A Multi-representation Spatial Data Model

Sheng Zhou and Christopher B. Jones

Department of Computer Science, Cardiff University
Cardiff, CF24 3XF, United Kingdom
{S.Zhou,C.B.Jones}@cs.cf.ac.uk

Abstract. Geo-referenced information is characterised by the fact that it may be represented on maps at different levels of detail or generalisation. Ideally a spatial database will provide access to spatial data across a continuous range of resolution and multiple levels of generalisation. Existing work on multi-resolution databases has treated generalisation control as one-dimensional. Here we extend the concept of multi-resolution spatial databases to provide support for multiple representations with variable resolution. Therefore the controls on generalisation become multi-dimensional with spatial resolution as one dimension and various types of generalisation style metrics as the other dimensions. We present a multi-representation spatial data model based on this approach and illustrate the implementation of multi-representation geometry in association with an online web demonstration.

1 Introduction

1.1 Multiple Representations of Geographical Phenomena

Geometric objects in spatial databases and GIS are representations of real world geographical phenomena. A single phenomenon may have multiple representations reflecting different perspectives of the observer. The observer's perspective has an aspect of **scale**, which is linked to **resolution** and introduces a limit on the maximum geometric information that may be represented for a particular phenomenon, as well as an aspect of **generalisation criteria** (**GC**) mainly reflecting the purposes of map compilation. Generalisation criteria may be adapted for example to map specifications for topographical maps or for different types of thematic map. They can be associated with **generalisation metrics** (**GM**) in the respective compilation and generalisation procedures, and may be interpreted as the degree of reduction relating to the maximum information that may be represented. Scale and generalisation criteria together control the form of the geometric objects representing a phenomenon as well as the nature of non-spatial properties associated with the phenomenon. In addition, there is also the issue of temporal multiplicity of representations that reflect change of geographical phenomena over time. While fully acknowledging the importance of this issue, we confine our focus here to the problems associated with space.

T. Hadzilacos et al. (Eds.): SSTD 2003, LNCS 2750, pp. 394-411, 2003.

Fig.1 presents an example of multiple geometric representations of a single geographical phenomenon (Isle of Wight, UK). Representation **A** preserves maximum information, as may be found in a topographical map. The series **A**, **B** and **C** are three representations at the same scale/resolution but under different generalisation criteria. **B** is generalised from **A** assuming criteria for thematic maps in which small details are removed. **C** is further generalised from **B** for use for example in a newspaper illustration, with only large details retained. The other series, of **A**, **D** and **E,** demonstrates the impact of scale/resolution change while the same generalisation criteria (i.e. criteria for a topographical map) remain in effect. As scale decreases, maximum information at a certain scale is preserved in the corresponding representations with only redundant data being removed.

Beyond a single phenomenon, scale/resolution and generalisation criteria will also determine what types of phenomena and which particular phenomena will be represented. This results in another level of representational multiplicity corresponding to object selection. Furthermore, the multiplicity of representations is reflected in differences in a phenomenon's non-spatial attributes and the values of these attributes.

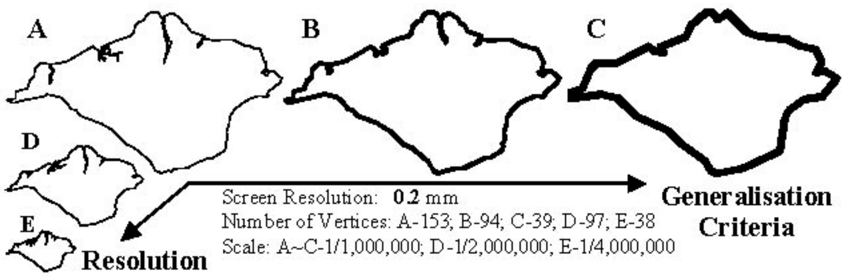

Fig. 1. Multiple geometric representations for a single geographical phenomenon (source dataset derived from original Ordnance Survey map, Crown copyright 2002)

1.2 From MV-SDB and MS-SDB to MRep-SDB

As different representations of geographical phenomena reflect different perspectives and serve different purposes, the efficacy of a spatial database will be increased significantly if multi-representation of geographical phenomena is supported. For example, a web map server could provide "active" maps that are adaptive to users' diverse interests as well as to the varying presentation characteristics of different browsing devices (e.g. printers, desktop CRTs, PDAs and WAP mobile phones).

To an extent, the functionality of multi-representation may be supported in current GIS and spatial databases by simply storing a collection of maps each of which reflects a pre-defined perspective. This approach, which we term **simple multi-version spatial database** or **MV-SDB**, may provide simple and efficient solutions to some applications. However, for many other applications it has severe problems such as potential inconsistency among different maps, high storage/update overhead, and most significantly, lack of flexibility in control over the user's perspective.

In recent years, multi-scale/resolution spatial databases (**MS-SDB**) have attracted increasing interest. On the theoretical side, for example, [1] described a formal model which represents the multi-resolution nature of map features through continuous functions over abstract cell complexes that model the combinational structure of maps. A model for a scale-based space partition hierarchy is described in [2]. “Stratified map spaces” are proposed in [3] as a formal basis for multi-resolution spatial databases. Graphs are used in [4] to describe amalgamation and selection operations caused by resolution change. Examples of experimental implementation include the PR-file [5] for multi-resolution access, and multi-scale geometry [6].

Although MS-SDB support representational multiplicity on the scale/resolution dimension, they do not provide support for variations in generalisation criteria as would be expected of a genuine multi-representation spatial database (**MRep-SDB**). The ideal MRep-SDB will be like every cartographer’s dream: given a query scale/resolution value drawn from a **continuous** range and an **arbitrary** set of generalisation criteria reflecting the purpose of the query, a map that exactly meets the requirements is retrieved automatically (and efficiently) with a quality matching (or at least close to) the quality of one generalised by an expert cartographer from a master map. In short, an ideal MRep-SDB should support **on-demand**, **on-the-fly** and **high quality** spatial data retrieval.

Progress (mainly on theoretical aspects) has been made in providing better support in spatial databases for multi-representations involving semantic criteria. For example, [7] discussed the issues of value and entity hierarchies relevant to multi-representation. [8] described a “VUEL” (View Element) model to handle multiple representations for applications such as spatial OLAP. In [9] a multiple representation schema language was introduced for modeling multiple representations, matching objects and maintaining consistency. An extensive survey on a wide range of issues relating to multi-representation was carried out by the MurMur project team[10].

A fundamental issue in designing an MRep-SDB is how to integrate (potentially very large numbers of) multiple representations of the same phenomenon at different levels of detail and under various criteria. The approach currently adopted by most schemes may best be described as a **linked multi-version** approach. Here links are provided between different fixed representations, while intermediate representations, which may be required by services such as intelligent zooming, are interpolated or extrapolated from existing representations [11] by online generalisation procedures.

