ACTIVE OBJECTS AND DYNAMIC TOPOLOGY FOR SPATIAL DATA RE-ENGINEERING AND RICH DATA MODELLING

Paul Hardy
Laser-Scan Ltd., Cambridge, UK
paul@lsl.co.uk

Dagstuhl Seminar 01191, May 2001
Computational Cartography and Spatial Modelling

ABSTRACT

Many organisations such as national mapping agencies have collected large datasets of spatial data, often by digitising existing map series. However, they are discovering that the current and future demands for spatial datasets and mapping do not match the simple data models and unstructured feature data that they have at present. To create the required rich data models and appropriate mapping using the established GIS and digital mapping software tools has been a major challenge, which has not often been economically viable. However, there are now appearing new software products based on active objects in intelligent spatial object databases, which provide the necessary tools and infrastructure to make these tasks economically possible.

This paper overviews the data modelling, dynamic topology, and active object capabilities of a commercial spatial data manipulation product, and shows how they are being used for large-scale spatial data re-engineering to create new national framework datasets and generalised mapping. It also covers some of the challenges remaining, such as conflation of existing datasets to improve quality.

1 INTRODUCTION

1.1 Spatial Data Holdings

Over the last twenty years, many organisations have built up considerable holdings of spatial data. In particular the National Mapping Agencies (NMAs) have captured existing map series (often the ones at largest scale) to build spatial ‘databanks’. The original purposes for this was often to provide economies in the production of traditional paper map sheet products, but they have become the principal sources for digital data products to fuel the growth of GIS. These databanks are often not true databases, but large collections of files, usually in one file equals one map sheet units.

One example of such a databank is the LandLine data of the Ordnance Survey of Great Britain (OSGB). This consists of 223,000 files, each one corresponding to a sheet in the large-scale map of Britain (1:1250 for urban areas, 1:2500 for rural area, and 1:10,000 for empty areas). The data is held in Common Internal Transfer Format (CITF), which is an OSGB variant of the BS7567 (NTF) national standard. The data model used is feature-based, with features having a feature code and a small number of attributes. Most feature codes represent linear features (walls, fences, kerbs, etc.).

Like many other organisations with such data holdings, the OSGB has realised that its databank, while adequate for its original purpose of paper map production and early GIS applications using maps as backdrops, is not suited to the generation of appropriate data products for today’s markets, nor to the derivation of new products. As such, it has recently undertaken a spatial data re-engineering task to produce a new Digital National Framework dataset (DNF).

The continuous dataset approach is not unique to Britain. The paper on “A National Topographic Database for the 21st Century - Paradigm Shifts in Business Process and Technology” [Howard et al 2000], gives an overview of the pressures and gains involved in building a national database for New Zealand.

1.2 What is Spatial Data Re-engineering?

Spatial Data Re-engineering is the process of adding value to existing data, by some or all of:

- Defining a data model (schema) that models the real world rather than a map-based cartography
- Enriching the data model with rules to check validity and ensure data integrity
- Adjusting the data model to facilitate efficient implementation of the target applications
- Loading existing data into the new data model, including reclassifying features
- Building a spatial continuum without artificial sheet or tile edges

- Cleaning the geometric structure of the data, such as removing overshoots, undershoots and slivers
- Cleaning the logical structure of the data, such as wrongly coded features
- Building topological relationships
- Adding internal structure and intelligence, such as grouping relationships
- Validating the data for self-consistency, both geometric and semantic
- Deducing new classes of data more appropriate to the tasks in hand.

Using newly available spatial processing technology, many of these can now be achieved as automated bulk processes, although the addition of a small amount of human interactive completion (guided by automated markups) can significantly improve the final quality.

1.3 What is Conflation

Conflation is the operation of amalgamating two or more datasets, so as to pick out the ‘good’ aspects of each and produce a combined result which is better than either of the originals. The primary source for much of today’s spatial base data was field survey, some dating back to the 19th century. Now there are many other positional data sources such as remote sensed analysis (satellite imagery, aerial photography, SAR, LIDAR), and new survey techniques like GPS. These have created alternative datasets with much better positional accuracy, and/or greater currency (up-to-dateness), and/or greater logical consistency, and/or greater attribute richness. However, it would be far too expensive to throw away the existing base data and start again from the new sources.

So what is needed is processes that can compare the original base data with the new alternative data, and extract the extra value from the new while retaining all that is valuable in the original, to produce a consistent and higher quality single dataset.

1.4 What is Generalisation?

Generalisation is the process of deriving data and mapping at a target scale and specification, from a starting point of existing data, usually at a more detailed scale. In the past, it has primarily been for map generalisation, where it is the science (and art) of exaggerating those aspects that are important for this particular map purpose and scale, and removing irrelevant detail that would clutter the map and confuse the user. Increasingly, however, the NMAs want to apply generalisation to derive new digital datasets for use in GIS, in location-based web services, or in e-government.

Generalisation has traditionally been a hard task to automate, being dependent on the skills of the human cartographer. People have tried for years to build centralised 'knowledge bases' of generalisation rules, with very limited success. New Object-Oriented (O-O) and active Agent techniques, which distribute the task onto the map features themselves, have opened major new possibilities for Generalisation, as covered in later sections of this paper.

