BUILDING A HARMONIZED BASE MODEL FOR GEO-INFORMATION IN THE NETHERLANDS

WILKO QUAK and Marian de Vries

Delft University of Technology
OTB Research Institute for Housing, Urban and Mobility Studies
Section GIS Technology
The Netherlands

ABSTRACT

In the Netherlands a base model for Geo-Information (NEN3610) has recently been published. This base model is designed as a starting point for spatial data providers in the Netherlands to create sector models. In a sector model a data providers models its own data according to the rules described in the base model. Because all sectors now share a base, interoperability between the sectors is greatly enhanced. This paper describes some design issues behind the model and describes the construction of the first sector-model TOP10NL.

The key features of the base model are: It is an object-oriented model and is fully described in the Unified Modelling Language (UML); it is based on international standards (the ISO/TC 211 model); it describes nationwide unique identifiers; it defines a mechanism to track changes of objects through time; a GML exchange model for the data can be automatically derived from the model; and the model is meant to be extended by sectors to fit the needs of that sector.

We discuss two mechanisms for extending the base model to create a sector model: Extension by subclass and extension by pattern. The pro's and cons of the different mechanisms are discussed.

INTRODUCTION

Within the Netherlands there are many public and private parties that exchange geo-information. More and more often the data is automatically processed, transformed or integrated in complex chains of data processing. Processing spatial data in such a way can only be done when the participants agree on both the semantics and the structure of the data that is handled. In order to achieve the interoperability various providers of spatial data in the Netherlands agreed to make a harmonized base model for geo-information in the Netherlands. This model was registered at the Dutch organization for standardization as NEN3610:2005 (2005). The model consists of a collection of base classes that are described in the UML modelling language. In UML these classes have a name and a collection of attributes that describe properties of the base object on which all parties agree. Data providers that wish to use the model can extend the classes in the base model with their own attributes as long as their extensions do not contradict with the definitions in the base model. Such an extension is called a sector model. In

Figure 1 an overview of how the sector models relate to the national and international models is shown.

