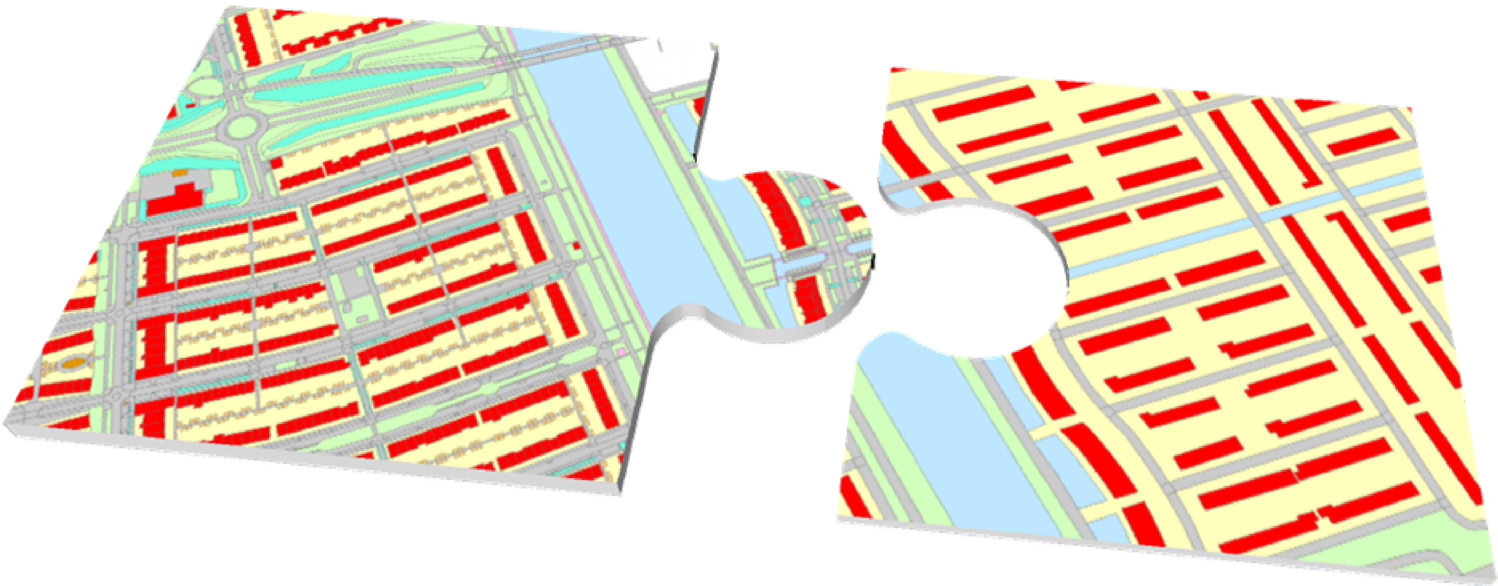
Developing a vario-scale IMGeo using the constrained tGAP structure



Gemeente Rotterdam
Gemeentewerken

TU Delft
Delft University of Technology

Master Thesis Arjen Hofman, 12-2-2008

Table of contents

Preface

The title of this report is 'Developing a vario-scale IMGeo using the constrained tGAP tree'. This report is the final result of the master thesis of Arjen Hofman. The master thesis is the author's final work to finalize the study on Geomatics at Delft, University of Technology. The work for this master thesis has been mainly executed at Gemeentewerken Rotterdam, the department of civil works in Rotterdam.

The main theme of this report is the generalisation of geographical map data. The report contains theoretical and practical parts. It deepens the current knowledge on differences between the large scale geographical standard IMGeo and the medium scale geographical standard Top10NL. After these theoretical differences the practice that is shown appears to have more difficulties; the test data of Almere shows some unexpected issues. Finally the pre-processed test data is to be generalised using the concepts of the constrained tGAP structure, which will be explained later in this report.

A word of thanks to my supervisors and graduation professor: Arta Dilo, the supervisor at Delft, University of Technology; Nicole Borkens, the supervisor at Gemeentewerken Rotterdam; Michiel Boelhouwer and Evert Bontenbal, the practical supervisors at Gemeentewerken Rotterdam and Prof. Peter van Oosterom, professor at the chair of GIS Technology at Delft, University of Technology.
Also thanks to Jonathan Damen, student at Utrecht University who did his master thesis at Vicrea at the same time and on the subject of generalisation, for the many discussions we had on the topic and the possibility to test my data with his algorithms.
Finally, thanks to my girlfriend Martine for her support during this master thesis.

Rotterdam, February 12th 2008
Arjen Hofman

Summary

The aim of this master thesis has been to answer the question:

How can a vario-scale IMGeo be designed and developed by applying the constrained tGAP structure with Top10NL as initial constraint?

This research has been done under authority of Gemeentewerken Rotterdam and TU Delft. For Gemeentewerken Rotterdam the main goal is to investigate the possibilities of automatic generalisation based on their project 'Basisregistratie Geografie', for TU Delft the main goal is to see the concept of the constrained tGAP structure being investigated.

The models IMGeo, a new large scale topographical standard in The Netherlands, and Top10NL, the authentic registration for medium scale topography in The Netherlands, are both derived from NEN 3610; they form the basis for the generalisation in this research. The main differences are explained by their backgrounds. IMGeo originates from the GBKN, whereas Top10NL is based on Top10Vector. The fact that these models are not made cooperatively is bad luck for this research; no object class hierarchy could be made from these models, because there are object classes in Top10NL which don't even exist in IMGeo. The Top10NL object classes therefore can't be seen as a generalisation of the object classes of IMGeo.

The generalisation approach in this research was the constrained tGAP structure, a concept from the universities of Hannover and Delft. In this research IMGeo objects were assigned to Top10NL regions. Four possible methods to assign IMGeo objects to Top10NL regions were developed:

- **Simple overlay method**
  An intersection between the models where every IMGeo object is split at the borders of the overlapping Top10NL object. In the end result only Top10NL geometry will be visible.
- **The maximum area method**
  The Top10NL object which overlaps the IMGeo object the most is the shape to which the whole IMGeo object is assigned to. The IMGeo geometry is kept in this method.
- **The 35%-split method**
  If an IMGeo object belongs for more than 35 % to two Top10NL objects we consider this Top10NL geometry as enrichment of the structure; therefore the IMGeo object is split and a new IMGeo object is created. For all IMGeo objects that don't have two Top10NL objects overlapping for more than 35% the maximum area method is applied
- **The building first method**
  This method assigns IMGeo-buildings to a building region in case of some overlap with a Top10NL building without considering the amount of overlap. The other IMGeo objects are selected as in the maximum area method.

The building first method has been developed because the results of the maximum area method and the 35%-split method were unsatisfying. This building first method gives the best results of the four methods. With this method further research was done.

For the test dataset from the municipality of Almere class weights and class compatibilities have been derived, which are input values for the tGAP structure. This has been done after tuning the weights and compatibilities of the constrained tGAP structure and comparing the end result to the Top10NL dataset.

Final conclusion of this report is that the constrained tGAP absolutely offers possibilities for automatic generalisation from large to medium scale data. However, the amount of pre-processing of the data and the state of development of the tGAP structure are reasons for a governmental organisation like Gemeentewerken Rotterdam to not yet develop a product like this. A topological structure of the models might possibly solve these problems. Cooperation between researching parties, industrial parties and governmental organisations in projects like DURP Ondergronden could be a good opportunity to develop a vario-scale IMGeo as described in this report.

Samenvatting

Het doel van dit afstudeeronderzoek was om een antwoord te vinden op de volgende onderzoeksvraag:

Hoe kan een IMGeo met een variabele schaal worden ontworpen en ontwikkeld volgens de 'constrained tGAP' structuur met Top10NL als voorwaarde?

Dit onderzoek is gedaan in opdracht van Gemeentewerken Rotterdam en de TU Delft. Voor Gemeentewerken Rotterdam was het belangrijkste doel om de mogelijkheden van automatische generalisatie te bekijken in het kader van het project 'Basisregistratie Geografie', voor de TU Delft was het belangrijkste doel om het concept van de constrained tGAP structuur verder uit te werken.

De modellen IMGeo, een nieuwe grootschalige topografische standaard in Nederland, en Top10NL, de Basisregistratie Topografie, zijn beide afgeleid van NEN 3610; deze modellen vormen de basis van dit onderzoek. De belangrijkste verschillen tussen deze modellen kunnen worden verklaard uit hun achtergronden. IMGeo is afgeleid van de GBKN en Top10NL heeft zijn wortels in Top10Vector. Het is jammer voor dit onderzoek dat deze modellen niet parallel zijn gemaakt, om deze reden kon er geen klassenhierarchie gemaakt worden tussen de modellen, omdat er object klassen zijn in Top10NL die niet eens voorkomen in IMGeo. De objectklassen van Top10NL kunnen daarom ook niet worden gezien als een generalisatie van de objectklassen van IMGeo.

De generalisatie aanpak in dit onderzoek was de constrained tGAP structuur, een concept van de universiteiten van Hannover en Delft. In dit onderzoek werden IMGeo objecten toegekend aan Top10NL regio's. Vier mogelijke methoden om dit te doen zijn onderzocht:

- **Eenvoudige intersectie methode**
  Dit is een intersectie tussen de modellen waar elk IMGeo object wordt gesplitst op de rand van een Top10NL object. Dit heeft als gevolg dat het in het eindresultaat enkel nog de Top10NL geometry waar te nemen is.
- **De 'maximum area' methode**
  Het Top10NL object dat de meeste overlap heeft met het IMGeo object is de Top10NL regio waar het IMGeo object aan wordt toegewezen. In deze methode blijft daarom de geometrie van IMGeo behouden.
- **De '35%-split' methode**
  Als een IMGeo object voor meer dan 35 % overlapt met twee Top10NL object, dan beschouwen we de Top10NL gemetrie als een verrijking voor de structuur. Daarom wordt het IMGeo object in dit geval gesplitst en wordt er dus een nieuw IMGeo object gecreëerd. Voor alle IMGeo objecten die niet 2 Top10NL objecten hebben die aan deze voorwaarde voldoen, wordt de 'maximum area' methode uitgevoerd.
- **De 'building first' methode**
  Deze methode kent IMGeo-gebouwen toe aan Top10NL gebouw regio als er sprake is van enige overlap tussen het IMGeo gebouw en de Top10NL regio ongeacht het formaat van het overlappende gebied. De andere IMGeo objecten worden weer geselcteerd volgens de 'maximum area' methode.

De ‘building first method’ is ontwikkeld, omdat de resultaten van de andere methodes niet goed genoeg waren. Deze ‘building first method’ geeft de beste resultaten van de vier genoemde methodes. Het verdere onderzoek is dan ook gedaan met deze methode.

Voor de dataset van de Gemeente Almere zijn gewichten en overgangsmatrices voor de verschillende objectklassen afgeleid, dit zijn ingangswaarden voor de tGAP structuur. Dit is gedaan door de eindresultaten van de constrained tGAP te itereren en deze te vergelijken met de Top10NL dataset.

Conclusie van dit rapport is dat de constrained tGAP absoluut mogelijkheden biedt voor automatische generalisatie van grootschalige naar kleinschalige topografie. De hoeveelheid voorwerk die de data vereist en de onderzoeksstatus van de tGAP structuur zijn echter redenen voor Gemeentewerken Rotterdam om dit nog niet in productie te kunnen gaan nemen. Een topologische structuur voor de topografische modellen zou een hoop datavoorbereiding mogelijk kunnen voorkomen. Samenwerking tussen onderzoekers, bedrijven en overheidspartijen zoals in een project als DURP Ondergronden kan goede mogelijkheden bieden om een IMGeo met variabele schaal, zoals beschreven in dit rapport, te ontwikkelen.

1. Introduction

Automatic generalisation of geographical datasets is the issue to be discussed in this master thesis report. If you mention this issue to a layman, he will probably reply saying: "Didn't that exist before?" Most people nowadays are known to products like Google Earth and Google Maps and they think that all geographical data can be just generalised in the way Google does it.

When looking at a more formal definition of generalisation, as stated by the ICA (ICA, 1973), we see that Google actually performs a good way of generalisation.

"Generalisation is the selection and simplified representation of detail appropriate to the scale and/or purpose of the map".

Google only wants to display that amount of roads which is appropriate for a certain scale and doesn't want to display any other information besides the difference between water, cities and other land. Google also uses fixed scales stored in giant databases. This requires lots of storage space and disables vario-scale zooming.

When talking about automatic generalisation of topographical maps more classes are involved which all have their importance in a map. If we want to extract topographical maps at all scales from the most detailed map and avoid data redundancy, this requires more advanced generalisation tools. To see whether it is possible to extract the medium scale topographical map from a large scale base map this research has been done within the municipality of Rotterdam.

1.1 Problem definition

The collection of geographical data at different scales seems to be more work than necessary. After all, the real world objects that are represented are still the same. Manufacturing geographical data products at different scales from one single data set should be possible, but why isn't this as easy as it seems? The question is: what information has to be shown at what scale?

In producing small and medium scale topographical maps the municipality of Rotterdam has already answered this question, but the question whether this can be automatised still exists. Within the framework of the project Basisregistratie Geografie, which identifies core registrations on geography within the municipality, the municipality asks itself the question whether the production of geographical data can't be more efficient. Data collection at different scales in Rotterdam is now done separately, while collecting data only once would be far more efficient.

This problem of inefficiency forms the basis of this master thesis. In this thesis it is shown whether it is possible to create a vario-scale geographical dataset using a large scale data set. The dataset that will be primarily used is a 1:1,000 dataset according to IMGeo, a new Dutch model on large scale topography. Through a constrained tGAP structure, which will be introduced in section 3.1, it is shown to what extent it is possible to create a vario-scale IMGeo. The constrained tGAP is an idea that builds on the topological Generalised Area

Partition (tGAP) structure as developed at the TU Delft. The constraint that will be used in this structure is Top10NL, the Dutch 1:10,000 map standard.

The main question of this master thesis is:

How can a vario-scale IMGeo be designed and developed by applying the constrained tGAP structure with Top10NL as initial constraint?

1.2 Research objective

Primary goal of this research was to make a vario-scale IMGeo model with constraints from the Top10NL using the tGAP structure which is developed by the TU Delft (Van Oosterom, 2005; Van Oosterom et al., 2006). It is known that IMGeo and Top10NL are both based on NEN3610, but have some different specifications. A related goal was to see whether it is possible to let the models (partially) match.

The used method is the constrained tGAP structure. This structure is a concept proposed by Jan Haunert from the University of Hannover during a visit to the Delft University of Technology. The method is enriched with methods for pre-processing data from different sources and with better weight and compatibility estimates.

1.3 Research issues

By studying the organisation of the municipality of Rotterdam answers have to be found to the following 5 questions:

- What are the requirements for one topographic base map from which all other products could be derived?
- What are the expectations of the cooperation of the large scale topography- and the medium scale topography section in the future?
- What are the current processes within the organisation?
- What are the differences between the IMGeo- and the Top10NL model?
- What connections can be made between the different models?

After answering these questions, it is tried to implement generalisation methods and algorithms in the test data from the municipality of Almere. The choice for data from the municipality of Almere was made, because at the time the research started IMGeo data of Rotterdam was not available, although the research has been done in Rotterdam. By comparing this model to the Top10NL of the same region and on the basis of pre-processing the following questions can be answered:

- Are IMGeo and Top10NL suitable as input datasets for the constrained tGAP structure?
- How can the constraints for the constrained tGAP be determined?
- How can the associated weights and compatibilities be determined?

According to the aggregations that are made in the Top10NL the tGAP structure is built. From the results of the constrained tGAP structure general rules can be defined for the generalisation parameters. The related goal is to get to a situation in which the tGAP tree can be built without constraints and to compare this result with the real constrained tGAP.

The representation at 1:10,000 should than be acceptable with respect to the current 1:10,000 map.

The tGAP class weights and class compatibilities are modified according to what is necessary for the IMGeo and Top10NL models.

Finally, after answering these questions and having studied the situation in Rotterdam and Almere a general answer can be given to the broader central question:

- How can the constrained tGAP structure be used to apply generalisation from large scale topography to medium scale topography?

1.4 Thesis structure

The thesis starts with background of the data models used and the current situation in the department of Surveying of Gemeentewerken Rotterdam in chapter 2. In chapter 3 the research methodology is explained. Chapter 4 introduces the test datasets. Chapter 5 describes the design and the implementation of the constrained tGAP structure. The results of the methods presented in chapter 5 are improved in chapter 6. Finally, chapter 7 gives conclusions and recommendations.