Because the *status quo* of automatic map generalisation is far from meeting the quality and performance demands to online (or even off-line) generalisation, while acknowledging that map generalisation procedures are the proper tools for generating multi-representations of spatial phenomena, we argue that as much generalisation workload as possible should be moved to a pre-processing stage in order to achieve good performance and maximum flexibility simultaneously. Results of generalisation can then be stored explicitly inside an MRep-SDB to facilitate retrieval and minimize requirements for any post-query generalisation.

In the remainder of this paper, we present a multi-representation spatial data model based on multi-representation geometry. Experimental results of a method to generate multi-representation geometry using a new line generalisation metric are reported and an on-line web demo has been set up. An approach to modelling multiplicity among a set of objects is also presented. Some issues relevant to the design of MRep-SDB design are also addressed briefly.

2 Cartographic Semantics and Multi-representation

Cartographic semantics describe the relation between geographical phenomena and their database representations. In this section, we will discuss those aspects of cartographic semantics relating to our multi-representation spatial data model.

2.1 Scale and Spatial Resolution

Scale (S_{rep}) is the ratio of the physical extent of the presentation medium (*ME*) and the real world extent (*FE*) of the presented contents. Resolution usually refers to the minimum dimension of a feature in a dataset (**database resolution**, R_{db}) or on the presentation medium (**presentation resolution**, R_{rep}). Normally when spatial data are collected for certain purposes, a R_{rep} will be specified. R_{db} is then related to R_{rep} as:

$$S_{rep} = ME / FE = R_{rep} / R_{db} \Rightarrow R_{db} = R_{rep} / S_{rep} = (R_{rep} * FE) / ME \quad (1)$$

Unlike paper maps, in a GIS/SDB environment, both *ME* and R_{rep} may have different values on different or the same presentation devices. For a fixed S_{rep}, different R_{rep} corresponds to different R_{db}. This is the reason why (spatial) resolution instead of scale should be used in the context of a spatial database [12]. In discussions below, “resolution” refers to database resolution unless stated otherwise.

2.2 Resolution, Generalisation Criteria and Multiple Representation

Increase in resolution value (i.e. coarsening) will result in simplification of a representation’s geometric form. In this context **simplification** refers to processes that normally remove only those elements (e.g. vertices) that are redundant from a resolution point of view while original information is preserved to a maximum. The well-known RDP algorithm [13] when used appropriately may be regarded as an example of such a process. Note that simplification may cause topological change in the original geometric form. In addition, two representations at different resolutions may have different values for a non-spatial attribute (e.g. Landuse). This would typically be due to the level of detail at which a classification hierarchy is applied (for example, *corn* could be generalised to *cereals*) [7].

On the other hand, **generalisation** refers to those processes that remove “details” from the geometric form (which may be the result of previous resolution-driven simplification) or eliminate entire objects in order to highlight the major geometric characteristics of phenomena or maintain a balance of information among different object types according to the criteria imposed.

For a single phenomenon, different GC may be associated with different metrics and/or metric values and will generate different representations at a fixed resolution. Details contained in the geometric form of each representation will vary, as illustrated in Fig. 1. Furthermore, different GC may result in retrieval of representations of different phenomena in the same type hierarchy. For example, at the same resolution, if more details are required, individual buildings could be retrieved; otherwise, the same location may be represented as a built-up area that is the aggregation of a group of buildings.

Two representations of the same phenomenon, such as a land parcel, under different GC may also vary in their attribute values. In this case, attribute values may be drawn from two different value hierarchies. For example, a parcel may be a cornfield under the criterion of current land-use but a residential area under the criterion of planned land-use in urban planning practice.

From a map-wide point of view, coarsening spatial resolution will generally cause a **selection** of geographical phenomena to be presented, as discrete objects fall below the resolution threshold. For a fixed resolution, variation of GC will affect the way phenomena are selected. In addition, change of either resolution or GC may cause **aggregation** of phenomena into new phenomena.

2.3 Incompatibility of Representations

Two representations are said to be incompatible if they should not present simultaneously. **Representation incompatibility** refers to incompatibility among multiple representations of a particular phenomenon or phenomena in the same perception hierarchy (e.g. a group of buildings aggregate to a built-up area). **Topological /proximal incompatibility** may exist between two representations of two different phenomena due to topological or proximal consistency constraints. For example, a generalised representation of a road may leave the initial location of a village on the wrong side of it and, therefore, these two representations are incompatible and a displaced new representation for the village should be retrieved along with this road representation.

3 Multi-representation Geometry and Spatial Object

In this section we introduce the concept of multi-representation geometry (**MRep-Geometry**) which we regard as the basic unit for representing spatial objects in a multi-representation context.

3.1 Generalisation Criteria, Generalisation Metrics and Presentation Space

As previously mentioned, a geometric representation of a spatial phenomenon is defined at a given resolution and under a certain set of GC which describe the purposes of query or map compilation. Generalisation procedures with numerical GM are applied to source data to generate results to meet these purposes. Therefore, a representation may be associated with certain GM values at a given resolution.

Assuming there are $n > 0$ generalisation metrics applied to a dataset (there may be multiple GMs applied to one type of phenomenon and different GMs for different phenomena), we may associate the resolution dimension *RES* with these n metrics to define an n+1-dimensional space (RES, GM_1, GM_2, …, GM_n) as the **presentation space** (**PS**) of the dataset. Consequently, any representation of the dataset (or a phenomenon in it) may be mapped to a point in this abstract space.

For the sake of simplicity, in the following discussion we will use a dataset containing one open polyline with one GM applied as an example. Therefore, the PS for this dataset is a 2-D space (*RES*, *GM*).

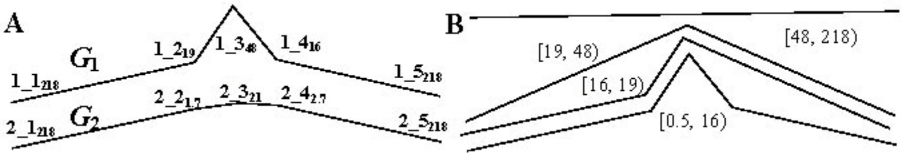

Fig. 2. **A**-Two Representations of the same phenomenon **B**- Multi-resolution retrieval on G_1

3.2 Multi-representation Geometry

Assuming at resolution *res* and with GM value *gm*, the geometric form of a spatial phenomenon is represented by an instance (G) of one of the well-known geometric types (point, open or closed polyline, chain, polygon, … or collections of various types), we may denote this representation of the phenomenon as $Rep = (G, PR=\{(res, gm)\})$. We regard such a geometry instance G as a **single-representation geometry** (**SRep-geometry**) and PR as Rep's **presentation range**, which is a point set in the dataset's PS and in this case contains a single point (res, gm).