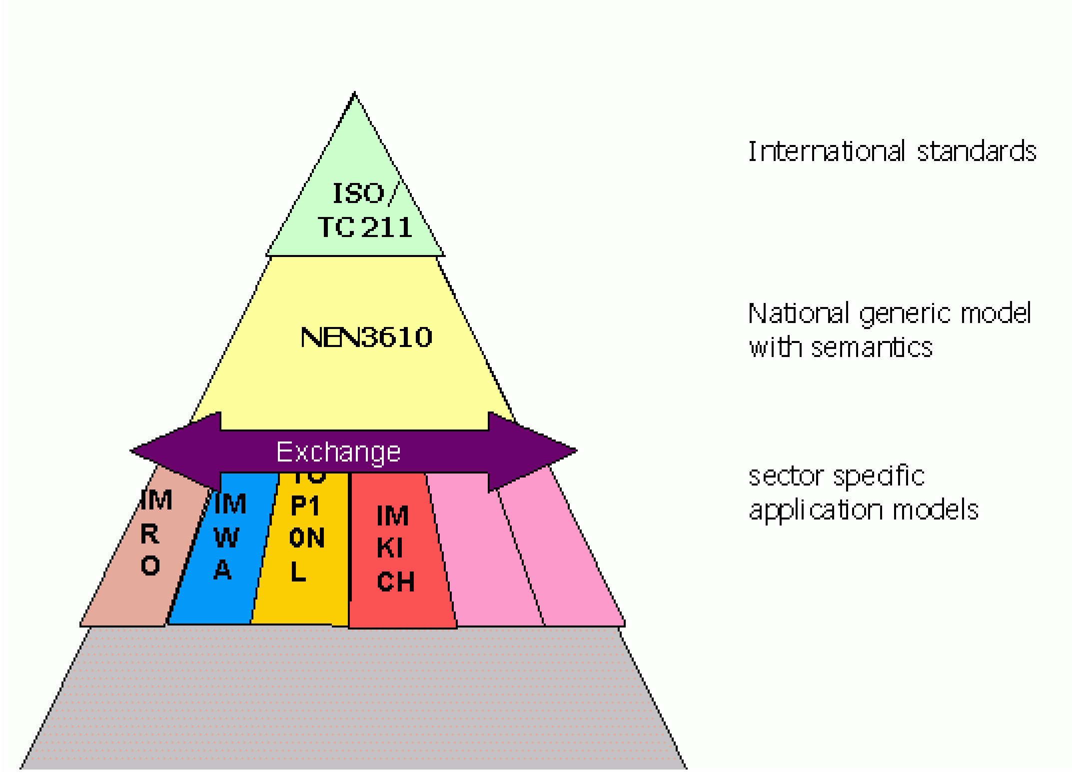

Figure 1: The NEN3610 base model for Geo-Information in relation with international standards and local implementations

In this paper we describe the complete process of making the harmonized base model and generating the first dataset according to this harmonized model together with lessons we learned during the process. This paper is structured as follows: First we give an overview of the main design issues of the NEN3610 model. Then we describe two different ways of how the base model can be extended to create a sector model; this is followed by an example of the first sector model (TOP10NL). Finally we discuss the advantages and disadvantages of the two different extension mechanisms that were detected in the process of making the TOP10NL sector model and give some conclusions.

NEN3610 – A Base Model for the Netherlands

The development of the NEN3610 base model started in 2003 when it was decided that the previous base model (dating from 1992) was due for a revision. In the revision process done by a small project group shortcomings of the old standard were overcome and new developments in geo-information were taken into account. A focus group of users of geo-information was formed to review the work of the project group. The main characteristics of the resulting model are given below:

The model adheres to international standards

One of the reasons for the renewal of the old base model was that the old model was mainly based on national standards. As a result exchange of the data outside the Netherlands was complicated. To overcome this the new model needed to link to international standards. Therefore the ISO/TC 211 standards for Geographic Information was used as a basis for our national model.

The model is fully described in the UML modeling language

Inside the complete project UML class diagrams are used to describe the data model. UML *(ISO/IEC 19501:2005)* is a graphical modeling tool with well defined semantics and an underlying computer model. The graphical representation of the model gives human users pictures to talk about; most figures in this document are UML diagrams. The underlying computer model enables software to reason about the model thus enabling a model driven approach (MDA).

The model defines a base class and a hierarchy of sub-classes that can be extended by sectors

The base class of the NEN3610 model is called GeoFeature and in this class all characteristics that are shared by all features in the Netherlands are defined. It is depicted in Figure 2. As can be seen it defines several attributes. Apart from the attribute ‘identification’ (described below) all attributes are optional. This lowers the implementation burden for sectors that wish to implement the model. The other attributes in GeoFeature are all optional. This means that the attribute can be omitted in an implementation of the model. If an optional is implemented its semantics are defined in the base model.

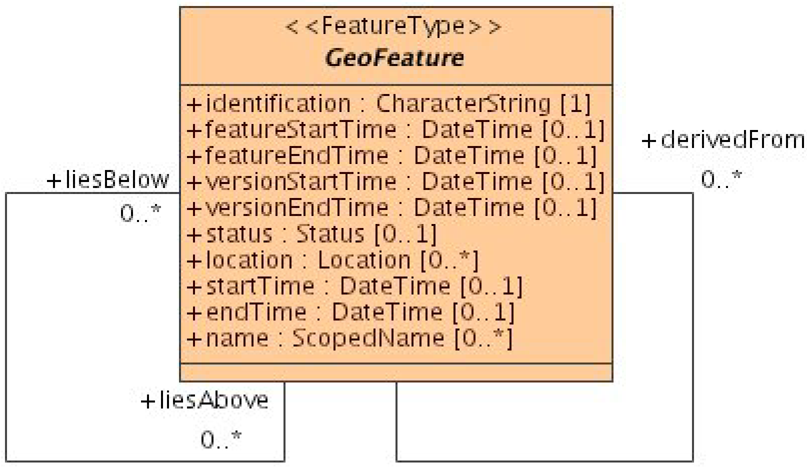

Figure 2: The base class of the NEN3610 Hierarchy.

