A prototype for ontology driven on-demand mapping of urban traffic accidents

Nicholas Gould[1], Jianquan Cheng[2]

Manchester Metropolitan University, Chester Street, Manchester, UK
Email: [1]nicholas.m.gould@stu.mmu.ac.uk
Email: [2]j.cheng@mmu.ac.uk

Can the concepts of cartographic generalisation be formalised in an ontology with sufficient detail to allow the process to be automated?

1 Introduction

Government, both national and local, is making increasing amounts of spatial data freely available. The DataGM website, for example, provides access to georeferenced data for road traffic accidents, fire and rescue incidents, bus stops, bus routes and traffic signals in Greater Manchester (Trafford Council 2012). However, how can thousands of road accidents be mapped legibly by the non-expert cartographer without obscuring the underlying road network? Tools such as the Google maps API provide only a partial solution in that they merely overlay data on base maps. There is no integration of user-supplied data. What is required is cartographic generalisation on-demand. But to automate the map creation process it is necessary to formalise the knowledge required for generalisation (Touya et al. 2010).

2 Why use an ontology?

The prevailing paradigm for automatic generalisation is constraint-based, in particular agent-based, modelling (Harrie and Weibel 2007). Agent-based systems require a knowledge base that has to be updated each time a new generalisation algorithm is introduced or when user requirements change (Taillandier and Taillandier 2012). What happens when an end-user wishes to map features of an unfamiliar type, such as road accidents? The knowledge required to generalise these features, their attributes, relevant operations and relations with other features, has to be encoded. In effect, cartographic knowledge is embedded in the configurations of sophisticated software applications. Ideally that knowledge, once defined, would be shared.

One possible option for representing domain knowledge in a sharable and reusable manner is to employ ontologies (Gruber 1993). An ontology captures the semantics of the concepts in a domain and is not merely a classification (Kavouras and Kokla 2008). It has been argued that all information systems are based on implicit ontologies and making the ontology explicit avoids conflicts between ontological concepts and their implementation (Fonseca et al. 2002). The use of ontologies to realise semantic interoperability in a distributed environment is well researched (Lemmens et al. 2007; Lüscher et al. 2007; Janowicz et al. 2010) and Regnauld (2007) proposed an on-demand mapping system based on ontologies.

Ontologies are more than taxonomies; we can reason with them and apply them to decision-making. Such ontologies, used in scientific workflows, are rare (Janowicz et al. 2012). In other domains ontologies have been used to match students to courses (Kontopoulos et al. 2008) and applicants to jobs (García-Sánchez et al. 2006). This paper describes an attempt to use an ontology to model the process of generalisation, in an effort to facilitate the automation of map generation at different scales.

The aim of this project is to determine the *why*, *when* and *how* of generalisation (McMaster and Shea 1992). The need to produce a legible map is the reason *why* we need to generalise. The existence of geometric conditions (congestion, imperceptibility) in the mapped data determines the *when*. The existence of these conditions can be determined by using measures such as the density and distribution of features (Stigmar and Harrie 2011). *How* is answered by the use of generalisation operations such as Amalgamation and Collapse. The goal is to use an ontology to help choose which measures and operations to apply.

There is no single correct way of modelling a domain and ontological engineering is necessarily an iterative process (Noy and McGuinness 2001). There are a number of methodologies available to guide the process (Sure et al. 2009) but in our case the "simple" method described by Noy and McGuinness (2001) was employed. This involves defining a set of *competency questions* that the ontology is expected to answer. These include: *what measure algorithm should be used for a particular condition*? *What operation will alleviate the condition*?

The ontology employs a (loose) medical analogy which describes *conditions* (such as feature congestion) that are characterised by *symptoms* (e.g. high feature density) which have *remedies* (e.g. feature count reduction). The remedies are implemented by operations which in turn are implemented by transformation[1] algorithms.

The applicability of an operation to a given condition of a given set of features is governed by a number of factors, primarily geometry (for example, point features cannot be collapsed; pruning only applies to line features, specifically a network of line features). Geometry alone is not sufficient, however. The choice of operation is also governed by a number of *requirements* and *restrictions*. SelectionByAttribute requires that the source data has an attribute (field) that can be used to rank the features by importance. Amalgamation is restricted from application to a road network since Amalgamation is a form of Abstraction, which is forbidden for a Network by the ontology.

