data are stored separately from geographic data, i.e. coordinates and geometry of objects to which the data refer[2]. Therefore the system needs additionally certain description of geographic data, or map index, to establish a link between thematic and geographic data.

3.2 Parameters and characteristic variables

The work with data in DESCARTES is based on distinguishing between **parameters** and **characteristic variables** made in the data index. **Parameters**, or **independent variables**, serve for referring data to some location, moment or interval in time, object or group of objects etc. Parameters are defined by specifying their names and sets of possible values. Specially distinguished in DESCARTES are parameters values of which are spatial objects, for example, "European countries" with the set of possible values {"Albania", "Austria", "Belgium", ...,

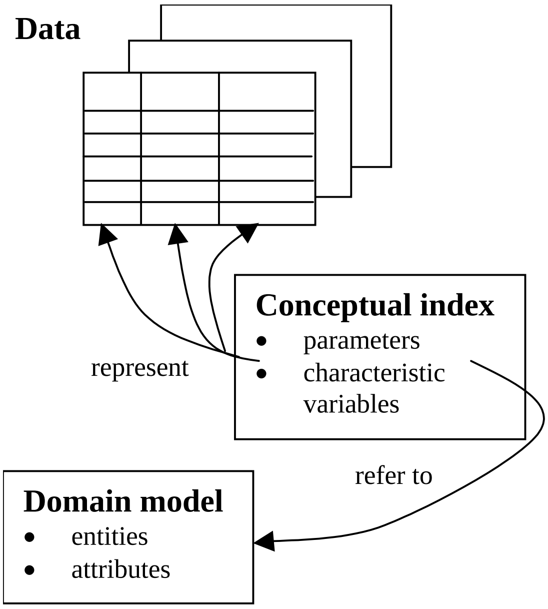


Figure 3. The approach to data characterization adopted in DESCARTES.

[2] The system currently supports table data in DBF (dBase), CSV (Excel) and ASCII formats, vector geographic information in SHP and E00 (ArcView), MIF/MID (MapInfo), BNA (AtlasGIS), DXF (AutoCAD) formats, and raster maps in GIF and JPG formats

"United Kingdom"}. Such parameters will be further referred to as spatial parameters.

The notion "**characteristic variable**" is used to denote results of observations, calculations, estimation etc. found for various values of a parameter or for various value combinations of several parameters. For example, if we have daily data about noontime air temperature and rainfall in Bonn, Paris, and Amsterdam, we can characterize this data set in terms of the parameters "date" and "city" and the characteristic variables "noontime air temperature" and "rainfall".

In this example one can notice that "noontime temperature" refers not only to some observed or measured phenomenon but also to certain time moment (noontime) at that the observation/measurement was made. Unlike other parameters, this reference is common for all data records. In DESCARTES it is possible, when necessary, to specify such common references for a data set in whole. They are called "**invariant parameters**". Invariant parameters have no influence on data presentation, but they are mentioned in map legends. As an alternative to the explicit specification, invariants can be implicitly reflected in names of attributes, like was demonstrated in the cited example.

Note that distinguishing between parameters and characteristic variables only makes sense in regard of some specific data set. For example, time of the day is a parameter in a data set with measurements of air temperature and pressure made at different moments during a day, but it is a characteristic variable in a data set with daily-recorded sunrise and sunset times.

DESCARTES deals only with data referring to **spatial objects**. In other words, one of the parameters characterizing a DESCARTES table should be a spatial parameter. Data in a table may refer also to other parameters. For example, if a table contains numbers of male and female population in different age groups (children, adults, and old people) in countries of Europe, we have one characteristic variable "Population number" and 3 parameters:

1. "Countries of Europe" having as its values the individual countries Albania, Austria, and so on;
2. "Gender" with values "male" and "female";
3. "Age group" with values "children", "adults", and "old people".

There are two opportunities of referring table data to values of a parameter:

1. Values of the parameter are explicitly contained in a column of a table;
2. Columns of a table implicitly refer to the values.

