

IMPROVING EFFICIENCY FOR DEVELOPING AUTOMATIC GENERALISATION SOLUTIONS

Nicolas Regnauld

Ordnance Survey, Romsey Road, Southampton, SO16 4GU, UK – Nicolas.Regnauld@ordnancesurvey.co.uk

©Crown Copyright 2005. Reproduced by permission of Ordnance Survey

KEY WORDS: Generalisation, Web services, Automation, Multiple Representation

ABSTRACT:

The Research & Innovation department at Ordnance Survey has been working for two years on a project to investigate the automatic generalisation of our data to derive a vector dataset matching the specifications of the current OS Landranger Map series (1:50 000 scale paper maps). This paper gives an insight of the results obtained, and highlights the complexity of the algorithms that we had to develop for this purpose. We then discuss the reason why after many years of research, there is so little available in commercial GIS platforms in terms of generalisation algorithms. The first generations of algorithms, performing transformation on independent features like simplification and smoothing. More complex algorithms like displacement, collapse, amalgamation, typification are rarely available, and when they are, perform well only for specific applications. We believe that the main reason for this is the difficulty to test algorithms which are developed in different research laboratories by others. We see some new ideas and experiments around generalisation web services as a potential way of overcoming this problem. We explain how their wide use could help the research community, and the potential users of generalisation algorithms.

1.INTRODUCTION

Ordnance Survey has in the last decade put a lot of effort into the creation of a single highly detailed geographic database, providing a complete seamless coverage of Great Britain. The current product that best illustrates the richness and potential of this database is OS MasterMap™, “the definitive digital map of Great Britain” (Ordnance Survey 2005). OS MasterMap is a large scale vector product. Ordnance Survey would now like to derive other products from its database, with a high level of automation. This requires generalising the data, which has proved difficult to automate, despite years of research in this domain.

The Generalisation team in Research & Innovation, is investigating the generation of small scale vector products from the main database using a high level of automation. In order to support the generation of our current product portfolio, plus future products, we need a generic approach that will allow us to reuse tools and mechanisms for different products. Given the complexity and the size of the task, we want to make the best use of what already exists, develop missing tools, and mix the appropriate components to build the generalisation solutions that fit our requirements. Combining algorithms from different sources to generate a given product is not easy. These algorithms may be based on different GIS platforms, they have been designed to work on different data, and they have been developed using different product specifications. In addition, if the first generations of algorithms, performing transformation on independent features like simplification (Douglas and Peucker 1973) and smoothing (Akima 1974), are often available on commercial GIS platforms, the more complex algorithms like displacement (Ruas 1998, Højholt 2000, Sester 2005), collapse, amalgamation, typification (Regnauld 2003, Burghardt and Cecconi 2003) are rarely available.

In this paper, we want to highlight the need for a standard way of publishing algorithms so that they can be combined into

custom applications. The first section describes a prototype that we have built to generate a 1:50,000 vector digital map from our data. It focuses on the heterogeneity of the solution built. The second discusses the reason why so much development has been necessary, why there are so few algorithms available to choose from when we want to build a full generalisation solution. The last section explains how the research on the generalisation web services, which have emerged in the last few years, could provide a solution to make prototypes developed in research environment available to potential users. Based on their evaluation, it could be decided to reengineer them into fully supported tools in commercial GISs.

2.PROTOTYPE TO BUILD A 1:50,000 VECTOR MAP FROM OS MASTERMAP

This project started in March 2004, and is nearing completion, scheduled for the end of December 2005. The aim was to use data from OS MasterMap, and try to derive automatically a cartographic vector dataset, following the specifications of our current 1:50,000 paper map series. It has been accepted from the beginning that we were not aiming for a finished product, manual corrections by a cartographer are expected after the automatic process.

2.1 Source data

The source data for the generalisation algorithm mainly comes from OS MasterMap, the largest scale vector dataset derived from Ordnance Survey’s main database. We have used two layers:

- The Topography Layer is a large scale digital database of the detailed surface features on the landscape. It contains features regrouped in 9 themes: roads, tracks and paths; land; buildings; water; rail; height; heritage; structures; and administrative boundaries. Figure 1 shows a small sample of an urban area.