The intention is to describe the operations sufficiently that they can be selected automatically. It is also necessary to define the output geometry of the operation since that will influence the selection of subsequent operations. Any unintended consequences of the operation also have to be described. For example the process of Amalgamation, in abstracting the original features, will remove any importance attribute from the features and thus prohibit the use of SelectionByAttribute in subsequent operations.

The ontology, stored as an OWL2 (Web Ontology Language) file, was developed using the Protégé ontology editor (Horridge 2011) and it could be tested by issuing queries from within the editor. However, to fully test the concept a prototype was developed.

3 Implementation

The prototype consists of using the ontology to select appropriate measure algorithms, applying those measure algorithms and, if a condition was identified, selecting an operation (and transformation algorithm) to remedy the condition.

In the current prototype (Figure 1) the measure and transformation algorithms, implemented as Java methods, form part of the system, but it is envisaged that these will be provided in future by web services such as WEBGEN (Burghardt et al. 2005). The following sections describe the main components of the prototype.

[1] The word transformation is used instead of generalisation because of doubts over what operations actually constitute generalisation (Foerster et al. 2007).

3.1 Source data

In the first instance the prototype is restricted to mapping traffic accidents with roads as a base. The accidents are in Greater Manchester, UK, over a 12 year period. Each accident has a severity attribute with values of 1 (fatal), 2 (severe), or 3 (slight). The base road network is Ordnance Survey's MasterMap road features (polygons) and their Integrated Transport Network (ITN) road network (lines)[2].

The source dataset is described as an *individual* (or instance) in the ontology using attributes such as Geometry and FeatureType. In the current implementation the source data is held as Shape files but future versions could use web services.

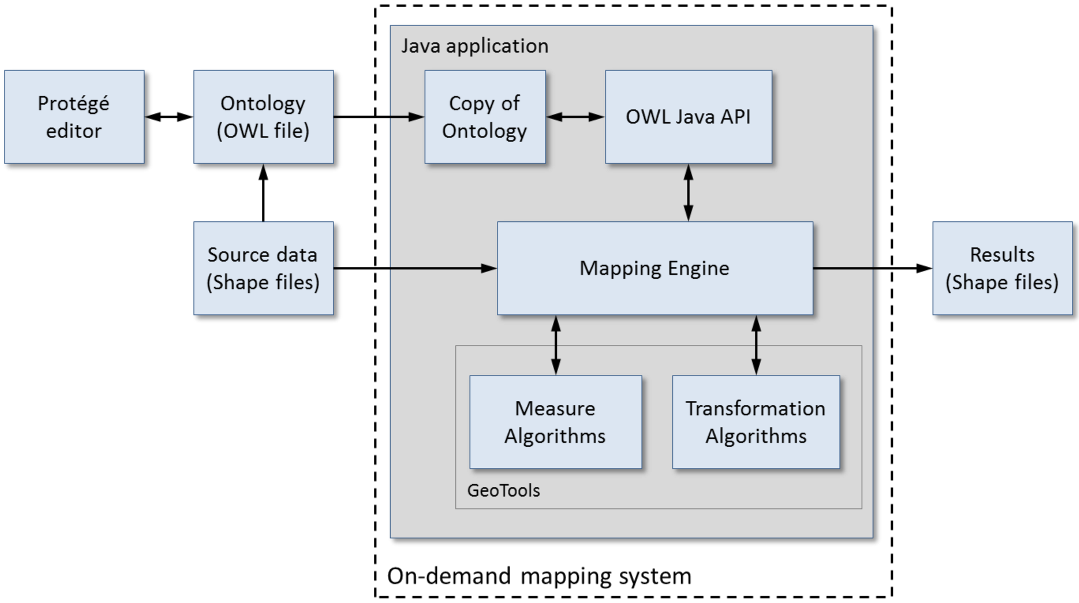


Figure 1. On-demand mapping system prototype architecture.

