

---

## **Dynamic multi-granularity semantic representations for navigation facility services**

---

Géraldine Del Mondo\* and  
Michel Mainguenaud

Institut National des Sciences Appliquées – Rouen/LITIS – EA4108,  
BP08 165 Avenue de l’Université,  
F76801 Saint Etienne du Rouvray, France  
Email: geraldine.del\_mondo@insa-rouen.fr  
Email: michel.mainguenaud@insa-rouen.fr  
\*Corresponding author

**Abstract:** The main purpose of this article is to tackle the management of alphanumeric data provided to end users in navigation facility services. We propose, via a top down and interpreted approach, to define a data model. The aim is to take into account the spatial relevance of alphanumeric attributes using a logical spatial representation (not using a cartographical representation). Based on a conventional database model extended with links to model relationships between alphanumeric attributes and spatial semantics, this model provides the management of several levels of abstraction (granularity levels). We describe a data manipulation operator to zoom into a deeper abstraction level and we define rules to build the data model associated with the results of this operator.

**Keywords:** geographical information systems; spatial databases; navigation facilities; specialisation.

**Reference** to this paper should be made as follows: Del Mondo, G. and Mainguenaud, M. (2016) ‘Dynamic multi-granularity semantic representations for navigation facility services’, *Int. J. Spatial, Temporal and Multimedia Information Systems*, Vol. 1, No. 1, pp.3–29.

**Biographical notes:** Géraldine Del Mondo received in 2011, after three years working at the Naval Academy Research Institute in Brest, France, her PhD in Geomatic/Computer Science from the University of Brittany. She is currently an Assistant Professor at the Department of Information System Engineering of the National Institute of Applied Science (INSA) at the University of Normandy. She is a member of the Transport Intelligent System group (STI) of the Computer Science, Information Processing, and Systems Laboratory (LITIS). Her research interests include geomatic, spatial and temporal reasoning, databases, applied graph theory and intelligent transport systems.

Michel Mainguenaud was the Director of the Information System Engineering Department and then the Dean for Education at the Institut National des Sciences Appliquées (INSA) of Rouen. He was the Director of the National CNRS/SIGMA-Cassini research group (a CNRS recognised national group to promote research in geographical information systems in France). His main interests are geographical information systems, multi-media database models and query languages: documents, images and maps.

## 1 Introduction

The overwhelming development of mobile internet and geo-localisation tools such as the Global Positioning System (GPS), leads to computer applications and interoperable services for navigation facilities. Whatever the relative speed of a displacement is, high with vehicles, high speed trains or low with pedestrians, the availability of semantic information (and not only cartographic information) is still an open problem. Numerous tools are already available within vehicles or on mobile phones for example. They are mainly based on a cartographic approach of a space representation. Data semantics is relatively poor in these applications.

While defining a path from one place to another one, for example, several levels of abstraction may be very useful. A driver may require information about the town (s)he is leaving and information about the town in which (s)he will arrive. During the journey using for example a highway, very detailed information are far less useful since the important point for the driver is the number of the exit. Therefore, providing an operator dealing with different levels of abstraction is important. We must be able to change the level of abstraction from a higher level to a lower level. This change may occur from the cartographical point of view (as it is very conventional in navigation facilities) but also from the alphanumeric point of view (which is far less studied).

In the literature, (Stell and Worboys, 1998) and the Tioga project (Aiken et al., 1995, 1996) are classified in the top-down approach. The main goal, from the database point of view, is to offer relevant data associated with a granularity that is compatible with the actual space perception. Granularity is defined as the degree of spatial relevance of available data. As an example, the National Gross Product of a country may not be very relevant for a navigation facility tool while entering in a town (even if this town is located in the country in which the end user is driving).

We propose to define a Data Model (DM) able to take into account the spatial relevance of alphanumeric data depending on the logical spatial representation (and not on cartographic aspects). As an example, the population of a town is valid for the whole spatial representation of a town although the mayor's name is valid for any (logical) subset of this town. Within a DM, information can be obtained from a calculus based on spatial or alphanumeric data. As an example, population density is built from a conventional alphanumeric data (e.g., the population), and a spatial information (e.g., the surface). Starting from the spatial characteristics of alphanumeric data, we define propagation rules (i.e., transfer from a DM to another one) from a granularity level to another one. The propagation may be completely reliable or an uncertainty may be introduced. This uncertainty may grow in relation to the decreasing of the space perception.

Section 2 presents a state of the art for the top-down approach. Section 3 presents a toy database model to illustrate our concepts. Section 4 presents the management of calculus functions to define calculated attributes. Section 5 presents the DSZI operator in order to provide the variation of granularity level, variations from the less to the more detailed representations and its propagation rules for alphanumeric data. Section 6 presents a conclusion and perspective works.

## **2 State-of-the-art**

To understand a spatial phenomenon, Geographical Information Systems must provide a relevant level of abstraction, i.e., granularity (Muller et al., 1995). The visualisation and the analysis are then better performed. Stell and Worboys (1998) define the granularity (i.e., level of granularity) in a spatial database as the set of objects with the same resolution. They explain that: "Resolution or granularity is concerned with the level of discernibility between elements of a phenomenon that is being represented by the dataset." Hornsby and Egenhofer (1999) introduce a difference between granularity and resolution. Resolution is related to the level of detail from the representation point of view. Granularity is defined, like in Vangenot (2004), on the semantic field.

In the early '80s, Minsky (1981) introduced the notion of multiple perceptions in a knowledge representation system. Such systems provide the representation of a single object with different perspectives. Navigating from a level of granularity to another one is a mandatory manipulation. Two operators are therefore defined: the generalisation operator and the specialisation operator. The generalisation operator can be divided into two specific approaches: cartographic and model-driven.

The cartographic generalisation is a set of graphical transformations of spatial data (Buttenfield and McMaster, 1991; McMaster and Shea, 1992; Mackaness et al., 2011). It implies a scale reduction and a specification of which features must be still represented in a less detailed map (Camossi, 2005). In this context, the term of multiple scales is more dedicated to cartographic generalisation. The model-driven generalisation does not consider the cartographical representation (Muller et al., 1995; Balley, 2007). This operator models several spatial and semantic levels. It is similar to the vertical navigation in Fonseca et al. (2002), provides a high vision of the phenomenon and facilitates data access. The specialisation operator increases data precision while changing of levels.

While changing from a level of granularity to another one, the data matching and semantic coherency (Bobzien and Morgenstern, 2003) between data are still an open challenge. As an example, these problems are tackled in Mustière (2006). This work studies the integration of two maps from databases with different levels of details and the associated consequences. This matching is frequently studied at the cartographical level (Follin et al., 2004; Volz, 2005; Stoter et al., 2008) but far less at the semantic level. Recently, Stoter et al. (2008) uses Object Constraint Language (OCL used in UML) to maintain the consistency between two maps with different scales in order to apply generalisation operators. Constraints are used to validate data at different scales (i.e., granularity). The selection of specific classes for a given level is simplified. Nevertheless, this generalisation is semi-automatic since frequent interactions with end-users are required. Furthermore, the study is mainly based on cartographic concepts.

Aiken et al. (1995, 1996) take into account semantic aspects. They ask end-users to create their own programs in a dedicated language to visualise data with a specific level of granularity. The basic idea is to navigate through multi-dimensional data using the concept of views (i.e., dimensions or granularity levels) depending on end-users' needs. The main drawback is that, even if end-users manipulate a graphical language to define a program, the definition is far from trivial. The maintain of the coherency between different contexts is not clearly defined. An approach based on ontologies has been defined in Fonseca et al. (2002). This approach is interesting as soon as the designer's interest is oriented towards the semantics an object can have depending on the context.

The authors define it as the horizontal navigation, i.e., the semantic change of the (attributes) names from a granularity level to another one. Nevertheless, an ontology is required at each level in order to manipulate generalisation, specialisation or goal extraction operators on the database.

Stell and Worboys (1998) present a conceptual framework to represent several levels of granularity. The authors define a  $\rho$  function to navigate from a level to another one. This framework does not require any specific data modelling as in Fonseca et al. (2002) and relies on the following concepts:

- **Map:** A collection of data (preferably attributes with semantic and geometric goals). It is a state of the database in the time interval where the compiled approach is performed.
- **Granularity lattice:** Provides an order between two maps such as the spatial and semantic granularity levels vary within minimum and maximum thresholds (two different lattices).
- **Map space:** A set of maps, i.e., all the instances of a database with a given schema. They are partially ordered depending on the semantic or spatial precisions. Precision refers to the level of granularity with respect to which data are registered.
- **Extent of representation:** All the elements involved in this representation. The extent may be semantic or geometric.
- **Stratified map space (SMS)  $\Sigma(E)$ :** A granularity lattice such as elements are map spaces. The idea here is to order within a lattice, different map spaces depending on their (spatial and) semantic levels of granularity. The variation of levels is provided with two operators defined on map spaces: the generalisation operator, Gen, and its opposite, Lift. A SMS is associated with a unique extent.
- **Sheaf of SMS:** A set of SMS. This notion avoids to be reduced to a single extent. It establishes relationships between different SMS. Each SMS is associated with an extent. The  $\rho$  function allows navigating within different SMS.