2 KEY CAPABILITIES OF OBJECT SOFTWARE

2.1 Object Data Modelling

Most existing GIS and spatial database systems have data model capabilities that are limited because they match their underlying feature-based or relational datastores. Many years ago, computer science and software engineering recognised the elegance, generality, power, and economy of Object Orientation (O-O). O-O is a way of looking at the world and modelling it inside a computer. The GIS world was generally slow to take up O-O concepts, but a new generation of O-O spatial databases, spatial toolkits, and spatial applications is now available, and fundamentally change the baseline for spatial processing [Warboys et al, 1990]. The examples used in this paper are all based on the Laser-Scan Gothic family of products, including the Gothic O-O spatial database and spatial toolkit, and the LAMPS2 mapping and geodata production application. O-O has a language of its own, and some of the main concepts are:

- **Encapsulation**
  - Data and behaviour are not separated, but held together inside an object class. This gives efficiency of data handling and ease of development, as effects of change are localised.
  - Objects respond to messages that are sent to them by carrying out method behaviours and returning values
- **Referencing**
  - Objects can have direct knowledge of related objects, of any class.
- **Inheritance**
  - The family structure of the world can be modelled (a church is a kind of public building, which is a kind of a building, which is a kind of a man-made structure).

- **Polymorphism**
  - Objects of different classes all respond to standard messages, but execute different appropriate behaviours for each class. This makes extensibility to new classes easy.

2.2 Dynamic Topology

An important capability of a spatial data processing product is its ability to create and handle topology (the knowledge of spatial relationships, such as adjacency, colinearity, connectivity, and containment). The Gothic object spatial database uses methods and object references to implement in-built support for topology structure. The user can choose Spaghetti or Structured for each class (with priority if structured), and then define topological model, and separate snapping tolerances between pairs of classes. Priorities are important to prevent accurately positioned features (e.g. survey points or graticule lines) being moved as a result of snapping onto other less precise features.

Topological model can be
- Share node - only end points of features are involved
- Node split link - at least one end of feature must be involved
- Link split link - any part of feature can be involved

The different models are appropriate to different classes; for instance, a motorway will often only have “node split link” topology with other roads to prevent unnecessary nodes being created where it goes over or under ordinary roads.

The database will then apply these rules and create the necessary links and nodes as objects are digitised, imported, edited, or processed. The effect is to clean up digitising errors, inaccuracies and inconsistencies, such as undershoots, overshoots and slivers, and to validate the spatial relationships at the time, prevent bad constructs (such as figure-of-eight areas) getting into the dataset.

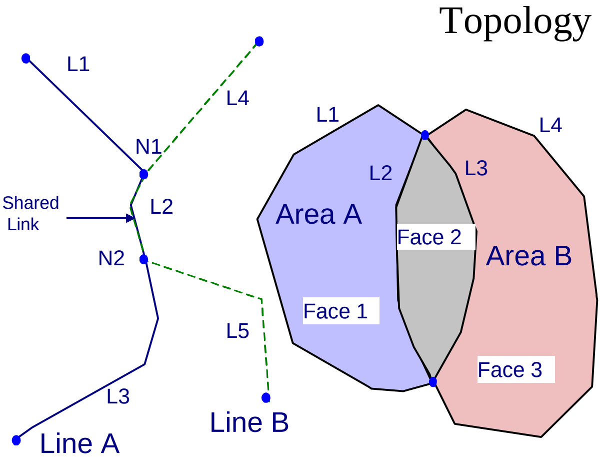
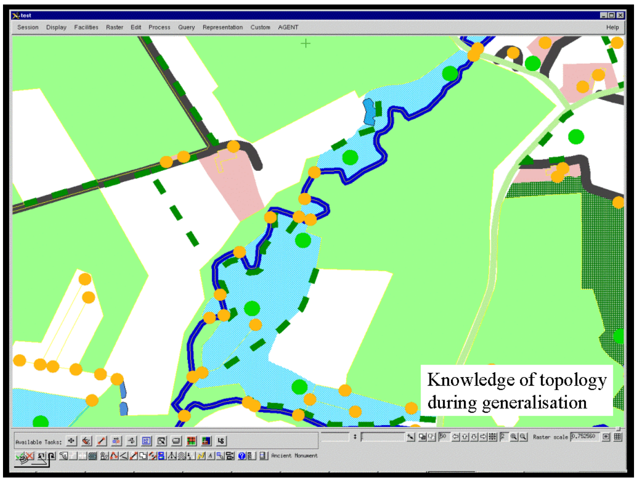


Figure 1 - Link/Node & Face topological structure



Knowledge of topology during generalisation

Figure 2 - Using Topology during Generalisation

In addition to line topology, which is made up of links, areas can be topologically structured, referencing links as their boundaries and one or more faces as their interior (Figure 1). Uniquely, the topology is maintained automatically as the data is edited, avoiding risks of overshoots, undershoots, slivers etc., and obviating the need for subsequent error-prone building of coverages. Polygons can be formed out of existing linework, either directly or as a set of faces (the atomic entities of area).

It is important to realise that topology relationships are established and maintained simultaneously between all chosen classes. There is no concept of ‘coverages’, so no risk of slivers and mismatches between themes, which are a problem for many traditional GIS products.

The created topology is a vital underpinning to many generated map and geodata products. Many data products (VPF, S57) require explicit topology references and single storage of shared edges. Topology is also key to good generalisation without destroying adjacency relationships. For example, if it is required during scale reduction to filter points from a river, and there are forests coming down to the river edge, then the filtering should be applied to the underlying topology link objects so as to avoid creating gaps or overlaps between the river and the forest (Figure 2).

2.3 Continuous mapping

Map-makers have always had to divide the world into rectangular tiles in order to make it fit onto sheets of paper. Traditional GIS continued the habit, using separate files to hold tiles of data. However, it seems to be a law of life (Hardy's law?) that the location you are interested in always seems to lie at corner of at least three map sheets or tiles!

One major strength of an object database as opposed to traditional GIS files, is that it is possible (indeed preferred) to store data as a continuous dataset with no sheet boundaries. The Gothic spatial object database does this, using an embedded spatial index (modified quadtree), and intelligent spatial storage clustering techniques to provide data retrieval performance, even for multi-gigabyte datasets.