Thus, the data about population mentioned above could be contained in a table of any of the 4 following structures:

Country	Gender	Age group	Population number
Albania	Male	Children	563953
Albania	Male	Adults	1104371
Albania	Male	Old people	86821
...	...	...	...

Country	Gender	Population number (children)	Population number (adults)	Population number (old people)
Albania	Male	563953	1104371	86821
Albania	Female	520186	1026321	112252
...	...	...	...	...

Country	Age group	Population number (male)	Population number (female)
Albania	Children	563953	520186
Albania	Adults	1104371	1026321
Albania	Old people	86821	112252
...	...	...	...

Country	Population number (male, children)	Population number (female, children)	Population number (male, adults)	…
Albania	563953	520186	1104371	…
...	...	...	...	...

DESCARTES is able to understand all such structures and to treat them as equivalent (i.e. process data in the same way irrespective of the table structure) provided that tables are correctly described.

3.3 Organization of domain notion base and table index

Notions in a domain notion base are linked by ***is-a*** relationship and form a hierarchy. There are 3 system-reserved general notions: "**entity**", "**property**", and "**spatial object**". Any domain-specific notion is required to be a descendant of one of these 3 notions. Spatial objects of an application should be descendants of the notion "spatial object".

Each notion in a domain notion base has a unique identifier. DESCARTES handles the identifiers completely internally hiding them from end users. The users see only full names of notions. A notion base can define names of notions in several languages. The system will select appropriate names depending on the language it communicates with the user (currently there are variants of the system with English, German, and Russian user interfaces). Besides names, the notion base contains information about types of properties (integer, real, string etc.) and relationships among notions. Besides the *is-a* relationship, it is very important for DESCARTES to know about inclusions. In the example cited above "Male" and "Female" are included in "Both genders". Inclusion relationships are also specified in the domain notion base.

A table index contains no information about names, types, or relationships. It merely links table columns to corresponding notions in the notion base by citing identifiers of these notions. This is essentially a collection of pairs (<column number>, <reference>) marked by the indicator P (standing for "parameter") or D (standing for "data", i.e. characteristic variable). <reference> is either a single identifier of a notion in the notion base or an identifier of a notion-attribute plus one or more modifiers P(<parameter>,<value>). For example,

- P(1, Country)
- P(2, Year)
- D(3, GNP)
- D(4, PopulationNumber,
  {P(Gender, male), P(age_group, 0-15y)})
- D(5, PopulationNumber,
  {P(Gender, female), P(age_group, 0-15y)})
- D(6, PopulationNumber,
  {P(Gender, male), P(age_group, 16-64y)})
- ...

Here "country", "year", "GNP", "PopulationNumber", "Gender", "male", "female", "age_group", and "0-15y" are identifiers of notions defined in the domain notion base. The notion “country” denotes a group of spatial objects, i.e. particular countries are descendants of this notion in the is-a-hierarchy. “Country” and “Year” are parameters in this table, and their values are contained in the columns 1 and 2, respectively. “GNP” (gross national product) and “PopulationNumber” are characteristic variables. The fourth column contains number of male population in the age group from 0 to 15 years. Its description uses the notions “PopulationNumber”, “Gender”, “male”, “age_group”, “0-15y” and states that “PopulationNumber” is the attribute values of which are contained in the column, and that these values refer to the value “male” of the parameter “Gender” and to the value “0-15y” of the parameter “age_group”. The fifth column differs from the fourth one in that it refers to another value of the parameter “Gender”, and the sixth - to another value of the parameter “age_group”. In all the three cases the characteristic variable is the same.

