

Improving Map Generalization for On-demand Mapping by Multiscale Databases

Alessandro Cecconi
Department of Geography, University of Zurich
Winterthurerstrasse 190
CH-8057 Zurich, Switzerland
acecconi@geo.unizh.ch

Keywords: *On-demand Mapping, Multiscale Database, On-the-fly Generalization, Levels of Detail (LoD), Hierarchical Linking*

1 Introduction

On-demand mapping is concerned with the generation of maps upon the user request and according to the user requirements. Users are able to produce their own maps and customize the process of generation. For the flexible creation of maps on-demand at arbitrary scales and for arbitrary themes cartographic generalization is needed. The generalization process critically determines how long it takes to create a map. Because automated generalization is only partially solved other solutions must be found 1) to substitute for missing automated generalization operations and 2) to speed up cartographically appropriate yet computationally intensive generalization algorithms. Multiscale databases (MSDB) offer a possible solution (Buttenfield 1993, Devogele *et al.* 1996). Using a multiscale database the automated generalization process can be divided into two steps. First, slow generalization algorithms can be precomputed in advance (offline) and the results stored as levels of detail (LoDs) in the multiscale database. Second, those generalization algorithms that are computationally efficient can be computed on-the-fly and used to refine the nearest LoD to the requested map scale.

2 Combination of Map Generalization and MSDB

This paper explores an approach which combines components of a MSDB (Devogele *et al.* 1996) with components of on-the-fly generalization (van Oosterom and Schenkelaars 1995). On the basis of user specifications the adequate object classes are selected from the database LoD appropriate for the desired map scale and refined further at run-time by on-the-fly generalization algorithms. In order to implement this architecture the main issues to be solved are:

- to study the generalization process in conjunction with multiscale databases;
- to develop a schema and structure for the MSDB which includes hierarchical linking of corresponding objects between different levels.

Jones *et al.* (2000) devote more attention to MSDB design spending less time on building generalization process. Glover and Mackaness (1999) experimented with on-the-fly generalization from a single scale database.

In our study case (we use two sets representing two LoD), the first level is equivalent to a 1:25'000 topographic database, the second level to 1:200'000. The definitions of the links are very important and needed for on-the-fly generalization and can be used to simplify the generalization process. For example, if the requested scale would be 1:100'000, the links given between the corresponding objects can be used for an 'interpolation' (or morphing) process between the two geometries.

3 Multiscale Database Schema and Generalization Process for the Feature Class Roadnetwork

The database schema was created for an object-oriented database and will be implemented in the commercial GIS-product Gothic Lamps2 (Laserscan). Following feature classes have been considered: road network, river network and buildings. As it looks similar for all features, it will be illustrated for the class *roadnetwork*. Figure 1 explains the idea of the implementation. The feature class *roadnetwork* inherits from different super classes methods for handling geometry and topology {*simple_line*, *topography*, *map_generalisation*}. Types (or attributes) and methods play in conjunction with the generalization process a very important role. For the hierarchical linking of corresponding objects or groups of object (where single objects cannot be linked) a special ID (GroupID) has been introduced. This ID is unique according the feature class and allows to use the information of the other data set to simplify and to speed up the on-the-fly generalization process. The class *roadnetwork* entails the information to two classes (depending on the given data set): *roadnetwork25k* and *roadnetwork200k*. From these instances of the objects (with the geometry and all other information) can be derived.

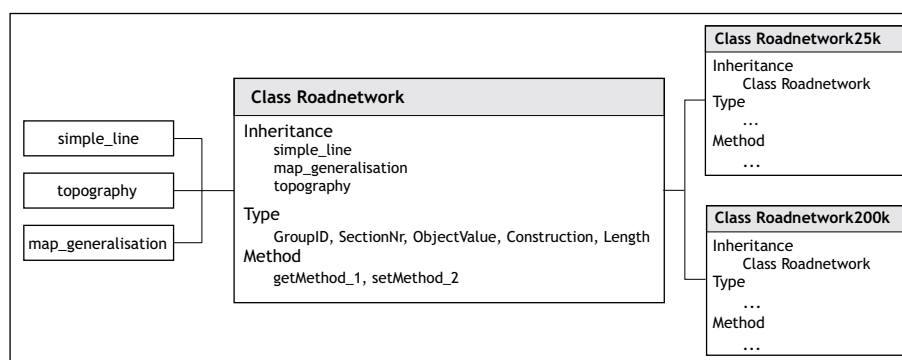


Figure 1: Database schema for the feature class *roadnetwork*.