As soon as one builds a continuous dataset, then there is an immediate need for version management and multi-user update locking through long transactions. The" copy, modify and replace whole tile" strategies used for traditional databanks cannot be applied to the whole of a 100GB dataset!

Gothic is a versioned database. In this, each user (or bulk-processing task) has a stable view of a 'version' of the dataset (Figure 3). Only changes made by that user are stored in that version, the unchanged objects are accessed from the previous version. This provides multi-user write access without lock bottlenecks.

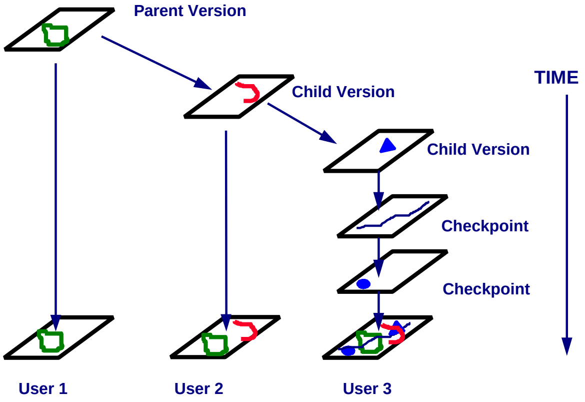


Figure 3 - Dataset versions

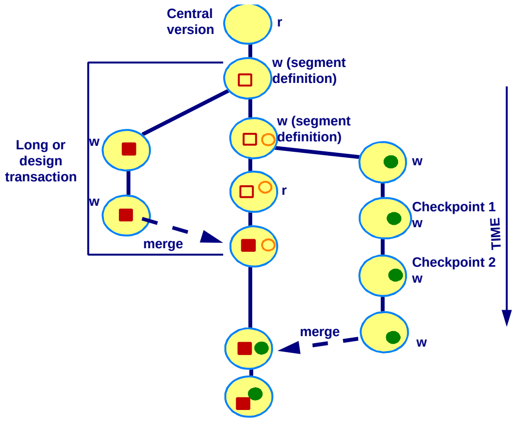


Figure 4 - Long transactions

In figure 4, two users each reserve a segment of a continuous dataset for update. After several checkpoints (e.g. stops for lunch or stages in bulk processing), the changes are merged to give an updated mainstream version.

Versioning solves the problem of local update, but the distribution of updates needs a different solution. As with local update, once one has a multi-gigabyte dataset, one can no longer send out complete copies when something changes. Neither do the tiles exist to localise change. The solution lies with incremental update (change-only update). This is made possible by the introduction of unique IDs, of a defined object lifecycle, and of flexible standards for intelligent transmission of data.

In particular, the use of the Geographic Markup Language (GML) which has been defined by the OpenGIS consortium (OGC, see http://www.OpenGIS.org/), is key. GML is based on the eXtensible Markup Language (XML) and is being extended with operators for incremental update. The first such set of extensions was prototyped by Laser-Scan for NIMA as SOTF - the Spatial Object Transfer Format [Hardy & Woodsford 2000]. Currently, the OSGB is defining a GML-based transfer format for DNF, which includes incremental update.

There is a recent realisation that the same GML-based techniques that are being deployed for transmitting updates to customers are also applicable to distributing the task of update of the central continuous database. In this strategy, the central database becomes a data warehouse or repository, and communicates by GML with the editing subsystems which take on the task of handling the editing processes and the subsequent validation. In the future, a three-tier approach to distributed update is likely to become prevalent, where the field updates units are very thin clients, communicating in real-time over a G3 mobile network with 'Intermediate Servers'. These handle versioning, editing and validation, and in turn communicate on a long transaction basis with the central repository.

2.4 Active Objects

The biggest difference between Gothic and a traditional GIS, is that the elements that are stored are not passive features but are active objects. Each object belongs to an object class, and has behaviours appropriate to that class. In O-O terminology, these are called 'methods'. Methods are invoked in response to messages, which are sent either automatically (reflex methods), or by operator request. Methods are of several types:

- Value methods return an answer to a message. The results appear as attributes on enquiry; e.g., what is my area? Similar value methods include length, description, and more complex answers like “what objects do I intersect with?”. Because value methods are only evaluated when they are needed, there is never a problem with out-of-date attributes as can happen when information like ‘area’ is stored as an attribute in a traditional GIS.
- Reflex methods occur automatically at milestones in an object's lifecycle: creation, modification or deletion (before and after). They are used to set up consequences of actions, and thereby ensure data integrity.
- Validation reflex methods enforce integrity, and allow you to put your own rules on each object class to prevent bad data ever getting into the system.
- Any change to a referenced object fires a reflex method, which can trigger propagation of effects from one object to another.
- Display methods give active representation. All drawing in Gothic is done by sending a message to objects saying, “draw yourself”. Objects can decide at the time how to draw themselves appropriately according to scale, position, and specification.
- Process methods happen at operator request. They are used to carry out data cleaning, data checking, polygon formation, and generalisation on defined sets of objects.
- Agent methods allow objects to ‘think for themselves’, and to increase their “happiness”. Agents behaviours are currently used for intelligent generalisation, but have potential for use in other processes such as conflation.

2.5 Object Processing

When applied to the large spatial data holdings described above, O-O provides a clean and consistent data model in which the individual geographic features (roads, houses, lakes) can be given much responsibility for their own behaviour. Object class methods can enforce data integrity, preventing bad data from being created. Unique object identifiers such as the Topographic Object ID (TOID) of DNF are key to defining a consistent object lifecycle, and thence to be able to handle change through time. Spatial processing methods can carry out spatial re-engineering in the context of the geographic feature, where all the necessary spatial information is at hand. Generalisation methods can make roads and rivers displace themselves to avoid conflicts.