Here is a fragment of a domain knowledge base that may correspond to the above-given fragment of a table index:

- spatial object
    - country
        - Albania
        - Austria
        - ...
- property
    - year: *integer*
    - PopulationNumber: *integer*
- entity
    - Gender
        - Both_Genders
        - male *part-of* Both_Genders
        - female *part-of* Both_Genders
    - age_group
        - All_Ages
        - 0-15y *part-of* All_Ages (children from 0 to 15 years)
        - 16-64y *part-of* All_Ages (people from 16 to 64 years)
        - 65more *part-of* All_Ages (old people of 65 years and more)

The texts in brackets are names of notions where they are different from the identifiers. Note that we do not intend to present here the syntax of domain notion definition and table description used in DESCARTES.

3.3.1 Types of attributes

DESCARTES recognizes 4 **primitive types** of attributes: numeric (integer or real); date; logical; string.

In addition, an attribute may be referred to a notion in the domain notion base. This means that values of the attribute are all descendants of this notion. For example, we may define an attribute “Dominant religion” and specify its type as “ISA religion”, where “religion” is the identifier of the notion “Religion” having “children” like “Catholic”, “Protestant”, “Orthodox”, “Muslim” etc.

In map design attributes referring to notions are treated in the same way as those having the primitive type “string”. What are then potential advantages of such linking?

- It is possible to describe relationships among values. For example, in the case of religions we can define a subgroup “Christian religions” containing “Catholic”, “Protestant”, and “Orthodox”, and in the case of tree species we can subdivide the species into coniferous, hardwood, and deciduous. Such structuring is potentially useful for **queries** (“*Select all countries where dominant religion is Christian*” instead of listing individual values) and **data aggregation** (“*aggregate timber districts according to their specialization in cultivating coniferous, hardwood, or deciduous species, and calculate some summary statistics for the groups*”) though these functions are currently not implemented in DESCARTES.
- It is possible to use codes as values of the attribute within a table and to decode them into meaningful texts before presenting to the user, as was described above.
- It is possible to specify naming for values of the attribute in several languages.

3.3.2 Relationships

Accounting for **relationships** is important in designing maps that present several table columns using **diagrams** (bar charts, pie charts, segmented bars etc.) In DESCARTES relationships are defined for domain notions rather than for table columns and values. This allows reusing the information about relationships for several tables with similar data. For example, we can define “Men” and “Women” as parts of “Whole population” and refer to these notions when indexing tables with data on population structure for different territories (e.g. Germany and France), different levels of administrative division of the same territory (communes, counties, states), different census years etc. The system will apply the information about the inclusion whenever it deals with columns referring to “Men” and “Women”, and we do not need to specify it for each table.

Important relationships from the visualization viewpoint are **inclusion** and **comparability** of attributes.

Two numeric attributes are ***comparable*** if their difference makes sense. For groups of comparable attributes presentation by parallel bars is allowable. Of course, attributes measured using different units are incomparable (e.g. length in m and distance in km). Currently DESCARTES has no “knowledge” about units of measurement. We plan to account for such information in the future.

Information about inclusions not only determines the applicability of presentation techniques like pie charts or segmented bars. It can be potentially used for intelligent assistance to the user in data analysis. For example, the system could automatically calculate percentages having absolute numbers of inhabitants in various population groups. Such development of the intelligent capabilities of the system will be done in the future.

3.3.3 Link to geographic data

Geographic data specify coordinates and geometry of spatial objects. The geographic specifications of objects are united in groups called ***geographic layers***, or simply layers. Each layer is contained in a separate file. An application may use any number of layers. For each table there should be a layer with geographical objects the data in the table refer to. DESCARTES establishes links between tables and layers through identifiers of spatial objects: they should be the same in a table and in the corresponding layer.

Some layers may be not linked to any table. Objects from them may serve as a background for data presentation. For example, rivers and forests may be shown on a map presenting data referring to districts of Bonn. Such layers are not required to contain identifiers.

A ***base map scenario*** is an enumeration of layers to be included in a background map, with specified order of their drawing, the type of objects (point or line or area), colors for borders and filling, and names to be shown in the map legend. Objects belonging to the same layer are shown uniformly, i.e. using the same color(s) for lines (borders) and contour filling (when appropriate).