As an example an open polyline is denoted as either an n-tuple $<p_1, p_2, \ldots, p_n>$ or a partial ordering $<\mathrm{P}, \leq >$ on a vertex set $\mathrm{P}=\{p_i|i= 1, n\}$ where $\leq$ reflects the vertex precedence in the polyline. Fig.2-A illustrates two simple representations of the same phenomenon as $Rep_1 = (G_1, \{(res_1, gm_1)\})$ and $Rep_2 = (G_2, \{(res_2, gm_2)\})$ where $G_1 = <\mathrm{P}_1, \leq >$ ($\mathrm{P}_1 = \{p_{1\_1}, p_{1\_2}, p_{1\_3}, p_{1\_4}, p_{1\_5}\}$) and $G_2 = <\mathrm{P}_2, \leq >$ ($\mathrm{P}_2 = \{p_{2\_1}, p_{2\_2}, p_{2\_3}, p_{2\_4}, p_{2\_5}\}$).

If we follow the multi-version approach, Rep_1 and Rep_2 are two distinct versions of the phenomenon and G_1 and G_2 will be stored separately. Alternatively, we may merge G_1 and G_2 into a single linear structure. Firstly a single graph may be generated from G_1 and G_2 by merging two vertices in G_1 and G_2 into one single vertex if their coordinates are identical and the resulting graph is a DAG. By applying topological sorting on the resulting DAG, we will obtain a linear vertex sequence $MG=<p_1, p_2, p_3, p_6, p_4, p_5>$. Geometrically, $p_i = p_{1\_i} = p_{2\_i}$ except that $p_3 = p_{1\_3}$ and $p_6 = p_{2\_3}$. A vertex p_i in MG has the form of (x_i, y_i, PR_i). PR_i is the vertex's presentation range, which is the union of the presentation ranges of all representations containing this vertex. In our example, $PR_i = \{(res_1, gm_1), (res_2, gm_2)\}$ except that $PR_3 = \{(res_1, gm_1)\}$, as p_3 is in G_1 only, and $PR_6 = \{(res_2, gm_2)\}$ as p_6 is in G_2 only. We regard MG as a **multi-representation geometry** (**MRep-geometry**). Obviously, an SRep-geometry corresponding to a query point (res_q, gm_q) in PS may be easily retrieved from an MRep-geometry by selecting those vertices p_i satisfying $(res_q, gm_q) \in PR_i$.

Specifically, if there exists a query that may retrieve all vertices in an MRep-geometry, we regard such an MRep-geometry as **subsetting** (vertices retrieved by any other queries will be a sub-set of that retrieved by this query); otherwise, it is **non-subsetting** (e.g. p_3 and p_6 in MG will never be retrieved simultaneously).

Several different representations of a spatial phenomenon may be merged into one multiple-representation (**MRep**) containing one MRep-geometry merged from geometries of these representations. Consequently, all representations of the phenomenon may be merged into one or several MReps and a single entity (**Multi-representation spatial object**, or **MRO**) may be used to represent this phenomenon as $\mathrm{MRO}=\{MRep_i|i=1, n\}$.

Note that although some representations of a phenomenon are geometrically merge-able, for practical reasons, concerning for example the maintenance of non-spatial attributes that may vary between representations, we may choose not to merge them into a single MRep. In addition, we cannot normally expect to have all the representations of a phenomenon immediately available to merge into MReps. Thus we may need to compute the multiple representations from one or a few detailed representations of a phenomenon.

3.3 Realisation of Simplification and Generalisation Metrics

3.3.1 Geometric Details and Metrics Values

The constituent details in a geometry can be defined in many application-dependent ways, e.g. through their correspondence with inflexion points or skeleton branches [14]. In general we may regard any three or more consecutive vertices in a geometry as a detail. For example, given an open polyline $pl = \langle p_1, p_2, \ldots, p_n \rangle$, any sub-tuple $\langle p_i, \ldots, p_j, \ldots, p_k \rangle$ ($i < j < k$; $1 \le i \le n\text{-}2$; $2 \le j \le n\text{-}1$; $3 \le k \le n$) of pl is a **detail** on pl, denoted as $dtl(p_i, p_k)$. In particular, points p_i and p_k are **base points** of the detail and points p_j are **detail points** of the detail. A detail should contain two base points and at least one detail point. A detail is **simplified** if some of its detail points are removed. If new detail points are inserted, the detail is **modified**. Normally we may regard a detail as **removed** if at least one of its base points or all its detail points are removed.

Various relations, such as separate, adjacent, identical, inclusive and nesting, may be defined between two details $dtl(p_a, p_b)$ and $dtl(p_c, p_d)$ on the same polyline. As these relations are not used in the current study, we omit their definitions here.

With the above definition, both simplification and generalisation may be viewed as processes of detail simplification and/or removal (detail modification may also be involved in some algorithms) although the intention of simplification and generalisation are very different. Simplification removes some details while retaining most "critical" points. On the other hand, generalisation normally removes some "critical" points to smooth an object. If we process a detailed representation of a phenomenon with different simplification (normally associated with resolution) and generalisation metric values, different less-detailed representations may be generated, with different vertices/details removed from the original representation (and possibly with some vertices/details added as well). By associating these metric values with the vertices/details removed (or added) and storing these values along with geometric data, different representations of a phenomenon may be retrieved directly without complicated simplification/generalisation processing at query time.

3.3.2 Spatial Resolution, RDP Tolerance and Simplification Metrics

RDP tolerance has been previously adopted in multi-resolution access schemes [5, 6] for representing the spatial resolution dimension. In this study, we will also use the extended RDP tolerance for this purpose, i.e. with tolerance promotion during the process to form a monotonically decreasing tolerance value hierarchy [6].