This base class GeoFeature is the top of a class hierarchy. Below the top level a list of subclasses is defined that represent objects that are commonly seen in the Netherlands. The main subclasses in the hierarchy are depicted in Figure 3; in this figure details are omitted. These main classes are the points where the sectors can connect to the base model. If they wish to model an object the best fitting class in the hierarchy should be extended. In case no class in the base model matches the needs of the sector, the base class GeoFeature should be extended.

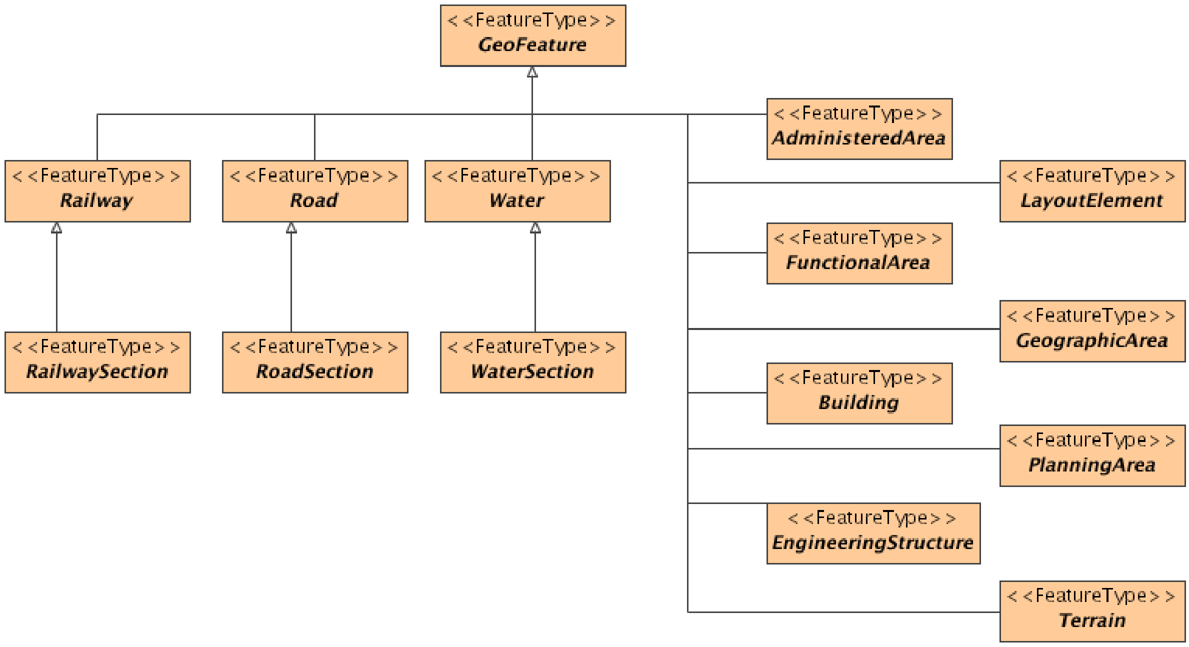

Figure 3: First level of the hierarchy of NEN3610 without details.

Every object has a unique nationwide identifier

In the base class of NEN3610 (GeoFeature) there is only one obligatory attribute. This is the identification attribute (see Figure 2). This string attribute should have a value that is unique across all sector models in the Netherlands. This uniqueness is achieved by the following procedure: Each identifier consists of three parts separated by colons: The first part is the country code “nl”. The second part is a unique sector identifier that will be provided by NEN3610. The third part is a sector specific identifier ensures unique identifiers within the sector. Together the three parts guarantee a nationwide unique number, nationwide unique number. An example of a unique identifier is: “nl.top10nl.road.4d696e6b65”.