3.2 Using the ontology

Interaction between the Java application and the ontology is done via the OWL API (Horridge and Bechhofer 2011). Most of the calls to the ontology consist of queries in the form of Manchester OWL syntax. The syntax is verbose but human-readable, which makes development easier. The following is an example of query string generated by the Mapping Engine and executed against the ontology, which will return a list of measure algorithms meeting the specified criteria:

```
MeasureAlgorithm and measures some HighFeatureDensity and
hasInputGeometry some AreaGeometry
```

The first usage of the ontology is to identify an appropriate measure algorithm for each *mapped feature collection*, where a mapped feature collection is defined as a set of features, of the same type, in the user's selected bounding box.

3.3 Measure algorithms

A number of measure algorithms were developed for the prototype, including one to measure the density of point features and others to measure the density of road features (as areas and as lines). There is also an algorithm to measure the density of generic area features, such as amalgamated clusters of accidents. The algorithms begin by

[2] The ITN network is used to simulate a collapse of the Master Map road features.

identifying clusters in the source data. What constitutes a cluster is determined by scale. If the density of features in a cluster exceeds a threshold then the clusters are returned flagged as high density (Figure 2). The focus on the density of features is because we are primarily interested in resolving the congestion of features.

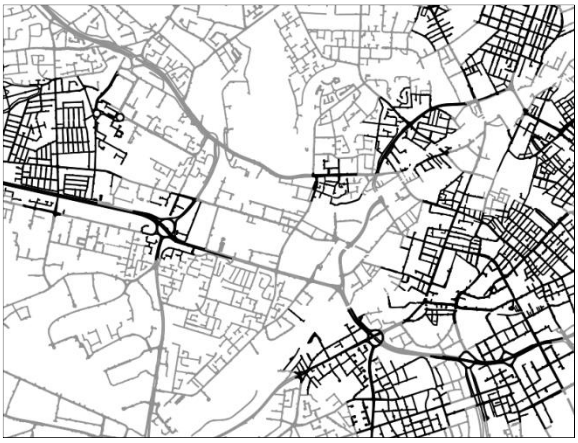


Figure 2. Regions of high crossroad density (in black).

If a mapped feature collection is deemed to have a particular condition (e.g. congestion) then the next stage is to select an appropriate operation to remedy the problem and then to choose an algorithm that implements that operation.

3.4 Transformation algorithms

The transformation algorithms implemented in the prototype are governed by a *DegreeOfGeneralisation* parameter (Zhou and Jones 2003). The value of this parameter (1 = low, 9 = high) is governed by the output from the respective measure algorithm. The higher the number of congested features found, the higher the value of the DegreeOfGeneralisation. The exception is the Collapse algorithm which is a binary - all or nothing - operation. A more sophisticated version of the prototype would consider then number of clusters and their distribution rather than simply the total number of clustered features.

The *SelectionByAttribute* operation will only be suggested by the ontology if the source data has an *ImportanceAttribute* defined in its ontology description. This is simply the name of a data attribute (field) that can be used to rank the features by importance. The accident data has a *severity* attribute that can serve as such.

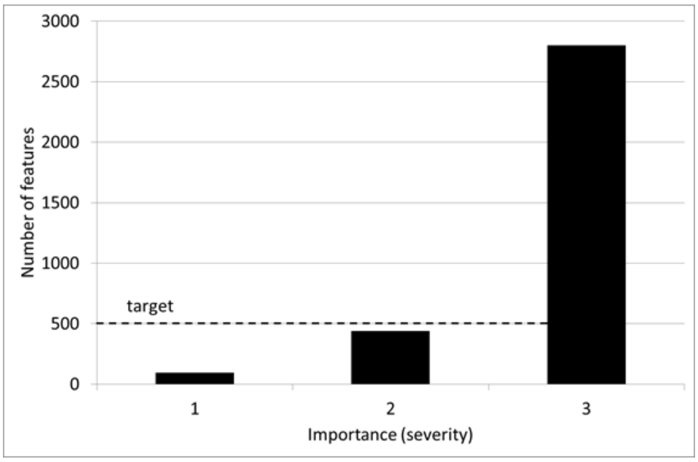


Figure 3. Distribution of accident importance

The algorithm that implements SelectionByAttribute uses the DegreeOfGeneralisation to determine the number of features to retain. This value is then used to determine which features to retain. For example, if the number of features to retain was 500 then the algorithm would return features of importance value of 1 and 2 (Figure 3).

