VISUALIZATION OF GEOMETRICAL AND NON-GEOMETRICAL DATA

Maria Beatriz Carmo[1], João Duarte Cunha[2], Ana Paula Cláudio[1](*)

[1]FCUL-DI, Bloco C5, Piso 1, Campo Grande 1700 Lisboa, Portugal
e-mail: bc@di.fc.ul.pt, apc@di.fc.ul.pt

[2]LNEC-CI, Av. do Brasil 101 1799 Lisboa Codex, Portugal
e-mail: jdc@lnec.pt

ABSTRACT

A prototype for the visualization of both geometrical and non-geometrical data is described. Special emphasis is given to flat structured or tabular data. Main features of this prototype are: a data independent architecture; filtering of information using degree of interest functions and establishing limits for the variation of one or more data attributes; interactive specification of attribute restrictions combining **and** and **or** operators; choice of representations using current scale factor and the result of the degree of interest function. The concept of information classes is introduced to associate a set of attributes with a spatial reference and a graphical representation. Derived information classes are used to combine attribute restrictions with **and** and **or** operators.

Keywords: visualization, degree of interest function, filtering, graphical representation

1. INTRODUCTION

Creating visualizations is a quite complex process and, as pointed out in [Campo97], visualizations tend to be hand-crafted for each particular case.

By identifying common requirements in a wide range of cases it is possible to develop generic systems. Among these requirements we can find the choice of representations, the need to accommodate different data formats, the provision of flexible filtering mechanisms and the ability to use different scales of representation in a meaningful way.

Our goal is to provide a generic system to visualize geometrical and non-geometrical data, in particular structured data with a flat structure (tabular or relational) using the terminology proposed in [Boyle93]. To demonstrate the basic features of such a system a prototype was developed.

To make the prototype independent of the data to be represented, a neutral file format was defined. This technique is found in

(*) This work was supported by project SARA PRAXIS /2/2.1/TIT/1662/95

most visualization programs and its main advantage is a clear separation between data syntax and data semantics. Different data formats can be dealt with, provided that a conversion module is available for them.

Geometrical data have an intrinsic spatial reference and an associated list of geometries. Each geometry defines the graphical representation to be used in a particular range of scales as explained in 2.1.

For non-geometrical data there is a need to define graphical elements to represent the data. The spatial reference or grid must be explicitly defined. When both geometrical and non-geometrical data are present, the spatial reference for non-geometrical data is strongly correlated with the intrinsic spatial reference of the geometrical data.

By defining lists of representations with different levels of detail it is possible to adapt the representation to the current scale as well as to simplify the representation of less relevant information.

The concept of information class was introduced in order to combine data associated to the same spatial reference with a list of representations. Information classes are described in detail in section 2.

As a main feature, the prototype provides different mechanisms to reduce the amount of information displayed. First, limits for the variation of one or more data attributes can be established and it is possible to combine interactively these restrictions using the **and** and **or** operators. Second, the representation of an entire information class can be suppressed. And finally, a degree of interest function [Furnas86] can be used both as a filtering mechanism to reduce the amount of information displayed, and as a means to choose simplified representations for less relevant information. The use of simplified representations is in fact a mechanism to reduce the amount of information effectively displayed without complete suppression.

The degree of interest function (DOI), as defined by G. Furnas, assigns to each data point a number which is a measure of the user's interest in seeing the data associated with that point. The value of the function in a point **x** depends on the *a priori* importance of the point, **API(x)**, and on the distance between **x** and the current focus, **y**: **DOI(x|y)=API(x)-D(x,y)**. The distance may correspond to the Euclidean distance or to a semantic distance. Only the points whose DOI exceeds a given threshold, **k**, are displayed. This means that the amount of information effectively displayed depends on the value of this threshold.

We have extended the use of a degree of interest function to situations where a focus is not specified. In this case it is assumed that the distance function is equal to zero everywhere, i.e., the DOI function exactly matches the *a priori* importance. A DOI can be associated with geometries as well as grid points.