An important issue when working with unique identifiers of objects is lifecycle information: How much can an object change while keeping the same identification. The amount of change that an object can endure without changing its identity is sector dependent.

If only an unimportant attribute of an object changes it can keep the same identification; there will a new *version* of the same object (see temporal modeling). How much an object can change while keeping its identity is left to the sector models. The only rule is that an object cannot change type without changing identity. A change of an object that does not change its identification results in a new version of the object.

Every GeoFeature can be traced through time

In the NEN3610 model every object has several temporal attributes (see Figure 2). The different temporal attributes are used in different cases and have different semantics. The startTime and endTime attribute are used to track the Feature in the real world whereas the other temporal attributes ,featureStartTime, featureEndTime, versionStartTime and versionEndTime refer to the lifetime of the object inside the computer. The distinction between the two sets of attributes is very important when querying the data: when a policy maker wishes to know how forests were destroyed in a given year the first two temporal attributes are used. The other attributes would give answer to the question how many road were deleted from the computer in that year.

The other four temporal attribute are use to track the lifecyle of an object and can be used for change only updates of datasets.

Geographic objects in the model are created at some instant, change during their lifetime and later cease to exist within the computer. The lifetime of a geographic feature in the model is stored in the featureStartTime and featureEndTime attributes. Objects that are still valid, have no featureEndTime (i.e the value is NULL).
If a geographic feature is destroyed or changes to such an extent that it becomes a different geographic feature, the feature is not removed from the system, but the featureEndTime is set so that it know not to be valid anymore. If a geographic feature in the system is changed, a new (modified) version of the geographic feature is created. The same value is entered as the versionEndTime of the old version and the versionStartTime of the new version. At any time during a geographic feature's lifetime it has exactly one valid version.

An GML application schema can be automatically derived from the model

Apart from using UML for harmonizing the data model, the UML can also be used to derive a file format to exchange the data between the parties. The exchange format is defined by a GML application schema. This XML document can be automatically derived from the UML model using the rules in the ISO19136 document.

Enumerations of well defined phenomena

For many attributes in the model, a predefined list of possible values is provided as part of the model in the form of enumerations or CodeLists. Depending on the level of agreement between the different parties in the model the list is a closed enumeration (only the given values are allowed, this is depicted by the <<enumeration>> stereotype) or the list is list of suggested values, that can be extended or changed in sector-models (with the <<CodeList>> stereotype. See Figure 4 for an example of a CodeList for the construction material of buildings in NEN3610.

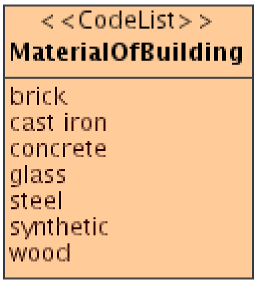

Figure 4: Enumeration of suggested values for the attribute MaterialOfBuilding.

Although a suggested list of values is not a powerful way of standardization the lists have proven to be very useful in the model as a starting point for the harmonization of the different sectors and as a base of well defined words.

MAKING A SECTOR MODEL INSIDE THE NEN3610 MODEL

The base model NEN3610 is only a template model. This means that no data can exist in the model; the model is a collection of definitions that need to be extended by a sector to become a data model in which data can be exchanged. Such a sector model is an extension of the base model. In a sector model all definitions and classes as defined in NEN3610 can be extended, refined or constricted in any way as long as the definitions in the sector model do not contradict with the base model. In this way any valid instance of an object in the sector model is also a valid according to the base model.

There are two different ways of extending a UML model to generate a sector model. Here we label these two ways as 'extension by subclass' and 'extension by pattern'. These two extension mechanisms result in different UML models and since the GML application schema's are automatically derived from the UML also the GML applications schema's are different. Below we first explain the two extension mechanisms, after that we discuss the advantages of the two mechanisms.