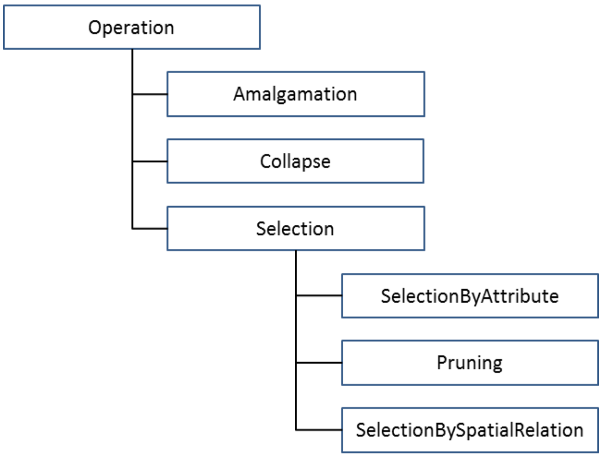


Figure 4. Operation sub-classes in the ontology

The same principle is used to govern the pruning algorithm applied to the road network. *Pruning*, like *SelectionByAttribute*, is a form of *Selection* (Figure 4). However, in this case the algorithm uses the DegreeOfGeneralisation to determine the *total length* of the features to be retained. The pruning algorithm uses strokes to determine which road features to retain (Thomson and Richardson 1999). The longer the stroke the higher its importance and the more likely it is to be retained.

3.5 Workflow

The user selects two data sources (accidents and roads). The order of selection is important; it means that the roads have to be generalised with respect to the accidents. The user is presented with a starting bounding box with the features displayed at a large scale. The workflow (Figure 5) is triggered when the user either pans or changes scale by zooming. The features in the bounding box are defined as a *mapped feature collection*, one for each feature type (i.e. one for road sections and one for accidents).

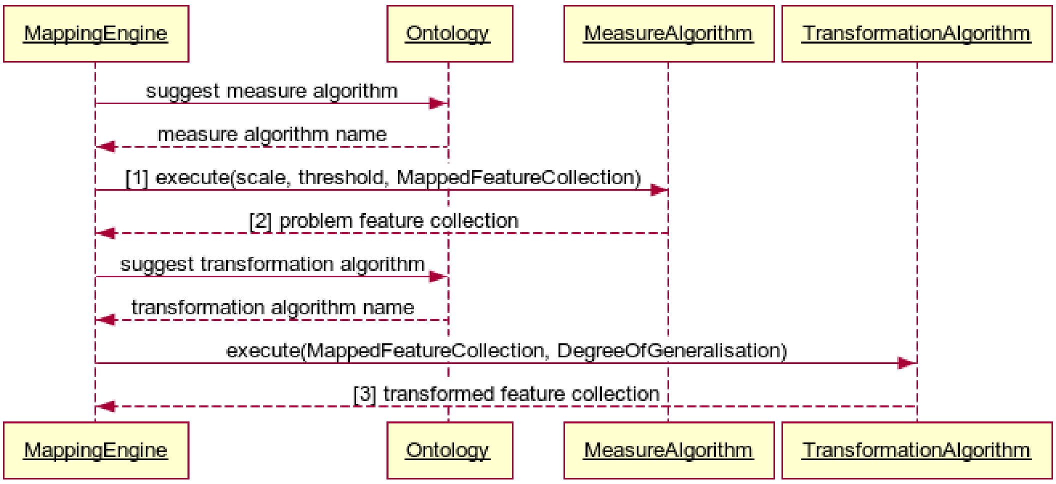


Figure 5. UML sequence diagram (single mapped feature collection)

The workflow is applied to each mapped feature collection in turn for a given scale. The entire process is complete when there are no *conditions* identified by the measure algorithms for any of the mapped feature collections. If the user chooses another scale then the process is repeated, starting with the original source data.