2. Models and projects at Gemeentewerken Rotterdam

The department Gemeentewerken Rotterdam is the organisation within the municipality of Rotterdam that has offered the Master Thesis project. In this chapter the reasons for the research are given. The central question that will be answered in this chapter is: why is this generalisation project interesting for the municipality of Rotterdam. Gemeentewerken Rotterdam is a very dynamical organisation in which a lot of developments take place and have taken place during this research which are of importance in this research. Interviews were done to be able to describe these developments properly.

This chapter will first describe the products of the organisation that are currently being maintained in section 2.1. After this the developments are being described. Section 2.2 introduces the authentic registration on Geography (Basisregistratie Geografie) for the municipality of Rotterdam. From this project other developments are derived. In section 2.4 the information model IMGeo and its role in Rotterdam is described, section 2.4 handles the steps towards Top10NL. Section 2.5 gives an overview of projects outside Gemeentewerken Rotterdam, to which this master thesis also contributes. Finally, in section 2.6 some conclusions are drawn.

Part of this chapter is taken from the literature study done by Hofman (Hofman, 2007). In this report the chapter is extended with more material on the introduction of IMGeo and Top10NL in the Rotterdam situation, based on the interviews.

2.1 Current situation at Gemeentewerken Rotterdam

The organisation of Gemeentewerken Rotterdam has about 1800 employees. The department in which this master thesis project takes place is the department of Surveying (Landmeten); in this department about 100 people are employed. This department consists of 2 subdepartments: measurements and geo-registrations. The two most important products maintained by the subdepartment of geo-registrations are the Large Scale Base Map (GBKN) with a scale of 1:1,000 and the Medium Scale Base Map (KBK Rotterdam) with a scale of 1:10,000. Within the production process of this KBK also smaller scales are derived; these are 1:20,000 and 1:50,000. In the next two subsections the large scale and medium scale data sets will be described.

2.1.1 Large Scale Topographic Base Map

The Dutch Large Scale Topographical Base Map (GBKN) is a map product with scales varying from 1:500 to 1:5,000 (productinformatie GBKN, 2006). The scale that is mostly used for this product is 1:1,000. All municipalities in The Netherlands are obliged to keep this GBKN up to date; they can also choose to contract this work to a coordinating foundation. Its use varies a lot; it is for example used as bottom layer for maintenance services within municipalities or as base map for utility companies. Also the collection rules vary per municipality.

Most municipalities use the GBKN as a basis for all soft and hard topography. The collection and maintenance of the data is done in two different ways:

- Terrestrial (field) data collection
- (Stereographic) Aerial photographic data collection

Terrestrial data collection is more accurate, but also more expensive. In lots of municipalities large parts of the GBKN are not measured terrestrially. Instead, they choose for the less accurate aerial photography to fill up the total municipal map. In Rotterdam this is not the case, all data for the GBKN is collected terrestrially. The Rotterdam version of the GBKN is called GBK-Rotterdam or GBK-R for short.

All large scale topography is currently being exchanged according to the standard NEN1878 (LSV GBKN, 2004). This is the Dutch standard for exchanging topography. It doesn't exchange polygons; it exchanges only points and lines. In section 2.4, when IMGeo is described, we will see that the use of polygons in future standards can become necessary.

2.1.2 Medium scale Topographic Base Map

The medium or small scale topographic base map, as the municipality of Rotterdam maintains it, is not a regular map. Most municipalities only maintain their 1:1,000 map and leave the maintenance of a 1:10,000 map to the Dutch Cadastre (Kadaster).
The Topographical Agency (Topografische Dienst) used to be the responsible organisation for this 1:10,000 map, called Top10NL or Top10Vector, its earlier version. After a reorganisation the Kadaster took control of Top10NL.

The 1:10,000 map from the municipality of Rotterdam (KBK-Rotterdam, or KBK-R for short) shows very much resemblance with the Top10NL. There are differences between the data models and the way information is presented. The 1:10,000 map is collected and drawn from aerial photographs and is therefore far less accurate than the 1:1,000 GBKN. Both models are currently being changed as part of the project Basisregistraties (authentic registrations). From 2008 Top10NL is an authentic registration and with that governmental organisations will be obliged to use it, except for municipalities that have their own production of the medium scale map; they have to conform to Top10NL from 2010. In the next subsection we take a closer look at this project and the changes this implies.

2.2 Basisregistraties

This section will introduce the main developments in the organisation of Gemeentewerken Rotterdam due to the project Basisregistraties. The project Basisregistraties is the translation to the Rotterdam situation from the national project 'Stroomlijning basisgegevens', which authenticates certain registrations in order to channel information within governmental organisations. Subsection 2.2.1 describes the national developments and subsection 2.2.2 describes the actions of the municipality of Rotterdam with respect to these developments.

2.2.1 Nationale Geo Informatie Infrastructuur

The Nationale Geo Informatie Infrastructuur (NGII) is the Dutch national Geo Information Infrastructure (GII). Within the framework of a GII data should be collected once and used for multiple purposes.

Within the framework of the project 'Stroomlijning Basisgegevens' 6 authentic registrations were indicated in the beginning. These registrations are connected to each other. One of these authentic registrations is the registration of Topography. An authentic registration is said to be the only official governmental registration on that particular area. For the registration on topography the government chose for Top10NL, the 1:10,000 map from the Topographical Service of the Dutch Cadastre (Kadaster) as authentic registration.

In figure 2.1 it is shown how the 6 authentic registrations are inter-related (Rietdijk and Verhoef, 2002). It is shown that the building registration and the cadastral registration are both related to the topography registration. The problem with the 1:10,000 map is that buildings are not detailed enough in the Top10NL; it can therefore not be related to the cadastral and building registration, which it would have to be according to figure 2.1. This makes that Top10NL is totally outside the system of authentic registrations. In (Schravendeel et al., 2005) it is stated that in the future the Medium Scale Map has to be derived from the Large Scale Map to be able to state that the system still has *authentic* registrations. The connection with the GBKN could be made, but this is not an authentic registration, because it is partly financed by private parties. In the original vision presented in figure 2.1 there were only 6 authentic registrations; at this moment there are 10 authentic registrations and three candidate authentic registrations.

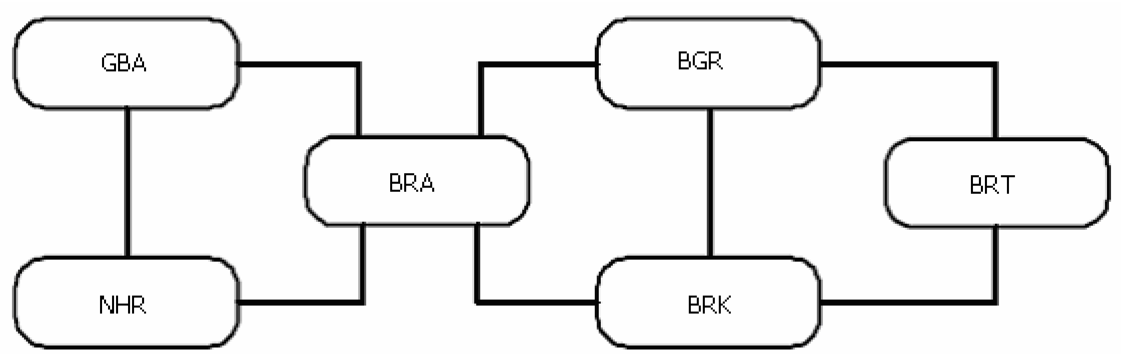


Figure 2.1 Authentic registrations in The Netherlands (Courtesy: Rietdijk and Verhoef, 2002)

The 6 authentic registrations in figure 2.1 are:

1. Municipal administration of citizens (GBA)
2. New company register (NHR)
3. Address registration (BRA)
4. Building registration (BGR)
5. Cadastral registration (BRK)
6. Topographical registration (BRT)

One of the candidate authentic registrations is the 1:1,000 large scale base map (GBKN). Not to replace the Top10NL as an authentic registration, but to become a separate authentic registration on large scale topography beside the small or medium scale authentic registration (LSV GBKN, 2002). The GBKN could be the topographical registration in figure 2.1 which connects the topography to the cadastral- and building registration. Figure 2.1 shows the original vision of the authentic registration in The Netherlands, figure 2.2 shows the current status and vision for authentic registrations in 2009.

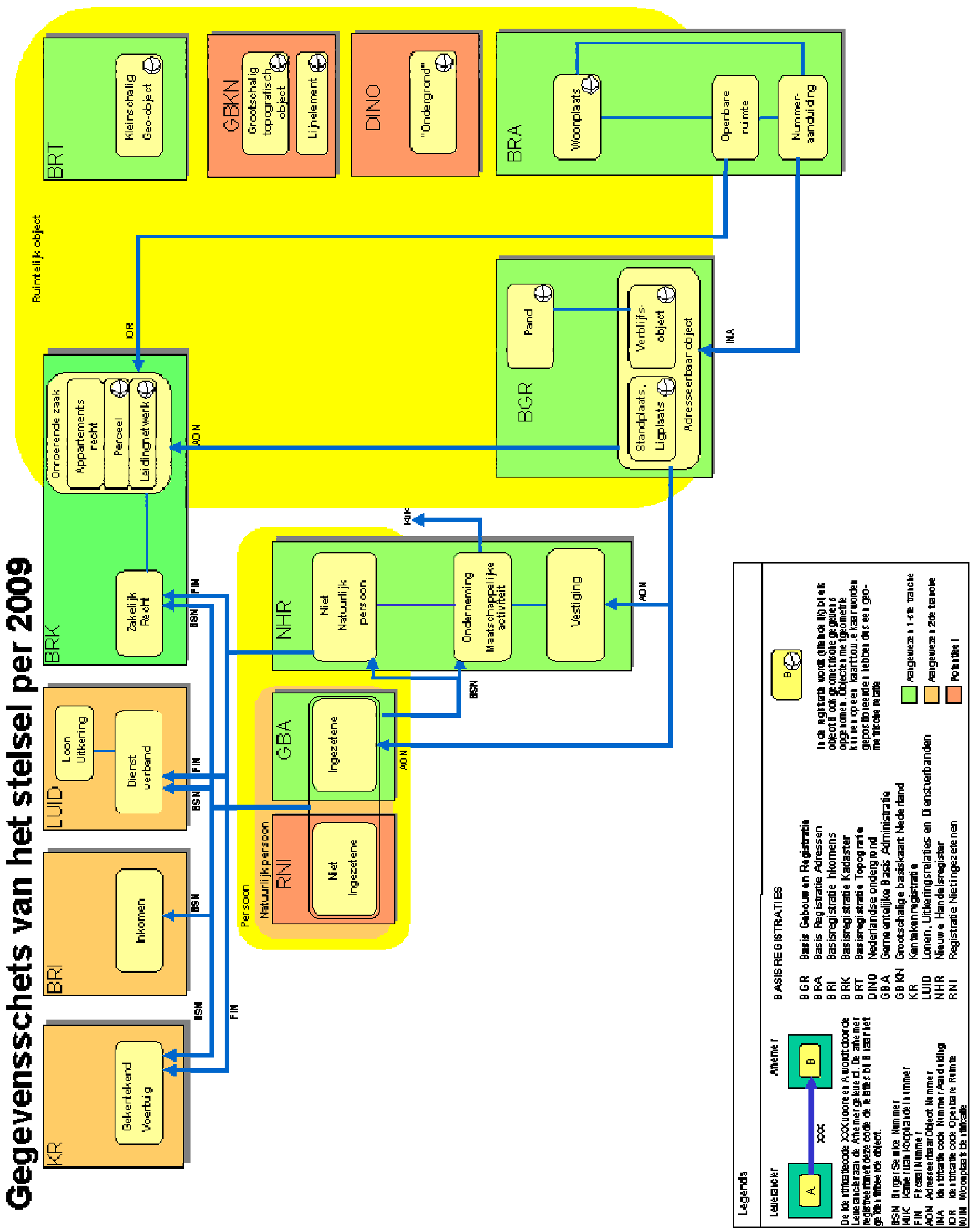


Figure 2.2: Vision for the authentic registrations in The Netherlands in 2009 (courtesy: Campschroer et al., 2006)

If the GBKN would join the system of authentic registration, this wouldn't be according to the definition of authentic registrations, because the vision of the authentic registrations is to collect the data once and to use it in all possible situations. When using the GBKN as well as the Top10NL the strange situation would arise that 2 separate authentic registrations show the same data at different scales with different specifications. One of the solutions could be to go to a vario-scale topographic model in which the data plays a central role and the visualisation is of minor importance; this option is investigated in this master thesis research.

The GBKN will be needed to make a connection between the registrations on topography on the one hand and the registration on buildings and addresses (BAG) on the other hand. BAG is the junction of the registration of buildings and adresses in figure 2.1. The Top10NL is not detailed enough to extract individual building geometry out of it, for this reason the geometry of the GBKN can be used for this. However, this connection is not (yet) indicated in figure 2.2.

In the next subsection we will see how these national rules have been translated to the municipal situation in Rotterdam.

2.2.2 Basisregistratie Geografie

The information in this subsection came from interviews with four people from the surveying department at Gemeentewerken Rotterdam. The leader of the project 'Basisregistratie Geografie' (authentic registration geography) for the municipality of Rotterdam is Nicole Borkens. Together with Edim Hadziavdic, who is responsible for the subdepartment of geo-registrations, she provided me with the necessary information for this subsection through interviews. Also Frank Kenselaar and Louis Smit were interviewed for this purpose.

In Rotterdam the governmental agency (Bestuursdienst) wanted to translate the system of authentic registrations for the national level to the municipal level. Therefore core registrations were created. In Dutch they have the same name (Basisregistraties), but they are not authentic because they are not the only source of the particular information. It was not just getting Top10NL in the organisation of the municipality, but it was also a matter of looking at the national developments and see what the municipality of Rotterdam can do with these developments.

In Rotterdam we can distinguish between a (most important) first order and a (less important) second order core registration on geography. The first order registrations are:

1. GBK-R (Large Scale Base Map Rotterdam)
2. KBK-R (Small/Medium Scale Base Map Rotterdam)
3. LVZK (Utilities registration)

Five other products can be considered to be registrations of the second order:

1. TIR grenzen (Borders registration)
2. Gemeentelijke eigendommen (Municipal Property Map)
3. Kadastrale kaart (Cadastral Map)
4. Luchtfoto's (Aerial Photographs)
5. Panoramafoto's (Panoramic Photographs)

The way the municipality of Rotterdam can distinguish between 'topography as a product' and 'topography as data' is through the way the finances of the products are organised. In the old situation the municipal customers, i.e. other services within the municipality of Rotterdam, payed directly by subscriptions to Gemeentewerken Rotterdam for the products they were delivered. Part of the project Basisregistratie Geografie is to organise the financing of the registrations in a central way. The total amount of money spent by the municipal customers in the year 2005 will be payed by the Bestuursdienst from 2007. The customers pay their share to the Bestuursdienst and in turn they are able to use all data provided within the Basisregistratie Geografie. Now the whole organisation of the municipality of Rotterdam is able to use all the geographical data of the surveying department, even without subscription.

The expectation is that through this opening of information sources the registration will be used more frequently by the own municipal services. Many agencies use data from TeleAtlas in for example Google Maps to search for information in the map; Gemeentewerken Rotterdam hopes to diminish the use of other data by providing the most reliable and actual data for the municipal organisation. With this move Gemeentewerken Rotterdam uses the produced data more efficiently and is still secured of the same amount of financial resources.

Another part of the project is to get other services involved in what information they want to see in the product in order to be able to use the map for their own registrations. Nowadays the department of roads and the department of public space (Openbare Ruimte) use the GBK only as bottom layer to make their own polygons in their own systems. In the new situation they will be able to export the geometry from the GBK and do their maintenance on the basis of that product. Customers from outside the organisation of the municipality of Rotterdam (e.g. utility companies) still have to subscribe to the data they want to have.

Reorganisation

As mentioned before since July 1st 2007 the department of surveying within Gemeentewerken Rotterdam has been reorganised. The main vision for the reorganisation is to continue with the direction of the project Basisregistratie Geografie and to work more effective.

In the old situation the maintenance of products and doing external surveying works were already the core businesses. The department is now divided into two sub-departments, one on surveying and one on geo-registrations. The structure within these sub-departments is not anymore that separate groups work on separate products, but the employees are based in teams. The teams execute the most urgent work. The utility work is work that needs to be done directly with an actuality of a month. The GBK-R has an actuality of three months and the KBK has an actuality of a year. This means that in times that there is less work to do on utilities the teams can focus more on the GBK or KBK and are therefore more flexible.

The gain of the reorganisation that has to do with the project Basisregistratie Geografie is that the process of product development is now officially set to registration maintenance. The department therefore needed to change the way of approaching geography. The visualisation (i.e. the making of products) is not the central issue anymore. The registration of objects takes a central place now, with the visualisation as a derivative of this.