Extension by subclass

In extension by subclass a sector model is created inside a new UML package. In a UML diagram the package name is the name between brackets below the classname, for convenience we use a color schema where every package has a separate color). A sector that wishes to model a specific Feature Type chooses the Feature Type inside the base UML model and creates a subclass of that specific class inside their own package. In the case of NEN3610 this means that either one of the predefined classes should be used or if there is no applicable class, the base class GeoFeature should be extended. Within the subclass the following things can be changed in relation to its superclasses:

- Attributes and relationships can be added.
- Attributes and relationships from superclasses can be restricted. For example, an attribute that is optional in a superclass can become obligatory in the newly created subclass. Restrictions are only valid if an instantiation of the restricted object would also be a valid instantiation of the superclass. In this way optional attributes can be omitted in a subclass. On the other hand, omitting an attribute that is obligatory in a superclass is not allowed, since the instatiation of the object in the subclass (without the attribute) is not a valid instatiation of the superclass anymore.

Below we give an example of how the Building object of NEN3610 is extended with a subclass in TOP10NL. In the subclass the following things are done:

- The location attribute (that was optional in NEN3610) will never be used in TOP10NL, so it is declared with [0] cardinality.
- Various new attributes are introduced (e.g. heightClass, height, geometry, …).
- Some attributes that were optional are made obligatory (featureStartTime)

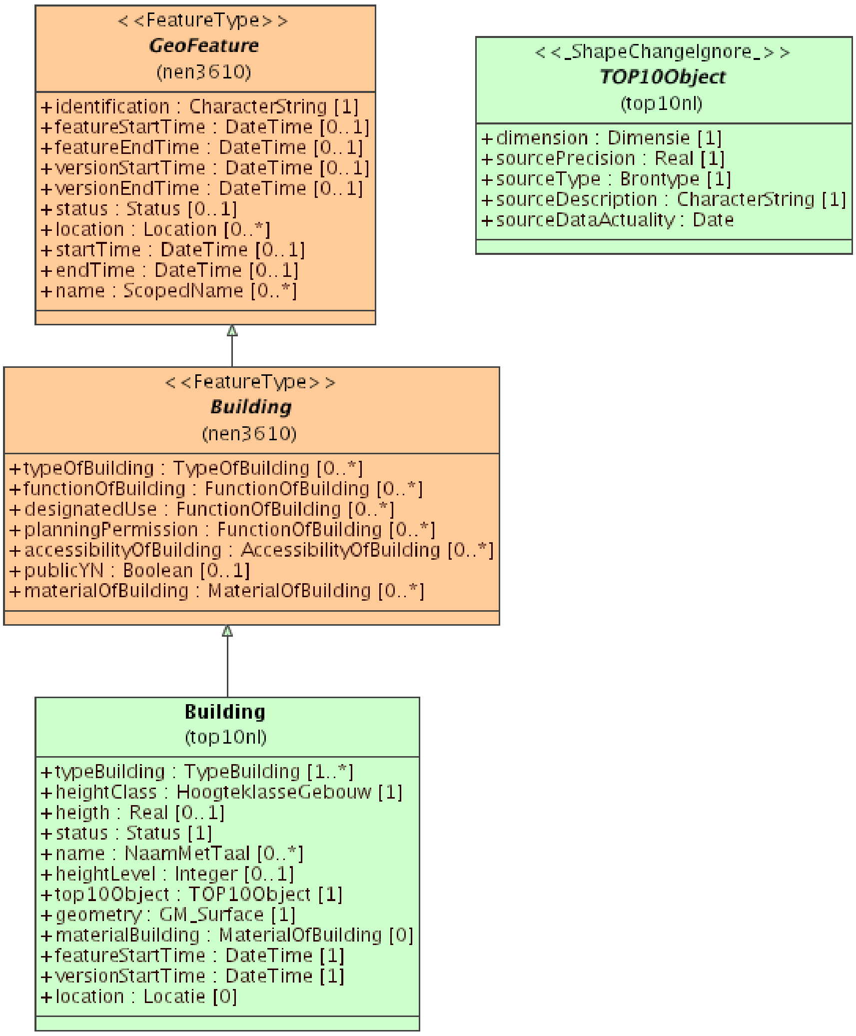

Figure 5: Extension of the NEN3610 Building base class by extension in TOP10NL.