Stell and Worboys (1998) use a descending compiled approach. It takes into account the whole database and defines the  $\rho$  function for the whole database. This process requires a knowledge of the complete database schema and its instances. The main advantage is to improve the response time to a query (all data are available). Nevertheless, an update on the database (schema or instances) leads to a re-compilation and therefore is less adapted to a multi-user context.

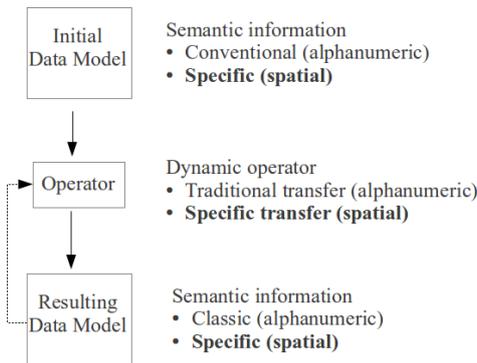
It would be interesting to offer in a single system, the two approaches (i.e., compiled and interpreted). Depending on applications, a database administrator may be in charge of choosing one or the other approach. The following sections present a proposition of an interpreted approach, starting from a toy database example to illustrate the different notions.

### 3 Model

We rely on the conceptual framework defined in Stell and Worboys (1998). We define  $\rho$  to provide a specialisation operation (i.e., DSZI operator) but only on a subset of the DM. In that sense, this operator is an interpreted one.

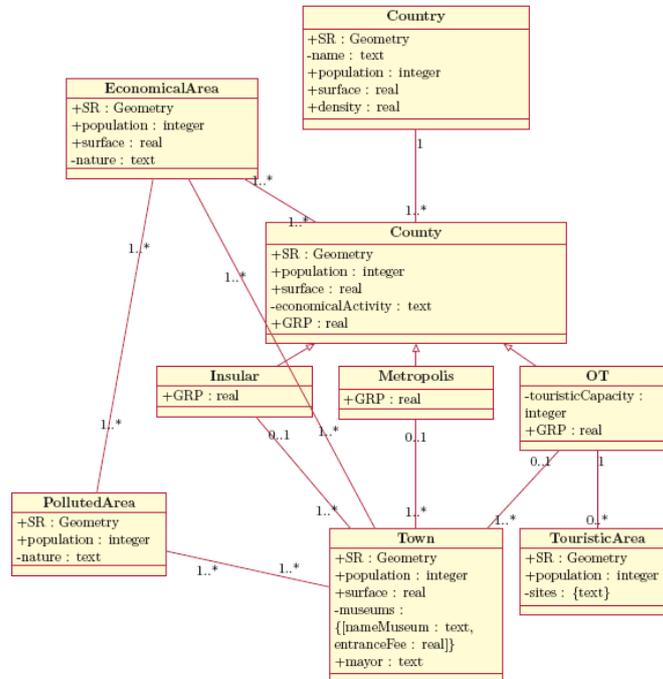
Figure 1 illustrates the dynamic process of an operator. Operators are closed. The result of an operator can be used as an operand of another operator. We start from an initial DM which contains semantic information. It describes information usually found in database schemas (e.g., alphanumeric) and spatial dimension related information. Since we manage a DM associated with a result of an operator, the relevant level of modelling is upon a database schema. In order to realise this process, we have to specify several concepts and rules that ensure the accuracy of the generated DM.

**Figure 1** Dynamic operator: operator application on DM



We present a toy class diagram with the UML formalism (Figure 2) to illustrate concepts developed in this article. Table 1 describes most important attributes of this DM.

**Figure 2** Class diagram of the toy application (see online version for colours)



**Table 1** Semantics of main DM attributes

<i>Attribute</i>	<i>Semantics</i>
name	Country name
SR	Spatial Representation of an entity
population	Population of an entity
surface	Surface of an entity
density	Density of population associated with a country
economicalActivity	Main economical activity of a county
nature	Nature of a polluted area/economical area
sites	Cultural heritage of a touristic area
touristicCapacity	Number of available housings
museum	City Museums, name of a museum and its associated entrance fee: {[name, entranceFee]}
mayor	Mayor's name of a city
GRP	Gross Regional Product of an entity

In this class diagram, Insular and Metropolis define an insular county and a metropolitan county. OT represents overseas territories. Starting from this conceptual representation, we propose to rely on a graph model to formalise interactions and then to define propagation rules from a granularity level to another one.

### 3.1 General structure of the database modelling

We rely on an ANSI/SPARC architecture, but we do not take into account the physical level. The conceptual schema may be represented with an UML formalism (Figure 2). Nevertheless, in order to keep a coherency between end users' data (e.g., transportation networks, maps) and the model, we prefer to handle interactions with a directed labelled graph model. Using a graph concept to formalise a DM (at the meta level) since data are modelled with a graph is similar to the definition of a meta-base with relations, in relational databases, to model relations (i.e., user defined relations are described by relations at the meta level). This modelling implies to formally define this graph, its nodes and its edges.

*Definition 1 (p-Graph):* A p-graph  $G = (N, E, \psi)$  is defined by its set of nodes  $N$ , its set of edges,  $E$  and its incident function  $\psi: E \rightarrow N \times N$ .

In the following, to facilitate the reading, even if  $p$  may be strictly greater than 1, we set  $G(N, E, \psi)$  is a p-graph, we note it  $G$  and name it graph.

*Definition 2 (reference model):* A reference (conceptual data) model  $M$  is represented as a graph. Nodes model classes, directed edges model their links. This graph is noted  $G_M$ . An edge is defined between two classes. The initial extremity of a directed edge is named a source class, the final extremity of a directed edge is named a target class.

*Definition 3 (Network):* Let  $\nu, \epsilon, \gamma$  be the labelling functions associated with respectively nodes, edges and graphs. A network is defined by  $Net = (G, \nu, \epsilon, \gamma)$  and corresponds to a labelled graph. To each network corresponds a reference model  $M = (\nu, \epsilon, \gamma)$ .

According to Stell and Worboys (1998), a network is a map, a reference model is an extent. The semantics of a graph is carried on by labelled nodes and edges.

### 3.1.1 Nodes

Node labels ( $\nu$ ) model the class semantics. To simplify the presentation, we denote an element of a label by its name (e.g., ‘mayor’). In case of ambiguity, we use the pointed notation using the class as a prefix (e.g., Town.surface vs. EconomicArea.surface). Each node is described by a list of fields defined by a structure (attributeName, domain, goal, visibility, family). Attributes are denoted using the conventional triplet: name of the attribute (‘attributeName’), its definition domain (‘domain’), its semantic goal (‘goal’). The field ‘domain’ represents the range of values that can be taken by an attribute. The field ‘goal’ describes the semantic meaning of an attribute. The ‘visibility’ characteristic can take two values: ‘Public’ or ‘Private’. A public visibility makes available the attribute for other nodes while a private visibility implies that the attribute is only available for this node. At the class level, two families of attributes may be distinguished: extension attributes and intension attributes (i.e., calculated). Values of extension attributes are directly accessible. To manage intension attributes, we define three principles for a calculus function (from the operational point of view, they may be combined):

- 1 Geometric: Defined from ISO13249 normalised functions applied to a spatial representation, e.g., *ST Area: Geometry*  $\rightarrow$  *Real*.
- 2 Alphanumeric, function on attribute(s): E.g., *Average Avg: Attribute*  $\rightarrow$  *Real*. As an example, the computation of the average fees for entrances in the museums of a town is a computation on a set-oriented attribute.
- 3 Aggregate upon classes: To count the number of instances (e.g., the number of county using a class variable at the class level).

Due to the fact that we take into account the spatial semantics of attributes (i.e., in relation to the spatial representation but completely independent from their cartographic representations), we must extend the classical representation of a DM. Therefore,  $M$  is described with the following n-uplet (attributeName, domain, goal, visibility, family, characteristic). The ‘characteristic’ property is defined upon the thematic (Mainguenaud, 1994) and network level (Claramunt and Mainguenaud, 1999). To simplify the presentation we focus on the thematic level.

In order to define the DM associated with the results of a spatial operator from a database model point of view, a classification of spatial semantics must be provided (i.e., topology but independent from cartographic representations). Three categories, which must be completed, are defined: Granule, Topology and Set Relationship. As far as we are concerned in this article, only the Granule category will be used. The Granule category defines the validity of an alphanumeric attribute upon the whole or a subset of the spatial representation. As an example, the ‘mayor’ attribute is set as valid for any subset of a spatial representation of a town. ‘Town.population’ is valid for the whole representation of a town. The value of the Granule category (‘Subset’ vs. ‘Global’) may change while defining the DM associated with the result of a spatial operator depending on the semantics of this operator.

*Definition 4 (Attribute properties):* An attribute is characterised as ‘Global’ whenever its semantic validity is relevant for the whole spatial representation initially available and is characterised as ‘Subset’ whenever its semantic validity is relevant for any subset of the spatial representation.

Table 2 presents examples of attributes and their Granule characteristic from the example defined in Figure 2.