To represent data from a table on a map, DESCARTES needs to know 1) what layer contains the geographical objects the data refer to; 2) which additional layers make a background and how to draw them (order, colors, and filling). To provide this information, a ***map index*** is made. The map index enumerates all layers with geographical objects table data may refer to. Each layer is related to the domain notion denoting the corresponding group of spatial objects (more exactly, to the common parent of the notions denoting the objects in the *is-a* hierarchy). Thus, a layer with contours of European countries would be linked with the notion "Country" (see the fragment of an example domain notion base above). Besides, for each layer the map index specifies which base map scenario will provide the geographical background for the presentation of thematic data referring to the objects in this layer. One and the same base map scenario may be used for presenting different groups of geographical objects (if, of course, they are located on the same territory). For example, data referring to districts of Bonn and data referring to sectors (larger territory units) may be shown on the same background made by forests, rivers, roads, and other kinds of geographical entities.

So, to enable DESCARTES present data from a table on a map, the following requirements should be met:

1. The table has a column with identifiers of spatial objects (let us designate this column for further reference as CSp).
2. The notion base contains a notion denoting this group of objects, for example, "Districts" or "Countries of Europe". We shall refer to this notion as NSp.
3. The table is indexed, and the index associates CSp with the NSp.
4. There is a layer with a geographic specification (i.e. coordinates and geometry) of the objects contained in CSp. The identifiers of the objects in the layer should be the same as in CSp.
5. The layer is mentioned in the map index with the reference to NSp.

It is the association with the same domain notion that allows DESCARTES to link table and geographical data. This approach makes it possible to link any number of tables with a single geographic layer.

3.4 Advantages of splitting data characterization into domain notion base and table index

1. Once defined notions may be reused in describing several tables or in describing different parts of big tables.
2. Relationships are described in a table-independent manner, therefore there is no difference in specification of relationships between table columns and between values within a column.
3. This paradigm easily supports descriptions of various structures of tables. It allows the system to understand the equivalence of structures with parameter values explicitly contained in a column and with implicit reference of columns to the values (see the examples in the section "Parameters and data"). It is possible to automatically do equivalent transformations of table structures (this function exists in DESCARTES's ancestor IRIS [2] for Windows but was not ported to DESCARTES yet).
4. It is easy to make multi-lingual applications by providing names in several languages for each notion in the domain notion base. The description of tables is not affected.
5. There is a potential opportunity of implementing a search function: in an application with a large number of tables the system can find a table with data required by a user according to a conceptual specification like "population number in countries of Europe". The user can make such queries by selecting appropriate notions from the domain notion base, and the system can select data by matching the queries with table descriptions in the table index. This function exists in IRIS [2] but has not been re-implemented in current DESCARTES.
6. A column in a table may contain some codes instead of meaningful texts. It is possible, defining a domain notion, to specify that a certain text is its code and some other text is its meaningful name. Then, if a table column contains this code and is appropriately indexed (i.e. referred to the "parent" of this notion in the *is-a*-hierarchy), the system will substitute the code by the meaningful name before presenting the value to the user. Moreover, it is possible to provide meaningful names in several languages, and the system will select the "right" name depending on the current interface language.

4. INTERACTIVE ACQUISITION OF DATA CHARACTERISTICS

4.1 Scenario of building an application

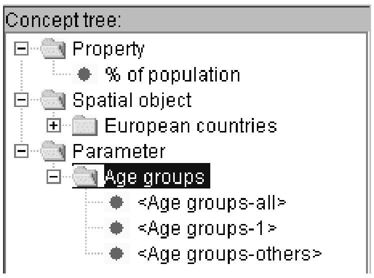
The information about a data set necessary for the work of DESCARTES has to be provided by a person familiar with the meaning of the data. An interactive module called APPLICATION BUILDER facilitates this activity. This module supports conceptual indexing of tables with thematic data, linking them to spatial data (coordinates and geometry of spatial objects), and description of semantic aspects. Interviewing the user about the data, APPLICATION BUILDER builds the domain notion base, the table index, and the map index in parallel. In so doing, it completely hides from the user the formal knowledge representation language and communicates with him through a convenient graphical interface. We describe the work of Application Builder using the data set shown in the table in Fig.1 as an example.

