A Model for the Representation of Evolving Road Features

Alex Lohfink
University of Glamorgan, Pontypridd, CF37 1DL
+44(0)1443 480480 ext. 3616
alohfink@glam.ac.uk

February 15, 2007

1 Introduction

The [1]Ordnance Survey MasterMap Integrated Transport Network (ITN) layer provides a topologically structured representation of the UK's driveable roads. This representation is continually updated, and as such can be regarded as composed of evolving road features. Road features in ITN data are composed of links and nodes, and changes to these components represent a change in the Road feature. These changes, however, are not adequately represented in the data, meaning that only the current state of the road network is recorded, and that past versions are inaccessible. It would be a significant benefit to a system utilising ITN data to be able to store and retrieve past versions of the road network, for example in accident analysis, or to project future or alternative representations to facilitate traffic management. To enable the development of such a system, a mechanism for the versioning of spatial objects is necessary, along with an understanding of methods of spatial data modelling that include a time dimension, known as spatiotemporal data modelling. Methods of object versioning have been developed most notably in Computer Aided Design (CAD), where compound design objects are constructed from versioned component objects. This translates well to ITN data where compound Road feature objects are constructed from component link and node objects, meaning that

[1]This project is supported by EPSRC and Ordnance Survey

techniques for versioning design objects should be applicable to Road feature objects.

This research aims to develop and implement a system that can represent evolving road features through the techniques of spatiotemporal data modelling and object versioning. So far these areas have been investigated and applied to devise the Static Feature Histories model, and future work will implement the data model and develop an efficient query model using JDBC strongly typed interfaces to allow the manipulation of versioned road features.

(This research is sponsored by Ordnance Survey, and as such is constrained to use ITN data and Oracle Spatial).

2 Previous approaches to versioning and evolving features

2.1 Object versioning mechanisms

The most common versioning model is that of the representation of a component as a generic object, with all revisions of this object being its derivations. The generic object also stores references to all its versions, and references to specific versions are then resolved at run-time by a process called Dynamic Reference Resolution (DRR). If the most recent version is copied or modified to create a new version, then a linear Version Derivation Graph results (VDG). If any other version is modified, a branched VDG results. Miles et al. (2000) present a versioning approach called Describe, which employs this model. VDGs can also be used for composite objects, but this can be problematical. Instead an ordered list can be employed to record the histories of composite object versions [Goonetillake (2004)].

2.2 Previous implementations

There have been several previous attempts to represent evolving features. Spatiotemporal GIS implemented for ArcInfo [Candy (1995); Raza et al. (1996)] record historical data, and systems to represent changes to historical boundaries [Ott and Swiaczny (1998); Winnige (2000); Gregory (2002)] have been developed using an underlying relational data model. The Tripod system [Griffiths et al. (2001)] is built on an object-oriented (OO) data model, represents spatial data using abstract data types (ADTs), and implements object behaviour in C++. However, Tripod's underlying database system was bespoke, meaning that it would be problematical to apply it to another system, and its development did not continue. Nevertheless, the OO model has been shown to be much more successful in representing complex objects that represent more real world entities, and spatial objects fall into this category.

3 Structure and limitations of ITN data

The basic unit of the Road feature is the Roadlink. Roadlinks are comprised of a polyline geometry with a Roadnode object at each end. A Road feature is an aggregate of Roadlink objects. The Road feature does not contain the road's geometry, only a reference to its constituent Roadlink objects. Roadlinks, Roadnodes, and Roads all have the attribute 'change history', which is a collection data type. Each element of this collection has a date and description, which is either 'new' or 'modified'. No other information on the changes are recorded.

Although the ITN data contains spatial objects, it remains essentially relational in nature, meaning that aggregation of Roadlinks is represented by relational joins, and Roadnode objects relating to Roadlink objects are similarly referenced in another join table. The version information provided means that the frequency of change can be seen, but previous versions of features cannot be retrieved from the data.