One of the fundamental tenets of O-O is polymorphism, in that different object classes may respond to the same message by different method behaviours. For spatial data re-engineering and generalisation, this has particular strengths in that methods such as the 'simplify outline' generalisation method may have very different behaviours defined for man-made objects like buildings, to natural objects like lakes, even though they are both area objects [Ormsby and Mackaness 1999]. The advent of the Object-Oriented paradigm therefore opens up new strategies for spatial data processing, and for generalisation [Buttenfield 1995]. A recent overview of generalisation techniques from an object viewpoint is covered in [Harrie 2001].

LAMPS2 includes an Object-Oriented generalisation facility, which allows the user to define the strategy for generalisation in terms of methods on the object classes [Hardy 1999a, Hardy 1999b]. Generalisation base classes are provided which supply generalisation process methods for multi-object combination operations (aggregation, typification and displacement) and for object generalisation (collapsing, refinement, exaggeration and simplification). Note that these are implemented as behaviours of the objects in the database, not as commands within a program.

As well as for Generalisation, process methods provide a rich spatial framework for carrying out standard and user-defined spatial data re-engineering tasks. They have been heavily used by the Ordnance Survey in the re-engineering to produce DNF, as described in a later section.

2.6 Active Agent techniques

The power of object generalisation has been greatly enhanced by the AGENT project on multi-agent generalisation. This project [Lamy et al, 1999] was a collaboration under the ESPRIT programme (LTR/24939). It involved Laser-Scan as providers of object technology together with a national mapping agency (IGN) as prime contractor, and academic partners (Universities of Edinburgh & Zurich, and INPG Grenoble). Some partners provide in-depth knowledge of generalisation algorithms, while others provide insight into multi-agent modelling. The contract involved 48 person years of effort over a 3-year period.

In this context, agents are self-aware active software objects that co-operate, subject to a set of constraints, to achieve a goal. For map generalisation, it is the geographic objects such as houses and roads, which become active agents and co-operate through simplification, typification and displacement of themselves to achieve a cartographically acceptable generalised result. Figure 5 shows co-operating meso-agents handling urban blocks, communicating with the micro-agents that are the buildings and roads.

Agent-based generalisation achieves two major steps forward:

- An individual micro-agent (e.g. a single building) tries different generalisation algorithms, keeping the results if the situation is better, discarding and trying different ones if the situation is worse. This avoids the inappropriate use of the same tools globally throughout a dataset.
- Meso-agents (e.g. an urban block which contains many buildings and borders several roads) coordinate generalisation across sets of objects, so avoiding consequential conflicts and retaining “gestalt” (the overall coherence) of the data.

The philosophy underlying the AGENT project generalisation mechanism is for the designer to define how the map should appear rather than defining how to achieve it (declarative rather than imperative). Each agent then tries to improve its happiness during an agent life-cycle, which encompasses:

- Measures – implemented as object methods, these assess the tangible state of the feature.
- Constraints – implemented as objects, these return the degree of unhappiness for each aspect of state.
- Algorithms – implemented as methods, these change the state, and (hopefully) improve the situation
- Plans – dynamic lists of values, these determine the choice and order in which algorithms are executed.
- Re-evaluate – part of the lifecycle method, this determines if the previous plan was a success or not, and if not will backtrack the state in order to try another plan.

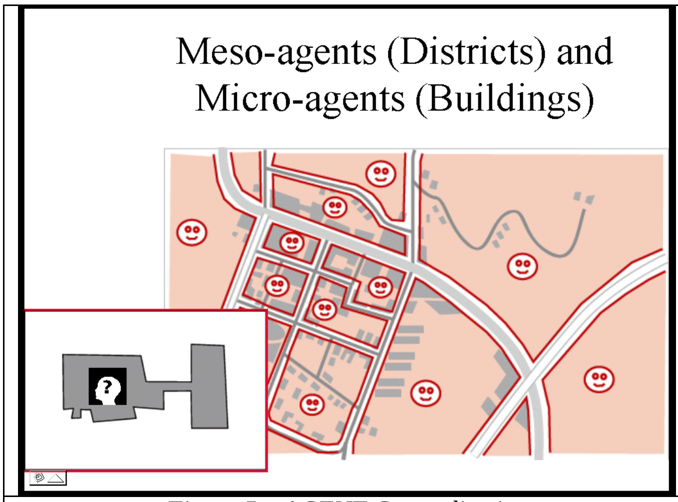

Meso-agents (Districts) and Micro-agents (Buildings)

Figure 5 – AGENT Generalisation

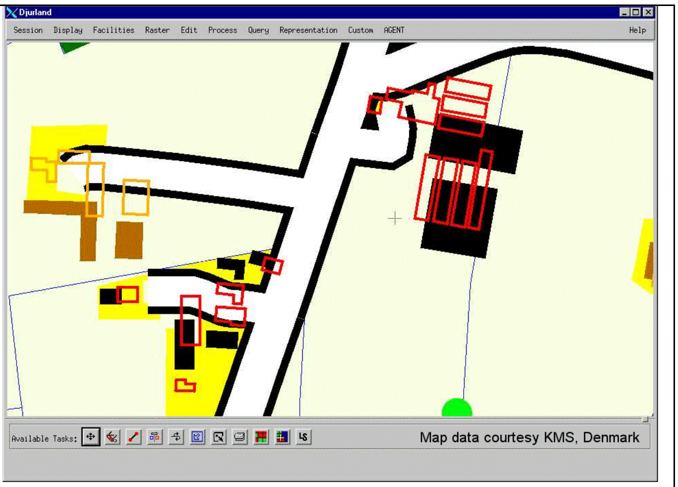


Figure 6 – KMS Agent Generalisation of farms

See the paper referenced at the head of this section and [Ruas 2000] for more information on the Agent project. See section 4.2 on KMS Denmark for a practical application of these techniques, as shown in Figure 6.