**Table 2** Examples for the ‘Granule’ characteristic

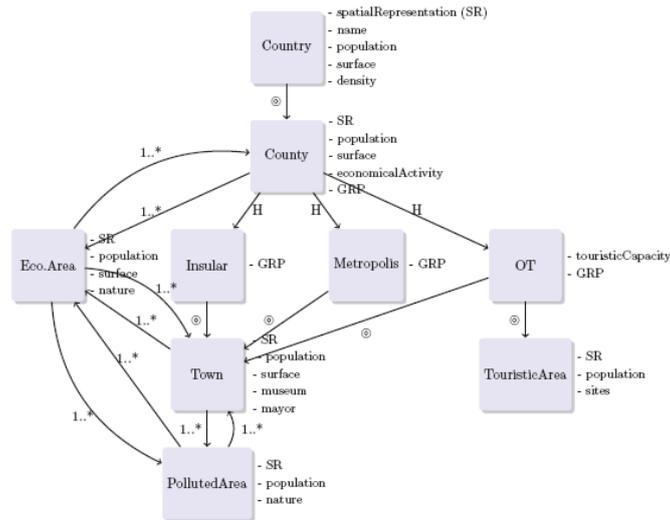
(Class.) attribute	Granule characteristic
name	Subset
Town.population	Global
PollutedArea.population	Global
Town.surface	Global
mayor	Subset

An attribute may have a different granule characteristic depending on its family: ‘extension’ or ‘intention’. As an example, when County.surface is defined as ‘intention’, its characteristic is ‘Subset’. The same attribute, defined as ‘extension’, would have been characterised ‘Global’ since it is not possible to apply this value to a subset of its spatial representation.

### 3.1.2 Edges

In the reference model,  $G_M$ , labels associated with edges ( $\epsilon$ ) model three relationships: heritage, (strict) spatial inclusion and cardinality (1 ...\*). Figure 3 presents an example of the reference model based upon the UML description of Figure 2. In this graph, the three relationships are respectively represented by labelled edges with H,  $\odot$  and 1 ...\*. Labels allow us defining propagation rules of attributes from a class to another one.

**Figure 3** Reference DM example (see online version for colours)



### 3.2 Propagation rules in $G_M$

The label H, models the notion of heritage. This represents the classic notion that a subclass derives from a super-class. That means that a sub-class recovers (public) attributes and functions from its super-class. The overloading of functions is possible within sub-classes. That means it is possible to define a function with the same name as a function in the super-class but with different parameters. So, the overloading allows calling the right function depending on parameters. The super-class is the source class. The sub-class is the target class.

*Rule 1:* Label ‘H’, visibility ‘Public’: Attributes are automatically propagated from the source class to the target class. They still keep the same characteristics.

The label  $\odot$  models the spatial inclusion. This notion implies that instances of a class are spatially included into instances of another one. The including class is the source class. The included class is the target class.

*Rule 2:* Label  $\odot$ , whatever the visibility is: Attributes are propagated from the source class to the target class only and only if they have the ‘Subset’ characteristic. They still keep the ‘Subset’ characteristic (Mainguenaud, 1994).

The label  $1 \dots *$ , models the link between two classes upon a cardinality. A priori, this cardinality let us suppose that a spatial inclusion relationship does not exist. Otherwise it would have been materialised by a  $\odot$  labelled edge. The propagation of an attribute is only possible if the visibility is set to ‘Public’. Two solutions are possible depending on the database administrator’s willingness (to take into account only a logical representation or to take into account the spatial representation).

*Rule 3:* Label  $1 \dots *$ , visibility ‘Public’, the spatial representation is not taken into account: Only attributes characterised as ‘Subset’ are propagated from the source class to the target class. The characteristic associated with the new attribute in the target class is moved from ‘Subset’ to ‘Global’.

In order to deal with the second case, where the spatial representation is taken into account, we need to introduce a new attribute property.

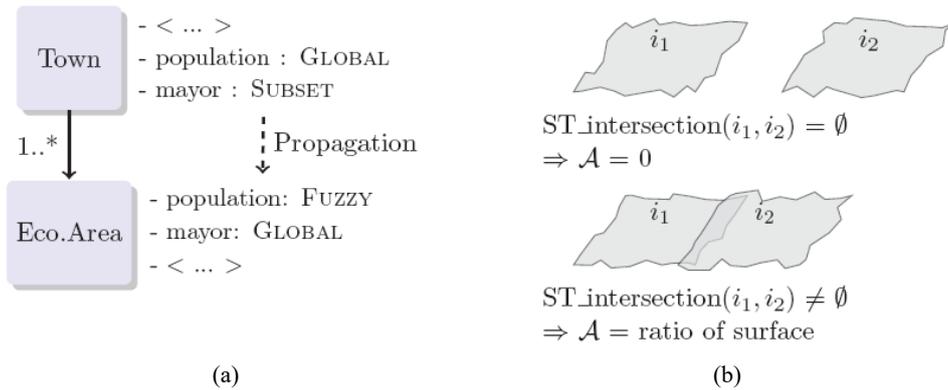
*Definition 5 (Property ‘Fuzzy’ for an attribute):* An attribute is characterised as ‘Fuzzy’ whenever its relevance for a spatial representation is not guaranteed. Its semantic validity is associated with a membership function  $\mathcal{A}$  defined in  $[0, 1]$ .

An example of the membership function  $\mathcal{A}$  may be an intersection ratio of surfaces. Let  $i_1$  be an instance of the target class,  $i_2$  be an instance of the source class. The value of  $\mathcal{A}$  is defined by  $ST\_Area(ST\_Intersection(i_1, i_2)) / ST\_Area(i_2)$ . This evaluation considers that the more important the proportion of  $i_1$  in  $i_2$  is, the more the attribute (from the source class) may be relevant for the target class. The value of the  $\mathcal{A}$  function may be improved by taking into account the semantics of involved attributes/classes or with the integration of external data (in order to precise the value of the function).

*Rule 4* Label  $1 \dots *$ , Visibility ‘Public’, the spatial representation is taken into account: Depending on the spatial intersection of an instance from the source class with an instance of the target class. Therefore, one should apply the same rules as in rule 3. Otherwise, attributes of the source class, characterised as ‘Subset’ are propagated to the

target class. The characteristic associated with the new attribute in the target class is moved from ‘Subset’ to ‘Fuzzy’. Attributes of the source class characterised as ‘Global’ are propagated to the target class. The characteristic associated with the new attribute in the target class is moved from ‘Global’ to ‘Fuzzy’. Attributes of the source class characterised as ‘Fuzzy’ are propagated and they are still characterised as ‘Fuzzy’.

**Figure 4** Attribute propagation and Fuzzy characteristic (see online version for colours)



As an example in Figure 4(a), the propagation of the attribute ‘mayor’ (respectively Town.population) to the EconomicArea class is:

- 1 without taking into account the spatial representation, only the ‘mayor’ attribute is propagated and it is characterised as ‘Global’
- 2 Taking into account the spatial representation, Town.mayor is propagated and it is characterised ‘Global’ and Town.population is propagated and it is characterised as ‘Fuzzy’.

For the ‘Town.population’ attribute, an empty intersection [Figure 4(b)] between the town and the economic area will lead to  $\mathcal{A} = 0$ , otherwise  $\mathcal{A}$  may be estimated as the proportion of the intersection between the two instances (it may be modulated depending on the choice of  $\mathcal{A}$ ).

When two classes are not directly connected (i.e., linked with a path, such as the number of edges is strictly greater than 1), the propagation of attributes does not depend any more on a unique label but it depends on the labels of edges defining the path between these two classes (i.e., a word based on H,  $\odot$ , 1 ... \*). The list of labels on edges defining the path forms a regular expression. The following grammar defined the rules (with an EBNF formalism):

RegularExpression: = Expression +

Expression: = Label Expression +

Label: =  $\odot$  | 1... \* | H

The recursive application, from the source class, of propagation rules associated with labels that constitute the regular expression, allows determining the propagation of an attribute from a class to another one. Tables 3 and 4 sum up the propagation rules for label  $\odot$  and  $1 \dots *$ .

**Table 3** Propagation rules for attributes of a class  $C_1$  to a class  $C_2$  connected with an edge labelled:  $\odot$

$C_1 \setminus C_2$	Characteristics
Subset	Subset
Global	-
Fuzzy	Fuzzy

**Table 4** Propagation rules for attributes of a class  $C_1$  to a class  $C_2$  connected with an edge labelled:  $1 \dots *$

$C_1 \setminus C_2$	Without spatial representation	Taking into account the spatial representation
Subset	Global	$\cap = \emptyset$ : Global $\cap \neq \emptyset$ : Fuzzy
Global	-	$\cap = \emptyset$ : - $\cap \neq \emptyset$ : Fuzzy
Fuzzy	-	$\cap = \emptyset$ : - $\cap \neq \emptyset$ : Fuzzy

In Table 3, the symbol ‘-’ means that an attribute characterised as ‘Global’ of class  $C_1$  will not be propagated to class  $C_2$ . Indeed, by definition, an attribute characterised as ‘Global’ is valid for the whole spatial representation. The  $\odot$  label means the strict spatial inclusion of instances of  $C_2$  in instances of  $C_1$ . Therefore,  $C_2$  represents only a sub part of  $C_1$ . An attribute characterised as ‘Global’ of  $C_1$  cannot be valid for an instance of  $C_2$ . Nevertheless, the propagation of an attribute characterised as ‘Fuzzy’, keeps its characteristic as ‘Fuzzy’ but its membership function  $\mathcal{A}$  is re-defined and gains in reliability. In Table 4, without taking into account of the spatial representation, it is not possible to propagate attributes characterised as ‘Global’ or ‘Fuzzy’ by definition of these characteristics. Taking into account of the spatial representation, the upholding of the characteristic from ‘Fuzzy’ to ‘Fuzzy’ leads to the re-definition of the membership function  $\mathcal{A}$  with a loss of reliability.