4 Methodolgy

4.1 The benefits of the object-relational model

Oracle Spatial provides important benefits when devising a spatiotemporal model. Firstly, Oracle Spatial is object-relational (O-R), and as such supports Abstract Data Types (ADTs). Spatial ADTs can have spatial and aspatial data associated with them, and as such allow geographic features, in this case roads, to be abstracted independently [Rigaux et al. (2002), Voisard and David (2002)]. ADTs also offer the ability to extend spatial objects in the form of User Defined Types (UDTs) to include version information and time-dependent attributes, which are important in the representation of evolving features. Further, Oracle's data types include collections, meaning that attribute histories can be recorded by combining a time element with a data value or object. The O-R model also means that aggregation and other associations can be modelled using pointers (REFs) instead of the more costly (in performance terms) relational joins.

4.2 The Static Feature Histories model

Although most versioning models employ generic types and dynamic referencing, this is not ideal for ITN data. The generic types mean that there are levels of indirection in the model, introducing additional complexity when it comes to specifying a feature. If we examine the accepted benefits of dynamic referencing in the CAD environment, and view them as they apply to ITN data, then these benefits are questionable. The two main benefits are the elimination of version percolation, and the facility to freely combine components to create new configurations [Goonetillake (2004)]. Version percollation applies when many configurations exist that contain particular components, meaning that a change to the components has a cascading effect on updates in the configuration version hierarchy. In a system where updates are relatively infrequent and the number of configurations is restricted, as is the case for ITN data, then this is not an issue. Further, the restriction on configurations means that we do not need the facility to freely combine components. Therefore, a model based on generic types has no particular advantages and one major disadvantage - layers of indirection.

For this reason, we have devised the Static Feature Histories model. This model distinguishes between the invariant attributes of the objects, their version attributes, and their spatial attributes. Invariant attributes are contained in the generic object [Ahmed and Navathe (1991)], while specific version information is stored in a version descriptor [Chou and Kim (1988)]. A road version is thus comprised of static references to a generic object, a version descriptor, and to its component link versions. Figure 1 shows the structure of the static references for road, link, and node objects. Previous versions of a feature are recorded in a histories object, which contains the version information for all versions of the feature (number of versions, next version number, default version) and a history attribute, which is a collection of version descriptors.

The benefits of the model are:

- no indirection in resolving specific versions
- efficient retrieval of a spatial feature for a given time or time interval
- invariant attributes are not copied when a new version is derived