3 SPATIAL DATA RE-ENGINEERING TO CREATE USEFUL STRUCTURED DATA

3.1 Digital National Framework (DNF)

As described in the introduction, the OSGB is nearing completion of a major spatial data re-engineering project to produce a new base dataset for Britain - the Digital National Framework or DNF. This mammoth project involves taking the 230,000 sheets of LandLine spaghetti data through a series of processes to build a continuous object dataset. See http://www.ordnancesurvey.co.uk/dnf/ for background on the DNF project.

The starting point was the definition of a new data model called TOPO96, and the planning of the processes needed to populate it, starting from the original data. The original idea was to do a small amount of development and then spend several years pushing the original tiles through the flowline. However, it was later realised that a ‘fast track’ approach would be preferable, and that the advent of new Object-Oriented tools (notably Gothic and LAMPS2) made possible the processing of the complete dataset in one year, once customised development of cleaning processes was done.

The precise sequence of the re-engineering flowline is under OSGB control, but the main steps include ones similar to the following:

1. The LandLine sheet-based tiles of data are imported in chunks into Gothic. The chunks include several original tiles, so that edge matching and area building can happen across the original tile boundaries.
2. Cleaning processes are run which detect and correct logical inconsistencies in the original data

3. The basic topology is built, by turning on the topological rules of the data model and then running a process that picks each feature up and puts it down again. As well as building the link-node network structure, this applies prioritised snapping tolerances, and hence cleans up undershoots, overshoots and slivers.
4. The 'faces' that are the atomic units of the area topology are built. These faces refer to the links.
5. Cleaning processes are run which use the topological information together with feature code and attribute information to detect and remove geometric errors, such as duplicated features, spikes, loopbacks, and circular features.
6. A process is run looking for free ends in the network of linework. Some of these are intentional, but others are not, and marks are added to assist in later stages.
7. The links surrounding each face are traversed and the feature coding of the linear features sharing the links are inspected. Chaining rules are applied to the sets of features, which can detect errors and inconsistencies, and can help in the classification of the faces to form area polygons.
8. Further information in the classification processes comes from attribute coding of seed points present in the original data. These indicate buildings and some types of land cover (vegetation). Gothic face topology has explicit knowledge of containment, so that matching seeds with polygons is easy. Faces with two or more seeds or no seeds are detected for further analysis.
9. Where faces have two seeds and the analysis of the links defining the face indicates that these really should be two polygons, then a process can insert an 'inferred link'. One example of a common situation where this can be done is pairs of urban properties where the front garden is not divided by a full fence, but the fence line can be extended to make two sensible polygons. Another is a road junction, where separate polygons are formed for each street by inferred links to the road centreline junction.
10. Where a sensible classification for a polygon is achieved, it is coded appropriately. The minority of polygons where the classification is not apparent (e.g., due to coding errors or lack of ground features) are marked as 'unclassified'.
11. The re-engineered data, consisting of the complete polygons and their supporting linear and point features are exported to the data repository, where the unique object IDs (TOIDs) are allocated.

Once formed, the DNF dataset will form a national framework for linking other data. The new polygon features have a defined lifecycle, and their TOID will be retained for as long as they are recognisably 'the same feature'. DNF will also be particularly valuable, as it will form a starting point for derivation of products at other scales, by generalisation. It is also capable of much better visual presentation, as shown in figures 7 and 8, taken from the OGC Web Mapping Testbed 2 site at http://wmt.laser-scan.com/.