Several observations may be made on the process of calculating RDP tolerance values (denoted as TOL in the following discussions) of vertices in an open polyline $pl = \langle p_1, p_2, \ldots, p_n \rangle$ (see example vertex subscripts in Fig.1-A). For an internal vertex

p_j ($1 < j < n$) whose TOL value d_j is calculated relative to vertices p_i and p_k (i.e. distance from p_j to line p_i-p_k and $i < j < k$):

- A detail $dtl(p_i, p_k)$ may be defined and measured by d_j which only depends on p_i, p_j and p_k and is irrelevant to other (if any) detail points p_m ($i < m < k$ and $m \neq j$);
- $d_j \le d_i$ and $d_j \le d_k$ (for end-points p_1 and p_n in an open polyline, *ad hoc* TOL definitions are required);
- If there is any other detail point p_m in the detail, $d_m \le d_j$;
- In a vertex filtering process to select any vertex p_j satisfying $d_j > d_q$ (d_q is a query tolerance value), p_m will never be selected if p_j is not selected; p_j will never be selected if p_i and p_k are not selected; in addition, assuming $d_j > 0$, p_j will be selected for any d_q falling in the range $[0, dj)$; for the whole polyline, some of its vertices will be selected for $d_q \in [0, d_{max})$ where d_{max} is its vertices' maximum tolerance value;
- Consequently, the above detail $dtl(p_i, p_k)$ may be represented by the internal vertex p_j and its tolerance value d_j;
- Each internal vertex corresponds to one and only one detail that is recognised and hence **effective** in a vertex filtering process.
- If there are m ($\le n$ if no vertices added) distinct tolerance values $\{d_1, d_2, \ldots, d_m\}$ generated from the polyline, for any query tolerance value $d_q \in [d_k, d_{k+1})$ ($k < m$), the same set of vertices (hence the same geometrical representation) will be selected (Fig.2-B).

A mapping mechanism may be defined between tolerance value d and resolution value res (e.g. simply let $res = d$). Therefore, the above tolerance range of a vertex $[0, d_j)$ may be mapped to a **resolution range** $RR = [r_f, r_c)$ ($r_f \le r_c$) where r_f and r_c are the finest and coarsest resolution bounds of the vertex. If the base resolution of a dataset is r_0 (which conceptually speaking should be greater than 0) and the polyline presents in the initial dataset, we have $r_f = r_0$. Note that for vertices newly added during some other simplification processes or for vertices representing a phenomenon not present in the initial dataset, we have $r_f > r_0$. In addition, under other different simplification algorithms, a resolution range may well possess a more complicated form (such as the union of a few non-overlapping intervals $[r_{f\_1}, r_{c\_1}) \cup [r_{f\_2}, r_{c\_2}) \cup \ldots$).

Similarly, we may also say that in the above discussion the representation retrieved by $d_q \in [d_k, d_{k+1})$ corresponds to a resolution range $[r_k, r_{k+1})$. This fact implies that a limited (but perhaps large) number of representations (or versions) of a phenomenon are sufficient to support continuous change of query resolution within the resolution range $[r_f, r_{c\_max})$ which is the resolution range of the polyline.

From a multi-representation point of view, the above resolution ranges may be regarded as presentation ranges for vertices and representations on the resolution dimension and the polyline can thus be converted to a polyline which is multi-representational on the resolution dimension.

3.3.3 Generalisation Metrics and Weighted Effective Area

An example of the many available line generalisation metrics is the so-called "effective area" (**EA**) introduced in the Visvalingam-Whyatt (VW) algorithm [15]. The EA of a point p_i ($1 < i < n$) in an open polyline $pl = <p_1, p_2, \ldots, p_n>$ is the area of the triangle formed by p_i and the two points p_{i-1}, and p_{i+1} (EA of endpoints p_1 and p_n requires *ad hoc* definition). Unlike the top-down RDP algorithm, the generalisation process of the VW algorithm works bottom-up to iteratively remove the point with smallest EA from the polyline until a predefined EA threshold is reached. When a point is removed, the EA of the two points adjacent to the removed point will be recalculated. Thus if p_i with EA value ea_i is removed, EA of p_{i-1} and p_{i+1} become the areas of triangles (p_{i-2}, p_{i-1}, p_{i+1}) and (p_{i-1}, p_{i+1}, p_{i+2}) respectively if the new values are **greater** than ea_i. Without setting a fixed EA threshold, we may repeat the process until there are only two (or three for closed polylines) points left.

Like the RDP tolerance, each internal vertex also represents one and only one detail effective in a vertex filtering process based on EA values and this detail may be measured by the *final* EA value of the vertex (i.e. prior to its removal). Consequently, each vertex p_i may be labeled with an EA range $[0, ea_i)$ and the polyline is converted to a polyline which is multi-representational on the EA dimension.

In our current experiment, we have used **weighted effective area** (**WEA**), a new metric based on EA. Instead of using the area value of the triangle (p_{i-1}, p_i, p_{i+1}) directly, we also take the shape characteristics of the triangle into consideration. A group of weight factors is used to adjust the initial area value to generate a WEA value. These factors are based on measures reflecting flatness, skewness and orientation of the triangle. This new metric (or indeed a family of metrics) provides much more control on detail removal by using different weights and/or weight mapping functions. As it is not directly relevant to the present study, we will not present the details of the WEA metrics here, but the effect of WEA-based generalisation is shown in our experimental results.

3.4 A Method to Compute MRep-Geometry

In the previous sub-section, we have treated resolution and generalisation metric (WEA) dimensions separately. In this sub-section we will demonstrate a method to integrate the two dimensions to compute vertex presentation ranges in the RES-EA space. A single detailed open polyline *pl* (i.e. G_1 in Fig.2-A) is used as an example. For the sake of simplicity, we use EA as a generalisation metric in the following explanation, while WEA is used in our experiment on a real dataset. In addition, we focus on the states of the three internal points (p_2, p_3 and p_4) in *pl*. We also assume that the base dataset resolution is $r_b = 0.5$ and the maximum resolution and EA values for the polyline (and the two endpoints p_1 and p_5) are *MaxRes* and *MaxEA* whose values may change but are always equal to or greater than those of the internal vertices.

3.4.1 Initial Presentation Ranges

Initial vertex RES and EA values computed from the original polyline are shown in Fig.3-A (labeled on each internal vertex as EA/RES). Therefore the RES value sequence $< r_b = 0.5, 16, 19, 48, MaxRes>$ and EA value sequence $<0, 593, 940, 2096, MaxEA>$ form two partitions on RES and EA dimensions and the polyline should occupy the region of $PR_{pl}=\{(res, ea) \mid 0.5 \le res < MaxRes \wedge 0 \le ea < MaxEA\}$ in the RES-EA space.

For each vertex p_i, it is natural to think its presentation range is $PR_i=\{(res, ea) \mid 0.5 \le res < r_i \wedge 0 \le ea < ea_i\}$. However, because RES and EA dimensions are not independent and different vertices are selected/removed according to RES and EA criteria, it is possible, for example, that a vertex is selected due to the RES criterion while one or both of the base points of the detail represented by this vertex are not selected due to the EA criterion. Consequently, the detail represented by this vertex under RES criterion will not be presented properly from a cartographic point of view.