#### 4 Intension attributes and their calculus function

Intension attributes are described by their calculus function (i.e., a value of an attribute is defined by a return value of a function). A calculus function is defined in a class with a set of formal parameters. These parameters belong the class itself or sub-/connected classes of this class. The calculus function returns a value (the value of an attribute for a given instance of a class). The value is compatible with the domain on which the attribute is defined. Figure 5 provides examples of such functions.

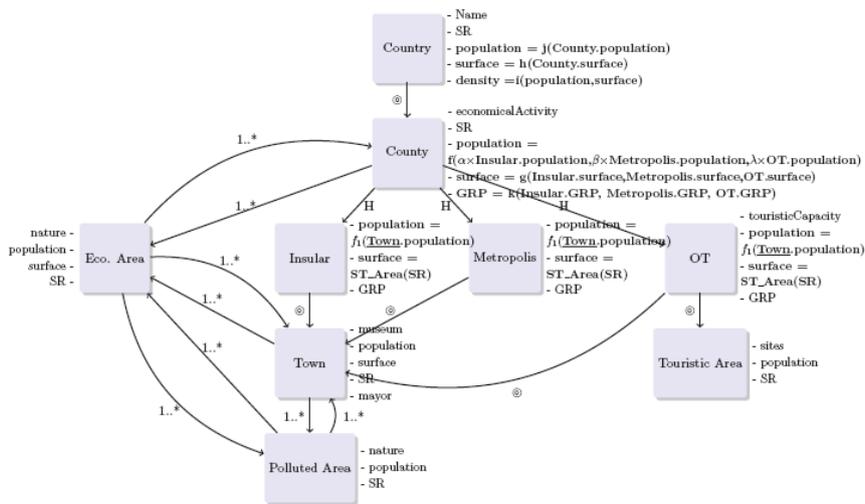
- The calculus function ST\_Area in the class Insular has a parameter that belongs to this class (i.e., spatialRepresentation).
- Using the pointed notation, the  $f_1$  function associated with the Insular class, calculates the value for the ‘population’ attribute. Its parameter is an attribute that comes from a connected class (e.g., linked with an edge labelled  $\odot$  : Town.population).
- The calculus function,  $f$  associated with the County class calculates the value for the ‘population’ attribute. Its parameters are the attributes: Insular.population, Metropolis.population and OT.population that respectively belong to sub-classes Insular, Metropolis and OT. These parameters are respectively coefficiented by  $\alpha, \beta, \lambda$ . These coefficients allow taking into account missing instances of the sub-classes (i.e., we consider that County is able to evaluate the relative proportion of the ‘population’ attribute in each sub-class).

Defining a calculus function implies to specify the operational and analytic specifications. We do not take into account the operational specification that deals with implementation matters. The analytic definition of a calculus function,  $f$ , relies on its signature and its formal specification.

- The signature is conventionally defined by  $f : T_1 \times \dots \times T_p \rightarrow T_{p+1}$  with  $T_i, i \in \{1 \dots p + 1\}$  taken into the set of domains associated with attributes. Each  $T_i, i \in \{1 \dots p\}$  is a formal parameter and could be the result of a recursive definition of a calculus function.
- The specification lists the different involved attributes, and for each of them its domain and its goal.

Calculus functions defined in a database schema have an application-dependent semantics (e.g., the calculus of a population in a demographic application). They can also be studied on their mathematical aspects.

**Figure 5** General structure of a graph  $G_M$  (see online version for colours)



*Definition 6 (Separable function):* A function is characterised as ‘Separable’ if it is possible to apply it to a subset of its parameters (e.g., add or product of its parameters).

As an example in Figure 5 (calculated attributes are in a bold font), the function  $k$  from the County class is ‘Separable’. It is possible to compute a relevant value for County.GRP by adding two of the three parameters (Insular.GRP, Metropolis.GRP and OT.GRP from the target classes Insular, Metropolis, OT) of  $k$ . This example corresponds to the definition of an external schema with a limitation to two classes: Metropolis and Insular.

#### 4.1 Principles for function parameter availability

The analytic definition of a function can be realised a-posteriori (i.e., all required attributes are available) or a priori. In both cases, an automatic availability of formal attributes for a calculus function should be provided. The link from the source class to target classes is materialised by a path in the graph  $G_M$ . The availability is provided by an operation named Inverted Heritage (noted IH). This operation requires the definition of the Inverted Overloading (IO).

*Definition 7 (IO):* The operation of IO allows the overload of ‘Separable’ function in a source class from its target classes.

In other words, this operation allows, if the function is characterised as ‘Separable’ and if formal parameters from target classes are not all available, modifying the signature in order to adapt it to calling conditions.

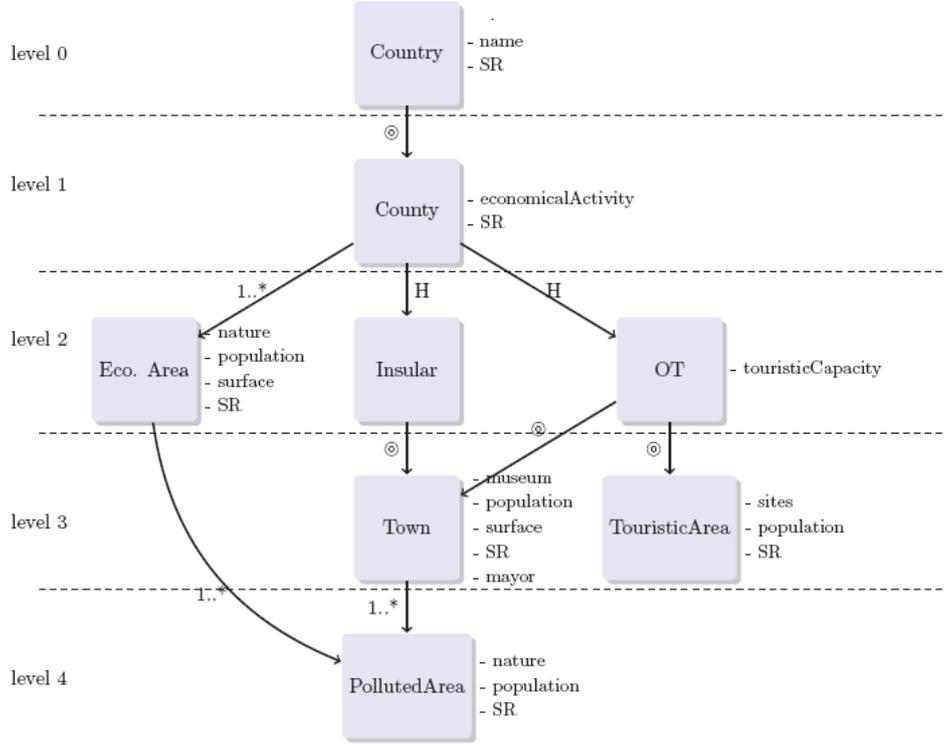
For each calculated attribute in a source class, the IH allows accessing to attributes of other classes (i.e., target classes) involved as a parameter of its calculus function.

*Definition 8 (IH):* The IH is similar to an inverted contravariance property (Cardelli, 1988) focussed on attributes in the graph  $G_M$ . It allows a source class accessing, possibly through the IO, to one or more attribute(s) of a target class (or a target class accessible using a path in  $G_M$ ).

As an example in Figure 5, to compute the attribute County.population, the IH allows getting attributes ‘population’ in sub-classes (Insular, Metropolis, DOM). This is the contrary of conventional heritage process in object model (i.e., a sub-class gets information from its super-class).

It is important to notice that the IH operator and the propagation of attributes from a source class to a target class (via relationships H,  $\odot$ ,  $1 \dots *$ ) are completely distinct. The IH does not manage characteristics of attributes (of a source class) since this characterisation is not required to compute the value of an intension attribute.

External schemas (i.e., view) can be defined depending on users’ needs using conventional data manipulation operators (e.g., selection, projection, join, aggregates) or spatial operators (e.g., intersection, inclusion). Such a schema is defined as a sub-model  $M'$  of a reference model  $M$  such as  $M' \subseteq M$ . An external schema is modelled with a Directed Acyclic Graph (DAG)  $G_{M'}$  (Figure 6). The problem now is to determine the relevant intensions attributes.

**Figure 6** A possible external schema (view)  $M'$  (see online version for colours)

Let  $\mathbf{C}$  be the set of classes involved in  $M'$ . It is possible to define levels in  $G_{M'}$  using a topological sort of its nodes. Therefore, it exists  $n \in \{1 \dots \mathcal{L}\}$  with  $\mathcal{L}$  the maximum number of levels in  $M'$ , such as  $n$  be the level of class  $C$  in  $M'$ . Let  $C^n$  be the class  $C$  of level  $n$  (Figure 6).

In order to illustrate the definition of IH let us consider Figure 6,  $\mathbf{C} = \{\mathbf{Country}, \mathbf{County}, \mathbf{Eco. Area}, \mathbf{Insular}, \mathbf{OT}, \mathbf{Town}, \mathbf{TouristiqueArea}, \mathbf{PollutedArea}\}$ ,  $n = 4$  and, for example,  $C^1 = \mathbf{County}$ .