The prototype also provides a zoom mechanism, following the model proposed in [Robert93] and [Cunha95]. In this model, the zoomed area is magnified without distortion but the global context is preserved.

Some of the features included in this prototype are also present in other systems namely IVEE [Ahlberg95], AutoIcon [Fairc93] and Iconographer [Gray90]. The IVEE system provides the automatic creation of dynamic queries applications attaching to each object in a database relation a graphical object. In AutoIcon each object type is associated with a list of representations and the visualization is determined by the value of the degree of interest function of a particular object, without any restrictions derived from the scale of the representation. In Iconographer attributes of application objects are interactively mapped onto icon attributes.

This paper deals essentially with the concepts of information classes and filtering mechanisms. In section 2 the concept of information class is defined. In section 3 filtering mechanisms are discussed, particularly the interactive combination of restrictions using the **and** and **or** operators.

2. INFORMATION CLASSES

The concept of information class was introduced in order to combine a set of attributes associated to the same spatial reference, with a list of representations. This concept is most relevant when dealing with non-geometrical information where it is necessary to associate to each data element a graphical representation and a spatial location. In this case, the spatial reference is represented as a grid composed of locations and implicit or explicit connections.

Information classes provide a convenient mechanism to associate data with the same spatial reference and semantically related. It is possible to have different classes sharing the same grid. We can have classes of information with:

- non-geometrical data only
- geometrical data only
- both geometrical and non-geometrical data

In the last case, it is not mandatory that geometrical and non-geometrical data share the same spatial reference, in the sense that they do not have to share the same locations. However, they must be defined in the same coordinate system.

The graphical representation of non-geometrical data is described in a list of representations. The representation displayed in each point of the grid is one of the representations described in the list of representations. For geometrical data there is a list of geometries. Both geometries and representations are described as lists of graphical primitives. However the list of geometries and the list of representations are treated in a different way, as explained later.

One of the attributes of the information class can be chosen as *a priori* importance value and called the API attribute. For information classes with geometrical data only, the API attribute qualifies each geometry of the list of geometries. In a non-geometrical information class the API attribute establishes the importance of each point of the grid. The same is true for a class with both geometrical and non-geometrical data. This means that, in this case, the elements of the list of geometries will not be affected by the degree of interest function. The visibility of a geometry depends solely on the current scale of representation. If none of the attributes is chosen as API attribute, the *a priori* importance is, by default, equal to one.

For non-geometrical data a grid must be defined. The grid classification proposed in [Speray90] was adopted. If the data does not include spatial information, only Cartesian or regular grids can be used. Otherwise, structured or unstructured grids can be specified. The connections of an unstructured grid do not need to be defined unless a surface is used to represent the values of the attributes.

For geometrical information there is no need to define a grid. The geometries themselves have specific locations for their elements. The list of geometries contains all the information necessary to display the geometry.

In summary, an information class can be composed of attributes, a grid, a list of representations and a list of geometries. One of the attributes can be selected as API attribute. The information class with its attributes and grid is defined in a data file. The list of representations and the list of geometries associated with an information class, i. e., the graphical data, are defined in a different file, the mapping file.

2.1 Lists of geometries, lists of representations and multiple representations

An information class with non-geometrical data must have a list of representations. The representations in the list may or may not depend on the attribute values in each point of the grid. The list contains a hierarchy of representations based on the smallest scale factor allowed for each one. One of the representations is chosen taking into account not only the current scale factor, but also semantic criteria expressed by the degree of interest function [Carmo97]. In this way less detailed representations can be selected to visualize information with less interest.

Classes with geometrical data have a list of geometries. Each geometry has two associated constants: the minimum and maximum scale factors allowed for that geometry. The geometries displayed are those in the list which can be represented with the current scale. This is indeed the main difference between lists of representations and lists of geometries: only one representation is displayed, but more than one geometry can be simultaneously displayed.

It is also possible to have multiple representations for geometrical objects. Two cases must be considered: classes with both geometrical and non-geometrical data and classes with geometrical data only.