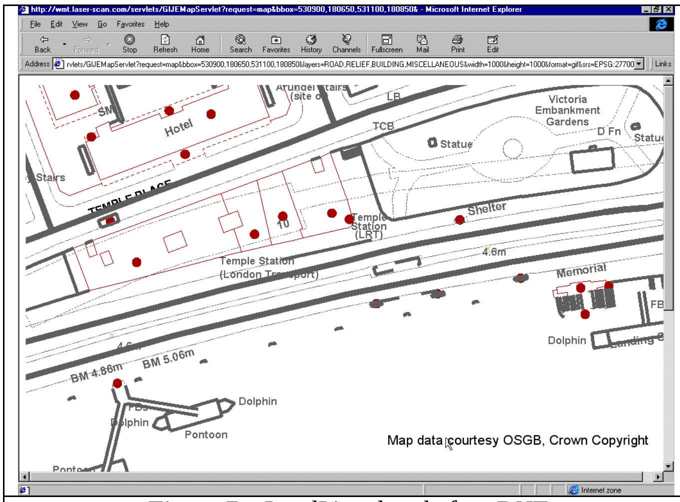


Figure 7 – LandLine data before DNF

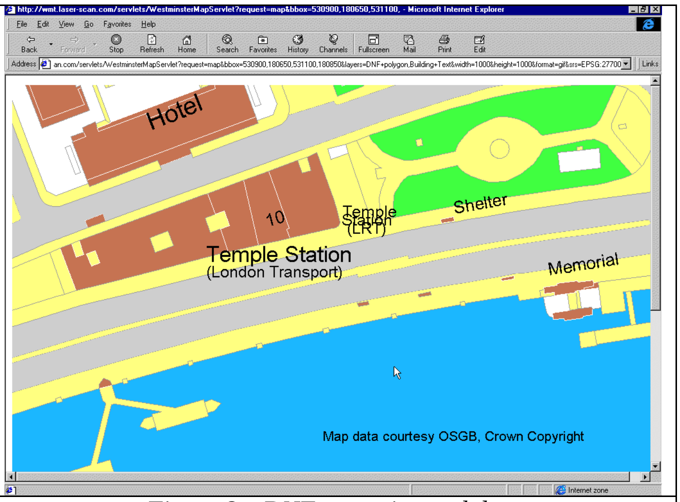


Figure 8 – DNF re-engineered data

3.2 Positional Accuracy Enhancement

Much OSGB detailed rural mapping is logically and thematically valid but positionally inaccurate (field lines date back to surveys of up to 150 years ago). A programme is underway to improve geometry while retaining attributes and logical structure for the areas of the country mapped at 1:2500. This is conflation task, albeit piecemeal. It is in parallel to the DNF re-engineering, but feeds into the resultant DNF dataset, improving its overall quality. It is being achieved partially by field resurvey, but largely by photogrammetric recapture, using Gothic LAMPS2 closely integrated with SOCET SET [Hardy 1999c]. The object techniques used involve retaining the topological connectivity and attributes from the original data, inserting new geometry captured by photogrammetry, and adding new metadata to the affected features, indicating that the change is due to positional accuracy improvement.

Other organisations have similar problems, e.g. U.S. Bureau of Census has large amounts of Tiger data which is attributed and structured, but not true to ground position. This mattered less in past, but as GPS becomes common, is in need of improvement by taking positional information from other sources and amalgamating it.

There are now new sources of positional information coming on stream, which will increase the scope for positional accuracy improvement in general. Some of these are imagery sources, such as the new 1m resolution commercial satellite imagery. Another very interesting source is the recent shuttle radar mission - the SRTM, which mapped almost the whole earth in a few days. It will take time before this data is processed and released, but it will be a significant free data source.

A very different source of future position data is the increasing use of Global Positioning System (GPS) receivers built in to cars and trucks. If the data could be collected (and valuable data tends to be collected), they would give very accurate information about the true positions of roads, including complex junctions.

3.3 CamMap Web Mapping

One freely accessible exemplar of the use of spatial data re-engineering and active object techniques in a web mapping environment is the CamMap site (http://www.CamMap.com/), [Hardy & Haire 2000]. This was built as a demonstrator from standard OSGB map linework datasets at different scales, imported into a consistent Gothic data model and re-engineered for the purpose of tourist information. Conflation techniques were applied to use a 1:10K dataset of important buildings to select detailed linework for those features from a more detailed LandLine dataset. These detailed lines were then used to build polygon objects. The display methods used by the web server application automatically switch between the coarse and detailed geometries for the object according to requested viewing scale. Note that these are not isolated independent objects, but the coarse object has direct references to the detailed geometry, and carries the single set of attributes that are accessible from either coarse or detailed geometry.

Display methods can do much more than just choose a representation for features. The spatial analysis available from the Gothic object toolkit accessible from within display methods can be used to do complex display techniques, such as label placement, which dynamically calculates the content, size and position for labels (Figure 9).

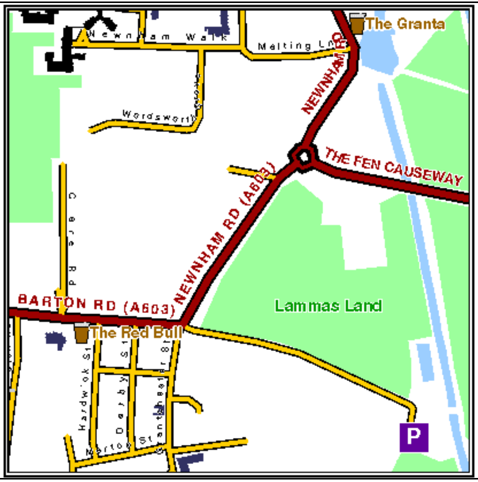
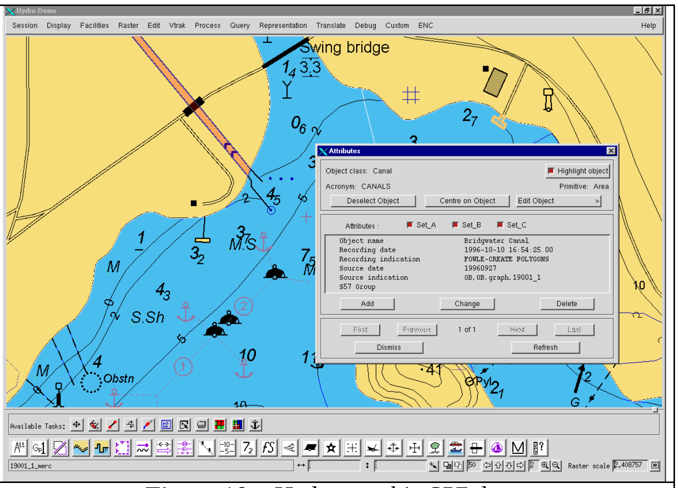


Figure 9 – Active text placement



Object class: Canal

Acronym: CANALS

Primitive: Area

Attributes :

Object name	Bridgwater Canal
Recording date	1996-10-10 16:54:25.00
Recording indication	FOWLE-CREATE POLYGONS
Source date	19960927
Source indication	GB,GB,graph,19001_1
S57 Group	

1 of 1

19001_1_merc

Raster scale 2.408757

Figure 10 – Hydrographic S57 data

3.4 Hydrographic S57 Electronic Navigation Charts

Ships traditionally navigated using paper charts, but there is a movement towards using Electronic Navigation Chart (ENC) data instead. Putting intelligent, attributed, structured data on board ship means that the console can alert the Captain that if the ship continues on the current course for 10 minutes it will run aground.

However, creating the necessary intelligent, attributed, structured data is a difficult task [Hardy 1998]. The main sources are usually digitised paper charts, where the features are dumb lines, coded according to their graphical representation. The United Kingdom Hydrographic Office (UKHO) produces the Admiralty series of paper charts which cover the world and are used for navigation by the majority of the world's shipping. When faced with producing S57, they have adopted Gothic LAMPS2 to create the ENC data, starting with their existing charts data.