The goal associated with attributes allows the unification between the specification of an attribute (at the level where the calculus function is defined) and an attribute of a sub-/connected class. The IH operator is formalised as:

- Let  $A_{C^n}$  be a calculated attribute of the class  $C^n$ . To simplify the reading, we note, in the following, a calculated attribute  $A$  instead of  $A_{C^n}$ .

Let  $A_{C^1} = A = \mathbf{County}.population$ .

- Let  $TC(C^n, G_{M'}) = \mathbf{C}_{FT} \subset \mathbf{C}$  be the set of classes obtained from a transitive closure from  $C^n$  on  $G_{M'}$ , and  $\mathbf{C}_A \subseteq \mathbf{C}_{FT}$  be the set of classes involved in the calculus function  $f$ .

$TC(C^1, G_{M'}) = \{\mathbf{Eco.Area}, \mathbf{Insular}, \mathbf{Town}, \mathbf{PollutedArea}, \mathbf{OT}, \mathbf{TouristiqueArea}\} \subset \mathbf{C}$

$$\mathbf{C}_A = \mathbf{C}_{\text{County.population}} = \{\mathbf{Insular}, \mathbf{OT}\}$$

- Let  $N$  be the number of those classes,  $N = |\mathbf{C}_A|$ .

$$N = |\{\mathbf{Insular}, \mathbf{OT}\}| = 2$$

- $\forall C_k \in \mathbf{C}_A, 1 \leq k \leq N$ , let  $Att(C_k, f)$  be the set of attributes of class  $C_k$  involved in the calculus function  $f$ .

$$C_1 = \mathbf{Insular}, C_2 = \mathbf{OT}$$

$$Att(C_1, f) = \{\mathbf{Insular.population}\}, Att(C_2, f) = \{\mathbf{OT.population}\}.$$

- Let  $Att_{\mathbf{C}_A}$  be the set of attributes involved in  $f$ .

$$Att_{\mathbf{C}_A} = \{\mathbf{Insular.population}, \mathbf{OT.population}\}$$

- Let  $L_f$  be the function that casts a set of parameters into a list of formal parameters for a given function  $f$ ,  $A = f(L_f(Att_{\mathbf{C}_A}))$ .

$$\mathbf{County.population} = f(L_f(Att_{\mathbf{C}_A})) = f(\mathbf{Insular.population}, \mathbf{OT.population}).$$

$$\rightarrow \text{The IH is defined by } IH(f, A, C^n) \rightarrow Att_{\mathbf{C}_A} = \bigcup_{k=1}^N Att(C_k, f).$$

#### 4.2 Characteristics of calculus functions

To define two properties that can characterise a function  $f$ , we define a subset of attributes involved in the definition of  $f$  as:

- Let  $\mathcal{P}(\mathbf{C}_A)$  be the set of parts of  $\mathbf{C}_A$ , we define  $SS(\mathbf{C}_A)$  a subset of  $\mathbf{C}_A$  such as  $SS(\mathbf{C}_A) \subseteq \mathcal{P}(\mathbf{C}_A) \setminus \{\emptyset \cup \mathbf{C}_A\}$ .

$$\text{Let } \mathbf{C}_A = \mathbf{C}_{\text{County.population}} = \{\mathbf{Insular}, \mathbf{OT}\}$$

$$\mathcal{P}(\mathbf{C}_A) = \{\{\}, \{\mathbf{Insular}, \mathbf{OT}\}, \{\mathbf{Insular}\}, \{\mathbf{OT}\}\}$$

$$\text{For example, } SS(\mathbf{C}_A) = \{\mathbf{Insular}, \mathbf{OT}\}$$

$$\rightarrow Att_{SS(\mathbf{C}_A)} = \bigcup_{C_k \in SS(\mathbf{C}_A)} Att(C_k, f) \text{ defines the set of attributes involved in classes of } SS(\mathbf{C}_A)$$

$$Att_{SS(\mathbf{C}_A)} = \{\mathbf{Insular.population}, \mathbf{OT.population}\}$$

*Definition 9 (Correlated parameter):* We name correlated parameter, a parameter of a calculus function,  $f$ , such as its expression is recursively defined with attribute of  $\mathbf{C}_A$ , with  $N > 1$ .

*Definition 10 (Correlated calculus function):* A calculus function associated with an attribute is characterised as ‘Correlated’ if at least one parameter is correlated.

*Definition 11 (Coefficiented calculus function):* A calculus function,  $f$ , associated with an attribute is characterised as ‘Coefficiented’ if  $\exists \alpha_1, \dots, \alpha_q \in \mathbb{R} \wedge \exists i \mid \alpha_i \neq 0 \wedge \alpha_i \neq 1$  such as  $A = f(\alpha_1 \times P_1, \dots, \alpha_i \times P_i, \dots, \alpha_q \times P_q)$  with  $P_i \in Att_{SS(C_A)}$ ,  $1 \leq i \leq q$  and  $q = |Att_{SS(C_A)}|$ .

*Definition 12 (Separability of a calculus function):*

- A calculus function  $f$  is characterised as ‘SEP’ if  $f$  is separable, that means:
  - a if it is possible to define  $L_f(Att_{SS(C_A)})$  and to apply  $f$  to this sub-list of attributes.
  - b or if  $f$  is characterised as ‘Correlated’ and whatever the correlated parameter of  $f$ , all (or none) of the attributes involved in this parameter belong to  $L_f(Att_{SS(C_A)})$ .
- In the opposite case or if  $f$  is characterised as ‘Coefficiented’,  $f$  is not separable and is characterised as ‘NSEP’.

The characteristic of a calculated attribute (i.e., Global, Subset, Fuzzy) is independent from the separability or not of its calculus function (e.g., Country.population and its calculus function  $j$  are respectively characterised as Global and SEP although Country.surface and its calculus function  $h$  are respectively characterised as Subset and SEP).

**Table 5** Characteristics of calculated attributes

<i>Attribute (per class)</i>	<i>Calculus function parameter(s)</i>	<i>Function separability</i>	<i>Characteristic of a calculated attribute</i>
<i>Country</i>			
population	$j$ : County.population	SEP	Global
surface	$h$ : County.surface	SEP	Subset
density	$i$ : population, surface	NSEP	Global
<i>County</i>			
population	$f$ : $\alpha \times \mathbf{Insular}$ .population, $\beta \times \mathbf{Metropolis}$ .population, $\lambda \times \mathbf{OT}$ .population	NSEP	Global
surface	$g$ : $\mathbf{Insular}$ .surface, $\mathbf{Metropolis}$ .surface, $\mathbf{OT}$ .surface	SEP	Subset
GRP	$k$ : $\mathbf{Insular}$ .GRP, $\mathbf{Metropolis}$ .GRP, $\mathbf{OT}$ .GRP	SEP	Global
<i>Insular/metropolis/OT</i>			
population	$f_1$ : $\mathbf{Town}$ .population	SEP	Global
surface	ST_Area, SR	SEP	Subset

Table 5 presents properties associated with calculated attributes. It is not possible to automatically associate a characteristic with an attribute from the characteristics of its calculus function parameters. The knowledge of the semantics of the function is mandatory. As an example, let us define two functions:

- Pollution (**Insular**.pollution, **Metropolis**.pollution, **OT**.pollution) as the ‘Indices of pollution by m<sup>2</sup>’, using as parameters indices of pollution by m<sup>2</sup> (respectively for each subclass of county).
- Climate (**Insular**.pluviometry, **Metropolis**.pluviometry, **OT**.pluviometry) as the ‘Climate of a total area’, using as parameters the pluviometry for a given area. Let us suppose this parameter is constant for an instance whatever the county class is. To simplify the definition of the function we do not consider the existence of micro-climate.

Characteristics associated with the attributes are:

- Pollution (Subset, Subset, Subset) → Global
- Climate (Subset, Subset, Subset) → Subset.

For two functions using parameters with the same characterisation, characteristics associated with calculated attributes may be different. Such a counter-example implies an a-priori definition for each calculated attribute. The database schema designer is in charge of this task.

