

GML for exchanging topographic data

Wilko Quak, Marian de Vries, Theo Tijssen, Jantien Stoter and Peter van Oosterom

Section GIS Technology

Department of Geodesy, Faculty of Civil Engineering and Geosciences

Delft University of Technology, Thijsseweg 11, 2629 JA Delft

the Netherlands

quak@geo.tudelft.nl

Introduction

The Dutch Topographic Service (TDN) is supplier of the TOP10vector, a digital vector file with topographical information of the Netherlands territory at a scale of 1:10,000. Since the creation of the first digital topographic data set by TDN there has been a need for exchange formats. This has always been a difficult issue and point of discussion. At the same time, users expressed additional requirements.

Most current topographic data products are a mixture of geographic data and cartographic presentation. In the design for a renewed product in the Netherlands, a clear distinction is made between a Digital Landscape Model (DLM) and a Digital Cartographic Model (DCM). The focus is on the design of the DLM meeting current and future user requirements. In parallel the exchange problem is tackled by using the new OpenGIS standard, Geography Markup Language (GML), which will be supported in future GISs.

At the moment the data in TOP10vector is maintained as polygons, lines, points, and text-features. The current TOP10vector is not maintained in a relational or object-oriented database, but in (Microstation design) files. Apart from this, other limitations exist, such as no unique object-ids. The only attribute information available is a feature code (road, building, pasture, river, etc.). User organizations in the Netherlands have asked the TDN to re-engineer their topographic data into a more object-oriented data model. For this purpose, a new data model is defined for the TOP10vector that meets these requirements. Important characteristics of the new conceptual model are: unique object-IDs, a partitioning of the surface as the basis for geometry (exceptions occur in case of overlap, e.g. road segments in tunnels or road segments on bridges), 2.5D objects with 3D coordinates, possibility of complex features (an aggregation of road segments into one - or more - 'named' roads) and the incorporation of metadata and temporal data for each object instance (versioning). The last characteristic opens the way for 'change only' updates distributed to user organisations (see also Badard and Richard, 2001).

Since there is also a growing demand to distribute the data in a more open transfer format, a prototype of the new data model is implemented using GML 2.0. GML 2.0 is accepted as recommendation by the OpenGIS Consortium in February 2001 (OpenGIS Consortium, 2001). The rationale behind the choice of GML is the fact that it is based on the world-wide accepted XML standard and that a rapidly increasing number of tools is available to generate, check and interpret XML/GML (van Oosterom, 2001). The article by Reichardt (2001) gives a short overview of GML 2.0 and motivation why one should use this.

The advantages of XML in general are that it is well readable by humans and machines (in contrast to binary formats), international (support of Unicode for non-western languages), method to convert XML documents is available (e.g. develop an XML Style Sheet Transformation, XSLT to convert a DLM into a DCM), extensible with own parts (using the 'XML Schema' language, (W3C, 2000)), and is very well supported by all kinds of software in the market (ranging from browsers to DBMSs). With respect to the specific spatial data, ISO (1999) and OpenGIS (1998) have harmonised their models and GML 2.0 is based on this.

The project 'Object Orientation TOP10vector' is carried out by the Section GIS Technology of Delft University of Technology (TU Delft) in co-operation with the Centre for Geo Information of Wageningen University (CGI), the International Institute for Aerospace Survey and Earth Sciences Enschede (ITC) and the Dutch Topographic Service (TDN). The Section GIS technology is responsible for an XML Schema definition as implementation of the new model and some GML prototypes with real-world data.

The new conceptual model, which was developed by ITC (in UML) has been converted into a technical model, also described in UML class diagrams and based on the OpenGIS Geography Markup Language (GML) (Open GIS Consortium, 2001) based on OpenGIS Simple Feature specification (Open GIS Consortium, 1999). This specification contains standards to represent geo-spatial features and geo-spatial feature collections. The conversion has been done manually. Research is going on to perform automatic conversions from UML to GML (Gronmo, 2001).

The developed technical data model as well as the data conversion process is described in this article. Besides the formal checking of the GML Top10vector prototype against the XML schemas, another method for checking is really using the GML data. For this reason visualisation efforts are also described.

This article is concluded with an overview of the main results and limitations of the TOP10vector GML prototype. It also gives an indication for future work

Data modelling

In order to make a GML prototype from the conceptual model, many engineering decisions have to be taken. The first step is to make a technical UML model of the conceptual UML model of the ITC (Knippers and Kraak, 2001). In this project, the Department of Geodesy created two alternative technical models. These are modeled after two models in the GML specification: one model with and one model without member restrictions. The details and rationale behind these two alternatives model will be described at the end of this section.