The S57 data model is complex (Figure 10):

- Spatial objects are nodes, edges, or faces, and describe geographic position information. They must conform to a strict topological structure of connectivity and adjacency.
- Feature objects describe the characteristics (attributes) of a real-world entity (a coastline, a sounding, or an anchorage area). They refer to spatial objects to locate themselves. More than one feature object can refer to a given edge or node, sharing a common set of (X,Y) coordinate geometry. An example is a depth contour and the two depth areas either side of it.
- A set of area objects cover the 'Skin of the earth', plus others that are superimposed.
- There are hierarchies of collection objects, only the bottom level of which will have geographical position information (e.g., a complex traffic separation scheme may be made up of a collection of simpler traffic separation schemes, each of which is made up of traffic lanes).

The main steps in re-engineering the chart data to S57 include:

- Importing the original cartographic data (mainly lines and points) into an appropriate Gothic object model
- Adding the many attributes required by S57, automatically where possible.
- Turning chart notes into INFORM attributes on relevant objects.
- Topologically structuring the linework, including cleaning overshoots, undershoots and slivers.
- Building area (polygon) objects such as depth areas, from the structured line objects such as depth contours and coastlines. This is done for Group 1 (skin of the earth), and for Group 2 polygons.
- Building the collection objects such as traffic schemes.
- Validation using process methods which can check things such as whether all mandatory attributes are populated and have valid values.
- Transfer from chart space (e.g. Mercator projection) into the WGS84 Lat/long space needed for S57 or for databanking. As part of this, several charts may be combined into a single S57 cell, or into a continuum of cells.
- Constructing and attributing metadata objects such as coverage and extent areas, or data quality areas.
- Pre-processing prior to S57 export, e.g. to build the collections of similar soundings into aggregated sounding objects, each of which will be output as a 3D coordinate record to S57.

A similar form of spatial data re-engineering of hydrographic data is done by the U.S. hydrographic agency NOAA. They use Gothic LAMPS2 to bring together chart data and relational data, which are then integrated and re-engineered to produce S57.

4 GENERALISATION

4.1 The AGENT project

The AGENT project ended in November 2000, having made some dramatic advances in the state of the art for generalisation. Results were presented on December 5th 2000 to the EU commission in Brussels, and the project was deemed a success.

The fruits of the project were implemented during the project in prototype form in the Laser-Scan Gothic LAMPS2 mapping software, and are now being productised and commercialised [Hardy 2000]. The first customer is KMS Denmark, and a variety of European NMAs have shown great interest in the new techniques.

4.2 KMS Denmark

The Danish National Mapping Agency "Kort & Matrikelstyrelsen" (KMS) has completed a 1:10K topographic database of the whole country. However, their customers (particularly the military) require up-to-date paper mapping at a scale of 1:50K. Deriving one scale and resolution from the other requires generalisation, traditionally a labour-intensive, expensive task. Late in 1999, KMS carried out a competitive test of automated generalisation capabilities, and chose Laser-Scan as the company to work with in building a solution to meet this requirement.

Following successful testing, Laser-Scan is working alongside KMS to produce the specific generalisation capability for developing the 1:50K data product for the Danish Defence from the original 1:10K topographic database. This is very much a joint effort, with some of the development work being done by KMS in Denmark, and some by Laser-Scan in Cambridge. Integration of this work is then completed by KMS on-site. The production flowline is scheduled to be in place and operational during the spring of 2001. An initial target has been set to produce 10 sheets at 1:50K by the end of 2001.

KMS is relying on the active agent capabilities for applying its rules on building generalisation, including handling of rural farms. Figure 6 shows the original data in outline and the generalised output in solid, including the effects of aggregation, elimination, typification, and displacement.

One of the major driving forces behind the project for KMS was the recognised need for a more holistic approach than has previously been available. Rather than operating on a feature-by-feature basis (“feature-separated generalisation”), automated generalisation of map data requires that features are treated collectively, as any changes taking place have a knock-on effect on surrounding features within the data. For example if a road becomes displaced without consideration for surrounding data, it may then find itself sitting squarely on top of a row of houses and passing through a hillside. Treating features collectively, within a delimited area, means that topological integrity can be taken into account, therefore maintaining connectivity within the data during and after generalisation, i.e. the process does not introduce more problems into the data or lose data integrity.

4.3 KMS Rules for Generalisation

The complexities of generalisation have been broken down, and where relevant, standard generalisation object processing methods have been applied. In other cases, new functionality has been implemented as object methods to deal with specific cases. Some examples of the generalisation issues addressed are listed below:

4.3.1 Urban Buildings

Lying within urban areas (such as industrial areas)

- Buildings smaller than a set size are deleted
- Buildings within user-defined size bands are represented as rectangles
- Buildings larger than a user-defined size are simplified
    - e.g. < 25 m2, delete
    - 25-80 m2, represent as rectangle size 15m x 15m
    - 80-400 m2, represent as rectangle size 13.3m x 30m
    - >400 m2, simplify
- Buildings closer than a user-defined distance should be merged together
- Buildings must not overlap with roads or each other
- Buildings must remain inside their original urban-areas (they should be displaced, rotated or deleted to remove the conflicts)

4.3.2 Farms

Clusters of buildings are identified as belonging to individual farms.

- The buildings should be enlarged, deleted or merged together (c.f. urban areas) depending on their size and proximity to one another
- Having been enlarged, new overlaps between buildings should be removed by dilation of the farm buildings (movement of all buildings away from the centre of the farm while maintaining the same relative positions)
- If any part of the group of farm buildings overlaps a road, the entire group should be displaced thus maintaining the overall shape (but not position) of the farm
- Farms should not conflict with other farms, other rural buildings or urban areas. They should be displaced to remove the conflicts

4.3.3 Rural Buildings

Buildings not in urban areas and not part of a farm are termed rural buildings.