For the reasons stated above, the initial valid presentation range of the polyline is $PR_{pl\_0} = PR_0 \cup PR_{res} \cup PR_{ea}$, where $PR_0 = \{(res, ea) \mid r_b \le res < MinRes \wedge 0 \le ea < MinEA\}$, $PR_{res} = \{(res, ea) \mid MinRes \le res < MaxRes \wedge 0 \le ea < MinEA\}$and $PR_{ea} = \{(res, ea) \mid r_b \le res < MinRes \wedge MinEA \le ea < MaxEA\}$. Here *MinRes* and *MinEA* are smallest vertex RES and EA values in the polyline (16 and 593 in this example). PR_0 is the region without simplification or generalisation effect, PR_{res} is the region with RDP-simplification effect only and PR_{ea} is the region with EA-generalisation only. The initial valid presentation range of each vertex may be defined similarly by replacing *MaxRes* and *MaxEA* with respectively vertex RES and EA bounds r_i and ea_i.

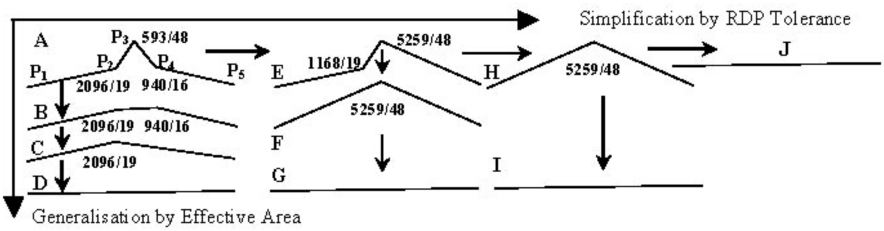

Fig. 3. Computing Mrep-polyline with RDP priority. Vertex labels are AE(final) /RDP values

3.4.2 Expansion of Presentation Ranges

A simple way to expand initial vertex presentation ranges into the region of $\{(res, ea) \mid MinRes \le res < MaxRes \wedge MinEA \le ea < MaxEA\}$ is to give priority to one dimension (e.g. RES) and recalculate metric values on the other dimension (e.g. EA) for vertices in representations (Fig.2-B) derived from the initial metric values on this dimension (i.e. RES). In other words, this is a process of "generalisation after simplification" (or "simplification after generalisation" if priority is given to the EA dimension).

Fig.3 illustrates this process while results are shown in Table 1 (which is a **PR-Table**, or presentation-range table, representing the Resolution-Generalisation metric plane). In Table 1, each cell represents a rectangular region (cell-PR) on the RES-EA plane as $\{(res, ea) \mid min\_res \le res < max\_res \wedge min\_ea \le ea < max\_ea\}$, e.g. the first (top-left) cell in Table 1 represents $\{(res, ea) \mid 0.5 \le res < 16 \wedge 0 \le ea < 593\}$. If a representation (or point, detail, etc.) occupies a cell, its presentation range should contain the cell-PR of this cell. Cells in Table 1 present representations in Fig.3 and labels of any internal points (p_2, p_3, p_4) in these representations, e.g. the first cell contains the original polyline (Fig.3-A) which has three internal points indicated in the form of (2/3/4). The *PR* of a vertex has a more complicated form. In the case of p_2 it is $\{(res, ea) \mid (0.5 \le res < 16 \wedge 0 \le ea < 2096) \vee (16 \le res < 19 \wedge 0 \le ea < 1168)\}$.

A consequence of this process is that in some vertices the value of the recalculated metric (EA in this case) is increased while the metric value on the other dimension (RES in this case) increases (e.g. p_3 has an EA value of 593 in representation Fig.3-A and the value is increased to 5259 in Fig.3-E). This may generate some odd query results. For example, for a query (r_q, ea_q) ($593 \le ea_q < 940$), when the value of r_q increases, we will retrieve representations B, E, H or J in Fig.3. Vertex p_3 does not present in B but is present in the simplified representations E and H. To solve this problem, a constraint may be applied to the process to make sure that recalculated metric values should never increase.

Table 1. PR-Table with RDP priority

0 \ 0.5	16	19	48	MaxTol
593	A(2/3/4)	E(2/3)	H(3)	J
940	B(2/4)	E(2/3)	H(3)	J
1168	C(2)	E(2/3)	H(3)	J
2096	C(2)	F(3)	H(3)	J
5259	D	F(3)	H(3)	J
MaxEA	D	G	I	J

It is also possible to simultaneously recalculate metric values on both dimensions with priority assigned to one dimension in order to create more sub-division of cells and generate more representations. However, due to the inter-dependency of the two dimensions, the process will be much more elaborate in order to maintain consistency of metric values.

3.4.3 Discussion

The above method is presented here only for the purpose of demonstrating the general procedure of generating MRep-geometry from detailed single-representation geometric objects. It is simple but the results are not of the highest quality. Some other generalisation metrics associated with other detail definitions may provide better support to handle metric inter-dependency and generate better results. It is also worth noting that results of different metrics may be stored together inside one MRep-geometry to support different generalisation requirements from a single source.

A characteristic of our current method is that there is a one-to-one correspondence between vertices and effective details. Consequently, we attach information relevant to the presentation of a detail to its corresponding vertex and do not need to define details explicitly. For other metrics in which details are defined differently (e.g. [14]), explicit detail definition and storage may be required and references to these details may be stored with vertices affected by these details in the form of constraints. At query time, these constraints can be tested to decide whether a vertex should be retrieved. Such constraints may also be used to resolve incompatibility among multiple representations of different objects.

4 Generalisation Metrics for Multiple Objects

The previous sub-section focuses on a single phenomenon. In this section we use a simple model for point object selection to demonstrate how generalisation metrics for selection can be integrated with spatial resolution to support representational multiplicity among a group of objects.

4.1 A Simple Model for Point Object Selection

Assume in a region with a real world extent of fe_0^2 (constant), there are N_{FLD} point phenomena of a certain type (e.g. residential site) and various other phenomena. The initial task is to construct a spatial dataset at a base dataset resolution of r_0 (derived from the pre-defined initial presentation extent me_0^2 and presentation resolution r_{rep_0} using equation (1) in 2.1).

From a presentation point of view, if we focus on these point phenomena only, r_0 imposes a restriction on the maximum number of phenomena (N_{max_0}) that may be represented in the dataset in order to maintain a reasonable **presentation object density** which may be defined as: $WD(N) = N^k / A_{disp}$. Here N is the number of retrieved objects, k is a weight factor and A_{disp} is the area of the presentation device (me^2 in this case, where me is normalized according to scale to fit fe_0). Therefore we may define $N_{max}(r)$ $(r \geq r_0)$ as the **dataset capacity** for this type of phenomena at r and **maximum** WD as (using (1)):

$$WDMax(r) = N_{max}(r)^k / A_{disp} = N_{max}(r)^k / me^2 = (N_{max}(r)^k * r^2) / (fe_0^2 * r_{rep}^2) \qquad (2)$$

Clearly, N_{max_0}=$N_{max}(r_0)$. Larger r corresponds to decrease of scale or increase of presentation resolution value and will result in smaller N_{max} to maintain the object density at an acceptable level. Therefore, $N_{max}(r)$ represents a resolution-oriented aspect of multiplicity in the process of object selection.