Other aspects of the project Basisregistratie Geografie are currently in process. The general vision of the project is described in this section; the details will in two cases be explained:

IMGeo and Top10NL are part of this project. These will be studied in respectively section 2.4 and 2.5. The intended end date of the project Basisregistratie Geografie is the end of 2010.

2.3 The standard NEN3610

NEN 3610 is the standard from the Dutch Institute for Standards (NEN: Nederlands Normalisatie-insituut) for geo-information. The title of the standard is therefore 'Basic model Geo-information (Basismodel Geo-informatie). This section is about the latest version of NEN 3610, which is NEN 3610:2005, which is the successor of NEN 3610:1995. If in this chapter, or elsewhere in the thesis, the term NEN 3610 is used, this refers to NEN 3610:2005. This section is based on the text of this report (NEN, 2005).

NEN 3610 is the Dutch version of the international 'General Feature Model'; the rules of this standard are defined in ISO 19109, titled 'Geographic Information - Rules for Application Schema'. The International Organisation for Standardisation (ISO) defines real-world objects and translates them to geographical features (ISO, 2005). These geographical features we want to model. This section describes how this modelling is done in NEN 3610.

Contents NEN 3610

Figure 2.3 shows the pyramid of Geo-information models. It shows that NEN 3610 is the general version of the geo-information model. From this other sector specific standards can be derived. In chapter 2 IMGeo and Top10NL were introduced.

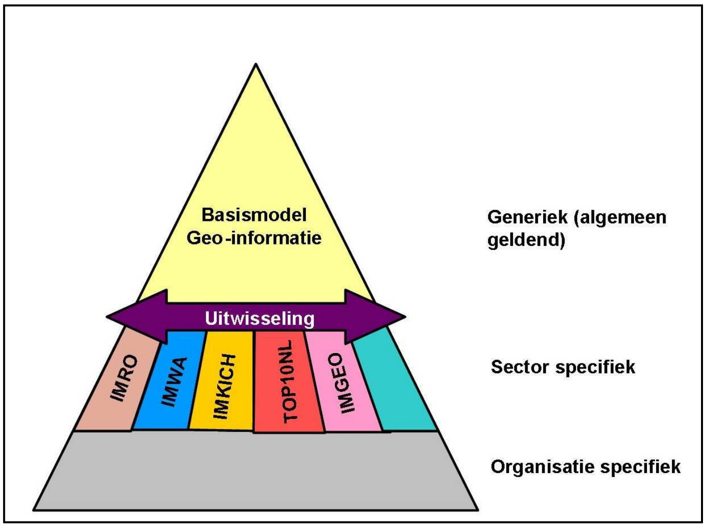


Figure 2.3: The Dutch Geo-information pyramid with NEN 3610 in top (Courtesy: NEN3610, 2005)

The models that are represented in figure 2.3 are:

- IMRO (Information model Spatial Development)
- IMWA (Information model Water)
- IMKICH (Information model Cultural History)
- Top10NL (Medium Scale Topography)
- IMGeo (Large Scale Topography)

The model is meant to make exchange easy. The bottom of the pyramid shows the organisation specific part. The standardisation in exchange is a matter of the organisations itself.

The description of classes and their relationships is done in a UML class diagram. In the Object Constraint Language (OCL), which is defined as part of UML, constraints to the model can be added. The exchange in NEN 3610 takes place in GML.

All objects in NEN 3610 are defined in the same way; for every object class there is a set of prerequisites, which can be related to the UML model, or give explanation to the position of the class in the model. Table 2.1 shows this standard presentation and description of a class.

Class content	Explanation
Definitie	Definition of the class
Herkomst definitie	The origin or source of the definition. This position is only filled if an existing definition is used.
Inwinningsregels	Description of collection rules with respect to this class. In NEN 3610 it is not possible to define collection rules. This can be done on sector level.
Generalisatie	From which class this class is a generalisation?
Specificatie	From which class this class is a specification?
Attributen	The attributes defined for this class.
Associaties	With what classes this class has associations.
Gebruik/voorbeelden	Explanation of the use of this class.

Table 2.1: Presentation of a class in NEN 3610

NEN 3610 defines a super class object GeoObject, in which all objects are identified and some general attributes are defined. These attributes are attached to all objects and shown in table 2.2.

Attribute name	Explanation
identificatie	A unique identifier for a geo-object
objectBeginTijd	System-time on which the object emerges
objectEindTijd	System-time on which the object becomes invalid
versieBeginTijd	System-time on which this version of the object emerges
versieEindTijd	System-time on which this version of the object becomes invalid
status	The status connected to the life-cycle of a geo-object
locatie	Reference to the location of the geo-object in terms of address or location description
beginTijd	Date on which the geo-object started to exist in reality
EindTijd	Date on which the geo-object ended to exist in reality
naam	Name of the object

Table 2.2: Attributes for the object GeoObject in NEN 3610

The geometry is not one of the attributes of the super class GeoObject. This is because at this level it is not possible to say how an object's geometry is defined. The geometry of objects is therefore always directly or indirectly defined in the subclasses.

NEN 3610 has 14 subclasses; not all these subclasses have to be used in the sector specific models. As we will see in section 2.6 sector models differ from each other. Only those classes in the sector models are used which are of use for the sector. NEN 3610 defines all subclasses or object classes which could emerge in the different sector models.
The 14 object classes in NEN 3610 are mentioned in Dutch with between brackets their translation in English.

- Weg (road)
- Spoorbaan (railroad)
- Water (water)
- Terrein (terrain)
- Gebouw (building)
- Kunstwerk (civil work)
- Waterkering (dike)
- Leiding (utilities)
- Inrichtingselement (topographical element)
- RegistratiefGebied (registration area)
- PlanologischGebied (planological area)
- FunctioneelGebied (functional area)
- GeografischGebied (geographical area)
- Meting (measurement)

As shown in figure 2.3 the topographical models are not the only models derived from NEN3610. Since IMGeo and Top10NL are both topographical models defined with respect to the same standard, we should expect that they resemble a lot. In section 2.6 we will see whether this indeed is the fact. First we take a closer look at IMGeo and Top10NL in sections 2.4 and 2.5.

2.4 IMGeo and its implementation in Rotterdam

This section discusses IMGeo. IMGeo is the new model according to which the Rotterdam GBK will be made in the future. This section describes the model and the current status of the model. The text in this subsection comes sometimes from the report on IMGeo, version 1.0 (IMGeo, 2007). The author translated and edited the text in those cases. Other information comes from interviews with Rinske van Gosliga and Edim Hadziavdic. The section starts with the structure of IMGeo in subsection 2.4.1; the status of the model will be described in subsection 2.4.2.

IMGeo is a model which in the first place has been created because 4 large municipalities in the Netherlands felt the need for large scale object-oriented geo-information. These municipalities initiated to make an information model, which would define and standardize the exchange of objects. These municipalities are Amsterdam, The Hague, Vlaardingen and Rotterdam.

2.4.1 Structure

The structure of objects in IMGeo will be described in this subsection. The focus will first be on the objects, later on the attributes. The model is fully described in IMGeo (2007).

Objects
In IMGeo a lot of different objects are taken into account. Some of them are area objects; others are line or point objects. All available main objects are listed; for every class the English translation is added between brackets. Some classes describe the situation as parts of an object. With roads for example normally the road objects are cut into pieces which go from one junction to the next; these are the road parts. The whole road object consists of

several road parts. The whole class diagram of the IMGeo model, which defines how the objects are related to each other, is provided in Appendix A.

IMG_ GeoObject (object)
This is the super class under which every object in IMGeo is situated. Its attributes are linked to every object in the model.

Weg (road), Wegdeel (road part)
These classes define all kinds of roads in the model; the object class road is the super class here. All roads can consist of several road parts.

Spoorbaan (railroads), spoorbaandeel (railroad part)
These classes define all railroads in the model; the object class railroad is the super class here. All railroads can consist of several railroad parts.

Water (water), waterdeel (water part)
These classes define all water in the model; the object class water is the super class here. All water objects can consist of several water parts.

Terrein (terrain), terreindeel (terrain part)
These classes define all terrains in the model; the object class terrain is the super class here. Terrains can consist of several terrain parts. Different types of land use are for example modelled in these classes.

Kunstwerk (civil works), kunstwerkdeel (civil works part)
These classes define all civil works in the model; the object class civil works is the super class here. All civil works can consist of several civil works parts. Examples of civil works to be modelled in this class are bridges and tunnels.

Pand, verblijfsobject (buildings, residence objects)
These classes define all buildings in the model. All definitions in this class are according to the regulations of the registration for buildings and addresses (BAG). In IMGeo only the geometry of the building and the ID of the buildings and their associated residence objects appear.

Inrichtingselement (topographical elements)
These classes define all elements which fit up the area. All kind of elements are meant here, for example traffic lights and lamp-posts. For the complete list of elements the reader is referred to the full report of IMGeo (IMGeo, 2007).

Registratief gebied (registration area)
The registration area is an abstract class. Here the space is subdivided in provinces, municipalities, places of residence, neighbourhoods, etc.

IMGeo objects are used in the format of figure 2.4. The figure represents the form of an UML-class diagram in IMGeo.

Klassenaam
+attribuutnaam : <attribuutdomein> [multipliciteit]

Figure 2.4 Representation of an object in IMGeo

Explanation of figure 2.4
- 'Klassenaam' = the name of the object class.
- 'attribuutnaam' = the name of the attribute.
- <attribuutdomein> = a reference to the acceptable values for this attribute, the domain.
- [multipliciteit] = the number of values the attribute can take.

Attributes
Every object in IMGeo has a unique ID; this ID is the first and most important attribute of an object. Most attributes of objects in IMGeo are stored in the super class IMG_GeoObject. These attributes are:

- Object ID
- Object Start time
- Object End time
- Version Start time
- Version End time
- Status
- Location
- Name

The start- and end time of objects are the time the object first appears and when it becomes invalid. The version time is the time an object is modified; the end time of this is when a new version of the object is created.

The status of an object is connected to the life-cycle of the object. This consists of planned objects, existing objects and former objects. This temporal aspect in IMGeo allows to see how situations will be in the future and to recall earlier situations for e.g. juridical procedures. 'Location' describes the location of the object, not in coordinates but in words. The name of the object is for example the name of the street.

In the different object classes several attributes appear frequently. These attributes are geometry and level. The attribute geometry defines whether the object can appear as area, line or point (or a combination of those). The attribute level defines on what relative level the object is situated in case more objects are situated on the same x-y-spot (e.g. when a bridge crosses a river). The ground level is taken as level 0. Objects underneath or above this level are numbered with respect to this ground level.

For a full overview of all attributes the reader is referred to the report IMGeo (IMGeo, 2007).

2.4.2 Status of IMGeo

IMGeo has been initiated apart from the project Basisregistratie Geografie, but it actually can fit in very well. On the national level the need was felt to have a GBK which represents more than just geometry. IMGeo is an object-oriented model to which attributes can be assigned. In the first place it has been designed to improve the GBK-model, but eventually it might be used for other scales as well.

The IMGeo model defines rules for data collection, but not for visualisation. This makes that the model is suitable as a basis for generalisation. At the Kadaster also plans are made for an information model, named IMTop, which is planned to start from scale 1:10,000. The current Top10NL is not seen as an information model by everyone, because it contains specific visualisation rules for specific scales; in information models visualisation rules play a minor role. It might seem that IMGeo already is the solution to the generalisation problem, but in the first place IMGeo will be used as a means to produce a new object-oriented GBKN.

For IMGeo currently (August 2007) pilots are being finalised which show to what extend the conversion of the current area data to IMGeo in GML works. These pilots are done in the municipalities of The Hague, Echt-Susteren and Almere. The Hague and Almere have been the most successful in executing the pilot. First a paper mapping was made and this has been implemented in GML. The data of Almere appeared to be the most useful, because it consists of a quite complete set of objects. This was used for further tests during this master thesis research.

After finalising the pilots the model was submitted in the 'GI beraad', an organisation within the ministry of Spatial Development (VROM). After this it was placed in the pyramid of geo-information models of GeoNovum as shown in figure 2.3.

GeoNovum is the organisation that will be responsible for the IMGeo model. The model will be frozen for at least two years, after this period it is possible to add or change things to the model. Rinske van Gosliga will probably take place in the committee to guard the model on behalf of Gemeentewerken Rotterdam.

Beside the submission in the GI beraad the model will also be submitted by the standards forum of the Dutch ministry of Internal Affairs (Binnenlandse Zaken).

2.5 Top10NL and its implementation in Rotterdam

Within the organisation of Gemeentewerken Rotterdam there is a process going on for some years to come to delivering the KBK-R to the Dutch Cadastre. The KBK-R is a map drawn from aerial photographs. Objects don't have any attributes, only a classification; this will be changed when going to Top10NL. This section describes the structure of Top10NL in 2.5.1 and the current status of implementing Top10NL in Rotterdam in subsection 2.5.2.

Top10NL is the 1:10,000 map product of the Kadaster; it is the successor of the vector model Top10Vector. This product was divided into separate map sheets, which didn't overlap. The new model Top10NL is object-oriented, has no separate sheets anymore and is defined under NEN 3610.

Top10NL is expected to be a product which can form a bridge between several external geographical products, because of its object orientation. A lot of effort is also done to the visualisation of the product.

2.5.1 Structure

In the Top10NL attributes are connected to the objects, which is not the case in the KBK-R. Top10NL is based on the standard NEN 3610 and therefore the major object classes resemble very much to the major object classes in IMGeo, which was presented in section 2.4. Figure 2.5 shows the Top10NL object classes.

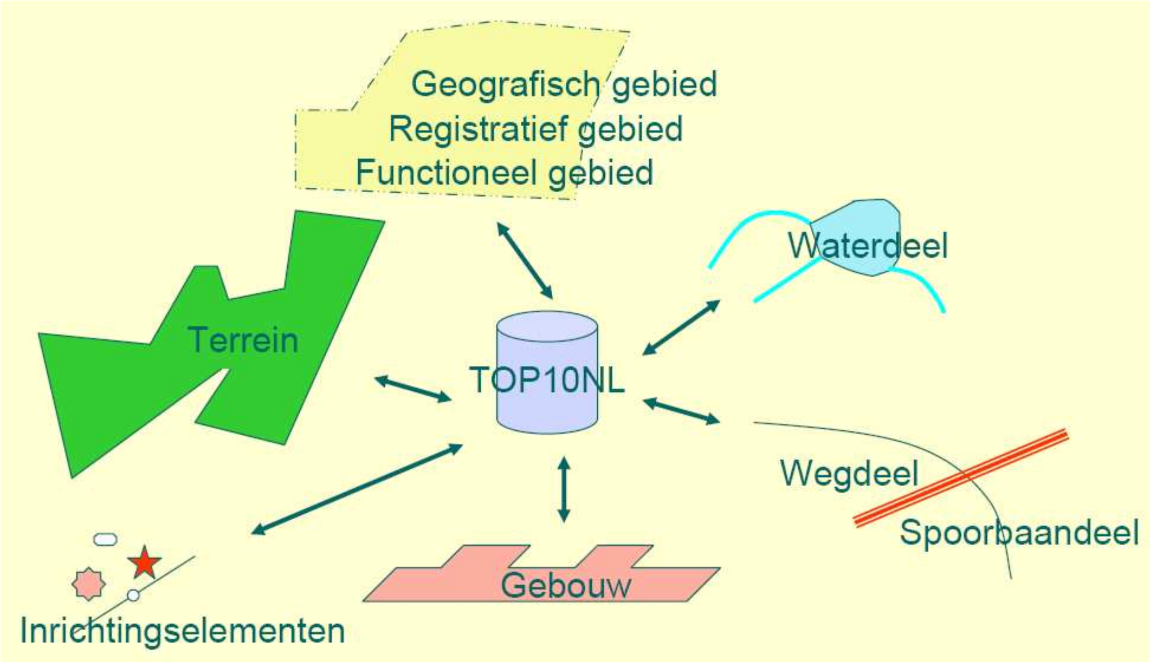


Figure 2.5: Objects in Top10NL (Courtesy: Lentjes, 2007)

The object classes defined in Top10NL are:

- Wegdeel (road)
- Spoorbaandeel (railroad)
- Waterdeel (water)
- Gebouw (building)
- Terrein (terrain)
- Inrichtingselement (topographical element)
- Reliëf (isolines)
- Registratief gebied (registration area)
- Geografisch gebied (geographical area)
- Functioneel gebied (functional area)

The full UML diagram of Top10NL is shown in Appendix B. Even though IMGeo and Top10NL are based on the same standard, the figure shows some differences.
In section 2.6 an extended comparison between the models will be made.