In the first case, as each geometry has its own range of allowed scales of representation, it is possible to have different representations for different scales. However it is not possible to associate an *a priori* importance to each representation, as already stated.

In the second case, classes with geometrical data only, it is possible not only to have multiple representations with different scale ranges, but also to split one representation into several ones with the same range of allowed scales and define for each a different *a priori* importance. In this way, the amount of information displayed can be controlled by the degree of interest function. A different approach is possible if the information class contains a single geometrical object. Instead of a list of geometries, a list of representations can be used to contain different representations of the object. This is possible since geometries and representations are in fact similar entities, treated in a different way. The appropriate representation will be chosen according to the scale of representation and the value of the degree of interest function of the object.

3. FILTERING MECHANISMS

Three types of filtering are provided:

- selection of classes to visualize
- domain restrictions
- use of degree of interest functions

The information classes which the user wants to display are marked as active classes. By default, all the information classes are active.

In the prototype, data attributes can be of type float or string. Interactively, the user can establish domain restrictions: limits for the variation of a float attribute or selected values of a string attribute. In the same class of information, the result is the conjunction of the domain restrictions. The disjunction of the results of all active classes will be displayed. To support the combination of domain restrictions with conjunctions and disjunctions in the same class, derived information classes were created as explained in 3.1.

Use of the degree of interest function is the third filtering mechanism. A data point of an active class is displayed only if the value of the degree of interest function is above a given threshold. For each information class one of the attributes can be used as *a priori* importance in each point. The distance function is defined as the Euclidean distance between the focus and the respective point of the grid, for non-geometrical data, or a point of the geometry, for geometrical data.

3.1 Interactive combination of restrictions on attribute values using and and or operators

One of the problems encountered in the interactive formulation of filtering criteria is the combination of restrictions involving both disjunctions and conjunctions. Different solutions have been proposed to solve this problem ([Kilger93], [Golds94], [Ogle95]). With information classes, the problem occurs for objects of the same information class when the simultaneous use of disjunctions and conjunctions is meaningful. For objects belonging to different classes, conjunctions are always meaningless, since the result would always be an empty set. This means that, for objects of different classes, only the disjunction of conditions can be used.

For each information class, the conjunction of the selection criteria for each attribute selects the objects satisfying simultaneously all the criteria. If, on the other hand, we wish to select the objects satisfying at least a given subset of the criteria, a possible solution involves the definition of derived information classes, one for each subset. Derived classes are copies of the base class, containing the same elements but subjected to different restrictions. The final result contains the elements selected in at least one class (base class or derived class).

By default, the representation of a given element will be independent of the class(es) where it has been selected.

To illustrate the concept, we will consider the representation of public buildings in a city map. The same type of representation was used for each class: an ellipse with a letter identifying the (base) class. Fig.1 shows the elements in the active classes, “hotels” and “stadiums”, without any restrictions. The minimum and maximum values of each attribute are shown. In Fig. 2 elements of class “stadiums” are still shown without any restrictions but only the elements of class “hotels” with one or five “stars” are displayed. To deal with the disjunction of conditions for elements of class “hotels” (one or five “stars”) a derived class was defined, “hotelsderv1”. The base class is then used to select elements of class “hotels” with one “star” and the derived class to select the elements of class “hotels” with five “stars”. The final result is the set of elements selected in each class, thus including both one and five “stars” elements of class “hotels”.