Figure 7 The three steps for the treatment of attributes

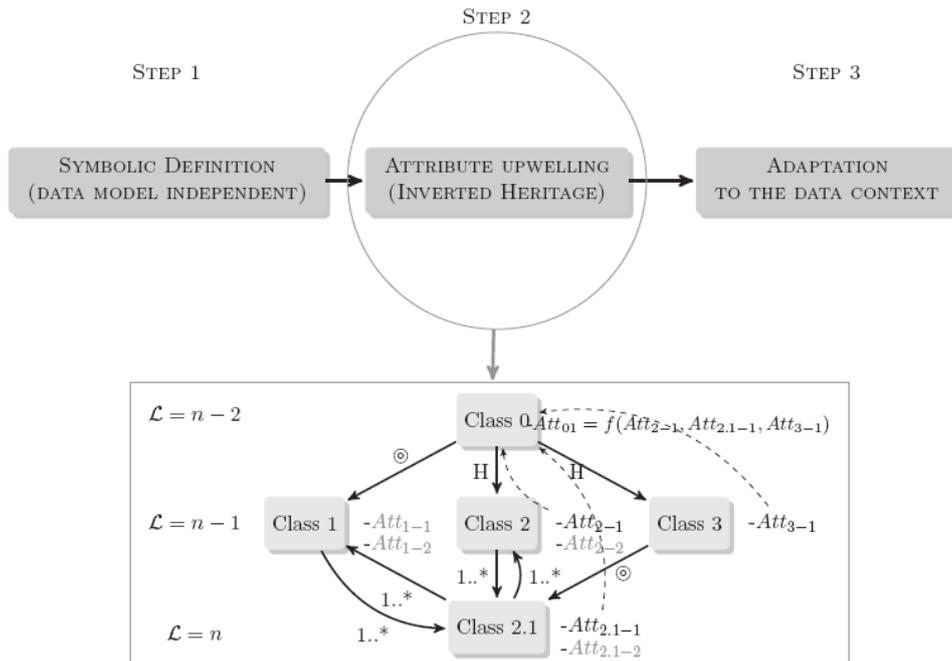
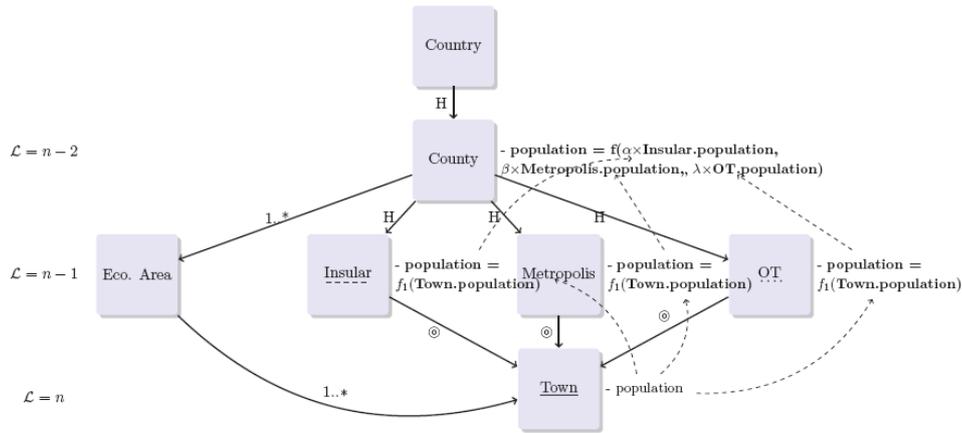


Figure 7 represents the three conceptual steps of our approach. The first step consists in specifying the view the end user wants to manipulate. It defines the extension attributes. The analytical definition of the different functions is provided as soon as all required attributes are available. The second step is the definition of analytical functions using the

IH. The IH automatically provides attributes used as parameters of a calculus function. The last step adapts results to the desired specificities of the DM (e.g., managing international unities).

This generic representation is illustrated Figure 8. Two cases of IH are defined: mono-class and multi-classes. The first one is provided by the function  $f_1$  which needs a single class (i.e., Town). The second one is provided by function  $f$  which needs three classes (Insular, Metropolis and OT). Dashed arrows represent the application of the IH. Starting from the data definition model and its dynamic, we propose a data manipulation operator to manage the variations of abstraction levels. We provide the *DSZI* operator to move from a level of granularity to another one.

**Figure 8** Step 2 applied to an example (see online version for colours)



## 5 Top down semantic information transfer

Complementary data is very often required when changing from abstraction levels. As an example, more precise information is provided with application such as ‘Google Earth’ when applying a zoom operation. More precise information is provided for example in route planning applications such as for example the translation from the name of a road (e.g., E138) to a street name in a town (e.g., Central Station Street). Our goal is to formalise the DM associated with the results of an operation providing the top-down granularity level change. Our level of definition is the conceptual model. In his compilation approach (Stell and Worboys, 1998) defines the relationship between several levels of abstraction (Sheaf of SMS) by the definition of the  $\rho$  function  $\rho : Net(G_1, v_1, e_1, \gamma_1) \rightarrow Net'(G_2, v_2, e_2, \gamma_2)$ . We propose to move from the representation of a network to another one by adding information. We define the dynamic semantic ZoomIn function (DSZI) with the same semantics as  $\rho$  but with a dynamic evaluation (interpreted). The dynamicity of this operation is based on selection criteria.

*Definition 13 (Selection criteria):* The selection criteria of the DSZI operator is the definition of properties that should be verified by elements involved in the networks used in the DSZI operator.

A selection criteria, associated with the DSZI operator, may be structural (i.e., based on the attributes of classes defined in the reference model  $M$ ) or based on instances. The involved operators may be a projection (i.e., nodes of the graph  $G_M$ ) or a selection (i.e., instances of the different classes). A selection criteria,  $c$ , may be:

- Semantic:
  - 1 Structural: e.g., the set of counties.
  - 2 Instance: e.g., towns such as their population are greater than 100,000.
- Graphic:
  - 1 Structural: e.g., surfacic elements included in OTs ('Town').
  - 2 Instance: e.g., Union of polluted area (whatever their natures are) in a given perimeter to provide a unique area.
- Mixed: e.g., counties such as the economical activity is 'Tourism' with a polluted area whatever the nature is.

This leads us to define a DSZI operator with selection criteria.

*Definition 14 (DSZI):* Let  $Net$  and  $Net'$  be two networks,  $Net = (G, \nu, \epsilon, \gamma)$  and  $Net' = (G', \nu', \epsilon', \gamma')$ , and a selection criteria,  $c$ , defined on the model  $(\nu, \epsilon, \gamma)$ . The DSZI operator is defined by:  $DSZI : Net \times c(\nu, \epsilon, \gamma) \rightarrow Net'$ .

The DSZI operator allows starting from a granularity level associated with  $M = (\nu, \epsilon, \gamma)$  to join a new granularity level associated with  $M' = (\nu', \epsilon', \gamma')$ . The DSZI operator takes into account configurations with a unique source class (i.e., target classes that belong to the transitive closure of a unique source class in the reference graph).

The first step is to define the principles of attribute availability, then we study the different cases of projection and selection.

### 5.1 Principles for attribute availability

Depending on attribute properties (Definitions 4 and 5), we showed that rules associated with the propagation process allow transferring or not attributes from a source class to a target class. When applying the DSZI operator, this propagation process has already been performed (for attributes in involved classes of the DSZI operator).

In the case of a calculated attribute, its value, at a given granularity level, may not be valid any more after the application of a DSZI operator. It may be relevant to delete, to update, to re-compute in a new attribute or to freeze an attribute value. Principles for attribute availability are used to solve this problem. Attribute availability principle depends on the property of a calculus function (Definition 12). This property determines the calculated attribute and parameters of the function. These principles do not rely on the directed edge labelling function, therefore to simplify the presentation these labels are erased in the following.

*Definition 15 (Frozen attribute):* The characterisation as 'Frozen' of an extension attribute forbids its update. An intension attribute is automatically characterised as 'Frozen' (due to its definition). A direct update of this attribute is therefore forbidden.

Nevertheless, the update of any parameter of the calculus function implies the automatic update of this attribute.

The notion of ‘Frozen’ attribute for an extension attribute is between the notion of snapshot and the notion of view in relational Database Management Systems. In a snapshot, values of attributes are available but updates are not allowed. Through a view, updates on database are allowed on certain conditions. In our case, the approach is similar to a snapshot since we do not allow updates on an extension attribute characterised as ‘Frozen’. It is similar to a view since updates on the view (not on the database) are allowed under certain conditions. The DSZI operation creates a ‘snapshot-view’ of the database, every frozen attribute in a view is not frozen in the database. If a new application of DSZI is performed, it takes into account the ‘frozen’ property for these attributes.

The management of a calculated attribute during a DSZI operator depends on:

- The set of attributes on which the projection/selection is defined (i.e., IH).
- The set of classes on which the projection/selection is defined. In particular, the number of such classes (attributes involved in a calculus function belong to these classes). This number is denoted by  $\#C_i < N$ .

We distinguish two cases: the projection and the selection in order to present different configurations. Projection and selection are classical database operations. The DSZI operator is defined on a subset of classes (Figure 6) with a projection operator. It can also be defined on a subset of instances (i.e., a selection operator). We illustrate these principles with the model defined Figure 8 (sub-model of the model presented Figure 3).

## 5.2 Projection case

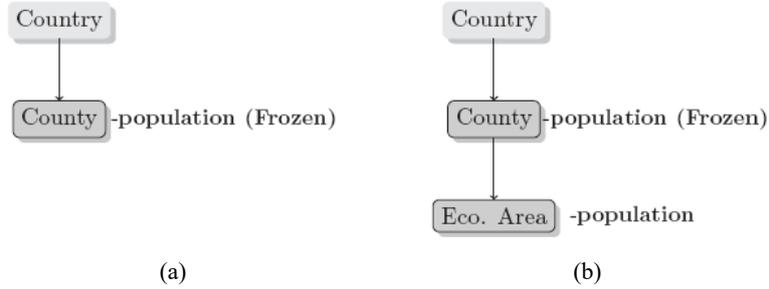
The projection, similar to the operator of projection in the relational algebra, consists in defining a subset of accessible classes using a path on the reference model from a unique class, name source class. Several cases are defined depending on the value of  $\#C_i$ :  $\#C_i = 0$ ,  $\#C_i = 1 < N$ ,  $1 < \#C_i < N$  and  $\#C_i = N$ .