First of all the user includes the data set in some DESCARTES application (new or previously existing). APPLICATION BUILDER loads the data and shows them in the table form. To link the table with corresponding geographic data, the user indicates the column that contains identifiers of geographic objects and points to the layer containing coordinates or contours of these objects.

On the next step the user is expected to describe relationships between data components. The system proposes to group together columns referring to the same attribute. Suppose that the user selects 3 columns with percentages of various age groups in the total population. After this the system asks about an appropriate name for the attribute. The user gives the name "% of total population". The system writes the new attribute to the domain notion base. It takes the type of the attribute and the range of values from the table.

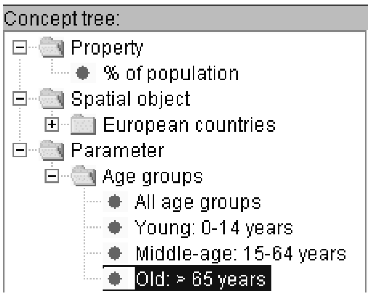
Then APPLICATION BUILDER asks which parameter can help to differentiate the columns forming the group. The user replies that the columns differ by values of the parameter "Age groups". The system proposes him a template to define the parameter with three default values as shown in Fig. 4. The user may change the proposed value names and to add more values. The role of the default names is to remind the user that she should consider existence of parameter values including others ("all") and that he should list all the possibilities ("others"). In our example, the user edited the set of the parameter values as is shown in Fig.5. On the next step the user is expected to specify which value of the parameter corresponds to each column in the group (see Fig. 6 and 7).

Going on with the data description, the user groups together three other columns with values of life expectancy at birth for



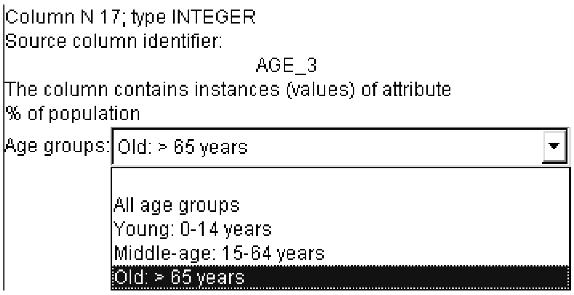
Concept tree:
- Property
  - % of population
- Spatial object
  - European countries
- Parameter
  - Age groups
    - <Age groups-all>
    - <Age groups-1>
    - <Age groups-others>

Figure 4. The template to specify a parameter



Concept tree:
- Property
  - % of population
- Spatial object
  - European countries
- Parameter
  - Age groups
    - All age groups
    - Young: 0-14 years
    - Middle-age: 15-64 years
    - Old: > 65 years

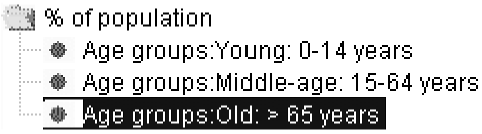
Figure 5. Specification of values of a parameter



Column N 17; type INTEGER
Source column identifier:
AGE_3
The column contains instances (values) of attribute
% of population
Age groups: Old: > 65 years

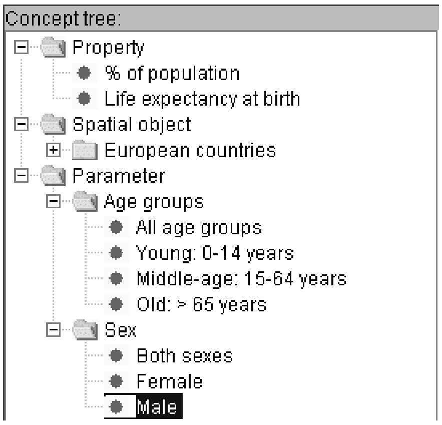
All age groups
Young: 0-14 years
Middle-age: 15-64 years
Old: > 65 years