2.5.2 Current situation

Since Top10NL is the authentic registration on topography, every governmental organisation is obliged to use Top10NL from 2010. The official release of Top10NL took place on January 1st 2008. The municipality of Rotterdam currently keeps its own KBK-R and the customers are satisfied about the quality and the actuality of this map product. Some features that are in the current KBK-R are not in the Top10NL.

The advantages of Top10NL with respect to the KBK-R are (Boelhouwer, 2006):

- Object orientation
- Based on both national and international standards
- Visualisation and objects are separated
- Easy exchange between municipalities
- History of objects is stored
- Data model can be extended
- Many applications due to large amount of attribute information

These are the advantages of Top10NL with respect to the KBK-R. When Top10NL was developed there was no object oriented model in The Netherlands. With IMGeo Top10NL is not unique in this anymore. In the first place Top10NL, as well as IMGeo, is a data model, therefore the visualisation of objects is separated from the object storage.

The information in Top10NL can be very easily exchanged between municipalities and other users, because the methods for data collection are the same everywhere.
What Top10NL doesn't have, or doesn't have filled, and what Rotterdam and its customers do require, is:

- Street names
- Inner areas
- High actuality
- Plan topography

The street names are attributes of the roads in Top10NL, but they are not displayed on the map as in the KBK. It also appears that most of the attribute fields with street names are not filled in the Top10NL. The inner areas are displayed as buildings in Top10NL, but this is information the fire brigade wants to have to see where they can get access to a buildings. Also subsidies for boroughs are based on the KBK; with no inner areas the results of this will be very different.

The actuality of the KBK is one year; the actuality of Top10NL is 2 years for roads and buildings; for the rest of the objects the actuality varies. Customers request an even more frequent update of the KBK. This is not possible because the aerial photographs are only made once a year. Plan topography could be inserted in the attribute 'status' in Top10NL, but it is not.

The differences between the products are discussed by both parties and the idea of an extra layer for Rotterdam came up. This would mean that Rotterdam would produce and exchange the Top10NL according to the specifications of the Kadaster, but for internal use would add its own layer.

A pilot with the Kadaster for these plans stocks, because the Kadaster is also in a reorganisation. However, it is a planned fact that Rotterdam will have Top10NL in their organisation from 2010, whether self-produced or produced by the Kadaster. Whether this will be with or without an extra layer and whether they will deliver the data to the Kadaster or vice versa remains unknown for now.

2.6 Differences between IMGeo and Top10NL

This section will discuss the differences there are between IMGeo and Top10NL. This will be done in two separate subsections, because the differences appear to be at two different levels. Subsection 2.6.1 discusses the differences in classes and attributes that are collected. Subsection 2.6.2 is on the differences in geometry.

2.6.1 Classes and attributes

IMGeo and Top10NL are based on the same standard: NEN 3610. Due to their different history they are still built up differently. IMGeo is built on the basis of the specifications of the GBKN and can be seen as a polygon-GBKN; Top10NL is the object-oriented successor of Top10Vector. To discover the differences between the models we take the top-down approach; we start at the top of the hierarchy and end with the most detailed attributes of both models.

GeoObject
The most important class in both models is the GeoObject. In section 4.1 we saw the attributes of the GeoObject being defined according to NEN 3610. Table 4.3 shows what attributes the GeoObject has in both sector models compared to the initial generic model. This table is to be read horizontally; the corresponding attributes in the other models are to be found on the same line.

Table 2.3 shows that IMGeo differs from NEN 3610 only in the attributes beginTijd and eindTijd; those are not included in IMGeo. For Top10NL there are more differences; the attributes beginTijd and eindTijd are not included as well, but also the status, location and name of the objects are not obligatory attributes in the Top10NL GeoObject. Instead Top10NL has other meta information: brontype (source type), bronbeschrijving (source description), bronactualiteit (source actuality), bronnauwkeurigheid (source accuracy) and dimensie (dimension). These attributes all tell something about the way the data has been collected.

The attributes status and name are in Top 10NL not in the superclass GeoObject, but in the individual objects. This is because the domain of values of these attributes can differ per object class in Top10NL.

NEN3610	IMGeo	Top10NL
identificatie	identificatie	identificatie
objectBeginTijd	objectBeginTijd	objectBeginTijd
objectEindTijd	objectEindTijd	objectEindTijd
versieBeginTijd	versieBeginTijd	versieBeginTijd
versieEindTijd	versieEindTijd	versieEindTijd
status	status	
locatie	locatie	
beginTijd		
eindTijd		
naam	naam	
		brontype
		bronbeschrijving
		bronactualiteit
		bronnauwkeurigheid
		dimensie

Table 2.3 Comparison of attributes in GeoObject

Object classes
As already seen in section 2.3 a sector model does not necessarily contain all classes mentioned. This is where we will see the first differences in the objects that are collected. Table 2.4 shows the classes that are defined in both models. The build up is the same as table 2.3.

NEN3610	IMGeo	Top10NL
Weg	Weg	Wegdeel
Spoorbaan	Spoorbaan	Spoorbaandeel
Water	Water	Waterdeel
Terrein	Terrein	Terrein
Gebouw	Pand, Verblijfsobject	Gebouw
Inrichtingselement	Inrichtingselement	Inrichtingselement
Kunstwerk	Kunstwerk	
RegistratiefGebied	RegistratiefGebied	RegistratiefGebied
Leiding		
FunctioneelGebied		FunctioneelGebied
GeografischGebied		GeografischGebied
PlanologischGebied		
RegistratiefGebied		
Meting		
		Reliëf

Table 2.4 Comparison of classes in IMGeo and Top10NL

Table 2.4 shows that different approaches have been taken to model the real world. In this respect IMGeo models buildings in the same way as NEN 3610, whereas Top10NL classifies them as they did in Top10Vector. This appears to give lots of differences in the list of attributes of both models. The class 'Reliëf' is also a class which is not modelled in IMGeo and NEN 3610 and can also be seen as being historical.

The classes Geographical, Planological and Functional area have all not been modelled in IMGeo; geographical area is too broad for municipal purposes, planological area's (spatial development areas) are interesting for other models under NEN3610. Functional area can be seen as an extension of the class 'Terrein', the geometry of these areas however is hard to determine. For example the boundary of a cemetery is not collected as area object in

Top10NL, but only as point object somewhere in the area to be able to label it. These are all reasons for less object classes in IMGeo.
The class 'Kunstwerk' is not modelled as a separate class in Top10NL. All objects belonging to this class in IMGeo are in the Top10NL class 'Inrichtingselementen'.

A remarkable fact is that in Top10NL the classes road, railroad and water are modelled as parts of the roads and not with a separate aggregation class above it. This means that separate road parts can't be identified to be belonging to the same road based on their common ID. However, this can be done by querying objects through their street names. Unfortunately, the attribute field with street names is hardly filled and therefore this is not a very good alternative in the current product.

Attributes and attribute values

We already discovered that the attributes under GeoObject in both models differ from each other. However not obliged in GeoObject, attributes like 'status' do exist in Top10NL on another place. This means that the individual objects normally have a status with values that can differ per object class in Top10NL.

In the appendices C and D an analysis is made of which attributes exist in which models. This analysis was done for the project IMTop, which will be introduced in section 2.7.

One of the most striking differences in the attributes is between the object class 'Gebouw' (Top10NL) and 'Verblijfsobject' (IMGeo); 'Verblijfsobject' is according to the definitions of the BAG and 'Gebouw' is mapped according to the mapping standards in Top10NL, which means that an inaccuracy of 4 meters is allowed. Due to the fact that BAG is being followed in IMGeo no attributes besides the building ID, the residence object ID, the geometry and the relative height are stored. In Top10NL also for instance the type of the building and the 'height class' (hoogteklasse) are taken into account.

Another important difference appears in the object class 'Inrichtingselementen' or topographical elements. The attribute values in this class differ very much from each other. The reason for this can be also found in the different backgrounds of the models. The build up of this class is also different in the different models. In Top10NL the topographical elements are all attribute values of the attribute 'type inrichtingselement', while in IMGeo the topographical elements are first subdivided in 11 subclasses. In these 11 subclasses the attribute 'type (name of subclass)' defines the attribute value of the subclass of the topographical element.

The 11 subclasses are:

- Bak (bin)
- Bord (sign)
- Installatie (installation)
- Kast (case)
- Mast (pylon)
- Overig Bouwwerk (other building)
- Paal (post)
- Put (well)
- Scheiding (separation)
- Straatmeubilair (street furniture)
- Spoorrail (railroad)

In the object class on topographical elements we can see very well that IMGeo is mainly based on the municipal registrations, on what they want to register. Top10NL has a more regional character, which can be seen through attribute values like high-voltage lines, which can cross through whole municipalities without a starting point and an ending point, but which have importance on a smaller scale.

2.6.2 Geometry

Appendices C and D show respectively the geometry of all attribute values in IMGeo and Top10NL. The distinction between points, lines and polygons is made here. For every object it is judged whether it is allowed to appear as point, line, polygon or a combination of them. The advantage with these models is that they both have lots of polygons. For generalisation purposes in the tGAP structure this is necessary.

The similarities are for example shown in the object classes building and terrain. These objects can in both models only appear as polygons. Also the objects in the class 'RegistratiefGebied' are all polygons in IMGeo, in Top10NL they are allowed to be point objects for labelling purposes; these point objects are easy to interpret as belonging to a wider area object, although this might not be collected.

The differences show up in the object classes with roads, railroads, water and topographical elements. The object class 'Kunstwerk' in IMGeo is for simplicity reasons taken into account with the topographical elements.

The elements in the classes roads, railroads and water have some similarities. In IMGeo they are always polygon objects, because at this large scale there is no need to simplify them. In Top10NL it depends on the importance and the width of the road, railroad or water whether it is represented as a line or a polygon object. Roads with a width smaller than 2 meters are point- or line objects in Top10NL. Railroad objects are always point- or line objects. Road- and railroad objects can be point objects in case of for example a railway- or a gas station. Water is a line object if its width is less than 6 meters. In the current tGAP structure there is no solution for the conversion of polygon features to line features. This could fit in the tGAP structure and progress is being made to implement this. Within this research this hasn't been taken into account.

There is another striking difference in the way railroads are collected in both models. In IMGeo the rail itself is collected as a topographical element and the area on which the rail lies is collected as an object in the sub-object class 'Railroad part'. In Top10NL this situation is exactly reversed. The lesson that can be learned from this example is that even the smallest details of the collection of objects can differ between models. This problem now applies when comparing Top10NL and IMGeo, both originating from NEN3610; the modelling of the models can really differ in every aspect. The geometrical appearance of lots of topographical elements also differs, mainly because the elements in the models itself differ a lot.

2.7 Related projects in The Netherlands

Outside Rotterdam also projects take place which focus on the same subject of generalising large scale topographical data. They don't take place in Rotterdam, but during this research a contribution has been given to and ideas could be obtained from some of these projects. It is worth mentioning them here, because they also give an indication of the relevance of this research within a broader perspective.

This section takes a look at three ongoing projects in the field of map generalisation in The Netherlands. The first two projects are executed by consortia, which will be introduced in this section. Subsection 2.7.1 is about the project IMTop, subsection 2.7.2 is about a part of the Ruimte voor Geo-informatie (RGI)-project DURP ondergronden. Subsection 2.7.3 takes a look at the project to insert GBKN buildings directly into the Top10NL.

2.7.1 IMTop

The project IMTop is a joint project by the Dutch Cadastre (Kadaster), ITC Enschede (International Institute for Geo-Information Science and Earth Observation) and TU Delft. Its aim is to combine the small scale datasets of the Kadaster through generalisation. All object classes should form at certain scale levels a logical and consistent set of topographical elements. The model therefore needs to know what classes need to be displayed at what scale and what level of detail is requested for an object class at a certain scale.

The scales that have to be modelled within IMTop are at least 1:10k, 1:25k, 1:50k, 1:100k, 1:250k, 1:500k and 1:1000k. These scales are necessary for the law on the authentic registrations, the scales 1:250k and 1:1000k are also needed for the European products respectively EuroRegionalMap and EuroGlobalmap.

The project has defined some requirements to which the model has to satisfy. A requirement of the Kadaster is that the model of Top10NL should be unchanged. There are also requirements with respect to the generalisation procedures. One of these is that IMTop should not only be suitable for the scales mentioned, but should have a possible vario-scale output for future products (Stoter et al., 2007).

The master thesis project of the author of this report is about generalisation of even larger scales than mentioned in this project. Because generalisation from a level 1:1,000 is even more interesting than when starting from 1:10,000 this master thesis project is seen as a useful addition to this generalisation project by this project group, because the scale 1:1,000 contains even more detailed geometry.

2.7.2 DURP Ondergronden

Another interesting development is to be found in the project DURP Ondergronden. This project is mainly about generalising topographical planning maps. It is executed by a large number of parties, which are:

- Bentley Systems Netherlands
- ESRI Nederland
- ITC Enschede
- Kadaster
- Landelijk Samenwerkingsverband GBKN (LSV-GBKN)
- NedGraphics
- Sense Organisatie & Coaching
- Technische Universiteit Delft

The objective of this research project is "to generate and use base maps for integrated querying of digital physical plans". Because the research was mainly focussing on maps of the Kadaster a subproject is defined in which the LSV-GBKN can participate. This subproject is about generating Top10NL from IMGeo, which is exactly the theme of the master thesis as well. This master thesis report therefore will be the starting point for this subproject.

A large portion of the work done for this project DURP Ondergronden can also be addressed to the RGI project 223 on Usable Mobile Maps (or MobiMaps).

2.7.3 GBKN buildings in Top10NL

In section 2.2 the BAG already was introduced. It was stated that the geometry of the buildings at scale 1:1,000 is needed, because this provides enough detail. For this reason the geometry of buildings in Top10NL is not accurate enough.

Since Top10NL is an authentic registration it preferably wants to have a connection to other authentic registrations like the BAG. Therefore currently it is being investigated whether it would be possible to put the GBKN buildings in Top10NL without any form of generalisation (Hidding and Uitermark, 2006).

Hampe states that intermediate scale layers will be necessary to have the correct amount of detail when the scale increases with a factor 2 to 4 (Hampe, 2003). The level of detail of GBKN buildings in the Top10NL is too much. If this project leads to implementation this has also direct consequences for this research. However, so far it is not taken into account.

2.8 Conclusions

This chapter introduced the situation in Rotterdam. It is shown that this is a dynamic situation in which several processes take place. The vision in Rotterdam is clearly to come to a situation in the future where data is collected once and used for many purposes, this is what the project 'Basisregistraties' is all about.

The progress of implementing IMGeo in Rotterdam is clearly there, but not enough in time to take place in this research. Nevertheless, Rotterdam is one of the municipalities that initiated IMGeo, this makes that the connection from Rotterdam to the subject remains. Pilot data from the municipality of Almere can be used instead and the results can be projected on the Rotterdam situation. For Top10NL we have seen that there is a necessity to implement this into the organisation of Gemeentewerken Rotterdam due to legal obligations in the near future.

To be able to use geographical data for many purposes generalisation is needed. Chapter 3 introduces the methodology to do so, chapter 4 and 5 describe how the models that were introduced in this chapter are used to build a way to reach this target of the project 'Basisregistraties'.

3. Research methodology

In this brief chapter the vision on the research methodology is explained. The chapter starts with an introduction to the constrained tGAP structure in section 3.1 and after that explains the research approach in section 3.2

3.1 The constrained tGAP structure

In a literature study (Hofman, 2007) Hofman arguments that for the generalisation proposed for this thesis work the tGAP structure is most suitable. The tGAP is fast, scale independent and stores the geometry of objects only once. These conditions make the tGAP very suitable. In the same literature study the proposal is done to work out the constrained tGAP structure (Haunert et al., 2007) with IMGeo as the basis and the Top10NL as constraint. This section will introduce this constrained tGAP structure.

The idea of the constrained tGAP is build upon the general idea of the tGAP structure (Van Oosterom, 1993 & 2005, Van Oosterom et al., 2006). The idea of the constrained tGAP tree is that the tGAP is not built from the largest scale alone, but that it is built between two map scales. Between these map scales the tGAP structure can than be built to show which objects are aggregated.