The main guidelines during the creation of the technical model have been:

1. the conceptual model
2. the data sets available (or 'creatable')
3. XML/GML principles
4. elegant and readable

In addition to the conceptual UML model of the new Top10vector, the required structure of a GML model (Open GIS Consortium, 2001) has quite an impact on the final GML prototype of the new Top10vector. GML is described by two XML Schema's: the geometry schema and the feature schema. As we will see in this chapter, the way GML organizes features in collections has a serious impact on the overall technical model (both UML and XML schema) of the Top10vector prototype.

The UML of the conceptual model is given per main feature category. That is there are (independent) UML schema's for road ('weg'), railroad ('spoorbaan'), water ('water'), building ('bebouwing'), terrain ('terrein'), providing element ('inrichtingselement'), and regions ('gebieden'). All these classes inherit from a generic class geographic object ('geografisch object') which has an id and two time attributes and a relationship to metadata.

The technical model integrates the different UML schema's in one overall schema. In this process, additional (inheritance) structure is created: road, railroad and water are all derived from the more general class infrastructure ('infrastructuur'), see Figure 1. The reason for this is that these classes have many attributes in common. In order to avoid repeating these all the time in the model (kind of redundancy, which may be difficult to maintain when changing the model), it was decided to introduce this additional class infrastructure. This is only visible in the (inheritance) structure of the classes and cannot be observed in the final instances (that is the GML prototype data).

The conceptual model is at class level only. The instances are from these classes. In addition, the technical model also recognises explicitly the sets of instances, which form the explicit collections of the classes. Five 'set' classes are modelled in the technical UML schema's:

1. spatial objects ('ruimtelijke objecten', which contain instances from the classes infrastructure, terrain and buildings)
2. providing elements ('inrichtings elementen', which contain instances from the class providing element; note the subtle difference between plural for the set and singular for the individual element)
3. functional regions ('functionele gebieden')
4. administrative regions ('administratieve gebieden')
5. geographic regions ('geografische gebieden').

Finally, there is one set of sets in the model and this is called Top10 themes ('Top10 themas'), which has as its elements instances. These instances are from the class Top10 theme ('Top10 thema'). Specialisations of these abstract classes are the basis set classes as described above: spatial objects, providing elements, functional regions, administrative regions and geographic regions. The reason for explicitly modelling is that an XML document can have only one 'root' element. This corresponds to the 'set of sets', that is an instances of the class Top10 themes. An alternative would be creating several GML documents, one for each collection with instances from a specific specialisation class (that is 5 in total), for one data set. This was considered an inferior solution.

The term 'Geografish object' from the conceptual model is replaced by the term 'Top10Object', but plays the same role. There are a few minor changes with respect to the conceptual model. The Top10Object now inherits from three classes: the abstract feature (GML basic class for a feature with geometry), metadata object (contains everything related to meta data) and temporal object ('temporeel object', which contains the temporal aspects).

restrictions) and the 'strict' technical UML model, is that the latter introduces restrictions to the values of the attributes (instead of allowing all character strings).

Data conversion

The process to create a GML prototype starts with the creation of a sample data set by the Dutch Topographic Service in accordance with the new conceptual model (see Figure 2).

In the next step this data is loaded into an Oracle 8i database at TU Delft. Then database views are created to model the data after the UML schemas. The data is retrieved from the database and converted into GML. This is done by means of a Java program that makes a JDBC connection to the database. Then it reads the data via SQL queries and writes the result to a GML file. The final step is to validate the generated GML file against the schema definition. Various standard XML tools can be used for this purpose. We used XML Spy and Turbo XML to check whether the files generated are 'well formed' and 'valid'.