If the *problem feature collection* (those features in the mapped feature collection identified as having the condition) is empty (step [2], Figure 5) then the sequence stops for that particular mapped feature collection. After step [3], step [1] is called again to assess the effect of the transformation. As depicted, the transformation algorithm is applied to all features in the mapped feature collection, not only those identified as problem features. This is, in fact, dependent on the transformation operation. For example, for *collapse* the transformation will apply to all features in the class but for *amalgamation* only those features in the problem feature collection will be affected.

The application of a transformation algorithm will change the data (e.g. from a cluster of point features to an area feature). However, the changes enacted by the transformation have to be reflected in the ontology; this is why a working copy of the ontology is made (Figure 1). The semantics of the features may have changed as well as the geometry. For example, the feature type of a set of amalgamated accidents is no longer AccidentFeatureType. The changes in the semantics may well effect what measure and transformation algorithms are applicable in subsequent iterations (if required). This process is known as *semantic propagation* (Janowicz et al. 2010).

4 Preliminary results

Sample results, exported to a Shape file, can be seen in Figure 6. The road network has been collapsed then pruned with a DegreeOfGeneralisation of 6. The accidents have been amalgamated. The pruning algorithm requires further work as there are dead-ends in the network. A combined stroke and mesh approach (Li and Zhou 2012) could be applied. More importantly, the spatial relation between the roads and the accident needs to be respected by the pruning algorithm as can be seen by the isolated accident cluster in the centre-right of the image. This can be done by giving a high weight in the pruning process to those roads intersecting an accident cluster.

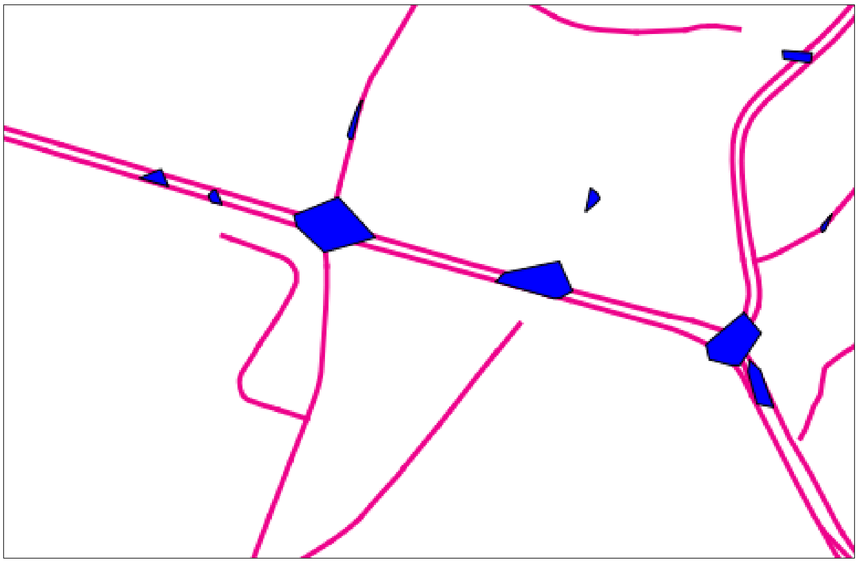


Figure 6. Generalised roads (lines) and accidents (polygons)

The amalgamation algorithm only returns features in a cluster; any features not in a cluster are discarded. In future this could be left to the discretion of the user. As yet no attempt has been made to formally evaluate the results from a user's point of view, the main concern being to see if the system could automatically identify congestion in the features and then automatically identify suitable transformation algorithms and apply them using appropriate parameter values. This was achieved with the provisos described above.