At a given resolution r and assuming that there are sufficient candidates for selection, we may select $N_r \leq N_{max}(r)$ phenomena of this type. This is because, when other types of phenomena are taken into consideration, the space for presenting each type of phenomenon is further restricted and a fine balance among the various types has to be maintained. Consequently, the selection rate (used as a general term here) assigned to a particular type may result in an under-capacity ($N_r < N_{max}(r)$) selection. As the selection rate is affected by the generalisation criteria used to make the dataset, different GC will result in different selection rates for the same resolution r, which represents a semantic aspect of multiplicity of object selection.

Note that the theoretical value of $N_{max}(r_0)$ may exceed the value N_{FLD}. We may either use a universal $N_{max}(r_0)$ for different datasets of the same size under the same GC set, or make $N_{max}(r_0)$ adaptive to the nature of a particular dataset. At present we simply assume that for $N_{max\_0} \leq N_{FLD}$ then $N_0 \leq N_{max\_0}$ phenomena are selected and presented in the initial dataset at r_0.

It is natural to think that objects are selected by their relative importance. When resolution value r increases, at $r = r_i$, the number of objects currently in the dataset will exceed $N_{max}(r_i)$ and the least important object O_i should be removed, leaving N_c objects remaining. Consequently, if the order of selection is fixed, we may say that O_i will present in the dataset within the range of $PR_i=\{(r, N_{sel}) \mid r_0 \leq r < r_i \wedge N_c < N_{sel}\}$, i.e. O_i will be selected only if there are $N_{sel} > N_c$ objects to be selected. PR_i is O_i's **presentation range for selection** in the 2D RES-NUM space (NUM represents object numbers).

To decide the resolution bound r_i at which O_i is removed with N_c objects left, by making a provision that maximum presentation density should remain constant at different resolutions (i.e. $WDMax(r_0) = WDMax(r)$), we have:

$$N_{\max}(r) = N_{\max\_0} * (r_0 / r)^{2/k} \Leftrightarrow r = r_0 * (N_{\max\_0} / N_{\max}(r))^{k/2} \quad (3)$$

$$N_{\max}(r_i) = N_c \Rightarrow r_i = r_0 * (N_{\max\_0} / N_c)^{k/2} \quad (4)$$

For the most important object in the dataset, we have $N_c = 0$ so that *ad hoc* definition is required to decide its coarsest resolution bound. We also assume r_{rep} is not changed in the process. Otherwise, (3) and (4) will possess more complicated forms.

Note that the above derivation has its roots in the so-called "radical law" for feature selection in cartography. From (3) we can also derive $r_0 / r = (N_{\max}(r) / N_{\max\_0})^{k/2}$. For $k = 4$, we have $r_0 / r = N^2_{\max}(r) / N^2_{\max\_0}$, which is exactly the basic "radical law" expression for object selection [16].

Number of objects to be selected is certainly not a user-friend query parameter in the process of querying a multi-representation object set described above. Instead we may use **degree of selection** (*DoS*) which is defined as:

$$DoS(r) = N_{sel} / N_{\max}(r) \Rightarrow N_{sel} = DoS(r) * N_{\max}(r) = DoS(r) * N_{\max\_0} * (r_0 / r)^{2/k} \quad (5)$$

Thus $DoS(r) \in [0, 1]$ as N_{sel} should not exceed $N_{\max}(r)$. Under the above provision of a constant maximum presentation object density, it is easy to prove that for the same *DoS* value, the retrieved presentation object density is also approximately constant (since N_{sel} is a discrete integer) at different query resolutions. On the other hand, at the same resolution and for different purposes, a user may use different *DoS* values to control the proportion of objects retrieved relative to the maximum number of objects that may retrieve at the resolution. Note that we have assumed the order of object selection is the same under resolution-driven selection and semantics-driven selection. Otherwise, an additional selection metric will be needed.

A single *DoS* may be of no great interest. However, for a dataset containing more than one type of object, by adjusting *DoS* values for different object types, we may "blend" the source dataset in many different ways to meet users various requirements.

4.2 Selection and Aggregation in Feature Hierarchy

The point object selection model presented above, albeit simplistic, illustrates a general approach for handling representational multiplicity among a group of objects. In normal cartographic practice, selection processing of objects with finite size (length, area, etc.) also obeys some sort of mathematic relations resembling the radical law. Consequently, models similar to the one in 4.1 but with the size and other characteristics of objects taken into consideration may be used to generate multi-representation datasets for these objects.

For an $MRO=\{MRep_i|i=1, n\}$, its presentation range for selection is the union of that of its *MRep*s (each of which could contain an MRep-geometry described in section 3). If its *MRep*s are defined according to different semantics, no special treatment will be needed as their retrieval is normally controlled by non-spatial attribute values.

If one *MRep* is derived from the other while resolution decreases, initially the two *MRep*s may be separated on the resolution dimension by the generalisation process. For example an MRO with a *PR* of $\{(r, N_{sel}) \mid r_b \le r < r_{\max} \wedge N_c < N_{sel}\}$ (number of objects is used as selection metric for convenience) contains two *MRep*s as $MRep_1$

with $PR_1 = \{(r, N_{sel}) \mid r_b \le r < r_1 \wedge N_c < N_{sel}\}$ and $MRep_2$ with $PR_2 = \{(r, N_{sel}) \mid r_1 \le r < r_{max} \wedge N_c < N_{sel}\}$. If this is the case, we may choose to move part of PR_1 into PR_2 to provide multiple representations with a certain resolution range $[r_1', r_1)$. Subsequently, we have $PR_1 = \{(r, N_{sel}) \mid (r_b \le r < r_1' \wedge N_c < N_{sel}) \vee (r_1' < r < r_1 \wedge N_c' < N_{sel})\}$ and $PR_2 = \{(r, N_{sel}) \mid (r_1' \le r < r_1 \wedge N_c < N_{sel} \le N_c') \vee (r_1 < r < r_{max} \wedge N_c < N_{sel})\}$, where $N_c < N_c'$. With this extension, for $r_1' \le r_q < r_1$, a smaller N_{sel} ($N_c < N_{sel} \le N_c'$), indicating a low selection rate, will retrieve $MRep_2$; otherwise, $MRep_1$ may be retrieved. Alternatively, a new metric may be used solely for this purpose. This technique also applies to the situation where one MRO is aggregated from a few other MROs residing in the same feature hierarchy (e.g. a built-up area aggregated from a group of buildings).