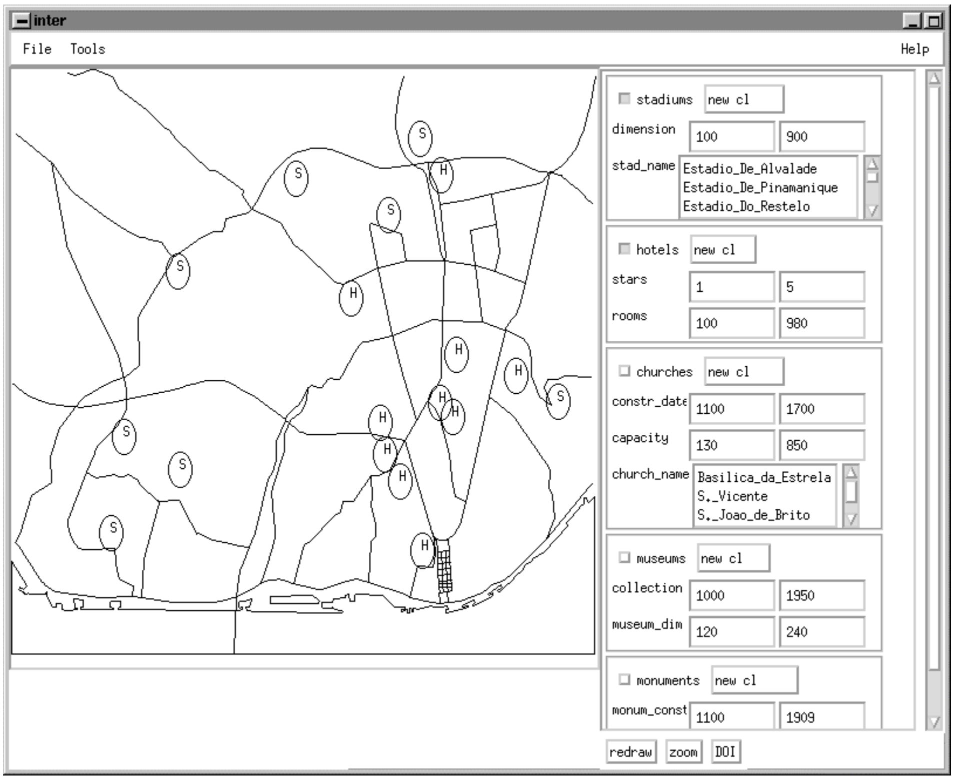

Fig. 1 - Elements in the active classes, “hotels” and “stadiums”, without any restrictions

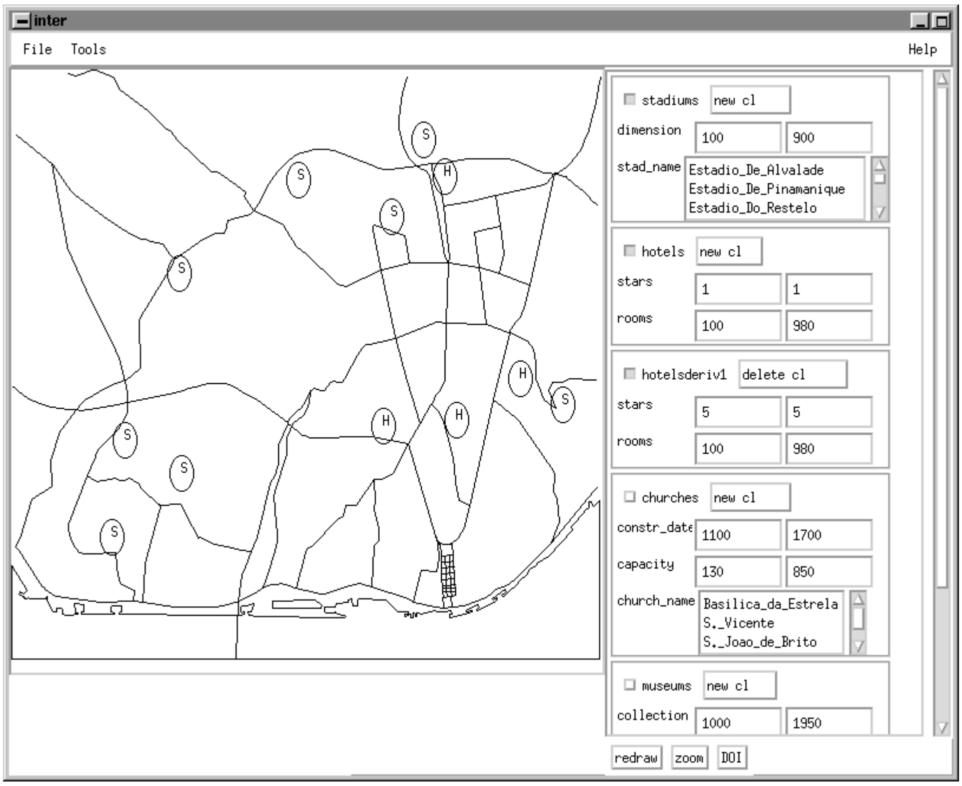

Fig.2 - Elements of class “stadiums” without any restrictions and elements of class “hotels” with one or five “stars”