- Clusters of rural buildings lying within a close proximity of each other should be identified
- Buildings should be merged, deleted, enlarged or simplified according to their size and proximity to other buildings
- Buildings should not conflict with roads, other buildings, urban areas or farms. The conflicts should be removed by displacing the buildings, maintaining the relative positions of buildings within clusters where appropriate.

4.3.4 Area objects

- Area objects such as forests or built-up areas should be merged together when they lie within a user-definable distance of each other
- The merging process should generate holes in the final area where appropriate, and should follow the outline of the original areas as closely as possible (i.e. not produce a convex-hull or shrink-wrap of the areas being combined)
- Narrow gaps in the areas being combined should be removed
- Area objects which are contained within a larger "name area" should be merged together, and if necessary broken across the outer edge of the "name area"

4.3.5 Roads

- Delete all minor roads less than 2km long which have a dead end and do not have buildings within 100m of the dead end
- Parallel roads (e.g. dual carriageways) should be collapsed to a single line
- Roundabouts should be identified and replaced with a point at their centre (to be replaced by a symbol)
- Road junctions should be identified and simplified
- Sliproads should be identified
- Bridges should be identified

5 CONCLUSION

- Spatial Data Re-engineering is a necessity for organisations such as National Mapping Agencies, which have built up holdings of spatial data from existing mapping, and now find that the structure and content do not match the needs of today's users.
- Re-engineering of spatial data has in the past been very difficult, given the constraints of feature-based GIS, static (or no) topology and map-sheet-based data storage.
- Now, powerful and cost-efficient spatial data re-engineering is possible, using active object software with rich data modelling capabilities and dynamic topology.
- There is still scope for improvement, e.g. by addition of new constraints and algorithms into the Agent generalisation, and by applying active object techniques to new types of re-engineering and other aspects of spatial data handling, e.g. text placement.
- The challenge of conflation remains, and will become a vital capability in next few years. Active objects and dynamic topology are likely to be key to solving conflation issues.

6 ACKNOWLEDGEMENTS

Thanks are due to the members of the AGENT Consortium, for material on Agent-based generalisation, to OSGB for DNF material and to KMS for material on their generalisation strategies.

7 REFERENCES

Buttenfield, B. P., 1995, "Object Oriented Map Generalization: Modelling and Cartographic Considerations", in J C Muller, J. P. L. a. R. W., ed., GIS and Generalization: Methodology and Practise, Bristol, Taylor and Francis, p. 91-105.

Hardy P.G. 1998, "S57 ECDIS Data Production and Update Using an Object-Oriented Spatial Database", International Hydrographic ECDIS Conference, Singapore

Hardy P.G., 1999a, "Map Generalisation - The Laser-Scan Way", on-line paper at http://www.Laser-Scan.com/papers/lamps2mapgen.htm

Hardy P.G., 1999b, "Active Object Techniques for Production of Multiple Map and Geodata Products from a Spatial Database", ICA/ACI Conference Proceedings, August 1999, Ottawa, Canada, or online at http://www.Laser-Scan.com/papers/ica99gwg/pghica99gwg.htm

Hardy, P.G., 1999c "Integrating Active Objects with Stereo Images for Map Production", RICS Geomatics Symposium, Nottingham, UK, October 1999, or online at http://www.Laser-Scan.com/papers/PGH_Kartdagar2000b.pdf

Hardy P.G. and Haire, K., 2000, "Generalisation, Web Mapping and Data Delivery over the Internet", ICA workshop, November 2000, Barcelona, Spain, or online at http://www.Laser-Scan.com/papers/barcelona2000pgh.pdf

Hardy P.G. 2000, "Multi-Scale Database Generalisation For Topographic Mapping, Hydrography And Web-Mapping, Using Active Object Techniques", IAPRS, Vol. XXXIII, ISPRS Amsterdam, Netherlands, July 2000, or online at http://www.Laser-Scan.com/papers/isprs2000pgh_1436.pdf

Hardy P.G. and Woodsford P.A., 2000, "Incremental Updating using the Gothic Versioned Object Database with the Hydrographic S57 ENC and SOTF Spatial Object Transfer Formats", ICA/ISPRS Workshop on Incremental Updating and Versioning of Spatial Databases, Amsterdam, Netherlands, July 2000, or online at http://www.Laser-Scan.com/papers/wsiuv_2000pghpaw.pdf

Harrie L., 2001, "An Optimisation Approach to Cartographic Generalisation ", Doctoral thesis, University of Lund, Sweden.

Howard G., Pickering R., Mole D., and Woodsford P., 1999, "A National Topographic Database for the 21st Century - Paradigm Shifts in Business Process and Technology", ICA/ACI Conference Proceedings, August 1999, Ottawa, Canada.

Lamy, S., et al, 1999, "The Application of Agents in Automated Map Generalisation", ICA/ACI Conference Proceedings, August 1999, Ottawa, Canada.

OSGB, 2001, "DNF - The Digital National Framework", http://www.ordnancesurvey.co.uk/dnf/.

Ormsby, D., and Mackaness, W. A., 1999, "The Development of Phenomenonological Generalisation Within an Object Oriented Paradigm": Cartography and Geographical Information Systems, v. 26, p. 70-80.
Ruas A., 2000, “Project AGENT: Overview and Results of a European R&D Project in Map Generalisation”, ICA Workshop, November 2000, Barcelona, Spain.
Worboys, M. F., Hearnshaw, H. M., and Maguire, D. J., 1990, "Object-Oriented data modelling for spatial databases": International Journal of Geographical Information Systems, v. 4, p. 369-384.

[Original 2001-04-22, Revised 2001-05-03]