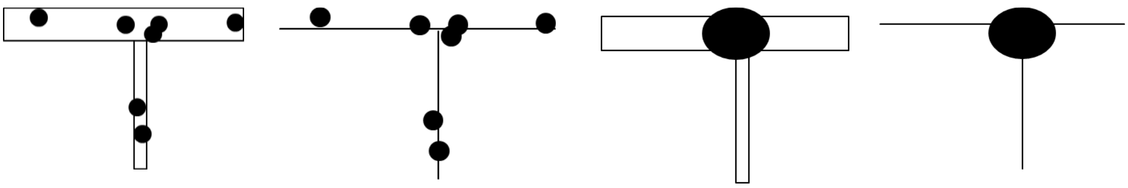
5 Discussion and further work

5.1 What are the expectations of the ontology?

What can we expect the ontology to do and what can it leave to the mapping engine? As it stands, the ontology may return more than one relevant operation, and hence more than one relevant transformation algorithm, for a given condition found in a given set of features. If this is the case then the user currently has to select which one to apply. For example, when congestion in the road accident features is identified, the ontology currently suggests Amalgamation and SelectionByAttribute. This is not ideal for an on-demand mapping system aimed at non-expert users. One possibility is to utilise an optimisation method to select the best operation from those suggested. But should the ontology do more? Can it indicate a preferred operation, perhaps influenced by user preferences?

Another possible deficiency of the ontology is that there may be conditions that are best solved by a sequence of operations (e.g. selection, pruning, and then smoothing). However, as it stands, the ontology only suggests atomic tasks.

A key requirement of an on-demand mapping system is that when generalising the topographic data it should respect any relationship it has with the thematic data. The relationship in our use case is initially semantic; a road accident, by definition, takes places on a road. This semantic relation can be expressed as a spatial relation. However, the exact nature of the relation is dependent on the current geometry of each feature type, which may change following generalisation. For example, when accidents are represented as points and road sections as polygons then accidents are *contained by* roads Figure 7a. If a cluster of accidents is represented by a polygon, then the relationship is *intersects* (Figure 7c and Figure 7d).



(a) contained by (b) adjacent (c) intersects (d) intersects

Figure 7 Spatial relation between accidents and roads for different geometries

The ontology is currently being modified to describe these relations (Figure 8) based on the models described by Jaara et al. (2012) and Touya et al. (2012).