Figure 3.1 shows the schematic process of the tGAP structure. In the tGAP structure the geometry of features is first to be transformed into a topological model, which consists of nodes, edges and faces. With the tGAP algorithm first the least important face is selected and merged with one of its neighbours. From this a new face originates. When repeating this process a tree of merged and emerged faces can be made. The bottom of this tree shows all the objects at the largest (most detailed) scale; going to the top of this tree details are left out and at the top only one object is left. Every level of detail in-between these scales can be requested and be presented as shown in figure 3.1

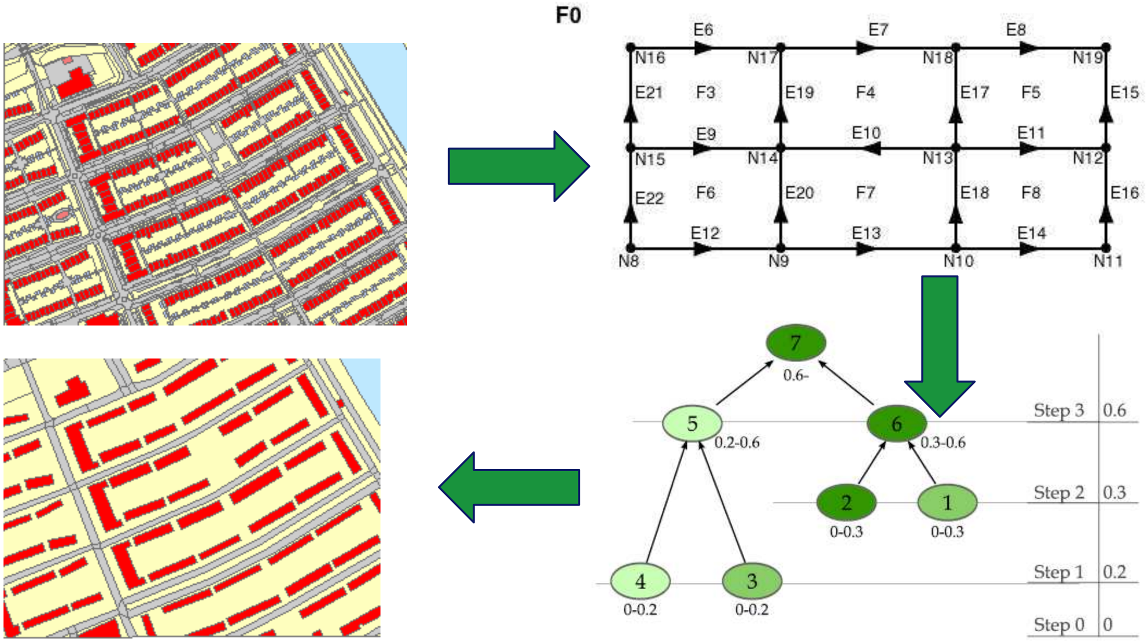


Figure 3.1: The process of the tGAP structure (figure courtesy lower right part: Haunert et al., 2007)

The constrained tGAP structure builds a collection of small tGAP trees; the merging of objects stops when the level of detail from the constraint has been reached. Figure 3.2 shows how this works for one part.

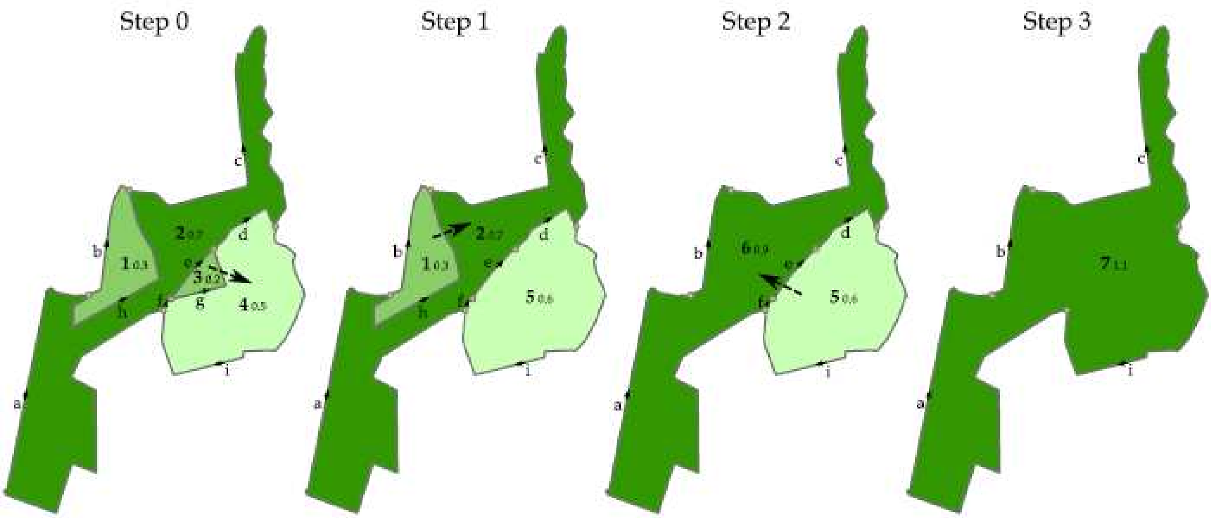


Figure 3.2: The build up of the constrained tGAP structure for a single region (figure courtesy: Haunert et al., 2007)

In step 0 of figure 3.2 the least important feature 3 is selected as the least important feature. Feature 4 is selected as the most compatible neighbour. These two features are merged to form a new feature 5. Feature 3 and 4 are not erased from the tGAP, they get the importance_high value 1, which means that they were merged after step 1. Feature 5 gets the importance_low value 1, which means that it originated after step 1. Not the importance value is stored as importance_high (or imp_high) and importance_low (or imp_low), but the iteration step; this is easier to implement. This process repeats until there are no merge able objects anymore in the dataset. The attributes imp_high and imp_low tell at which step an object appears or disappears from the map. Therefore, by querying the database for a certain importance value, the map can be viewed at every requested level of detail.

Objects in a smaller scale appear as large objects when put in a large scale map. They can be seen as regions in that map. The purpose of this research is to assign objects in the large scale dataset to regions in the small scale dataset. Because the region also is an object, the region-object can be seen as the top object of a small tGAP-'branches' and the objects inside the tGAP can be forced to merge to the class of this top object.

If datasets are combined the geometry will not match everywhere. To show this statistics can be made from the overlay of the maps; they show where objects differ and which object combinations are aggregated more than others. With this information on aggregations the importance factors and class compatibility functions can be better determined; in the current tGAP they have to be estimated.

Class weights
In the tGAP structure the weights of classes are hard to determine. The importance of an object is currently being calculated according to equation 3.1.

$$importance(x) = area(x) \cdot classweight(x)$$

Equation 3.1: Equation of the importance function in the tGAP structure

In the current tGAP structure (Van Oosterom, 2005) the class weights are all set to one, which means that the importance is only determined by the area size. When inserting a constraint the structure knows what the final top object of the tGAP branch has to be. With information of the output we improve the class weights for at least this specific case with IMGeo and Top10NL. Van Putten (1998) also investigated proper class weights. Future research will nevertheless be necessary to determine whether these class weights are also of use for other datasets.

Compatibility functions
For the compatibility functions the tGAP structure uses a table with similarities. This table defines for all classes a certain cost for transition. For every possible transition class a 'cost' is to be determined, which is done in chapter 5.

$$MostCompNeighbour(y) = min_y(cost(class(x)–class(y)) \cdot area(y))$$

Equation 3.2: Equation of the compatibility function in the tGAP structure.

This cost will first be estimated for all possible combinations of object classes in the table compatibilites. If the results are not satisfactory, the values of this table might be changed by statistical and empirical analysis. Other compatibility functions also make use of the length of the common boundary, this is not done in this function.

Final goal
From the constrained tGAP structure general rules can be defined for the generalisation parameters in the tGAP structure. The related goal in this is to get to a situation in which the tGAP structure can be better built and maintained without constraints.

For the master thesis the 1:1,000 IMGeo map and the 1:10,000 Top10NL are used to build this constrained tGAP. IMGeo is in this case the basis and Top10NL forms the initial constraint. The representation resulting from the tGAP at 1:10,000 should than be acceptable with respect to the current 1:10,000 map.

The full code of the constrained tGAP structure used in this research is provided in Appendix G.

3.2 Application of the constrained tGAP structure in this research

The approach of this research has already been explained in the introduction chapter. This section is a guide through the different steps that have been taken during the research.

Data preparation
The IMGeo- and Top10NL datasets that were used for this research needed to be prepared to go into the tGAP structure. Taking into account the specifications of IMGeo and Top10NL it was supposed that it wouldn't take to much effort to get to an area partition, which is necessary for the tGAP structure. This hasn't been the case. Chapter 4 shows what operations were necessary to come to a good basic dataset.

Assign a region to an IMGeo object
A region needs to be assigned to an IMGeo object, because the data preparation and the comparison of the two models showed that there were a lot of geometrical differences.
It was supposed that the assigning of a region could be done by making a model in ArcGIS. Part of the joining of the objects was done in an ArcGIS model, but because this program offered not enough possibilities the programming language Python was used further. With two scripts the IMGeo objects were assigned to a final region. This is all to be read in chapter 5 in which four methods are described giving different results in the generalisation structure.

Assigning class weights and creating the compatibility matrix
The first results of the tGAP tree are collected with estimates. After these results are compared to the existing Top10NL conclusions can be drawn with respect to values in these matrices which need to be changed. This is described in chapter 6. These matrices are implemented as Oracle tables.

Getting results using the constrained tGAP structure
The results using the constrained tGAP structure are described in sections 5.4 - 5.7. Using a PL/SQL script the tGAP-branches are built. In chapter 6 the results are interpreted for the assigning of better class weights and compatibility values. This process endures until the results are satisfying. Finally the methods are tested on a larger dataset.

4. Description of the Almere testdatasets

Chapter 4 introduces the test data from the municipality of Almere, as presented in both the IMGeo- and the Top10NL model. This test data will be used as input for the constrained tGAP structure.

In this chapter the datasets that are used for the generalisation are described. The data is all from a part of Almere which is shown in a perspective view in figure 4.1. The region indicated with a red rectangle is the area on which this research has focussed.

Figure 4.1: The test area and its region (figure courtesy: Google Earth)

For a first test only a small part of the indicated region is used. The specifications of the two datasets will be discussed in this chapter; it will turn out that not all content is according to the specifications of the datasets. Section 4.1 discusses the IMGeo test dataset of Almere, which was the result of the IMGeo pilot there; 4.2 is about the corresponding Top10NL test dataset. Sections 4.1 and 4.2 also show a part of the pre-processing, which has to be done for both models; section 4.3 describes other pre-processing that has to be done before entering the tGAP structure, 4.4 shows some statistics of the test area. The chapter will be closed with concluding remarks in section 4.5.

4.1 IMGeo dataset of Almere

As already mentioned in previous chapters, Almere was one of the three pilot areas in which IMGeo was tested. The results of this pilot are the input for this research. In this section the details of this dataset and the modifications that had to be made are discussed.

In figure 4.2 the part of the IMGeo test area is represented, which is used as input data for the constrained tGAP structure. Because no rules for visualisation are yet specified, the visualisation is the authors own reflection of the data.

This dataset is just a fraction of the area mentioned in figure 4.1. Its size is about 1 x 0,5 km. The choice for this small dataset has been made to not make the computations very large in intial testing. The map fragment with the black frame is enlarged and visualized in figure 4.3 at its normal scale 1:1,000 to give the reader a better impression of the details of IMGeo. In chapter 6 it is shown that the methods used are also tested on the larger area shown in figure 4.1.

Figure 4.2: Visualisation of the test area in IMGeo

Figure 4.3: Visualisation of a part of the test data at scale 1:1,000

Objects
The IMGeo objects available in this test area are not all objects which are mentioned in chapter 2 and Appendix C. Only a selection of these objects appears in this test area. Point- and line objects will not be taken into account in the research. The objects available are:

- Residence object
- Road
- Water
- Terrain
- Topographical elements

The object class terrain is a very general one. Therefore this class is further split based on the attribute value 'type terrain', these types are:

- Fallow land
- Lot
- Grass
- Plants
- Terrain (to be determined)

The object class topographical elements can be subdivided in two of its subclasses, they are:

- Other building
- Bin

This leads to the legend of figure 4.2 which is shown in figure 4.4.

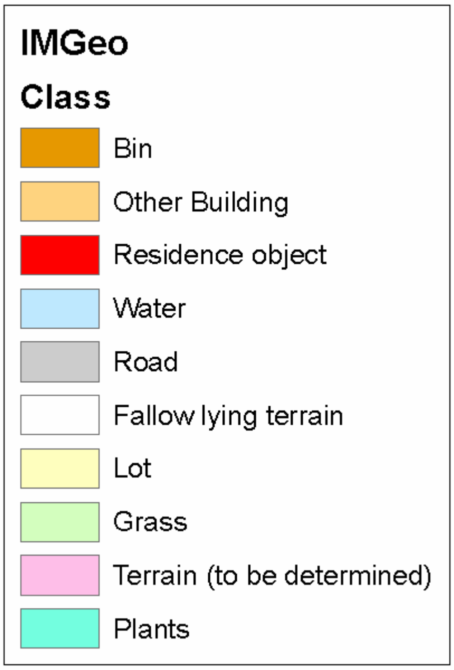


Figure 4.4: Legend of the IMGeo visualisation shown in figure 4.2

Pre-processing
To use the data in the tGAP structure the data needs to be an area partition, which means that there are no overlapping areas and no gaps in the dataset. This is not the case in the pilot IMGeo data of Almere; this data set consists of separate layers containing one object class, which overlap in several places. Therefore modifications to the data where necessary to make it a partition. Operations in ArcGIS were done to do this, these operations will be discussed.

In the final version of IMGeo it is stated that the polygon objects at level 0 have to form an area partition. Some objects are not part of the area partition at level 0, they are:

- Registration area
- Buildings (roof print)
- Civil works
- Other objects

Registration areas are not part of the partition at level 0. There is no decision yet about the level they will be at, but probably this will be about level -10.

The footprint of buildings is part of the partition in IMGeo. However, the roof prints of buildings, which are collected in Top10NL, are not. Therefore, if for tGAP generalisation the roof prints of buildings have to be taken into account, a new partition has to be made. The roof print of buildings will standard get about level 10. In the test dataset of the municipality of Almere these roof prints are not available.

Civil works are normally placed at the level 1. If for example a bridge is going over the water, then the water has level 0, the bridge has level 1 and the road on the bridge has level 2. For all other objects the same holds as for civil works; if the object is no part of level 0, then the object is no part of the area partition as well.

The above given examples are not provided in this test dataset; only a small amount of objects is provided and all objects got level 0. The following rules were applied when making the IMGeo test data an area partition. The hierarchy is in short:

- If ‘Terrain’ and another object overlap, the terrain part is erased.
- If ‘Water’ and ‘Road’ overlap, the water part is erased.
- If ‘Road’ and ‘Other building’ overlap, the road part is erased.

The hierarchy is mainly based on the fact that a map user gets a view from above; this is basically the idea of this hierarchy. A building stands on the terrain, a road lies over a canal (exception: aqauduct), etc.

The details of these actions are explained with use of figure 4.5 and 4.6.