Finally, due to the flexible nature of an MRep-SDB, potentially there are a great number of incompatible cases among objects, representations of objects or even details in representations caused by topological, proximal or semantic inconsistency. While there is no room to address these issues in detail, we believe most of these cases may be detected and resolved when the multi-representation dataset is generated. Solutions may be stored along with geometric data in the form of persistent constraints and checked at query time to retrieve the correct objects/presentations. For example, in the road-village case presented in 2.3, which representation of the village should be retrieved depends on which road representation is retrieved. Consequently, a constraint recording information on the condition of road retrieval may be attached to the village object and tested at query time.

5 Discussion and Experimental Results

5.1 Experimental Implementation and Results

The method in 4.1 to compute MRep-geometry geometry has been implemented in C++. The RDP tolerance criterion is used as resolution value and WEA is the generalisation metric. Priority is given to RDP and the WEA values are recalculated. As neither of the original RDP and VW algorithms guarantees topological consistency in their output, we also use a fully dynamic 2-D Constrained Delaunay Triangulation package (MGLIB-2) to maintain a triangulation of the dataset during the whole process for detecting topological inconsistency. Currently we do not remove a point causing inconsistency but raise its RES or WEA value and wait until the removal of other points makes this point delete-able. This is the simplest but certainly not the best solution. Without topological consistency checking, the time complexity of this process is roughly $O(n^3)$ for a polyline with n vertices.

Currently the data structure for a vertex in an MRep-geometry is $MRPoint\{x, y, r_{min}, N, R[N], SRWEA[N]\}$. The first three data items are coordinates and the finest resolution bound. $SRWEA[n]$ (n = 0, N-1) is the square root of WEA value at resolution range $[R[n-1], R[n])$. In particular, for n=0, the resolution range is $[r_{min}, R[0])$. Also, $R[N-1]$ is the coarsest resolution bound for the vertex. Note that if the WEA value is identical at two adjacent resolution intervals, the two intervals are merged. In addition, this is the form for subsetting points with WEA-adjustment only. For other cases, the PR-Table for a point is in the form of a more complicated sparse matrix, for which more sophisticated data structures have to be used.

The test dataset (see Fig.1, 4 and 5) is derived from several Ordnance Survey Land-Form PANORAMA map sheets at a field resolution of 1m. There are five closed objects and 2376 vertices in total while the largest object contains 2336 vertices. We have computed a few statistics on the generated multi-representation dataset. If data items in the above structure *MRPoint* are stored in double precision for coordinates, long integer for *N* and single precision for others, on average 45.42 bytes (i.e. 29.42 bytes for MRep-data and the average of *N* is 3.18) are required for a multi-representation vertex. By exhaustive enumeration scanning through all resolution and WEA interval bounds, we also obtain the number of distinguishable representations that we may retrieve from the dataset as 539,066 versions. Note that any geometrically identical representations at separate presentation ranges are **not** counted repeatedly. The total number of vertices in these 539,066 versions is 556,908,339. If the WEA non-increasing constraint is applied (see 3.4.2), the results are 524,212 versions and 540,711,604 vertices.

Although differences between many of these versions are minimal, all these versions are required in order to claim that genuine continuous change of resolution and generalisation metric values is supported under the generalisation procedures we adopt. It is true that due to selection, an object may be deleted at a resolution much smaller than its original coarsest resolution bound. Therefore, many of these versions (i.e. at coarser resolutions and hence containing fewer vertices) would be unnecessary. On the other hand, the size of MRep-geometry would also be reduced as *R*[*N*] and *WEA*[*N*] are smaller as well.

To demonstrate the representational multiplicity of MRep-geometry, we produced a JAVA applet-based web demo (http://www.cs.cf.ac.uk/user/S.Zhou/MRepDemo/).

In this web demo (Fig.4), we provide an operation mode of "Intelligent Zooming". Under this mode, a screen resolution value and a "**degree of generalisation**" (*DoG*) value is preset.

As subsequent zooming-in/out operations are carried out, the retrieved representations will show a resemblance in generalisation style. Very similar to *DoS*, *DoG* defines a mechanism to map WEA values to a real range of [0, 1]. A *DoG* value of 0 represents "minimum" WEA-based generalisation and 1 the "maximum" generalisation. When the required dataset resolution changes (due to zoom-in/zoom-out), the same *DoG* value will be mapped to a different WEA value accordingly to maintain roughly the same generalisation effect. Therefore, a user may avoid the trouble of having to compute a proper WEA value in order to make an intended query. Fig.5 represents three zooming series under different *DoG* values. From left to right, DoG = 0, 0.5, 0.75 and, from top to bottom, scales are $1{:}10^6$, $1{:}2.5\times10^6$, $1{:}5\times10^6$ and $1{:}10^7$ respectively. The screen resolution (R_{rep}) is 0.1mm. At each resolution *r*, *DoG* value is mapped to a WEA value $wea = r^2 * (1 - DoG)^{-2}$ ($DoG < 1$). The number of vertices in each retrieved representation is marked in the figure.

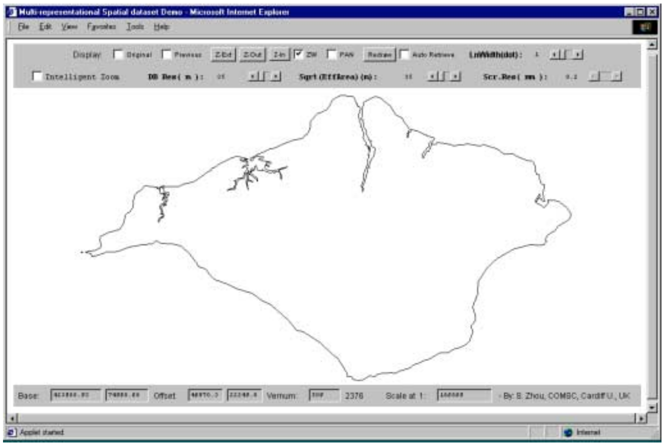

Fig. 4. A Web Demo for MRep-Geometry

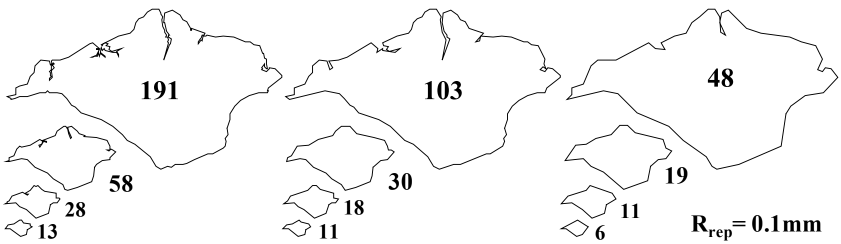

Fig. 5. MRep-Geometry and Intelligent Zooming (drawn at real scale. source dataset derived from original Ordnance Survey map, Crown copyright 2002)

5.2 MRep-SDB Design and Implementation Issues

In this sub-section we will briefly discuss a few key issues relevant to the design and implementation of an MRep-SDB. In an MRep-SDB, a spatial index on a column containing MRO has to support at least four dimensions: the two spatial dimensions, resolution dimension and one or more dimensions representing object selection metrics. As a 3D-Rtree can be used for a multi-resolution SDB [6] with good results, it is natural for us to use *N*-D-Rtree (*N*>3) to index an MRO column.