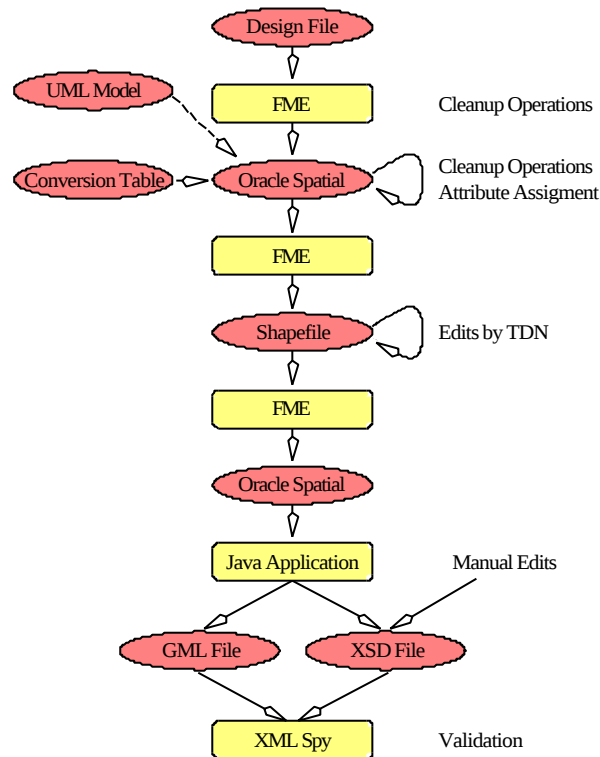


Figure 2: Architecture of conversion process

GML viewing

One of the basic principles of the conceptual model behind the GML specification is the separation of content and presentation. This of course fits one of the design principles of the Top10vector project: the distinction between the Digital Landscape Model and the Digital Cartographic Model. Consequently, the GML prototype files do not contain styling information (color, fill, line width, cartographic symbols) for the feature types. (note: It seems the Ordnance Survey did make another design decision for their Digital National Framework GML products: their GML files also contain cartographic features, beside the topographic base feature types.) There are several ways to add style to the GML features:

- when the GML files are imported in / converted into 'normal' GIS or CAD application software, color and style can be applied after the conversion process, with the standard tools that belong to the software;
- when the GML files are meant to be viewed without conversion into the (native) format of existing GIS or CAD software, there are again several possibilities:
 - a. the organization that provides the GML files also provides one or more files with default styling information. This could itself be an XML file (for example based on the OGC Styled Layer Descriptor recommendation [OGC 2001]), a cascading stylesheet (.css) file (for direct

- use in Internet browsers) or another kind of parameter file. The viewer software reads both the GML (content) data and the styling parameters and generates the cartographic model.
- the viewer software has an interactive module by which users can select the appropriate color, fill and line width.
 - a combination of a. and b.

Because GML is still very new, there are currently not many viewers that can display GML. At the Geodesy department a small simple viewer for spatial data is available. This viewer that is completely written in Java and can easily be extended. This is done by writing a new 'Loader' module for this new type of data. As a part of his Master's project (Monteiro, 2001), a student of the Section GIS technology wrote a loader for GML data. The GML Loader makes use of the XML Parser Xerces (Apache, 2001). Note that this loader is not a general GML viewer. It can currently only view GML files with a very specific structure. Note that this viewer was developed as a side product of a Master's project and is in no way a generic GML viewer. The speed with which the application was developed however does show that already very useful (free) software support is available on the internet. In Figure 3 a screenshot of this viewer is shown.



Figure 3: GML viewer

Conclusions

With the project described in this article, the first steps have been taken to improve the (model for) topographic data in the Netherlands (Van Asperen, 2000; Bakker and Kolk, 2001).

The Topographic Service in the Netherlands has enriched a sample data set, which was converted into a GML (2.0) prototype by TU Delft. This prototype cannot cover certain aspects in a standard manner, because these are not yet part of GML 2.0. So, they are currently either a part of the application part of the XML schema (such as the temporal attributes at this moment) or are excluded from the prototype (such as topology structure support for a planar partition).

The drawback of putting 'knowledge' (functionality) in an application specific schema and not using a standard schema is that one can not expect implementations of the standard 'understand' the semantics of application schema. The best one can hope for is that the implementation shows the values of the attributes and relationships correctly. It is too much to ask for, e.g. in example of the temporal attributes, that any GIS edit software understands how to use and fill the application specific attributes begin date ('begindatum') and end date ('einddatum'). However, once part of the standard, one can expect correct semantical (functional) support for this.

The drawback of not having topology structure in the GML prototypes of the Top10vector is obvious: there is quite a lot of data redundancy and it is now very difficult to make sure there are no unwanted gaps or overlaps between the different features within an intended partition. However, it was explicitly decided to keep the model simple (by the TDN) and not to introduce topology structures support in the application schemas. It is interesting to note the Ordnance Survey in Britain did choose to put topology structures in their application

schemas (Ordnance Survey, 2001). However, once the next version of GML does support topology structures, they will have to convert their schema to the standard GML (if they want to be able to use the full benefits of the standard).

In order to prove the concept of interoperability, the Netherlands Society for Earth Observation and Geoinformatics (KvAG) organized a GML-relay in June 2001. As could be concluded from this relay, the support of the GML 2.0 standard in commercial products was somewhat disappointing and in contrast with the activities of the major Geo-ICT vendors within the OpenGIS consortium and ISO TC211. The used GML test data set, a predecessor of the TOP10vector GML prototype in this report, was described in (De Vries et al., 2001). However, one has to realise that it will take a certain time before a standard is implemented in a certain product. Therefore, the KvAG decided to organise another GML-relay in December 2002.

The support of GML by data producers looks very promising. One proof of this was the OEEPE Workshop on XML/GML in November (19-20 at Marne-la-Vallee, Nr Paris, hosted by IGN France). Besides the Ordnance Survey (Ordnance Survey, 2001) and, of course, the Dutch Topographic Service, other geo-data providers are starting to use GML, for example US Census Bureau (Daisey, 2001), Northrhine-Westphalia (Brox, et al., 2002; Riecken, 2001) and again in the Netherlands the Dutch Cadastre. Further, GML is used in many web map/feature server environments created or now being created all over the world.