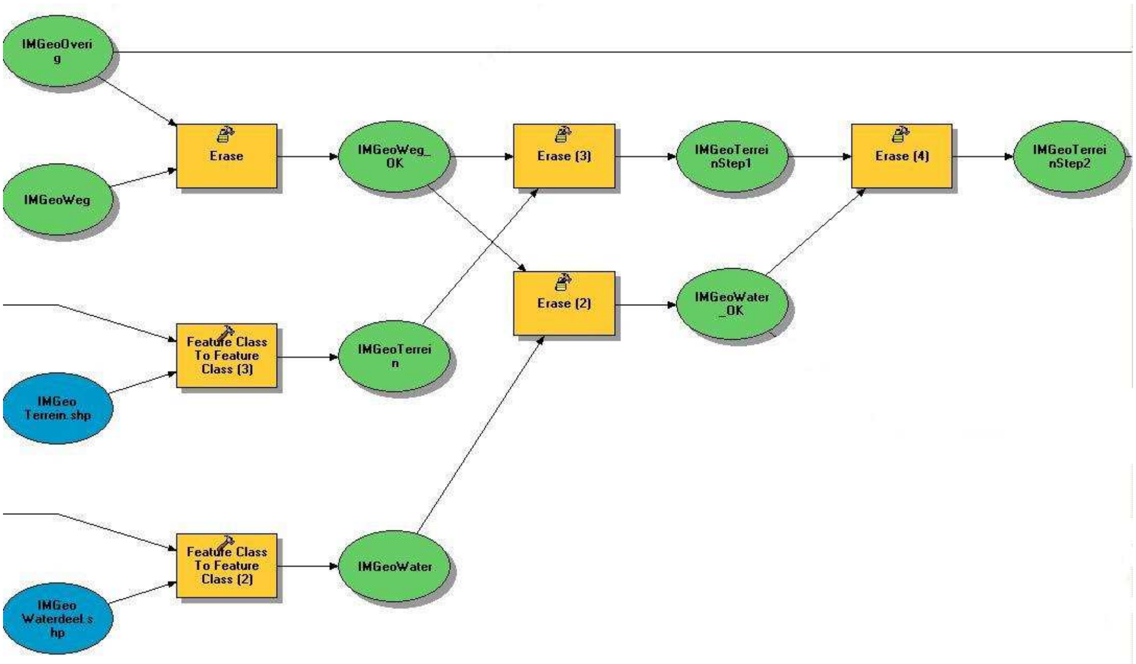


Figure 4.5: The first four erase actions in the ArcGIS model

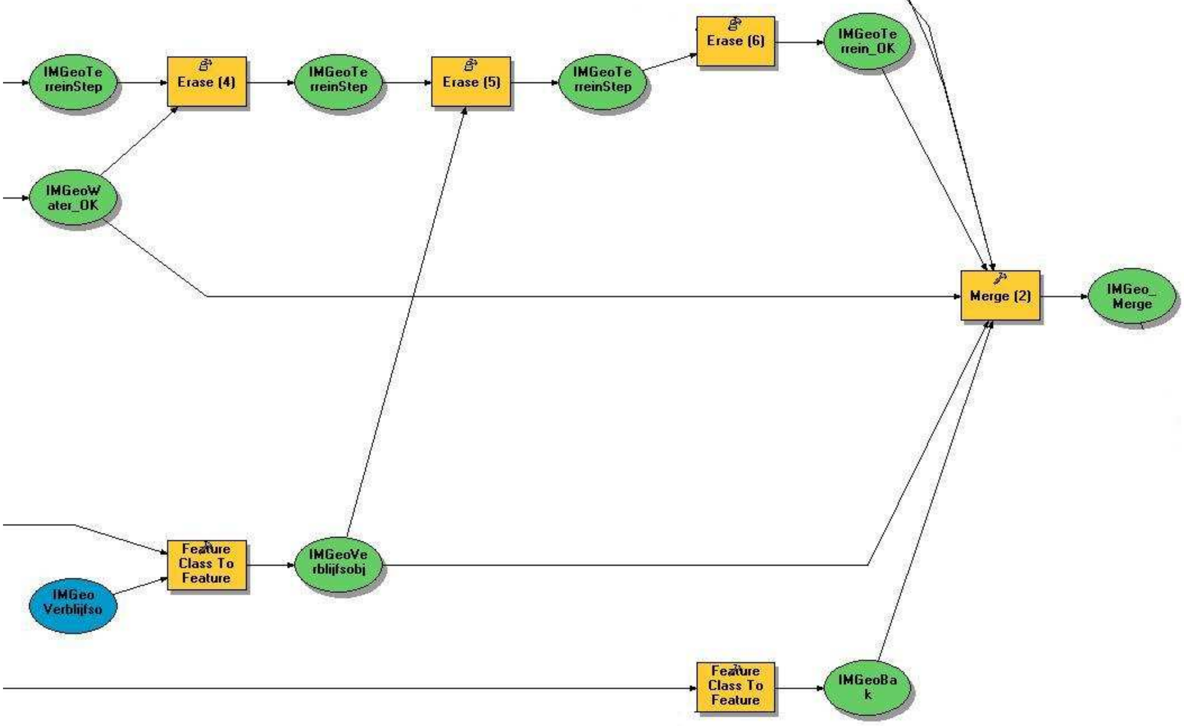


Figure 4.6: The other two erase actions and the merge operation in the ArcGIS model

The operations which add a feature class to a geodatabase do not need discussion, they are the initial load of a feature class to the geodatabase. The other operations in the figures 4.5 and 4.6 and the hierarchy they imply will be discussed beneath:

- Erase: Road objects are erased under the other building objects. This is an exceptional overlap; it only appears 2 times in the test data set.
- Erase (2): Water objects are erased under road objects. This means that bridges get priority above the water underneath it.
- Erase (3): Terrain objects are erased under road objects. The road is lying on the terrain.
- Erase (4): Water objects are erased under terrain objects. This is the only position where terrain is put higher in the hierarchy, because it is supposed that the water runs under the terrain here. This is a very rare occasion as well. It only happens near bridges.
- Erase (5): Terrain objects are erased under residence objects. Residence objects get priority here, because they lie on the terrain.
- Erase (6): Terrain objects are erased under other building objects. Other building objects get priority, because they lie on the terrain.
- Merge (2): The merge operation merges all the processed IMGeo shapefiles to one shapefile.

What was also discovered was that about 100 objects existed double in the database with other ID's. These are errors in the data due to the fact that no relative height levels were used. The double objects were for the largest part small area objects on crossings. To still make the area a partition one of these doubled objects was removed. To check whether the area really was a partition after all these actions the ArcGIS topology check was done. This operation checks for overlapping areas. Since most municipalities that want to work with IMGeo still have to do the conversion, this is one of the things that needs to be checked. For Almere it appeared that the data didn't apply to all IMGeo regulations at once.

4.2 Top10NL dataset of Almere

The Top10NL dataset of the same part of Almere as in figure 4.3 is shown in figure 4.7.

Figure 4.7: Visualisation of the test area in Top10NL

Like in section 4.1 with IMGeo the details of the Top10NL dataset are discussed.

Objects
As in the IMGeo case the set of objects is not the possible set of classes from the model as mentioned in chapter 2 and Appendix D. This is just because most objects don't exist in this area.

The object classes that are available in the test area are:
- Building
- Water
- Road
- Terrain

Also here the object class terrain is split into several terrain types based on the attribute values of 'type terrain', they are:
- Grassland
- Wood
- Other terrain

This leads to the legend of figure 4.7 which is shown in figure 4.8.

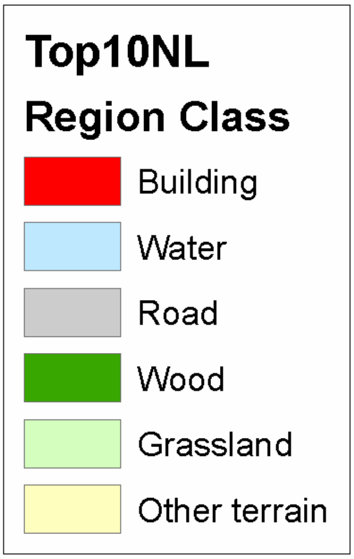


Figure 4.8: Legend of the Top10NL visualisation shown in figure 4.7.

Area Partition
The objects in Top10NL are not an area partition. The difference with the previously treated IMGeo is that Top10NL is not even meant to be an area partition; the different object classes are separated layers which can overlap. For the tGAP structure however this is required. To make this dataset an area partition, operations in ArcGIS were done.

The above described situation leads to a serious problem. We want to generate Top10NL data from IMGeo data, but through the tGAP structure we will never be able to return to a dataset which is not an area partition. The Kadaster will never accept this as a good replacement for the current Top10NL. The product specifications of theTop10NL might be changed to come to a situation in which the tGAP can produce the Top10NL.

The following rules were applied when making the Top10NL test data an area partition

- If 'Terrain' and another object overlap, the terrain part is erased.
- If 'Water' and 'Road' overlap, the water part is erased.

The details of these rules are explained with use of figure 4.9

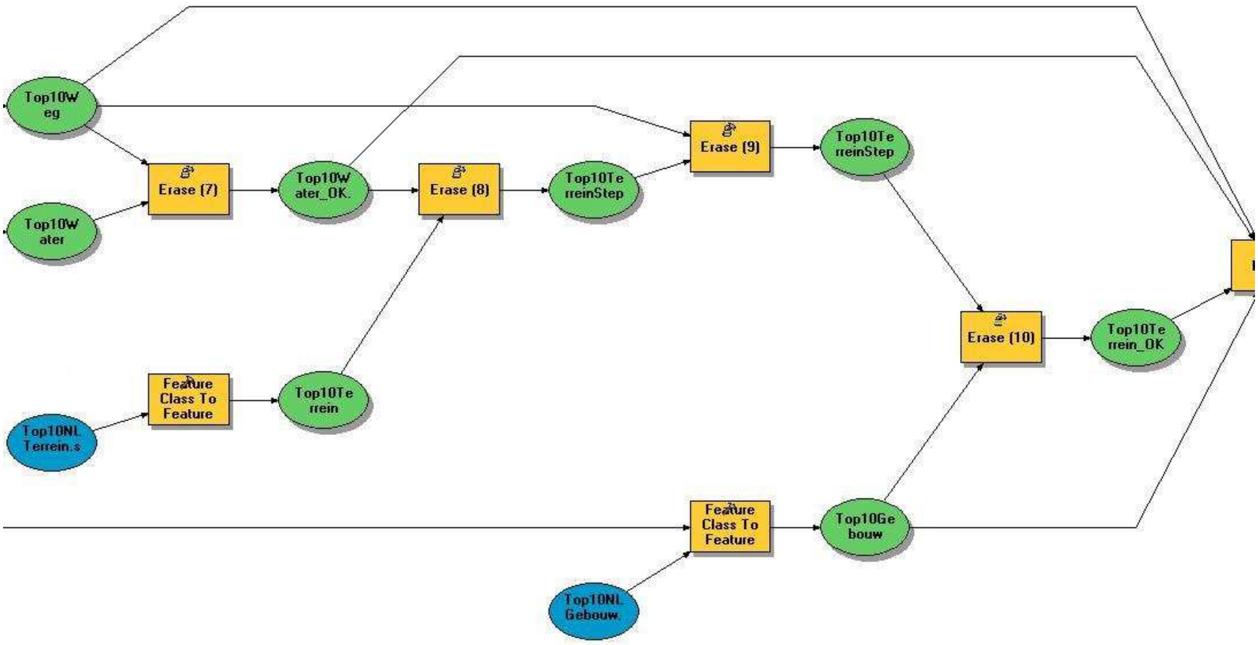


Figure 4.9: Erase operations in Top10NL in ArcGIS Modelbuilder

- Erase (7): Water objects are erased under road objects. This means that bridges get priority above the water underneath it.
- Erase (8): Water objects are erased under terrain objects. This is the only position where terrain is put higher in the hierarchy, because it is supposed that the water runs under the terrain here. This is a very rare occasion as well. It only happens near bridges.
- Erase (9): Terrain objects are erased under road objects. The road is lying on the terrain.
- Erase (10): Terrain objects are erased under building objects. Building objects get priority here, because they lie on the terrain.
- Merge: The merge operation merges all the processed Top10NL shapefiles to one shapefile.

4.3 Further pre-processing of the test datasets

Class names
Before being merged the shapefiles need a class name. This seems obvious, but because of the fact that the shapefiles are separated files, the mentioning of a class, which is necessary in the tGAP structure, has to be added. This also makes that the objects are distinguishable when they are merged.

Object ID's

With every operation in ArcGIS the feature ID's of the new shapefile change. For this reason we need stable identifiers. In the IMGeo model this could be the GML-ID or the 'Identificatie'. Because an identifier for residence objects was not yet implemented for the test area, this part of the column was empty and therefore this column was useless as identifier. The column with GML-ID's was not filled in the test data set. Normally the value of this attribute would serve well as an identifier. Because working without an identifying attribute is not an option, the feature ID's of the merged IMGeo objects were manually copied to the column GML-ID and this column is during the rest of the operations used as identifier.

In the Top10NL model there is the identifying column 'IDENT'. These identifiers have the form NL.TOP10NL.(9 digits). Because this identifier is used as input for a loop in the further processing in chapter 5 and files are given names with this identifier in it, it was not good to have identifiers with dots in it. Because the first part doesn't really identify the objects, except for mentioning that it is a Top10NL object, the object identifier for Top10NL has become a column named Top10ID, which only has the last 9 digits of the former identifying code. Within this research the first part of the identifier wasn't needed; in other cases of course this can be necessary.

Object ID's are known to be unique. However, there are some situations in the test data set where the object ID's of the Top10NL dataset are not unique, this is of course not allowed to happen. One of these situations is at the only bridge in the test data set as is shown in figure 4.10. The reason why these situations appear is because of the relative height. There is another road going underneath the bridge which has the relative height -1. Because we want to end the pre-processing with an area partition all the objects with a relative height ≠ 0 have to be removed. What remains is 3 objects with the same ID, as can be seen in figure 4.10; the parts that cross the intersecting road are separate objects. These remaining objects are manually merged with the ArcGIS operator 'dissolve' to solve this problem.

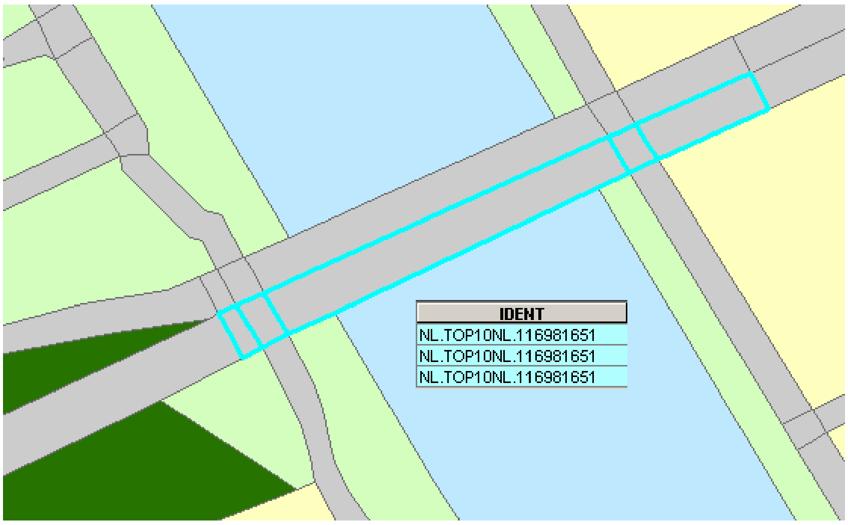


Figure 4.10: 'Non-unique' Top10ID's in the test data set

4.4 Geometrical and semantical differences between the models

The statistical findings in this section are based on the statistics of Appendix E, which show statistics of overlapping objects between the two datasets of IMGeo and Top10NL. If anywhere in this section is referred to statistics, it is referred to that appendix.

Buildings
The buildings in Top10NL should form an overlay with the buildings in IMGeo. The buildings should be generalised versions of the ones in IMGeo, but in lots of them there is a shift in the geometry. As shown in figure 4.11 the buildings look to be irregularly shifted compared to IMGeo. This can be due to the projection of the aerial photographs, but the deviations seem to be too large to attribute this to causes like parallax. It seems to be more appropriate that these deviations are the standard inaccuracy of the Top10NL, which can be up to 4 meters (Bakker et al., 2005). What can also be the case is that the irregularities have to do with the choice of the cartographers, because different cartographers make different choices and that's why we see irregularities. Looking at the corners of the blue (Top10NL) building blocks, we see that some buildings in the underlying (red) IMGeo model are hardly overlapped with the Top10NL buidling objects.

For this reason the buildings can't be connected one-to-one and smart rules need to be defined to connect them. Chapter 5 describes these smart rules.

Also a building generalisation methods were investigated during the research to replace the Top10NL buildings provided by the Kadaster. The generalisation method of Damen (Damen, 2008; to be published) is investigated for this purpose. This method generalises individual buildings to building blocks using the Minkovski sum. This method hasn't been further investigated within this research, it is only referred to here.

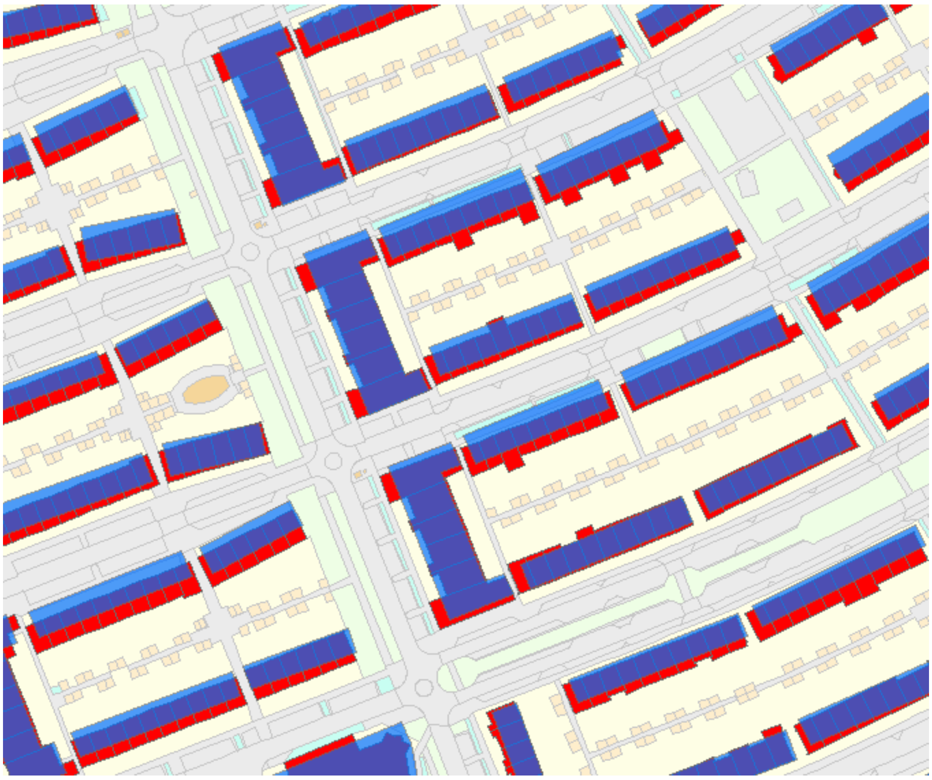


Figure 4.11: Overlay of Top10NL buildings (in blue) iwith IMGeo buildings (red).