### 5.2.1 Case 1: $\#C_i = 0$

In this case, no class involved in parameters of the calculus function is present in the projection. The concerned attribute is characterised as ‘Frozen’. As an example in the Figure 9(a),<sup>1</sup> the calculus function associated with County.population attribute involves attributes of the classes Insular, Metropolis and OT. In fact none of these classes are defined in the DSZI operator. In this case the County.population attribute is characterised as ‘Frozen’. This solution is also applied in Figure 9(b). The calculus function associated with the County.population attribute cannot be computed since Insular, Metropolis and OT classes are not available (EconomicArea class is not a class involved in the calculus function, i.e.,  $\text{EconomicArea} \notin C_A$ ). The attribute economicArea.population is completely independent of County.population, it could be updated (partial snapshot).

*Rule 5:* If  $\#C_i = 0$  then the attribute of the source class is characterised as ‘Frozen’. None of the attributes of the target class are concerned.

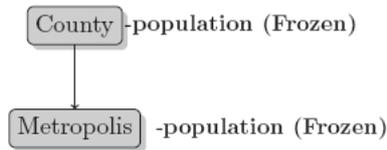
**Figure 9** Case 1: two examples of  $\#C_i = 0$



### 5.2.2 Case 2: $\#C_i = 1 < N$

As soon as the DSZI operator is applied to a single target class, no re-computation, in case of a calculus function characterised as ‘SEP’ is needed for the class on which the projection is applied. A re-computation at the class level (e.g., County, Figure 10) leads to duplicate data in comparison with the target class (e.g., Metropolis.population). The attribute of the class (e.g., County.population) is characterised as ‘Frozen’ to keep information obtained by the initial set of classes involved in the calculus function. This choice enforces the coherency since the characterisation by ‘Frozen’ forbids any update on this attribute. As an example, County.population and Metropolis.population cannot be updated (integral snapshot) since the value of County.population is evaluated from Metropolis.population with a calculus function that cannot be applied any more since its parameters are not available.

**Figure 10** Case 2:  $\#C_i = 1$



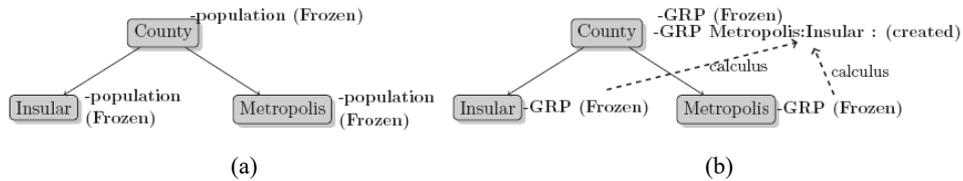
*Rule 6:* If  $\#C_i = 1 < N$  then the attribute of the source class is characterised as ‘Frozen’ and attributes of target classes involved in the calculus function are also characterised as ‘Frozen’.

### 5.2.3 Case 3: $1 < \#C_i < N$

As soon as the number of classes involved in the DSZI operator is strictly greater than one and strictly less than  $N$ , it may be interesting to evaluate the new value of an attribute depending on selected classes. To be evaluated, the calculated attribute must be defined with a calculus function characterised as ‘SEP’. In the opposite case, the concerned attribute must be characterised as ‘Frozen’ [e.g., County.population, Figure 11(a)]. The value of the (re-calculated) attribute is managed by a new attribute. Its goal depends on target classes of the projection. The new attribute is linked to the subset of initially

available classes while calculating the function in the source class. As an example in Figure 11(b), the goal of the GRP at the County level should be: ‘GRP of the county’. In case of a re-calculus of its value in a DSZI operator applied to Metropolis and Insular classes, for example, its new goal would not be anymore relevant with its new representation (i.e., after the DSZI operator, only the GRP of metropolitan and insular counties) GRP at the County level should be: ‘GRP of the county’. In case of a re-calculus of its value in a DSZI operator applied to Metropolis and Insular classes for example, its new goal would not be anymore relevant with its new representation (i.e., after the DSZI operator, only the GRP of metropolitan and insular counties).

**Figure 11** Case 3:  $1 < \#C_i < N$



While applying the DSZI operator, the attribute of the source class (e.g., County.GRP) is characterised as ‘Frozen’ for the same reasons as the previous case (case 2). Another attribute of this class (e.g., Metropolis:Insular.GRP) with the same goal and same calculus function of the ‘Frozen’ attribute is created. Nevertheless, only attributes of target classes involved in the DSZI operator are used. As an example in Figure 11(b), the goal of the new attribute is GRP of the county (reduced to Metropolis and Insular). Its value is the result of the calculus function applied to Metropolis.GRP and Insular.GRP.

The goals of County.GRP and Metropolis:Insular.GRP represent the same concept and their difference is only based on classes on which they are applied. The calculus functions of these two attributes are similar, except that Metropolis:Insular.GRP is evaluated with target classes of the DSZI operator.

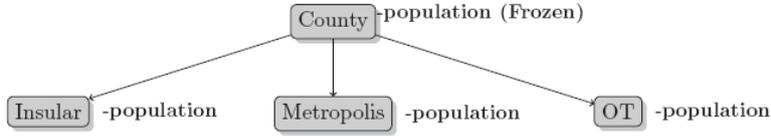
*Rule 7:* If  $1 < \#C_i < N$  then the attribute of the source class is characterised as ‘Frozen’. If the calculus function is characterised as ‘SEP’, a new calculated attribute is defined in the database schema from the subset of attributes of involved classes in the projection. The attributes of the target classes are characterised as ‘Frozen’.

#### 5.2.4 Case 4: $\#C_i = N$

As soon as, the DSZI operator involves the whole set of classes present in a calculus function, the update is authorised on target classes but it is forbidden on the source class.

As an example in Figure 12, the calculus function defined for County.population attribute involves all classes defined in the DSZI operator (i.e., Insular, Metropolis, OT). For example, an update on OT.population can be possible but an update on County.population is not permitted due to its definition (i.e., intension). The first update leads to a re-computation of the County.population attribute. In the second update, it cannot be possible to propagate the new value (or part of the new value of County.population) within target classes.

Figure 12 Case 4:  $\#C_i = N$



Rule 8: If  $\#C_i = N$  then the attribute in the source class is characterised as ‘Frozen’ and none of the attributes of target classes are concerned.

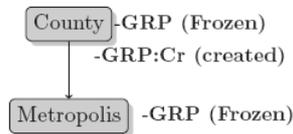
As an example, the attribute GRP:Insular of the Metropolis class [Figure 11(b)] is characterised as ‘Frozen’ (due to its definition -intension-). It could not be updated since parameters of the calculus functions (Insular.GRP and Metropolis.GRP) are characterised as ‘Frozen’. They could not be updated (due to their characterisation).

### 5.3 Selection case

To be relevant, the selection case must concern classes that are involved in the calculus function of the defined attribute. Therefore, we do not consider the case:  $\#C_i = 0$ .

Figure 13 presents the case:  $\#C_i = 1$ . Let Cr be a selection criteria on instances of the target class Metropolis. The calculated attribute County.GRP is characterised as ‘Frozen’. A new attribute County.GRP:Cr is created. This attribute models instances that respect the selection criteria Cr. The attribute of the target class defined with the calculus function (i.e., Metropolis.GRP) is characterised as ‘Frozen’.

Figure 13 Selection case with a selection criteria – Cr



Cases such as  $1 < \#C_i < N$  and  $\#C_i = N$  are similarly treated. The only difference is that the calculus of the value for the new attribute is based on several classes.

### 5.4 Operational architecture for DSZI

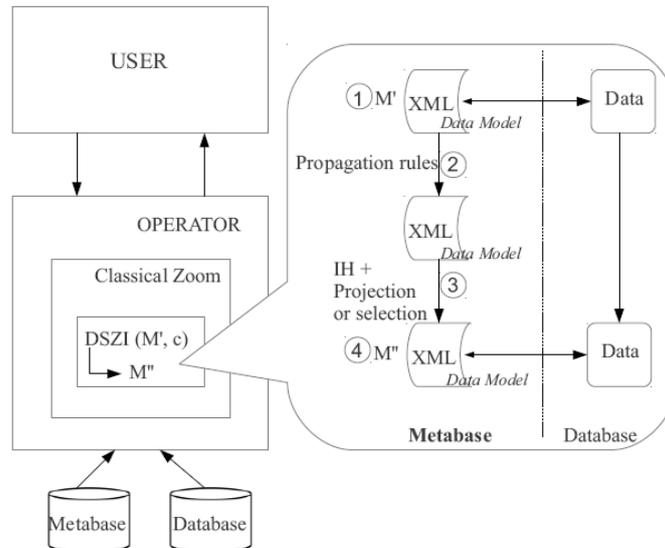
Figure 14 presents an operational architecture in order to validate the operation of DSZI. A user chooses to apply an operator on its data. In our case, this operator is the DSZI. The process from the application of this operator is as follow (steps are related to numbers in Figure 14):

- 1 an XML document contains the DM  $M'$  related to the user view of the metabase
- 2 apply propagation rules on  $M'$  in order to get the right attributes characteristics according to the available information in the view

- 3 perform IH and apply the criteria  $c$  (based on a projection or a selection)
- 4  $M''$  is the generated DM by the *DSZI* operator.

At this time we have implemented this part, with a criteria based on projection (left part of *DSZI* in Figure 14). Next step is to connect this component applied only on the metabase with the database, in order to get the instances (right part of *DSZI* in Figure 14).