4. CONCLUSIONS AND FUTURE WORK

The concept of information classes provides a convenient structuring mechanism to combine data, graphical representations and spatial references.

The definition of derived classes makes it easy to interactively combine restrictions expressed by **and** and **or** operators. We are currently considering the possibility of defining derived classes with less attributes than the base class, thus creating new entities. This possibility can be combined with the addition of graphical elements to the base class representation to obtain modified representations, or even to define completely new representations for the objects in a derived class.

A data file format was defined in order to make the prototype independent from the data.

Currently the representations of non-geometrical data are defined in a mapping file, but we intend to develop an interactive editor to define these representations.

Alternative types of representation, like 3D coloured surfaces, are also being developed.

REFERENCES

[Ahlberg95] Ahlberg, C., Wistrand, E.: IVEE: An Information Visualization & Exploration Environment, Proceedings '95 Visualization Information, Atlanta, IEEE Computer Society Press, pp 66-73, 1995

[Boyle93] Boyle, J., Eick, S., Hemmje, M., Keim, D., Lee, J. P., Sumner, E.: Database Issues for Data Visualization: Interaction, User Interfaces, and Presentation, Proceedings ‘93 Database Issues for Data Visualization, Lee, J. P., Grinstein, G.G. (eds), pp 25-34, 1993

[Campo97] Campo, M., Orosco, R., Teyseyre, A.: Automatic Abstraction Management in Information Visualization Systems, Proceedings 1997 IEEE Conference on Information Visualization IV'97, London, pp 50-56, 1997

[Carmo97] Carmo, M. B., Cunha, J. D.: Visualization of Large Volumes of Information Using Different Representations, Proceedings 1997 IEEE Conference on Information Visualization IV'97, London, pp 101-105, 1997

[Cunha95] Cunha, J. D., Carmo, M. B.: Modelo da Pirâmide Truncada, VII Encontro Português de Computação Gráfica, pp 35-46, 1995

[Fairc93] Fairchild, K. M.: Information Management Using Virtual Reality-Based Visualizations, *Virtual reality applications and explorations*, edited by Alan Wexelblat, Academic Press, pp 45-74, 1993

[Furnas86] Furnas, G.: Generalized Fisheye Views, Proceedings CHI '86, pp 16-23, 1986

[Golds94] Goldstein, J., Roth, S. F.: Using Aggregation and Dynamic Queries for Exploring Large Data Sets, ACM CHI '94 Conference Proceedings, Boston, pp 23-29, 1994

[Gray90] Gray, P. D., Waite, K. W., Draper, S. W.:Do-It-Yourself Iconic Displays: Reconfigurable Iconic Representations of Application Objects, INTERACT ’90, D. Diaper et al. (eds) Elsevier Science Publishers B. V.(North-Holland) pp 639-644, 1990

[Kilger93] Kilger, J., Swanberg, D., Jain, R.: Concept Clustering in a Query Interface to an Image Database, Proceedings UIST'93, Atlanta, pp 11-21, 1993

[Ogle95] Ogle, V., Stonebraker, M.: Chabot: Retrieval from a Relational Database of Images, IEEE Computer, vol.28, No.9, pp 40-48, 1995

[Rober93] Robertson, G., Mackinlay, J. D.,: The Document Lens, Proceedings UIST’93, pp 101-108, 1993

[Speray90] Speray, D., Kennon, S.: Volume Probes: Interactive Data Exploration on Arbitrary Grids, Computer Graphics, vol 24, No.5, pp 5-12, 1990