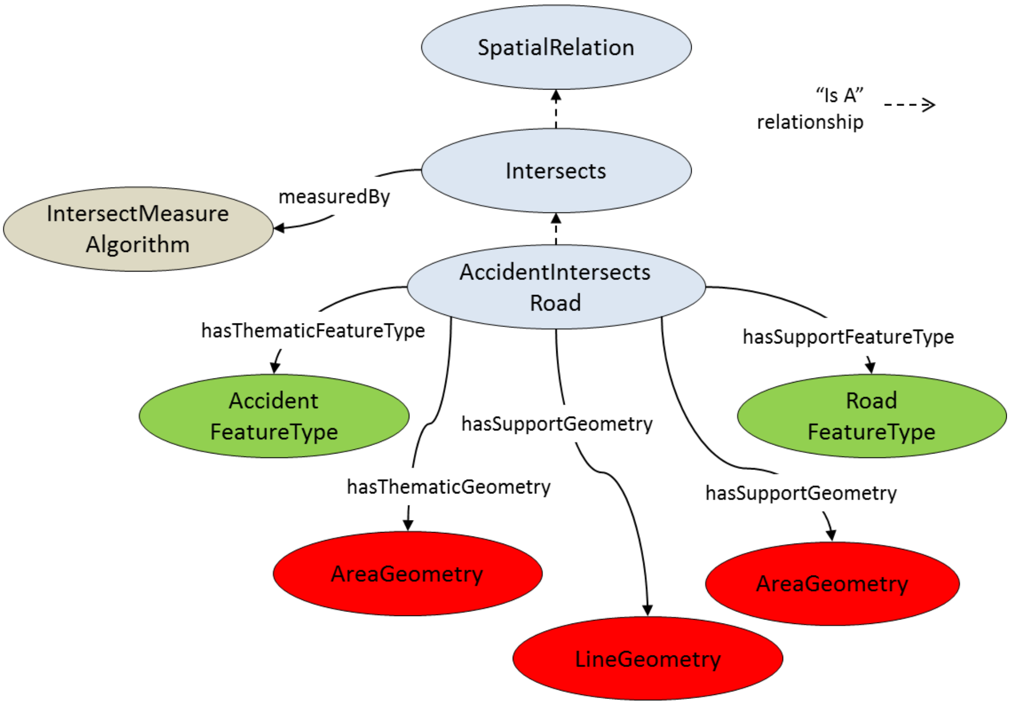


Figure 8 Spatial relations model for describing the intersects relation

The spatial relations model in Figure 8 describes only the intersects relation between accidents and roads depicted in Figure 7c and Figure 7d. The ontology is describing relationships between classes of features and not individuals. The terms *thematic* and *support* (Jaara et al. 2012) were adopted as they are more expressive than term such as *member1*, *member2* (Touya et al. (2012). Given a current feature type and geometry for both mapped feature collections it should be possible to determine any relevant spatial relations and then determine how it is measured. This information can then be passed to a road pruning algorithm, say, to ensure that the algorithm respects any relation when pruning.

In general, further consideration needs to be given to the relative roles of the ontology and the mapping engine. Ideally we would want to expect as much as possible of the ontology since it is a formalisation that can be shared whereas the mapping engine is local and proprietary.

5.2 Scale

The prototype has only been tested over a range of relatively large scales. At smaller scales, where there are a high number of features to be mapped, processing times for the measure algorithms become very long. One solution could be to make the selection of the measure algorithms scale dependent; for example, at small scales a quick estimate could be used to assess a condition.

Similarly, for a given condition for a given set of features, the ontology suggests the same operation(s) whatever the scale; all that changes is the DegreeOfGeneralisation. It would be useful to investigate the possibilities of making the choice of operation scale-dependent.

The processing speed of the transformation algorithms is also affected by scale. Each time the user changes scale then the process (Figure 5) is repeated starting each time with the raw source data; as the user zooms out there is no progressive generalisation. This may offer a further route to optimisation, perhaps by utilising a Multiple Representation Database (MRDB).

5.3 Further work

One advantage of the ontological approach can be shown by considering the Selection operation. As the prototype was developed it was realised that the Selection operation was too general and sub-classes were added (Figure 4). This refinement of the ontology did not require any changes to the mapping engine.

This raises the question of the applicability of the approach. Can it be extended beyond mapping traffic accidents? If, for example, we wished to map bus routes would this merely require the additional description of bus routes and their properties (line geometry, routes follow roads etc.) in the ontology? The next stage, after refining the transformation algorithms, is to test the prototype with different use cases and to test more conditions, not just congestion, such as feature imperceptibility.

Finally, given the limitations of the model described above, it may be that the role of ontologies in generalisation is to support other models such as agent-based systems by providing a shared, formalised knowledge base. This application of ontologies requires further investigation.

Acknowledgements

This project is funded by the Dalton Research Institute at Manchester Metropolitan University and the Ordnance Survey of Great Britain. Thanks are due to Nicolas Regnauld, formerly of the Ordnance Survey, and William Mackaness of the University of Edinburgh for their support and guidance.

References

Burghardt D, Neun M and Weibel R, 2005, Generalization Services on the Web - Classification and an Initial Prototype Implementation. *Cartography and Geographic Information Science*, 32(4) pp. 257-268.

Fonseca FT, Egenhofer MJ, Agouris P and Câmara G, 2002, Using Ontologies for Integrated Geographic Information Systems. *Transactions in GIS*, 6(3) pp. 231-257.

Foerster T, Stoter J and Köbben B, 2007, Towards a formal classification of generalization operators. In *International Cartographic Conference 2007*. Moscow, 4-10 August, 2007

García-Sánchez F, Martínez-Béjar R, Contreras L, Fernández-Breis JT and Castellanos-Nieves D, 2006, An ontology-based intelligent system for recruitment. *Expert Systems with Applications*, 31(2), pp. 248-263.

Gruber T, 1993, A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2) pp. 199-220.

Harrie L and Weibel R, 2007, Modelling the Overall Process of Generalisation. In Mackaness WA, Ruas A and Sarjakoski L (eds), *Generalisation of Geographic Information*. Amsterdam: Elsevier Science B.V., pp. 67-87.