Extension by pattern

In the second way of generating a sector model no subclasses of the base model are created, instead a copy of the model is made and the all classes in the base model are edited to fit the needs of the sector model. See Figure 6 for a UML fragment of the Building where the building class by pattern according to the NEN3610 model.

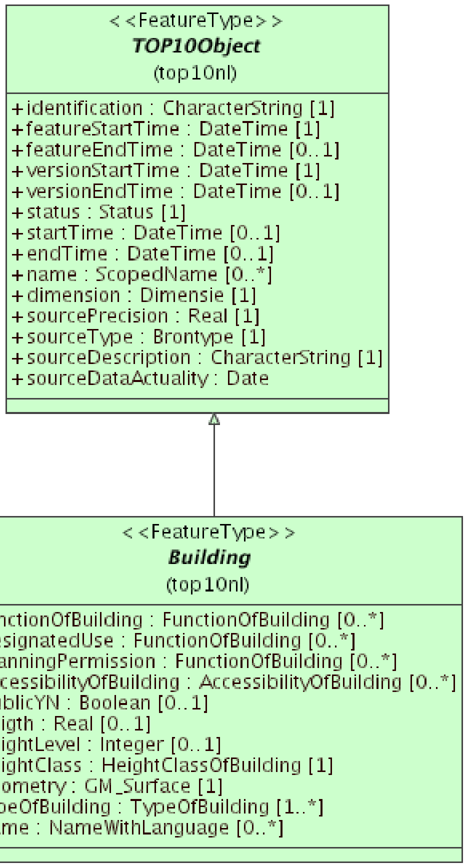

Figure 6: Extension of the NEN3610 Building base class by pattern.

Please note the following points:

- The NEN3610 base class. GeoFeature has been renamed TOP10Object and changed according to the needs of TOP10NL. Attributes that were optional in GeoFeature and are not used in TOP10NL are just deleted.
- The problem of solving multiple inheritance is solved by integrating the share attributes of the GeoFeature and the TOP10NL feature in the new base class.

TOP10NL – The First Sector Model within NEN3610

The first sector model that was generated inside the NEN3610 model is the TOP10NL model. This is a model of the scale 1:10 000 topographic data of the Dutch Cadastre. Just before the NEN3610 model was defined the Topographic Service had completed a data model for their data *(Bakker et al, 2005)*. Since the model was published at about the same time that the base model NEN3610 was developed, it also as a major source of input for the development of NEN3610. As a result, there are many similarities between the two models and the subsequent event of making TOP10NL a sector model of NEN3610 was quite straightforward.

The TOP10NL model was created using the extension by subclass paradigm and is depicted in Figure 7.