Figure 1: Extract of OS MasterMap data
 Ordnance Survey © Crown Copyright. All rights reserved.

- The Integrated Transport Network™ (ITN) Layer contains a network of road centre lines, and some routing information (like one way restrictions for example).

We have also used Ordnance Survey's Digital Landscape Model Land-Form PROFILE® Plus, which contains detailed height data defining the physical shape of the landscape of Great Britain.

2.2 The prototype generalisation process

After analysing the current 1:50,000 map series, we have identified a few generalisation process on which to concentrate. We have decided to investigate the following areas:

- Generalisation of buildings.

No suitable algorithms have been found in existing GIS to perform the generalisation required. We have identified the need for two different types of algorithms to generalise buildings in urban and rural areas. Building generalisation in rural areas tends to amalgamate small numbers of buildings and give the result a building-like shape. In urban areas, the high density required a higher level of amalgamation that shows built-up areas where buildings are present. These algorithms, as well as the tools to partition the space into rural and urban areas, are described in . They have been developed on the platform Clarity developed by Laser-Scan , and make use of its AGENT functionality. These algorithms rely on some spatial analysis tools that have been developed to analyse the proximity relationships between features. They are based on a Delaunay triangulation, from which different types of proximity graphs can be derived .

- Collapse of dual carriageways into a single centre line.

Dual carriageways had to be collapsed in order to apply a single symbol to depict the road at the target scale. explains how the dual carriageways are identified, and their component paired together, before being collapsed into a centreline, to which adjoining roads are reconnected. This has also been developed on the platform Clarity.

- Generalisation of woodlands.

The generalisation of the woodland has been investigated, and a theoretical solution proposed in . An automatic solution has then been developed using the tools available in the platform ArcGIS of ESRI. The solution is not as flexible as the original design but the results are good enough so the cost of developing the more complex and flexible method were not justified at this stage.

- Generalisation of the hydrology.

No existing GIS package provides a tool set which can handle the generalisation of our hydrology data. The main reason is that our hydrology layer does not form a proper structured network, it is disconnected wherever other features lie on top of the water (like bridges). So specific algorithms had to be developed. Algorithms have been developed to build a network from our data, reusing the ideas developed in an earlier research project described in . Then, we have used a variation of the Horton ordering to prune the network.

- Contour generalisation.

Contours have been generated from Land-Form PROFILE Plus, and generalised to avoid overcrowding of contour lines in dense slope areas. This has been done by interrupting selected lines to improve dense areas. This has been developed in ArcGIS in the context of a MSc project (student) at the University of Edinburgh (UK).

- Road displacement.

When representing roads at 1:50,000, the symbol used to display them covers an area significantly larger than the original road. This creates conflicts between symbols that could often be resolved with a slight displacement of the roads involved. We have investigated this problem, and tried using two different solutions. We have tried an algorithm based on beams , which is available on Clarity. We have also tried an algorithm developed by the University of Hannover, available as a stand-alone platform, PUSH . To obtain good results, both methods require that the conflict to be solved is first properly isolated. A method has been developed to achieve this. It has been implemented on Clarity, and can successfully trigger the beams-based displacement algorithm in a number of configurations (not all conflicts are solvable at the moment). The PUSH algorithm has not been used yet with this automatic detection, because we need to develop an interface that will convert and exchange the data between the two environments.

The result of these process is a vector dataset ready for printing at 1:50,000 scale. All the generalised features contain a reference to the original features from which they have been derived, which should allow for incremental update of the generalised dataset. The result map is shown on figure 3, it shows a good first approximation of our current LandRanger map series (1:50,000), as shown on figure 2. Additional data (like coastal data, names, cycle paths) have been put directly on the map without generalisation.

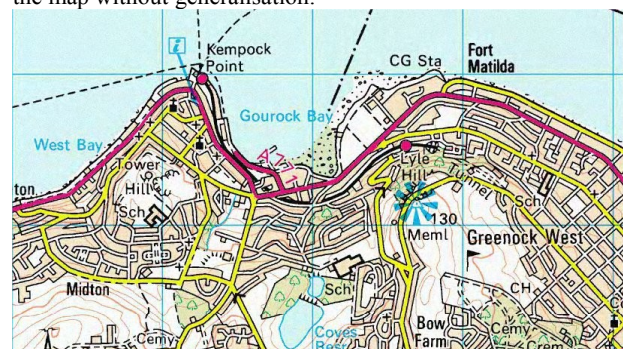


Figure 2: Extract from OS Landranger Map 63, Firth of Clyde
 Ordnance Survey © Crown Copyright. All rights reserved.