Another option is to put the GBKN buildings in the Top10NL without generalisation (Hidding and Uitermark, 2006). One of the major problems in this is that the GBKN buildings should not overlap road objects in Top10NL. In our test area only two small pieces of residence objects intersect the road in Top10NL. They have a total area of 13 m^2, as can be seen in the statistics. Also this option is not further investigated in this research, but it is well possible that this will happen in the near future. For more information on the project 'GBKN gebouwen in Top10NL' the reader is referred to subsection 2.7.3.

Roads
The IMGeo pilot in Almere is done to see whether it was possible for a municipality to convert its current GBKN registration to IMGeo. For the municipality of Almere it has been very hard to define the object 'Road'. In the current registration only the attribute 'pavement' is known; this can be roads, playground, sidewalks etc. In the IMGeo test dataset all 'pavement' in the Almere registration has been converted to road objects. This has also to do with what the municipal administrators of the municipality want to be collected in IMGeo. These differences in the collection rules lead to inconsistencies with the Top10NL dataset. Figure 4.12 shows what difference it makes when pavement is added to the IMGeo object class road.

It is right that the sidewalks are classified in IMGeo as roads. In Top10NL this subdivision is not made. The road is drawn with a certain standard deviation by which the sidewalks are sometimes taken into account and mostly not. The statistics confirm this view. The total road area in Top10NL is about 60% of what it is in IMGeo. For this reason only 51% of the area which is classified as road in IMGeo is also road in Top10NL.

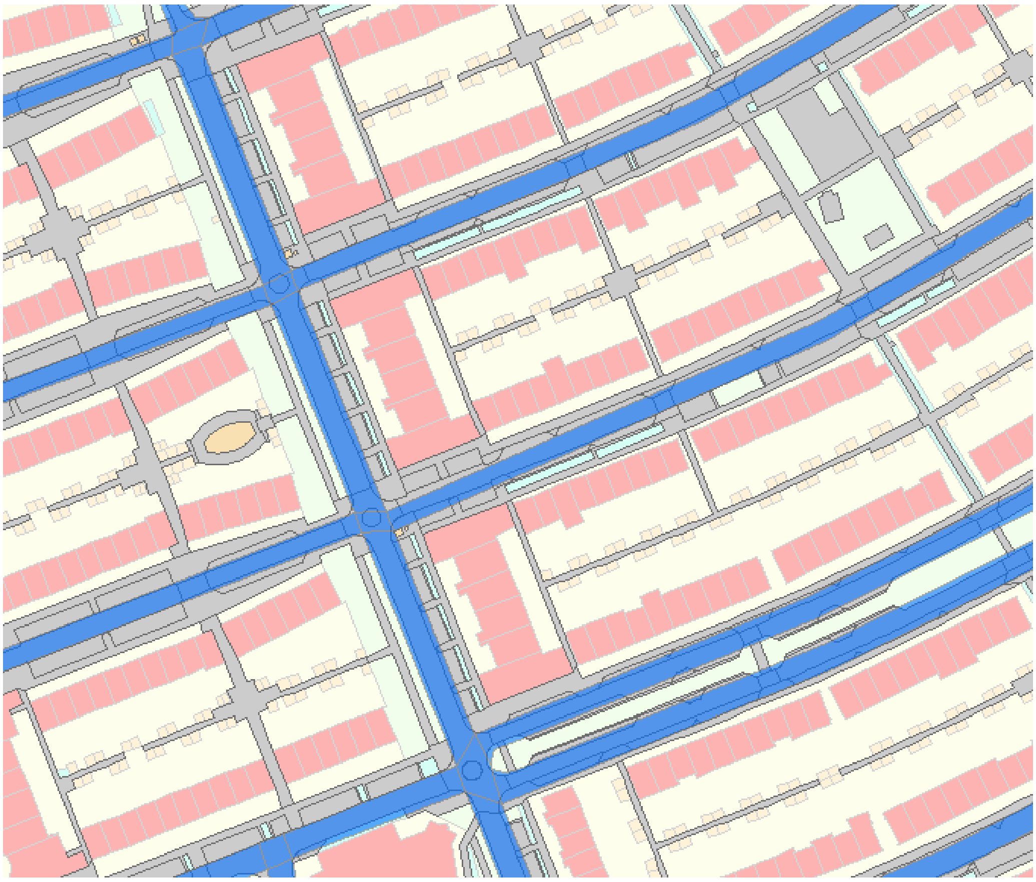


Figure 4.12: IMGeo roads (grey) overlaid with roads from Top10NL (in blue).

Semantics
Besides the geometrical differences which can easily lead to semantical differences, there are also objects in the models which are classified differently. In the test data there is a number of objects classified as 'Plants' in the IMGeo model whereas it is 'wood' in the Top10NL model. Nevertheless, the comparison of objects that is made in table 4.5 can be stated as the leading semantical model. In chapter 5 this table is to be extended with the values for the compatibility function; this is a matrix with values for all possible transitions between classes in the different models.

IMGeo	Top10NL
Residence object	Building
Road	Road
Water	Water
Fallow land	Other terrain
Lot	Other terrain
Grass	Grassland
Plants	Other terrain / Wood
Terrain (to be determined)	Other terrain
Bin	Other terrain
Other Building	Building / Other terrain

Table 4.5: Semantical comparison of IMGeo and Top10NL test data

The semantical comparison is based on a combination of what is to be expected looking at the object classes and what is in the dataset. For example the IMGeo objects 'Other Building' are mostly small sheds behind the residences. Semantically they can be classified as buildings, but because they are almost all classified as terrain in Top10NL the choice is made to mention them under 'other terrain' in table 4.5 as well.
These semantical considerations are all to be translated into compatibility functions. To make the model more readable to the computer, the class names are transposed to codes by the author. Table 4.6 shows which codes are used for the different objects.

Class	Code
Residence object / Building	1001
Other Building	5003
Road	2001
Water	3001
Lot	4001
Fallow land	4002
Plants	4003
Terrain (to be determined)	4004
Grass / Grassland	4005
Wood	4006
Other terrain	4007
Bin	5001

Table 4.6: Classes with their codes

Object classification hierarchy
We would like a structuring of objects in IMGeo and Top10NL which indicates a hierarchy. IMGeo object classes would be a specialisation of the Top10NL object classes and vice versa Top10NL object classes would be a generalisation of the IMGeo object classes. Table 4.5 gives an indication of a proposed hierarchy which is slightly elaborated in table 4.6.

A complete hierarchy of classes will be hard to make with models that are not initiated as connecting models (IMGeo, 2007). Further research and cooperation is needed to determine what hierarchy could be made between the models.

Additional Top10NL features
Some objects in Top10NL contain more information than their IMGeo equivalents. This seems strange, because Top10NL is the smaller scale and should be less detailed. In fact, in for example road classification Top10NL is more detailed than IMGeo. Another example is that Top10NL uses also centerlines of roads and canals in case they have to be collected as line objects, whereas they are area objects in IMGeo. The centerlines could be useful additional information for the generalisation process.

These forms of additional information could be an enrichment for the IMGeo model. However, because the geometrical accuracy of these additional Top10NL objects is too poor compared to the IMGeo model, the choice has been made not to push down objects or attributes from the Top10NL model to the IMGeo model at this moment.

4.5 Conclusions

This chapter introduced the two test datasets of Almere. We can conclude that although these models originate from the same source, NEN3610, they differ due to their different origins. These differences lead to a lot of practical problems combining the datasets.

For IMGeo the conclusion can be drawn that a lot needs to be checked within the municipalities before really implementing it in the production systems. The test data of Almere shows a lot of things which are not according to what they should be with respect to the specifications of IMGeo mentioned in chapter 2. A lot of modifications had to be done to the test data to get it prepared for the tGAP structure. If the IMGeo model would have been an area partition at level 0 still modifications should have been necessary, because in that case the choice would have to be made whether the objects at level 0 are the objects required for a topographical map at smaller scales.

For Top10NL the conclusion is that it will be impossible to get to a situation in which we can extract exact Top10NL data using the tGAP structure. This is because the Top10NL model is not an area partition and the result of a tGAP generalisation always will be an area partition. The Top10NL model as an authentic registration will not be changed in the next few years, but looking at developments like IMTop a harmonisation of models is likely to be possible. With a working generalisation structure proposals to change either the IMGeo model or the Top10NL model might have a good chance of success.

As for the test datasets: 21% of all objects are classified different in the other model. For this reason we can’t just copy the object classes of the Top10NL to the IMGeo model.
A lot of pre-processing still has to be done in order to assign the right Top10NL ID’s to the IMGeo ID’s. Chapter 5 will explain what has to be done to get to a solution for these problems.

5. Generalisation: design and implementation

The pre-processing of the test data used for this research has been described in chapter 4. In this chapter the methods, the design and the implementation of the constrained tGAP structure are elaborated. Section 5.1 describes the software and the programming languages that are used. Section 5.2 introduces the class weights and compatibility values that were used in the first tests with the constrained tGAP. Section 5.3 describes the conversions done in FME. In sections 5.4 – 5.7 four methods to assign IMGeo objects to Top10NL regions are described and executed. These 4 methods are:

- **Simple overlay method (5.4)**
  An intersection between the models where every IMGeo object is split at the borders of the overlapping Top10NL object. In the end result only Top10NL geometry will be visible.
- **The maximum area method (5.5)**
  The Top10NL object which overlaps the IMGeo object the most is the shape to which the whole IMGeo object is assigned to. The IMGeo geometry is kept in this method.
- **The 35%-split method (5.6)**
  If an IMGeo object belongs for more than 35 % to two Top10NL objects we consider this Top10NL geometry as enrichment of the structure; therefore the IMGeo object is split and a new IMGeo object is created. For all IMGeo objects that don't have two Top10NL objects overlapping for more than 35% the maximum area method is applied
- **The building first method (5.7)**
  This method assigns IMGeo-buildings to a building region in case of some overlap with a Top10NL building without considering the amount of overlap. The other IMGeo objects are selected as in the maximum area method.

These methods are all tested with the weights and compatibility values of section 5.2 and they are evaluated in section 5.8. Here the decision to continue with one of the methods will be argumented.

5.1 Software and programming languages

The software and programming languages that are used in this research already showed up in the opening paragraph of this chapter. In chapter 4 the reader already saw ESRI ArcGIS being mentioned. In this section the programs and languages used in this phase of the research are explained. The programs that are mentioned are ArcGIS, IDLE, FME and SQL Developer. The languages mentioned are Python and PL/SQL.

5.1.1 Software programs

ArcGIS
ArcGIS is a software package produced by ESRI. The version used for this research is ArcGIS 9.2. In ArcGIS shapefiles and geodatabases can be produced and modified. The shapefiles contain the geometry and the attributes of features. For large and repeated processes ArcGIS Modelbuilder is used. The features are stored in a geodatabase and the processes

which have to be done as pre-processing are put in a sequence. Because the possibilities of the ArcGIS Modelbuilder were not sufficient, because looping was not possible, the model of ArcGIS Modelbuilder was exported to Python script.

IDLE
Python's Integrated DeveLopment Environment (IDLE) is the environment where Python scripts can be executed without compiling, because Python is an interpreted language.

FME
The Canadian software producer Safe Software makes the program Feature Manipulation Editor (FME). In this research this program is used to translate IMGeo's GML files to ESRI Shapefiles and to translate ESRI Shapefiles to Oracle Spatial tables and vice versa. The tables in Oracle Spatial should give a representation of the topology of the area. Therefore the topology builder in FME is used.

SQL Developer
This program is used to write, compile and execute PL/SQL code in. It is used as an entrance to the Oracle Spatial Database in which the topological data, produced by FME, is stored.

5.1.2 Programming languages

Python
The history of the name Python lies in the British series of 'Monty Python's flying circus'. Python is an interpreted, interactive and object-oriented programming language. For this reason it is also very good to implement all kind of ArcGIS methods in. The ArcGIS methods can be called in Python through the geoprocessing module. How to implement ArcGIS scripts in Python is described in the manual on writing geoprocessing scripts (ESRI, 2004). For more info on Python the reader is referred to (Downey, 2007).

Oracle PL/SQL
PL/SQL stands for Procedural Language / SQL, it is a language that extends the normal Structured Query Language (SQL) with the fact that procedures are called and that blocks are nested into each other. This makes PL/SQL more powerful than SQL. The structure of PL/SQL is a block, which performs a logical action. It starts with a 'declare' section in which the variables are declared, after this there is a section with executables under 'begin'. Finally exceptions can be made under 'exception'; the blocks end with 'end'.

5.2 Assigning class weights and creating the compatibility matrix

For both the assigning of class weights and the creation of the compatibility matrix first of all assumed values have been used. These assumptions have been altered after the tests with the constrained tGAP structure.

Class weights
The first proposal for the weight values are presented in table 5.1. These values basically are a first estimate based on common sense and previous findings (Van Putten en Van Oosterom(2), 2000). After the best method choice at the end of this chapter these values will be altered in chapter 6.

Class	Code	Weight
Residence object / Building	1001	0,9
Other Building	5003	0,4
Road	2001	0,6
Water	3001	0,5
Lot	4001	0,3
Fallow land	4002	0,1
Plants	4003	0,3
Terrain (to be determined)	4004	0,1
Grass / Grassland	4005	0,3
Bin	5001	0,1

Table 5.1: IMGeo classes with their proposed importance values

Compatibility functions
The compatibility values are shown in table 5.2. The compatibility values are actually computed the other way around as the title suggests. The minimum cost to merge the least important object with one of its neighbours is computed in this function. Table 5.2 shows for every class 1 what the transition costs are for this object to be absorbed by its possible neighbours (class 2).

	Class 1→	**1001**	**5003**	**2001**	**3001**	**4001**	**4002**	**4003**	**4004**	**4005**	**5001**
Class 2 ↓	***Cost***										
1001		0	1	50	100	1	10	50	50	100	100
5003		1	0	50	100	50	10	50	50	100	100
2001		50	50	0	100	50	50	5	10	50	5
3001		100	100	100	0	100	100	100	100	100	100
4001		10	5	5	100	0	10	10	10	20	10
4002		10	5	10	100	5	0	10	10	50	20
4003		50	50	50	100	20	50	0	5	10	10
4004		10	5	50	100	20	10	100	0	50	20
4005		50	50	10	100	20	50	5	5	0	5
5001		100	100	100	100	100	100	100	100	100	0

Table 5.2: IMGeo compatibilities table with the transition costs

For table 5.2 the situation is the same as it is with table 5.1; the values are estimates and after the first computations they can be altered according to the findings of these computations. These initial values are determined by comparing the classes in this research to the work of Jan Haunert on ATKIS data (Haunert et al., 2007).

5.3 Translation of the test data to Oracle tables in FME Workbench

This section shows the conversions made in FME. The FME schema's are based on the work of Jan Haunert (Haunert et al., 2007). Before doing the PL/SQL computations on the topological tables in Oracle Spatial the input shapefiles need to be transformed from a geometrical model into a topological model and translated to Oracle Spatial tables. The FME Workbench which is shown in figure 5.1 performs both these actions. In the topology builder the shape is transformed from geometry to topology and after that the results are written to tables in Oracle Spatial.