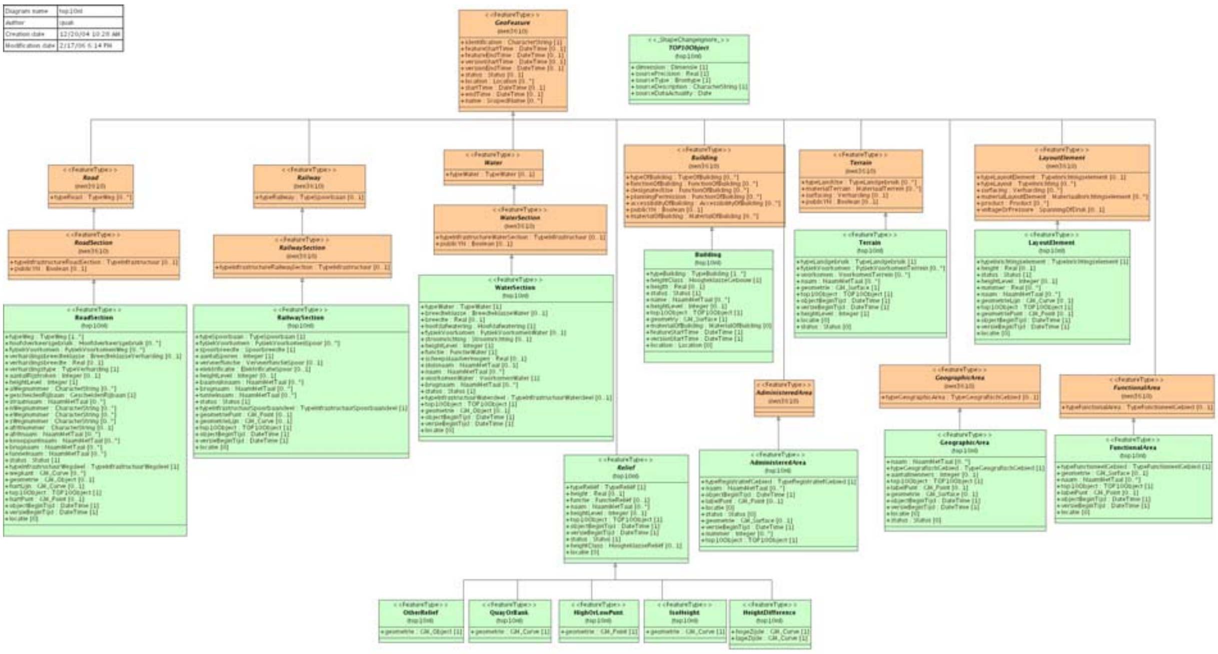

Figure 7: The complete TOP10NL model as an extension of NEN3610..

EXTENSION BY SUBCLASS VS EXTENSION BY PATTERN

An advantage of the extension by subclass is that there it is completely clear from the model which elements of the new class are extensions with respect to the base model and which elements from the new model are inherited.

The main disadvantage of this way of modeling is that because a copy of the original model, is edited the separation between the base model is destroyed: It is not clear anymore which part of the sector-model is sector-specific and which part is still the same as in the base model. When the base model changes it will be very hard to propagate the changes to the sector model.

GML Relay

An important source of information regarding the evaluation of the different extension mechanisms is the GML Relay (*GML Relay, 2006*). In this event a GML document based on the TOP10NL model (i.e. extension by subclass) was provided. This document was processed by various providers of GML software. Although the TOP10NL data was processed correctly by most GML providers, it turned out that many of the features of the model that were considered as complex by the participants turned out to be a result of the extension mechanism used. So the benefit of a modular model comes with the cost of a structure that is harder to handle.

More sector models in NEN3610 to come

TOP10NL is the first model developed as a sector model in NEN3610 since its publication in December 2005. Currently many other sectors are working on their sector model:

- IMRO: A model to exchange spatial plans between the government, the provinces and the communities with the different levels of detail that are needed for the different government agencies.
- IMKICH: A model for the historic objects found in the Netherlands. This model integrated 4 datasets of different organizations. (*IMKICH, 2004*)
- Parallel to the TOP10NL model the community of Amsterdam is modeling their 1:10000 topographic datasets. Although this set is very similar to the TOP10NL dataset, there are a few differences since the data is surveyed for a different purpose (city maintenance). It will be interesting to see how the two models will interoperate.
- The Dutch Cadastre is working to fit their model in NEN3610.