Figure 6. Linking of a column with a parameter value



% of population
- Age groups:Young: 0-14 years
- Age groups:Middle-age: 15-64 years
- Age groups:Old: > 65 years

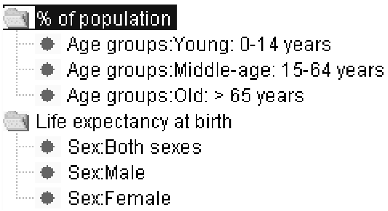
Figure 7. An indexed group of columns



Concept tree:
- Property
  - % of population
  - Life expectancy at birth
- Spatial object
  - European countries
- Parameter
  - Age groups
    - All age groups
    - Young: 0-14 years
    - Middle-age: 15-64 years
    - Old: > 65 years
  - Sex
    - Both sexes
    - Female
    - Male

Figure 8. A fragment of a concept tree with 2 parameters

the total population and for males and females. He enters a new attribute, “Life expectancy at birth”, and a new parameter, “Sex”, see Fig. 8. After describing the correspondence between columns and values of the parameter, the index of the table looks as shown in Fig. 9.



% of population
- Age groups:Young: 0-14 years
- Age groups:Middle-age: 15-64 years
- Age groups:Old: > 65 years

Life expectancy at birth
- Sex:Both sexes
- Sex:Male
- Sex:Female

Figure 9. Two groups of columns associated with domain notions

At the next step the user unites nine columns referring to the absolute population number. These columns differ by values of both previously defined parameters. The user selects these parameters as relevant to the group, and specifies the correspondence between the columns and the values of the parameters.

Having indexed all the table columns, the user specifies the inclusion and comparability relationships among the defined notions. Both tasks are fulfilled through grouping of related notions.

The information provided by the user is sufficient for the presentation of the table in the form shown in Fig.1 and for adequate visualization of various subsets of the table columns. It is important that the system guided the user in the process of formalization of his knowledge about the data set. He was requested to do rather simple and clear procedures of grouping and comparison that stimulated knowledge expression.

4.2 Knowledge acquisition techniques applied in the scenario

The task of revealing relationships among data components is a complex cognitive task. It is related to the problem of acquisition of expert’s knowledge. A human expert usually cannot be requested to describe his knowledge using a formal language. Previous research in knowledge engineering proposed to use psychologically based interview techniques for interactive knowledge elicitation. Thus, the systems Aquinas [7], Kitten [13], and AFORIZM [1] used the techniques of grouping and comparison. We have applied these ideas for knowledge engineering in DESCARTES APPLICATION BUILDER.

The APPLICATION BUILDER facilitates the description of data semantics by proposing the following procedure:

1) The user groups columns of the table referring to the same attribute;
2) The user gives a name to the common attribute;
3) The user compares the columns within the group with the goal to reveal parameters which express the difference in meanings of the columns;
4) The user defines the parameters. For each parameter the system stimulates the user to think about the possible values and relationships among them by proposing a pattern containing special values "All" and "others";
5) The user establishes the correspondence between the columns and the values of the parameters.

In the course of specification of a parameter the system applies special rules to the values “all” and “others”:

- if the user wants to rename the value "all", the system asks him if the new name represents the value that includes all others. Depending on his answer, the new value will replace "all" or will be added as an ordinary value;
- if the user wishes to rename or delete the value "others", the system asks the user if all possible values of the

parameters are already listed. For example, if we have a parameter "population division by employment" with values "all", "employed", "unemployed", and "others", such question may remind the user about the existence of people not related to employment (children, old, disabled, etc.)

So, the system guides the user in the process of describing a data set and tries to stimulate careful consideration of related concepts.