The methods are used for the generalization process and for the visualization of the objects in the display.

For the feature class *roadnetwork* the selection of generalization operators plays an important role. At a scale of 1:25'000 usually all roads are represented. At smaller scales only main roads and important minor roads are displayed. Figure 2 shows the proposed generalization process. Road selection is carried out in a two-step selection. The first selection step is based on the comparison with the smaller data set. Every road section that appears at both scales must appear. The second step decides on the basis of the road attribute which road section is shown. Next, it has to be decided whether a morphing transformation (morphing does not belong to the generalization operators in the closer sense (McMaster and Shea 1992)) will take place or a weeding operator is directly applied. If the corresponding road section is contained in both data sets a morphing process can be started.

Otherwise the weeding operator is used. The rendering step is responsible for the representation of the elements on the screen. If unsolved problems remain after rendering and subsequent displacement the process may return to the symbolization restart with smaller symbol size.

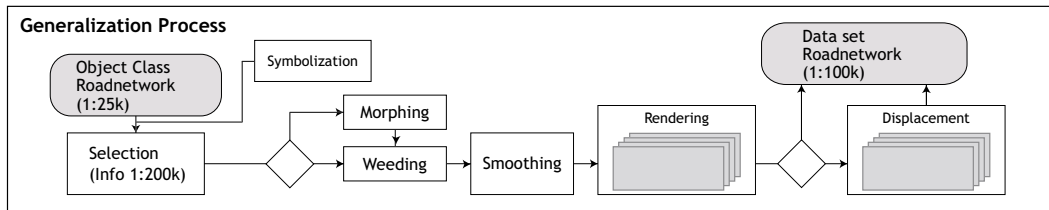


Figure 2: Generalization process for the feature class *roadnetwork*.

4 Conclusions and Outlook

The creation of maps on-demand is a time-critical task. Existing examples that offer zooming and scaling functions (e.g. www.mapquest.com) are often for a single purpose and use multiple (independent) LoDs. As consequence, they do not allow flexible generation of maps.

Our proposition allows on one hand the simplification and the speed up of the generalization task. On the other hand generalization renders MSDB more flexible and leads to better cartographic results.

The experimental case study that we are pursuing focuses on on-demand topographic map creation (using Swiss maps as an example). The defined schemas for the selected feature elements will be implemented and tested. Next, different generalization algorithms need to be adapted for our purposes and integrated into the MSDB context. The ultimate goal is to expand the specifications for the specific test scenario and build a generic module for on-demand mapping.

References

- BUTTENFIELD, B. (1993). Research Initiative 3: Multiple Representations. Closing Report. Technical report, National Center for Geographic Information and Analysis.
- DEVOGELE, T., TREVISAN, J., AND RAYNAL, L. (1996). Building a multi-scale database with scale-transition relationships. In: *Proceedings 7th International Symposium on Spatial Data Handling*, volume 1 of *Advances in GIS Research II*, pp. 6.19–6.33. Delft.
- GLOVER, E. AND MACKANESS, W. A. (1999). Dynamic generalisation from single detailed database to support web based interaction. In: *Proceedings of the 19th International Cartographic Conference*, pp. 1175–1183. Ottawa.
- JONES, C. B., ABDELMOITY, A. I., LONERGAN, M. E., VAN DER POORTEN, P., AND ZHOU, S. (2000). Multi-scale spatial database design for online generalisation. In: *Proceedings 9th International Symposium on Spatial Data Handling*, pp. 7b.34–7b.44. Beijing.
- MCMMASTER, R. AND SHEA, S. (1992). *Generalization in Digital Cartography*. Association of American Geographers, Washington, USA.
- VAN OOSTEROM, P. AND SCHENKELAARS, V. (1995). The Development of an Interactive Multi-Scale GIS. *International Journal of Geographic Information Systems*, **9** (5), pp. 489–507.