The OpenGIS consortium has started GML 3.0 developments with work items such as Topology, Temporal, Geometry Extensions, Units of Measure, Spatial Locator, Meta-data Mechanisms, Default Styling and Points of Interest/Areas of Interest. It is obvious that standardisation in these areas will further improve true interoperability. Very important, but not without difficulty and discussion, is the requirement to keep GML in harmony with the ISO TC211 standards. The work item 'Default Styling' may form a bridge between the DLM and DCM.

References

- Apache Software Foundation, <http://xml.apache.org/xerces2-j/index.html>. *Xerces2 Java Parser*
- Badard, T. and D. Richard. Using XML for the exchange of updating information between geographical information systems. *Computers, Environment and Urban Systems*, 25(1-5):17-31, 2001.
- Bakker, N. and Bert Kolk. A new generation TOP10vectordata in the netherlands. In *proceedings International Cartographic Conference ICA*, August 2001.
- Brox, C., Yaser Bishr, Kristian Senkler, Katharina Zens, and Werner Kuhn. Toward a geospatial data infrastructure for northrhine-westphalia. *Computers, Environment and Urban Systems*, 26(1):19-37, 2002.
- Buehler, K. and L. McKee. The opengis guide -introduction to interoperable geoprocessing. Technical Report Third edition, The Open GIS Consortium, Inc., June 1998.
- Carlson, D. Modeling XML Applications with UML. Addison Wesley, 2001.
- Daisey, P.. Implementing GML schema for us census tiger / line files. In *EOGEO 20001/Digital Earth Congress*, Fredericton, New Brunswick, Canada, June 2001.
- De Vries, M., Wilko Quak, Theo Tijssen, Jantien Stoter, and Peter van Oosterom. Topographic data, object orientation and GML. In *EOGEO 20001/Digital Earth Congress*, Fredericton, New Brunswick, Canada, June 2001.
- Gronmo, R. Supporting GI standards with a model-driven architecture. In *Proceedings of the 9th ACM international symposium on advances in Geographic Information Systems*, Atlanta, USA, November 9-10 2001.
- ISO TC 211/WG 2. Geographic information - Spatial schema. Technical Report second draft of ISO 19107 (15046-7), International Organization for Standardization, November 1999.
- Knippers, R. and Menno-Jan Kraak. Objectgerichte beschrijving TOP10vector - concept ontwerp gegevensmodel, versie 1.0. Technical report, ITC, Mei 2001. (in Dutch).
- Monteiro, F. GML and complex features. Master's thesis, Delft University of Technology, August 2001.
- Open GIS Consortium, Inc. OpenGIS Simple Features Specification for SQL. Technical Report Revision 1.1, OGC, May 1999. GIS Technology GML prototype
- Open GIS Consortium, Inc. Open GIS specification - geography markup language (GML). Technical Report version 2.0 (01-029), OGC, March 2001.
- Ordnance Survey. DNF data in GML - DNF release 1 product data: a description of how DNF data is represented in the geography markup language. Technical report, OS, May 2001. version 1.0.
- Ordnance Survey. DNF geometry and topology - DNF release 1 product data: specification of the geometric data types used in the DNF data product, and other geometry and topology information. Technical report, OS, May 2001. version 1.0.

- Ordnance Survey. Life cycles of DNF features - DNF release 1 product data: the life cycles rules applied to topographic features in the maintenance of the DNF data by ordnance survey. Technical report, OS, May 2001. version 1.0.
- Reichardt, M. OGC's GML 2.0 - a new wave of open geoprocessing on the web. *GeoInformatics*, 4:18, 21, July/August 2001.
- Riecken, J.. The improvement of the access to public geospatial data of cadastral and surveying and mapping as a part of the development of a nsdi in northrhine- westphalia, germany. In *Proceedings of the 4rd AGILE Conference on Geographic Information Science*, Brno, Czech Republic, pages 215-221, April 2001.
- Van Asperen, P.. Objectgerichte beschrijving TOP10vector - concept ontwerp gegevensmodel, versie 1.0. Technical report, Topografische Dienst Nederland, Augustus 2000. (version 4.0, in Dutch).
- Van Oosterom, P.. Opengis technologie als basis voor de nieuwe tdn datastructuur. In Nederlandse Vereniging voor Kartografie (NVK) studiemiddag, Alterra, Wageningen, April 2001. (in Dutch).
- W3C. Xml schema part 1: Structures and xml schema part. Technical report, World Wide Web consortium, October 2000. Candidate Recommendation.
- W3C. Xml schema part 2: Datatypes. Technical report, World Wide Web consortium, October 2000. Candidate Recommendation.