Horridge M, 2011, *A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools Edition 1.3*. [Online] [Accessed on 12th March 2013] http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP4_v1_3.pdf

Horridge M and Bechhofer S, 2011, The OWL API: A Java API for OWL ontologies. *Semantic Web*, 2(1), pp. 11-21.

Janowicz K, Schade S, Broring A, Kessler C, Maue P and Stasch C, 2010, Semantic Enablement for Spatial Data Infrastructures. *Transactions in GIS*, 14(2) p. 111.

Janowicz K, Scheider S, Pehle T and Hart G, 2012, Geospatial semantics and linked spatiotemporal data – Past, present, and future. *Semantic Web*, 3(4) pp. 321-332.

Jaara K, Duchêne C and Ruas A, 2012, A Model for Preserving the Consistency between Topographic and Thematic Layers throughout Data Migration. In *Proceedings of the Short Papers of the SDH2012*. Bonn, Germany, 21st-24th August 2012

Kavouras M and Kokla M, 2008, *Theories of geographic concepts: ontological approaches to semantic integration*. Boca Raton: CRC Press.

Kontopoulos E, Vrakas D, Kokkoras F, Bassiliades N and Vlahavas I, 2008, An ontology-based planning system for e-course generation. *Expert Systems with Applications*, 35(1-2), pp. 398-406.

Lemmens R, De By R, Gould M, Wytzisk A, Granell C and Van Oosterom P, 2007, Enhancing Geo-Service Chaining through Deep Service Descriptions. *Transactions in GIS*, 11(6) pp. 849-871.

Li Z and Zhou Q, 2012, Integration of linear and areal hierarchies for continuous multi-scale representation of road networks. *International Journal of Geographical Information Science*, 26(5) pp. 855-880.

Lüscher P, Burghardt D and Weibel R, 2007, Ontology-driven Enrichment of Spatial Databases. In *10th Workshop of the ICA commission on Generalisation and Multiple Representation*. Moscow, Russia.

McMaster RB and Shea KS, 1992, *Generalization in digital cartography*. Washington, D.C.: Association of American Geographers.

Noy NF and McGuinness D, 2001, *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford University.

Regnauld N, 2007, A distributed system architecture to provide on-demand mapping. In *International Cartographic Conference*. Moscow, 4-10 August 2007.

Stigmar H and Harrie L, 2011, Evaluation of Analytical Measures of Map Legibility. *The Cartographic Journal*, 48(1) pp. 41-53.

Sure Y, Staab S and Studer R, 2009, Ontology Engineering Methodology. In Staab S and Studer R (eds), *Handbook on Ontologies*. Berlin/Heidelburg: Springer-Verlag.

Taillandier P and Taillandier F, 2012, Multi-criteria diagnosis of control knowledge for cartographic generalisation. *European Journal of Operational Research*, 217(3) pp. 633-642.

Thomson R and Richardson D, 1999, The 'Good Continuation' Principle of Perceptual Organization applied to the Generalization of Road Networks. *17th International Cartographic Conference*. Ottawa, Canada.

Touya G, Duchêne C and Ruas A, 2010, Collaborative generalisation: formalisation of generalisation knowledge to orchestrate different cartographic generalisation processes. In *Proceedings of the 6th International Conference on Geographic Information Science*. Zurich, Switzerland: Springer-Verlag.

Touya G, Balley S, Duchêne C, Jaara K, Regnauld N and Gould N, 2012, Towards an Ontology of Spatial Relations and Relational Constraints. In *15th Workshop of the ICA commission on Generalisation and Multiple Representation*. Istanbul, Turkey, 13th-15th September 2012

Trafford Council, 2012, *DataGM*. [Online] [Accessed on 10th March 2013] http://datagm.org.uk/

Zhou S and Jones C, 2003, A Multi-representation Spatial Data Model. In Hadzilacos, T, Manolopoulos, Y, Roddick, J and Theodoridis, Y (eds), *Advances in Spatial and Temporal Databases*. Vol. 2750. Berlin / Heidelberg: Springer, pp. 394-411.