Although in this study we have demonstrated that a great deal of generalisation workload may be practically moved to a pre-processing stage, some type of online generalisation may still be required to process results retrieved from a MRep-SDB. One such process is so-called online graphic conflict resolution. Although we might be able to guarantee topological consistency (and proximal consistency to an extent) in the database, proximal inconsistency caused by user actions such as specifying a large symbol size is difficult to handle on the DBMS side and online generalisation procedures are consequently required. Experiments that achieve real-time performance for conflict resolution have been carried out and reported in [17].

Another issue is related to the importance ranking of objects for selection processing. We have assumed such ranking is carried out dataset wide. However, for queries retrieving objects in an area small in comparison to the extent of the dataset, whether such a ranking is still applicable is questionable if the importance of an object depends on whether some other objects are retrieved or not. We may design some more complex metrics to reflect this aspect of locality or use constraints to handle the issue. Alternatively, we may choose to use a query *DoS* value larger than the initially intended value to retrieve some more objects and subsequently apply some online selection procedures upon this relatively small set of objects.

6 Summary

In this paper we have discussed various aspects of the causes of representational multiplicity of geographical phenomena from a map generalisation point of view. We introduced the concept of multi-representation geometry (MRep-geometry) as the basic unit for representing the geometric form of multi-representational geographical phenomena. A practical method has been presented for generating topologically consistent MRep-geometry from a single-resolution source data using a new generalisation metric (weighted effective area). The resulting MRep-geometry presents representational multiplicity with continuous changes of both resolution and generalisation metric. We presented an approach for handling representational multiplicity of an object set which contains multiple multi-representation objects with a potential object type hierarchy. Experiments on MRep-geometry were carried out on a real dataset and results presented in an online web demo. Finally, we discussed various important issues relevant to designing, implementing and using a multi-representation spatial database.

Effective support for representational multiplicity in a spatial database will benefit a wide range of users. We believe the approach presented here holds a clear advantage over previous approaches in storage efficiency, performance and, most significantly, flexibility. Our implementation has (to an extent) demonstrated the practicality of this approach. However, we also acknowledge the huge difficulties in building such an MRep-SDB for even the simplest real-world applications. Predominately, these difficulties concern the automated map generalization required for dataset generation. Issues that will be addressed in more detail in future studies include resolution-dependent proximal consistency, the selection process for linear and areal objects and incompatibility among multiple representations of different objects.

References

[1] Puppo, E. and G. Dettori. Towards a formal model for multiresolution spatial maps. in Proc. Advances in Spatial Databases (SSD95), LNCS 951. p. 152-169. 1995

[2] Rigaux, P. and M. Scholl. Multi-scale partitions: Application to spatial and statistical database. in Advances in Spatial Databases (SSD95), LNCS 951. p. 170-183. 1995. Portland

[3] Stell, J.G. and M.F. Worboys. Stratified map spaces. in Proc. The 8th International Symposium on Spatial Data Handling. p. 180-189. 1998. British Columbia: Taylor and Francis

[4] Stell, J.G. and M.F. Worboys. Generalizing Graphs using Amalgamation and Selection. in Proc. Advances in Spatial Databases (SSD'99), LNCS 1651. p. 19-32. 1999: Springer

[5] Becker, B., H.-W. Six, and P. Widmayer. Spatial priority search: an access technique for scaleless maps. in Proc. ACM SIGMOD. p. 128-137. 1991. Denver, Colorado

[6] Zhou, S. and C.B. Jones. Design and implementation of multi-scale databases. in Proc. 7th International Symposium on Advances in Spatial and Temporal Databases (SSTD 2001). p. 365-384. 2001. Redondo Beach, CA, USA: Springer

[7] Rigaux, P. and M. Scholl. Multiple Representation Modelling and Querying. in Proc. IGIS'94, LNCS 884. p. 59-69. 1994. Ascona, Switzerland

[8] Bedard, Y. and E. Bernier. Supporting multiple representations with spatial databases views management and the concept of <<VUEL>>. in Proc. ISPRS / ICA Workshop Multi-Scale Representations of Spatial Data. 2002. Ottawa

[9] Friis-Christensen, A., et al. Management of Multiply Represented Geographic Entities. in Proc. International Database Engineering and Applications Symposium (IDEAS'02). p. 150-159. 2002

[10] The MurMur Consortium. Supporting Multiple Representations in Spatio-Temporal databases. in Proc. 6th EC-GI & GIS Workshop. 2000. Lyon, France

[11] The MurMur Consortium, MurMur Project Workpackage 3 (Services) Deliverable 9: Other Services. 2001

[12] Spaccapietra, S., C. Parent, and C. Vangenot. GIS Databases: From Multiscale to MultiRepresentation. in Proc. 4th International Symposium on Abstraction, Reformulation, and Approximation (SARA-2000), LNCS 1864. p. 57-70. 2000. Texas, USA: Springer

[13] Douglas, D.H. and T.K. Peucker, Algorithms for the reduction of the number of points required to represent a digitised line or its carcature. The Canadian Cartographer, 1973. **10**(2): p. 112-122

[14] van der Poorten, P.M. and C.B. Jones, Characterisation and generalisation of cartographic lines using Delaunay triangulation. International Journal of Geographical Information Science, 2002. **16**(8): p. 773-794

[15] Visvalingam, M. and J.D. Whyatt, Line generalisation by repeated elimination of points. Cartographic Journal, 1993. **30**(1): p. 46-51

[16] Robinson, A.H., et al., Elements of Cartography. 6th ed. 1995: Wiley

[17] Li, M., S. Zhou, and C.B. Jones. Multi-Agent Systems for Web-Based Map Information Retrieval. in Proc. 2nd International Conference on Advances in Geographical Information Sciences (GIScience 2002), LNCS 2478. p. 161-180. 2002