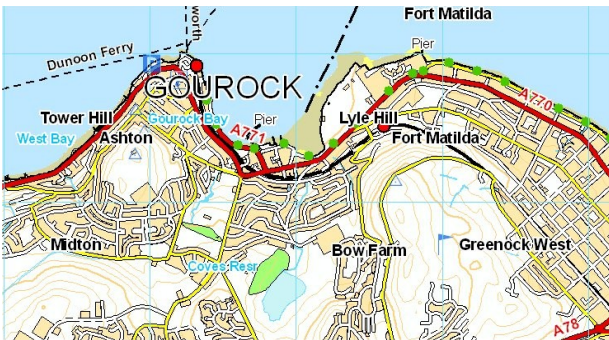


Figure 3: Results obtained using a research prototype generalising automatically OS MasterMap data
Ordnance Survey © Crown Copyright. All rights reserved.

3.LESSONS LEARNED DURING THE DEVELOPMENT OF THE PROTOTYPE

We have learned a lot during this project about cartographic rules, generalisation algorithms, generalisation process, the functionalities of current GIS platforms, and even about our own data. But what we want to emphasise here is the cost of developing generalisation solutions and the difficulty to reuse existing tools.

3.1 Cost of developing generalisation algorithms

Generalisation algorithms can be very complex to develop, particularly those which involve several features, like displacement, amalgamation, typification. Even simplification, which is often seen as an operation on a single feature, should, to produce good results, analyse the local shapes of the feature, rank them by importance and then decide which ones should be kept as they are, enlarged or eliminated, to keep the true character of the complete feature. The algorithms mentioned above took two years with a team of three people to develop.

3.2 Difficulty to reuse existing algorithms

Innovative generalisation algorithms are usually developed at universities, in research laboratories, although some National Mapping Agencies like IGN France, Ordnance Survey GB and probably a few others have some research teams working on it as well. Ideally, these algorithms, once they have been tested and recognised as valuable, should be implemented in a GIS platform, that customers (mainly National Mapping Agencies and map publishers) could use. This rarely happens, and the possible reasons could be:

- Limited testing. The algorithm prototypes developed during research projects are difficult to test. The reason is that they are usually designed to prove a concept, and work on a few chosen test datasets. Allowing the algorithm to work with other data (not another extract of the same type of data but data from another data provider) would take significant development time for each other input data format supported.
- Limited life time. Algorithms developed by researchers at universities are usually published but the prototype (executable program) itself is often short lived. The reason is that the researcher moves on to other places, other research areas, changes development platform, etc. In any case, researchers do not have

time resources to support prototypes. When a prototype has proved an algorithm, it has done its job. However, there is often much more in the code of the prototype than in the paper that describes it. This knowledge usually dies with the prototype, as the tricks included in the implementation to manage particular cases do not last long in the developer's mind. The only way to capture it efficiently at the moment is to involve the developer of the prototype into porting it into a supported system (commercial GIS). This rarely ever happens.

- Limited interoperability. Testing one algorithm which has not been designed to be run on a particular source data is already difficult. But to evaluate its true potential for fulfilling a given purpose, testing it in combination with other algorithms is often required. Small problems with results that initially don't look important can however completely spoil the action of the next algorithm to be applied in the generalisation sequence. This needs to be tested. This means that to test the integration of one algorithm with others, the algorithm must be available from the environment where it is planned to be used.

The main reason why algorithms that are developed in research are short lived and often need to be re-implemented to be tested properly at a later stage, is that we do not have a standard way of packaging them to make them widely available for testing. The difficulty of reusing an existing algorithm is a great obstacle to the development of more complex solutions. Instead of building on past developments, new researchers (like PhD student), need to start by spending a lot of time building their own toolbox, leaving them with significantly less time to innovate and report on their findings.