**Figure 14** Operational architecture



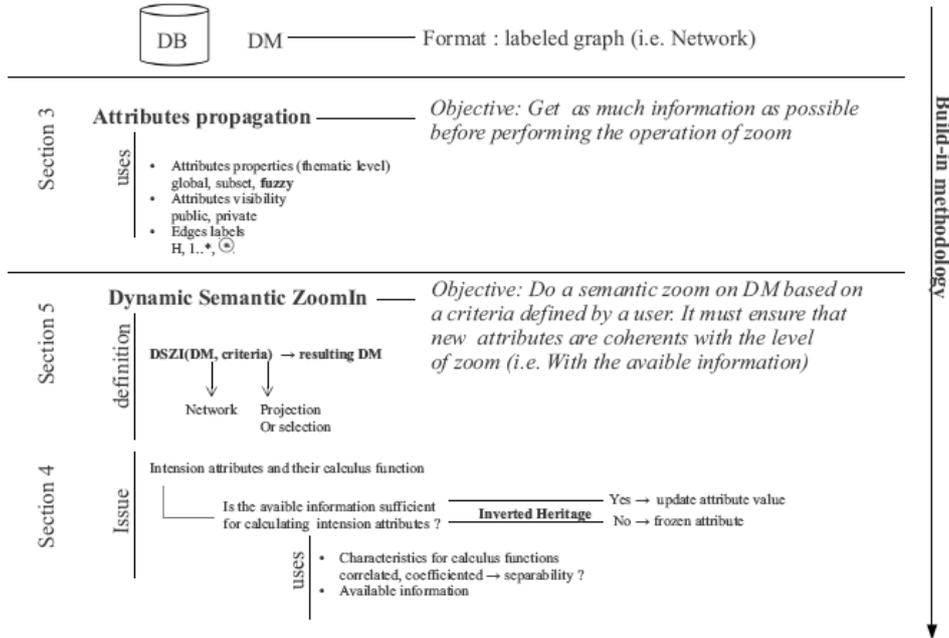
## 6 Conclusions

Due to the overwhelming development of navigation facility services on different devices (e.g., car, mobile phone), the availability of semantic data to enrich cartographic representation is a major challenge. Nowadays commercial propositions focus on cartographic representation of the navigation, providing factual data (conventional alphanumeric data) is still an open problem. The state of the art proposes numerous solutions with a cartographic starting point. The alphanumeric DM is less studied. We propose in this article a contribution in that sense.

The whole methodology and concepts discussed in the article are illustrated Figure 15. We start from a conventional database model. We propose in a first step a modelling with a labelled graph and characteristics on alphanumeric data associated with their spatial validity (Global, Subset, Fuzzy). We introduce three specific relationships (edges labels): heritage, strict spatial inclusion and the cardinality relationship  $1 \dots *$ . On this model, propagation rules are defined. They are applied as soon as a change in the semantic granularity occurs. These propagations depend on the spatial semantics of involved data and involved relationships. Calculated attributes (i.e., the definition of a calculus function) lead us to define rules depending on characteristics of their function (separable or not). In order to access to required data to evaluate a function, we define the

operator of IH. This operator provides required parameters to evaluate a function (from the lower classes).

Figure 15 Detail of the building methodology



Starting from the model and its propagation rules of alphanumeric data, we introduce the DSZI operator. This operator allows defining semantic information that should be transferred from one level of granularity to another one respecting a set of rules. This operator is closed under the DM (i.e., the results of an operator may be used as an operand of another operator). This property allows composing operators to define more complex queries or derived operators.

Once the model is defined, we can extend available operators, such as, for example, the ‘opposite’ DSZI operator, the Dynamic Semantic ZoomOut operator, or an operator that could provide a merge between top-down and bottom-up approaches, a sticking operator.

## References

- Aiken, A., Chen, J., Lin, M., Spalding, M., Stonebraker, M. and Woodruff, A. (1995) ‘The tioga-2 database visualization environment’, in Wierse, A., Grinstein, G.G. and Lang, U. (Eds.): *Database Issues for Data Visualization, IEEE Visualization ‘95 Workshop, Lecture Notes in Computer Science*, 28 October 28, Vol. 1183, pp.181–207, Springer, Atlanta, Georgia, USA.
- Aiken, A., Chen, J., Stonebraker, M. and Woodruff, A. (1996) ‘Tioga-2: a direct manipulation database visualization environment’, in Su, S.Y.W. (Ed.): *Proceedings of the Twelfth International Conference on Data Engineering*, 26 February–1 March, pp.208–217, IEEE Computer Society, New Orleans, Louisiana.

- Balley, S. (2007) 'Description of data structures and interoperability: Why and how to describe spatial datasets structures?', *23rd International Cartographic Conference (ICC'07)*, 4–10 August, Moscow, Russia [online] [http://icaci.org/files/documents/ICC\\_proceedings/ICC2007/html/Proceedings.htm](http://icaci.org/files/documents/ICC_proceedings/ICC2007/html/Proceedings.htm) (accessed 12 February 2015).
- Bobzien, M. and Morgenstern, D. (2003) 'Abstracting and formalizing model generalization', *7th ICA Workshop on Progress in Automated Map Generalization*, Paris, France, 28–30 April [online] <http://generalisation.icaci.org/index.php/prevents/52-workshop-2003-papers>.
- Buttenfield, B. and McMaster, R. (1991) *Map Generalization: Making Rules for Knowledge Representation*, 245pp, Longman Scientific & Technical, Wiley eds., Essex, UK.
- Camossi, E. (2005) *Spatio-Temporal Multigranularity in an Object Data Model*, PhD thesis, University of Milan.
- Cardelli, L. (1988) 'A semantics of multiple inheritance', *Information and Computation*, Vol. 76, Nos. 2/3, pp.138–164.
- Claramunt, C. and Mainguenaud, M. (1999) 'A revisited database projection operator for network facilities in a GIS', *Informatica*, Vol. 23, No. 2, pp.187–202, Slovenia.
- Follin, J.M., Bouju, A., Bertrand, F. and Boursier, P. (2004) 'Visualization of multi-resolution spatial data in mobile system', *Geographic Information Sciences Journal of GIS*, Special issue on LBS and GIS, Vol. 10, No. 2, pp.106–116.
- Fonseca, F., Egenhofer, E., Davis, M. and Camara, G. (2002) 'Semantic granularity in ontologydriven geographic information systems', *Annals of Mathematics and Artificial Intelligence – Special Issue on Spatial and Temporal Granularity*, Vol. 36, Nos. 1–2, pp.121–151.
- Hornsby, K. and Egenhofer, M.J. (1999) 'Shifts in detail through temporal zooming', *DEXA '99: Proceedings of the 10th International Workshop on Database & Expert Systems Applications*, Florence, Italy, pp.487–491, IEEE Computer Society, Washington, DC, USA.
- Mackaness, W., Ruas, A. and Sarjakoski, L. (2011) *Generalisation of Geographic Information: Cartographic Modelling and Applications*, International Cartographic Association, Elsevier Science, Amsterdam, The Netherlands.
- Mainguenaud, M. (1994) 'Consistency of spatial database query results', *Computer Environment and Urban Systems*, Vol. 18, No. 5, pp.333–342.
- McMaster, R. and Shea, K. (1992) *Generalization in Digital Cartography*, Association of American Geographer, Washington, USA.
- Minsky, M. (1981) 'A framework for representing knowledge', in Haugeland, J. (Ed.): *Mind Design: Philosophy, Psychology, Artificial Intelligence*, pp.95–128, MIT Press, Cambridge, MA.
- Muller, J.C., Weibel, R., Lagrange, J.P. and Salgé, F. (1995) *Generalization state of the art and issues in GIS and Generalization: Methodology and Practice*, Chapter 1, pp.3–17, Taylor and Francis, London, UK.
- Mustière, S. (2006) 'Results of experiments on automated matching of networks at different scales', *ISPRS WG II/3, II/6 Workshop 'Multiple Representation and Interoperability of Spatial Data'*, 22–24 February, Hanover, Germany [online] <http://www.isprs.org/proceedings/XXXVI/2-W40/> (accessed 12 February 2015).
- Stell, J. and Worboys, M. (1998) 'Stratified map spaces: a formal basis for multi-resolution spatial databases', *SDH'98 Proceedings 8th International Symposium on Spatial Data Handling*, pp.180–189, Vancouver, Canada, International Geographical.
- Stoter, J.E., Morales, J.M., Lemmens, R.L.G., Meijers, B.M., van Oosterom, P.J.M., Quak, C.W., Uitermark, H.T. and Brink, L. (2008) 'A data model for multi-scale topographical data', in Ruas, A. and Gold, C. (Eds.); *Headway in Spatial Data Handling, Lecture Notes in Geoinformation and Cartography*, Chapter 14, pp.233–254, Springer, Berlin Heidelberg.

- Vangenot, C. (2004) 'Multi-representation in spatial databases using the MADS conceptual model', *ICA Workshop on Generalisation and Multiple Representation*, Leicester, UK, 20–21 August 2004 [online] <http://generalisation.icaci.org/index.php/prevents/11-previous-events-details/54-workshop-2004-papers-and-program>.
- Volz, S. (2005) 'Shortest path search in multi-representation street databases', in Gartner, G. (Ed.): *Proceedings of the 3rd International Symposium on Location Based Services and Tele-Cartography*, 28–30 November, pp.1–10, Eigenverlag, Vienna, Austria.

## **Notes**

- 1 The class Country is not directly relevant in this example its purpose is to fix ideas in conjunction with Figure 8.