CONCLUSIONS AND FUTURE WORK

The process of making a base model for geo-information and the first sector models for the Netherlands has been an interesting process that has given many new insights in modelling spatial nationwide with the use of base models and sector models. Some of the conclusions are:

- Although creating sector models using the 'Extension by Subclass' results in a more modular model, in which a clear distinction between the sector and the general model still can be distinguished is also has some disadvantages: The UML models become a bit harder to interpret and generating a GML application schema from the model is very tricky.
- A base model to which many parties have contributed and feel responsible for is very valuable.

Now that there is momentum behind the development of a base model and sector-models it is time to look back and ahead. Currently in Europe the INSPIRE project (*INSPIRE, 2004*) is having goals on a European schale that are similar to the goals of NEN3610 for the Netherlands. It is a current topic of interest of how to cope more than one base model with possibly conflicting modelling constraints in each base model. A possible solution lies in creating mappings from one model to another. These mappings describe how the underlying data can be converted from one model to another without loss. This model mapping has parallels to the Query / Views / Transformations paradigm that is current in developement in Object Modeling (*OMG, 2002*). Another interesting topic that will come up now that there is a base model is how to cope with model changes: A change in the base model will have implications for the sector models that cannot follow the base model immediately. As a result there will be different versions of model in use at the same time. A good insight in what impact this has on interoperability needs to be studied before the first model change is there.

References

Bakker, N. and Bruns, B. and Storm, M. (2005): Gegevensmodel TOP10N, versie 2.3. (In Dutch). URL: http://www.kadaster.nl/top10nl/download/docs/gegevensmodel_top10nl_2.3.pdf

GML Relay (2006): The fourth GML relay on 2006-01-26 in Emmen (NL). URL: http://www.gdmc.nl/events/relay4/

IMKICH (2004): Informatiemodel Kennisinfrastructuur Cultuurhistorie, RAVI, 21 Sep. 2004.

NEN3610:2005 (2005): Basic scheme for geo-information – Terms, definitions, relations and general rules for the interchange pf information of spatial objects related to the earth's surface. (In Dutch), Nederlands Normalizatie-instituut, Delft, the Netherlands.

INSPIRE (2004): Proposal for a DIRECTIVE OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL establishing an infrastructure for spatial information in the Community (INSPIRE), Commission of the European Communities, 2004.

ISO19109:2005 (2005): Geographic Information: Rules for applications schema, ISO, Switzerland, 2005.

ISO/IEC 19501:2005 (2005): Information technology -- Open Distributed Processing -- Unified Modeling Language (UML) Version 1.4.2, ISO, Switzerland, 2005.

OMG (2002): OMG Request for Proposal: MOF 2.0 Query / Views / Transformations RFP. Object Modeling Group, URL: http://www.omg.org/docs/ad/02-04-10.pdf.

CVs of the Authors

Wilko Quak is a researcher at the Section GIS Technology of the OTB Research Institute for Housing, Urban and Mobility Studies at Delft University of Technology. His research interests include the development of Spatial Database Management Systems, benchmarking of Spatial DBMSs, spatial data modeling and standardization. He was co-developer of the NEN3610 base model of geo-information for the Netherlands and of the GML prototype of the TOP10NL topographic map of the Netherlands.

Marian de Vries is scientific researcher at the Section GIS Technology of the OTB Research Institute for Housing, Urban and Mobility Studies at Delft University of Technology (the Netherlands). Focus of her research is on Web service/client architectures that are based on interoperable software components and open (data exchange) standards. She was co-developer of the GML prototype of the new TOP10NL product of the Dutch Topographic Service. At the moment she is involved in a number of geo-information integration projects for large data providers in the Netherlands.

Co-ordinates

Drs C.W. Quak and drs M.E. de Vries
Delft University of Technology
OTB Research Institute for Housing, Urban and Mobility Studies
Section GIS technology
Jaffalaan 9
2628 BX Delft
The Netherlands

Telephone number	:	+31 (0)15 2783756
Fax number	:	+31 (0)15 2782745
E-mail address	:	c.w.quak@tudelft.nl
		m.d.Vries@otb.tudelft.nl
Website	:	http://www.otb.tudelft.nl