4.WEB SERVICES TO BOOST THE RESEARCH AND THE USE OF AUTOMATED GENERALISATION

In the last few years, interest has grown among the community involved in research in generalisation, for a more open platform for developing tools and processes. The aim is to share the tools developed and access them through a common platform. have proposed an architecture where tools (spatial analysis tools, generalisation operators, or even more complex generalisation processes) could be packaged into services, that could be called-up remotely over the Web. Such an approach has also been demonstrated, although using only basic generalisation tools, during the GiMoDig project, which aimed at providing mobile devices with Geospatial data coming from multiple sources, integrated and generalised in real-time through web services. A web service approach to share a wide range of generalisation tools would provide some benefits but would also bring some constraints:

Benefits of a web services approach:

- Availability. Algorithms published as Web services would be available for people to test them on their own data, from their own environment (as long as their platform has a plug-in to call this type of service). They could then be combined with other algorithms (present in the client platform or via other services). From a map producers point of view, when trying to design a flowline for improving a map production line or creating a new one, it would provide an efficient way of building a prototype reusing as

much as possible what already exists, and only develop the missing parts. The prototyping stage allows the existing tools to be tested and the appropriate ones chosen. The decision of how they should be included in the final production system is a separate issue which is much easier to tackle when the value of the algorithm for the task has been proved.

- Portability. Designing algorithms with the service concept in mind will help developers to produce algorithms which will be much more easily portable than if they are embedded in a GIS platform, making heavy use of internal tools. The availability of libraries like JTS (Java Topology Suite) should allow developers to stay away from some internal GIS libraries, when alternatives are available.
- Quicker prototyping of new generalisation ideas. By having access to a rich set of tools to support the development of new ones, the research community will be able to spend more time on developing the innovative part of any new algorithm or process, and less in redeveloping the basic functions that are required to support it.

Constraints:

- Packaging. Each algorithm will have to follow strict standards to read its data and export its results. This can be a problem for packaging existing algorithms, but once a standard has been accepted, then the design of a new algorithm would follow it from the beginning, and the constraint would not be a problem.
- Describing the algorithms. If people start using these services to publish algorithms, it could soon become difficult to use them, because of the large number of services available. Meta data will then need to be added to the services, based on some kind of classification. They should provide information about the granularity of the service (for example local generalisation action or generalisation process for a full dataset). Describing dependencies between services will also be critical. High granularity level generalisation services will probably orchestrate a number of lower level services. What happens when one gets removed? Should we define some classes of "similar" services (same type of data and parameters, same type of action)?
- Managing spatial data structures (triangulations, graphs, Voronoï diagrams, etc.) is not easy. Spatial structures are important tools that should be available through services, but they are different from the classic transformation algorithms. These structures are heavy to compute, but can be reused and updated. How could this be achieved in a service-based environment?
- Plug-ins. Each platform from which we want to access the generalisation services will have to have a plug-in written, so that it can do the interface between the local data format and the standard data format used by the service. This is a one-off cost which does not amount to much if we consider that at the moment, users basically need to write a sort of interface for each external algorithm that they want to call from their platform.
- Transfer time and accessibility. Obviously, when dealing with large amounts of data, the transfer might prove too costly, and there is also the issue of being blocked when a server is down. This is where it is im-

portant to differentiate production environment and research environment. From the research point of view, the loss of processing speed is often not so important compared to the time saved during the development. In the production environment, the speed is critical. But then, when an algorithm has been identified as being required in a production line, investment can easily be justified to have it redeveloped in the production environment, or even just ported when this is possible.

Overall, the idea of generalisation web services is very attractive and should enable much better use to be made of the existing research than we do at the moment. At Research & Innovation, we would like to contribute to the development of this approach. We have started testing the existing prototype developed in Zurich. So far we have managed to:

- Call the existing services from the open source GIS platform JUMP (<http://www.jump-project.org/>).
- Set-up a server that is a copy of the one developed in Zurich.
- Write a java algorithm based on the Java Topology Suite, and published it on our local server.
- Call the service from Clarity which required us to write the foundations of a client for accessing the services from Clarity.