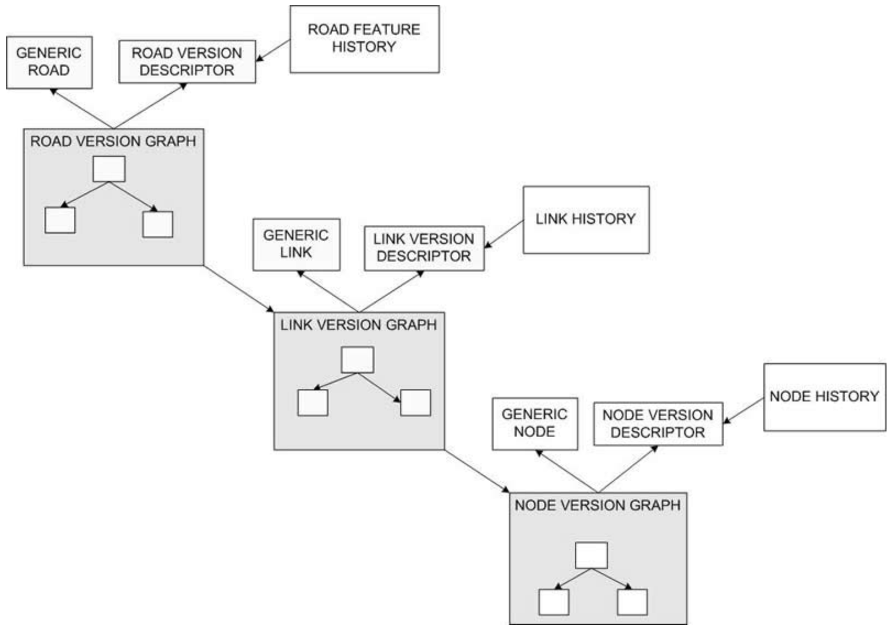

Figure 1: Static feature histories

One disadvantage is that the model relies on storing all versions exhaustively. However, storage hardware is now capacious and inexpensive, the degree of versioning required is relatively low, and ITN data is lightweight (in storage terms), and therefore this is not considered to be a significant drawback.

5 Conclusions and future work

Feature evolution involves the asynchronous evolution of numerous spatial objects and their attributes, and therefore to effectively represent feature evolution we must version-enable both spatial objects, their attributes, and features, unifying all within a temporal context. Currently, ITN data cannot provide adequate representation of evolving road features, and Oracle Spatial provides an object-relational framework within which to design and implement a spatiotemporal data model to version-enable ITN data. Future work will implement the Static Feature Histories model and develop a query model to manipulate versioned road features.

Biography

The author is currently a second year PhD student at the University of Glamorgan, under the supervision of Dr Nathan Thomas, Dr Tom Carnduff, and Dr Mark Ware. PhD topic: The Management and Representation of Evolving Features in Geospatial Databases.

References

Ahmed, R. and Navathe, S. B. (1991), Version management of composite objects in cad databases, *in* 'SIGMOD '91: Proceedings of the 1991 ACM SIGMOD international conference on Management of data', ACM Press, New York, NY, USA, pp. 218–227.

Candy, J. (1995), A prototype vector temporal gis, Master's thesis.
URL: *http://giswww1.bcit.ca/jcandy/thesis/*

Chou, H.-T. and Kim, W. (1988), Versions and change notification in an object-oriented database system, *in* 'DAC '88: Proceedings of the 25th ACM/IEEE conference on Design automation', IEEE Computer Society Press, Los Alamitos, CA, USA, pp. 275–281.

Goonetillake, J. (2004), The Management of Evolving Constraints and Integrity Validation in a Computerised Engineering Design System, Phd, University of Wales Institute, Cardiff.

Gregory, I. N. (2002), 'Time-variant gis databases of changing historical administrative boundaries: A european comparison', *Transaction in GIS* **6(2)**, 161–178.

Griffiths, T., Fernandes, A. A. A., Paton, N. W., Huang, B., Worboys, M., Johnson, C., Mason, K. T. and Stell, J. (2001), Tripod: a comprehensive system for the management of spatial and aspatial historical objects, *in* 'GIS '01: Proceedings of the 9th ACM international symposium on Advances in geographic information systems', ACM Press, New York, NY, USA, pp. 118–123.

Miles, J., Gray, W., Carnduff, T., Santoyridis, I. and Faulconbridge, A. (2000), 'Versioning and configuration management in design using cad and complex wrapped objects', *Artinficial Intelligence in Engineering* **14**, 249 – 260.

Ott, T. and Swiaczny, F. (1998), 'The analysis of cultural landscape change.', *History and Computing* **10**, 37–49.

Raza, A., Kainz, W. and Sliuzas, R. (1996), 'Design and implementation of a temporal gis for monitoring land use change: A case study of dar-es-salaam'.
URL: *citeseer.ist.psu.edu/raza96design.html*

Rigaux, P., Scholl, M. and Voisard, A. (2002), *Spatial Databases with Application to GIS*, Morgan Kaufmann.

Voisard, A. and David, B. (2002), 'A database perspective on geospatial data modeling', *Knowledge and Data Engineering, IEEE Transactions on* **14**(2), 226–243. TY - JOUR.

Winnige, N. (2000), 'Historical atlas of prussia', *WWW document* . Accessed 14th April 2006.
URL: *http://www.geog.port.ac.uk/hist-bound/*