5. CONCLUSIONS

To build a correct graphical or, in particular, cartographic presentation of data, one should account for numerous characteristics of data and semantic relationships among data components. For automated graphics design the required knowledge about data should be provided to a design system. The knowledge has to be represented in some formal, machine-readable language. This is an obstacle to the wide use of the existing automated data visualization systems, which still remain research prototypes. We tried to overcome this challenge through creation of APPLICATION BUILDER, a program that facilitates and stimulates data characterization.

The feasibility of the proposed solution has been proved by successful use of APPLICATION BUILDER by several people who made applications of DESCARTES in various domains, including national, regional and urban population statistics. However, communication with the users has shown that the process of application building requires understanding of the key concepts “attribute” and “parameter”.

Further development of the tool will be done within the EU-funded CommonGIS project (ESPRIT Project 28983, November 1998 - April 2001, URL http://commongis.jrc.it/) aimed at creating a knowledge-based system to support exploratory analysis of spatially referenced data.

The proposed data characterization schema can be used not only for visualization design but also for intelligent information retrieval [2] and for supporting data analysis with a joint use of interactive visualization and computational data mining methods [4].

REFERENCES

[1] Andrienko, G. and Andrienko, N. AFORIZM approach: creating situations to facilitate expertize transfer. In Steels, L., Schreiber, G., and Van de Velde, W. (Eds.) *EKAW’94: A Future for Knowledge Acquisition.* Lecture Notes in Artificial Intelligence. Springer-verlag, **867**, 1994, pp.244-261.

[2] Andrienko, G. and Andrienko, N. Search and representation of information in multimedia system: knowledge-based approach. *Programming and Computer Software*, **22** (1), 1996, pp.45-52.

[3] Andrienko, G. and Andrienko, N. Interactive maps for visual data exploration. *International Journal Geographic Information Science*, **13** (4), 1999, 355-374.

[4] Andrienko, G. and Andrienko, N. Knowledge-based visualization to support spatial data mining. In Hand, D.J., Kok, J.N., and Berthold, M.R. (Eds.) *Advances in Intelligent Data Analysis*, Lecture Notes in Computer Science, Springer-verlag, **1642**, 1999, pp.149-160.

[5] Andrienko, G. and Andrienko, N. Data characterization schema for intelligent support in visual data analysis. In Freksa, C. and Mark, D.M. (Eds.) *Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science*, Lecture Notes in Computer Science, Springer-verlag, **1661**, 1999, pp.349-366.

[6] Bertin, J. *Semiology of graphics. Diagrams, networks, maps.* The University of Wisconsin Press, Madison WI, 1983.

[7] Booze, J., Shema, D., and Bradshaw. J. Recent progress in Aquinas: a knowledge acquisition workbench. In *Proceedings of the European Knowledge Acquisition Workshop (EKAW'88)*. GMD Studien, **143**, 1988, pp.2.1-2.15.

[8] Jung, V. Knowledge-based visualization design for geographic information systems. *Proceedings of the 3rd ACM International Workshop on Advances in Geographic Information Systems* (Baltimor, 1995), ACM Press, pp.101-108.

[9] Mackinlay, J. Automating the design of graphical presentation of relational information. *ACM Transactions on Graphics*, **5** (2), 1986, pp.110-141

[10] MacEachren, A.M. *How maps work. Representation, visualization, and design*. The Guilford Press, NY, 1995.

[11] Roth, S.M. and Mattis, J. Data characterization for intelligent graphics presentation. *Proceedings of SIGCHI’90: Human factors in computing systems conference* (Seattle WA), ACM Press, 1990, pp.193-200.

[12] Senay, H. and Ignatius, E. A knowledge-based system for visualization design. *IEEE Computer Graphics and Applications*, **14** (6), 1994, pp.36-47.

[13] Shaw, M. and Gaines, B. KITTEN: knowledge initiation and transfer tools for experts and novices. *International Journal Man-Machine Studies*, **27** (3), 1987, pp.251-280.

[14] *Using ArcView GIS*. ESRI, 1996.