This was all done within a period of two days, with the help Moritz Neun from the University of Zurich who is doing a PhD on the subject. We did not hit any major problems. We now intend to push the technology by trying to publish complex algorithms as web services, as well as spatial analysis tools. The aim is to define a standard that can cope with all the requirements of the generalisation processes. We also want to develop clients and servers for some existing commercial GIS platforms and demonstrate how these platforms could interoperate using this technology.

5.CONCLUSION

During our generalisation project to derive 1:50,000 representation from our detailed data, we have found very few algorithms that we were able to try, in order to reuse them. We had to invest significant effort to develop algorithms, some completely new, some that had been studied in the past. Reusability of developments is not easy to achieve, but is worth investing in. If we always start from scratch, we will never be able to build the complex solutions required to achieve high level of automation of the generalisation process.

Research has started in recent years to address this issue through generalisation Web Services (Neun and Burghardt 2005). We now need to support this initiative and through it create a set of standards that will be used in the future to package generalisation algorithms. Then the library of available generalisation tools could grow, and allow for the development of more advanced generalisation solutions.

This article has been prepared for information purposes only. It is not designed to constitute definitive advice on the topics covered and any reliance placed on the contents of this article is at the sole risk of the reader.

Reference List

- Badard, T. and Braun, A., 2003, OXYGENE: An open framework for the deployment of geographic web services. *Proceedings of the 21st International Cartographic Conference, CD-ROM*.
- Bader, M. and Barrault, M., 2001, Cartographic Displacement in Generalization: Introducing Elastic Beams. *Fourth Workshop on Progress in Automated Map Generalization*, Beijing, 2-4 August, http://www.geo.unizh.ch/ICA/docs/beijing2001/papers/bader_barraultv1.pdf.
- Edwardes, A., Burghardt, D., Bobzien, M., Harrie, L., Letho, L., Reichenbacher, T., Sester, M., and Weibel, R., 2003, Map Generalisation Technology: Addressing the need for a common research platform. *Proceedings of the 21st International Cartographic Conference, CD-ROM*.
- Horton, H., 1945, Erosional development of streams and their drainage basins. *Bulletin of the Geological Society of America*, **56**, 275-370.
- Laser-Scan, 2003, Clarity. <http://www.laser-scan.co.uk/technologies/foundation/clarity/index.htm>.
- Neun, M. and Burghardt, D., 2005, Web Services for an Open Generalisation Research Platform. *8th ICA WORKSHOP on Generalisation and Multiple Representation*, A Coruna, Spain, July 7-8th, http://ica.ign.fr/Acoruna/Papers/Neun_Burghardt.pdf.
- Regnauld, N., 2005, Spatial structures to support automatic generalisation. *Proceedings of the International Cartographic Conference*.
- Regnauld, N. and Mackaness, W. A., 2005, Creating a Hydrographic Network from a Cartographic Representation: A Case study using Ordnance Survey MasterMap Data. *International Journal of Geographical Information Science*, **To be published**.
- Revell, P., 2005, Seeing the Wood from the Trees: Generalising OS MasterMap® Tree Coverage Polygons to Woodland at 1:50 000 Scale. *8th ICA WORKSHOP on Generalisation and Multiple Representation*, A Coruna, Spain, July 7-8th, <http://ica.ign.fr/Acoruna/Papers/Revell.pdf>.
- Revell, P., Regnauld, N., and Thom, S., 2005, Generalising OS MasterMap® topographic buildings and ITN road centerlines to 1:50 00 scale using a spatial hierarchy of agents, triangulation and topology. *Proceedings of the International Cartographic Conference*.
- Sarjakoski, T., Sester, M., Sarjakoski, T., Harrie, L., Hampe, M., Letho, L., and Koivula, T., 2005, Web Generalisation Service in GiMoDig - Towards a Standardised Service for Real-Time Generalisation. *Proceedings of the 8th AGILE Conference on GIScience*.
- Sester, M., 2005, Optimizing Approaches for Generalization and Data Abstraction. *International Journal of Geographical Information Science*, to be published.
- Thom, S., 2005, A Strategy for Collapsing OS Integrated Transport Network™ dual carriageways. *8th ICA WORKSHOP on Generalisation and Multiple Representation*, A Coruna, Spain, July 7-8th, <http://ica.ign.fr/Acoruna/Papers/Thom.pdf>.