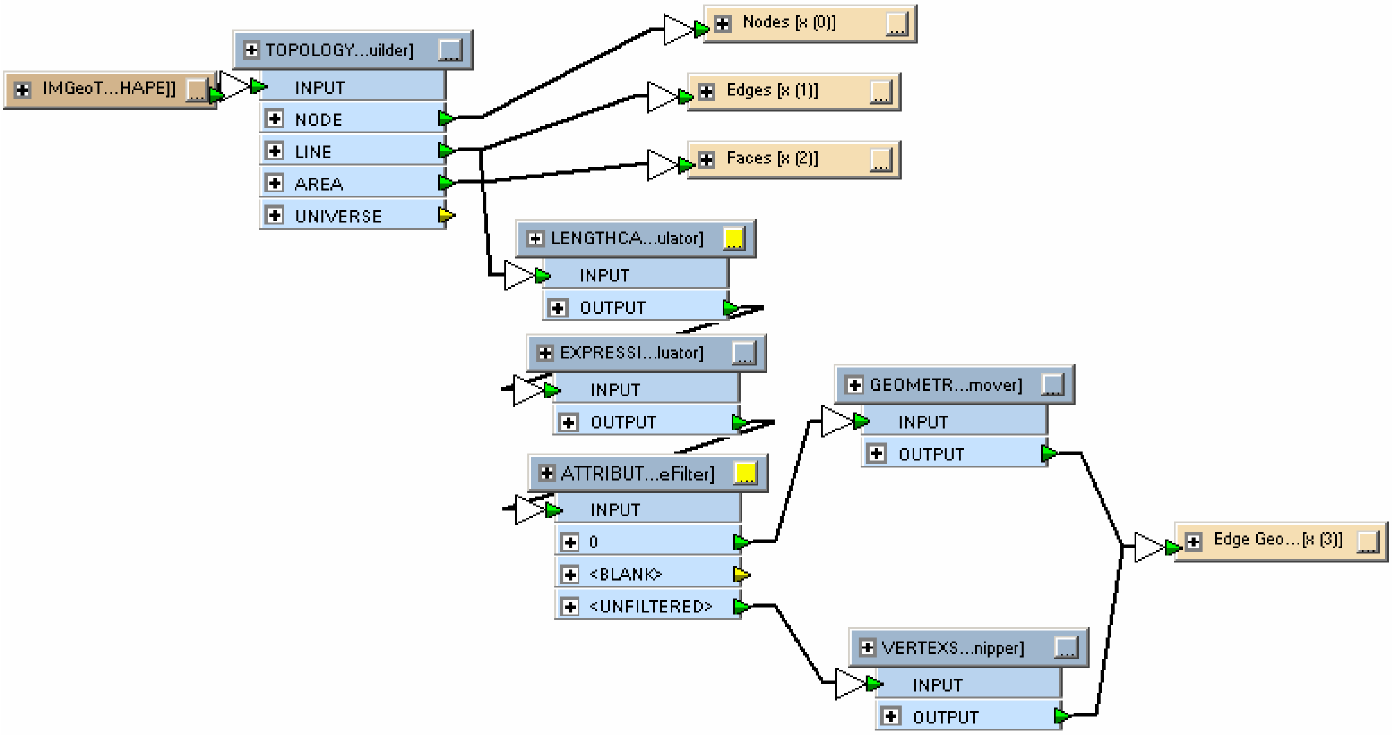


Figure 5.1: FME translation schema for translation to topological tables in Oracle Spatial

The topology builder converts the polygons from the shapefile into nodes, edges and faces. The nodes contain the geometry, the edges contain a reference to the face left and right of them and the faces contain all the attributes the shape had.

The edges take a special position, they are stored in two different tables; one with and one without geometry. In the table with edge geometry, which is right beneath in figure 5.4, a reference is given to the start and the end node of the edge, with this there is enough geometrical information to build the topological model. If the edge is no direct line from one node to another node, also the intermediate points are stored in the edge geometry table.

We now know that the geometry of the topological model is in the nodes, the edges are referenced by a start node and an end node in the edge geometry table and the faces are referenced, because the left- and right face of each edge is stored in the edge table.

The translation from Oracle Spatial tables back to ESRI shapefiles looks more complex, but actually does the same work backwards. Extra complexity is also added to this schema, because it can output shapefiles with all the different importance values the tGAP-tree has. Figure 5.2 shows this FME translation schema.

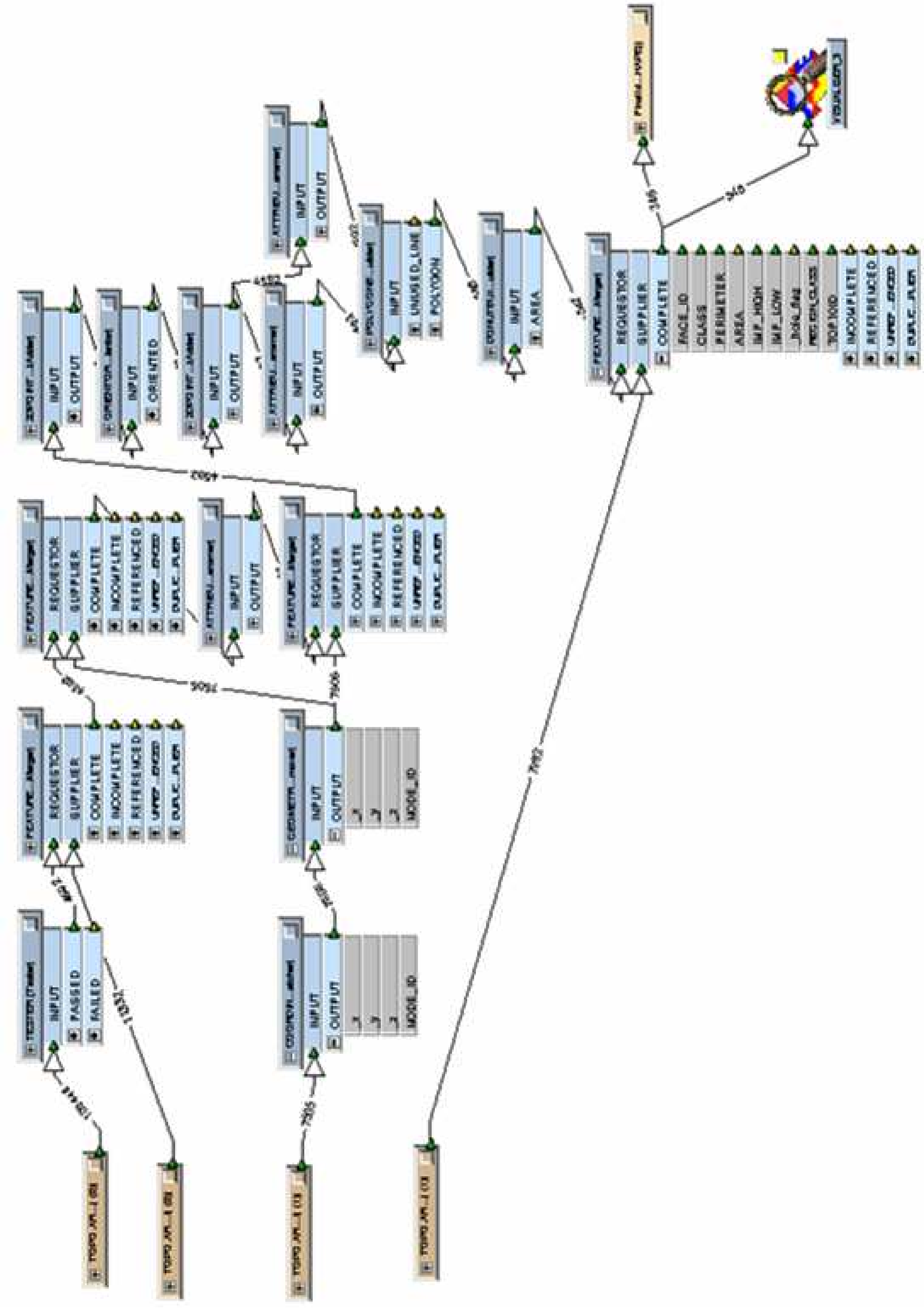


Figure 5.2: FME translation schema for the translation of Oracle tables to shapefiles

5.4 The simple overlay method

The simple overlay method takes the end products of chapter 4 as starting point. The merged IMGeo and the merged Top10NL files are intersected in ArcGIS, this is a polygon intersection. All IMGeo objects are split in case of an intersection with a Top10NL object boundary. The split IMGeo objects are then assigned to the corresponding Top10NL region. This method gives a very smooth zoom from IMGeo objects to Top10NL objects, but it doesn't keep the geometry of the IMGeo objects.

Due to the intersection operation some IMGeo objects might be split in separated sections, because several parts of objects might be assigned to the same Top10NL region. Figure 5.3 shows an example of this phenomenon. To solve this problem, these objects are separated with the ArcGIS 'Multipart to Singlepart', which creates new records for all separated objects. In total 817 new records had to be produced by this operation.

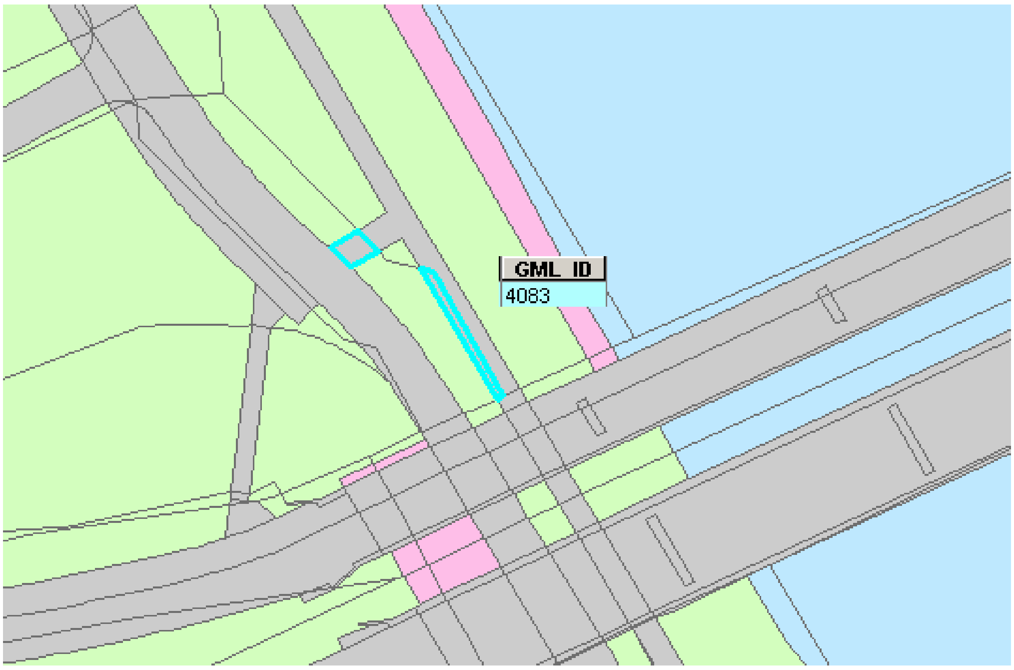


Figure 5.3: Two parts of the same road object in IMGeo belonging to the same Top10NL object; in the table they are one record.

With the resulting shapefile the translation in FME to Oracle tables is done. In SQL Developer the tGAP procedures were executed; once with and once without weights. Step by step features were merged; the situation after the last step gives a good indication of what happened with the objects during the generalisation and whether this is a desirable result. In total 7702 merges were executed before coming to the end result. Figure 5.4 gives the Top10NL as it should be. This can be compared to figure 5.5 shows the end result of the constrained tGAP generalisation without weights (or all weights equal to 1) and figure 5.6 shows the end result with the weights of table 5.1.

Figure 5.4: Visualisation of Top10NL

Figure 5.5: Visualisation of the end result using the simple overlay method without weights.

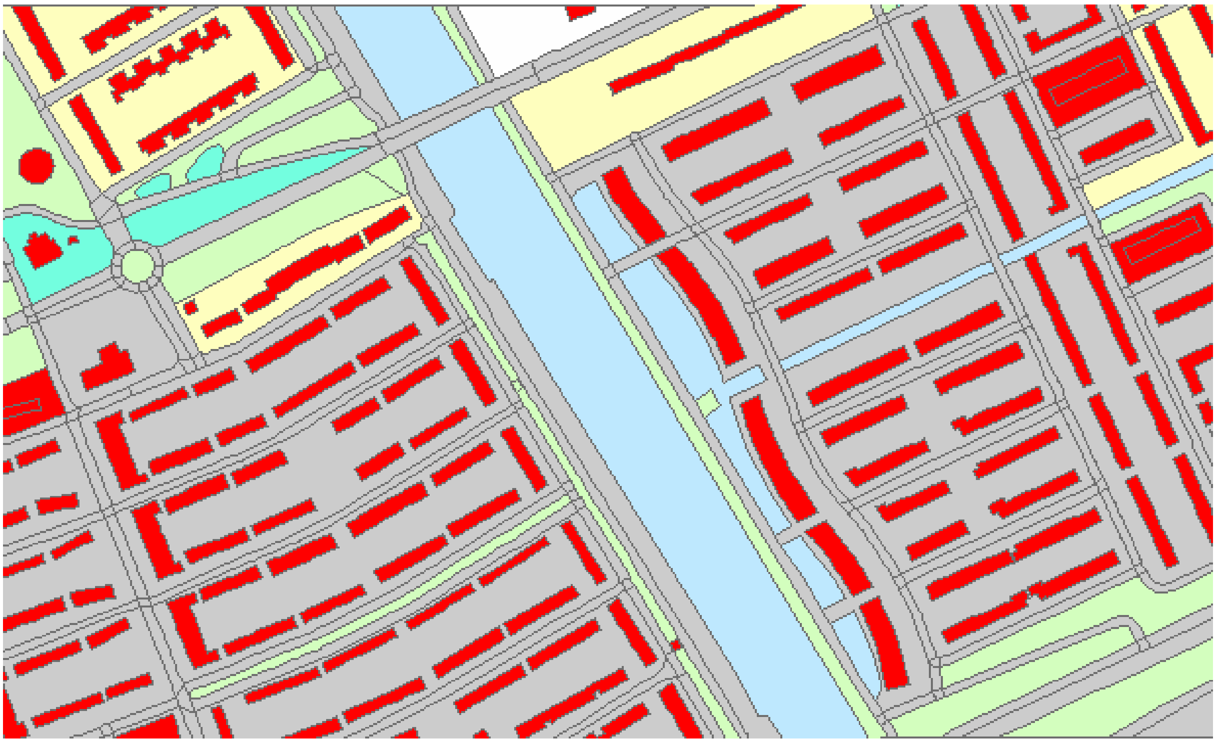


Figure 5.6: Visualisation of the end result using the simple overlay method with weights.

Comparing figures 5.5 and 5.6 to figure 5.4 the amount of road objects is directly striking. The explanation for this is the fact that far more area in IMGeo is classified as road as in Top10NL; this is stated in section 4.4 and Appendix E. Also considering the weights in figure 5.6 doesn't give satisfying results.

Figures 5.5 and 5.6 are only the end results of the tGAP generalisation. An overview of all intermediate results of all methods is provided in Appendix H. This shows how the build up of the end result took place.

The result of this method is not satisfying, because it admits the Top10NL geometry too much. The geometry of the final result is the same as the geometry of Top10NL, because all IMGeo objects are split at the borders of a Top10NL object. The purpose is to come to a vario-scale IMGeo in which Top10NL plays a smaller role, therefore other methods are investigated.

5.5 The maximum area method

The maximum area method requires a lot of pre-processing before converting the data to Oracle tables. The reason for the creation of this method is that the geometry of the IMGeo data is far more accurate than the geometry of Top10NL. Therefore the geometry of the IMGeo objects will be kept in this method. The constrained tGAP structure only merges objects and currently doesn't generalise lines. However, this is being developed and will be mentioned in the future work of this thesis. For this reason the results of this method will not be optimal, because the level of detail of the lines is too high for the representation at scale 1:10,000.

The approach to come to an input dataset for the tGAP structure using the maximum area method has the following steps:

- Joining the IMGeo objects with Top10NL in ArcGIS
- Selecting the maximum overlapping Top10NL object to be the region of the IMGeo object

These steps are treated in separate subsections.

5.5.1 Joining the IMGeo objects with Top10NL in ArcGIS

The joining of IMGeo objects with Top10NL is done with the ArcGIS operator Spatial Join. Figure 5.7 shows what options were used in this join operation. The join operation is a one-to-many operation, because we want to see all the possible connections between the IMGeo objects and Top10NL objects in the resulting shapefile.

The result of this action is a table in which an IMGeo object is mentioned with all its overlapping Top10NL objects. The difference between this table and the intersection in section 5.4 is that in this table the objects are kept as a whole, not as parts. We need this because we finally want to append a whole object to the input dataset for the constrained tGAP. This table is called 'IMGeoSpatialJoin' in the next section.

The table with the intersections from section 5.4 is called 'AppendTop10ID'.

5.5.2 Selecting the right region using the maximum area method.

After all IMGeo objects are assigned to the geometrical right Top10NL objects, the IMGeo objects are to be assigned to just one region, to one single Top10NL-ID. The IMGeo data is first selected with respect to its GML_ID, which is the unique identifier of the IMGeo objects. The selected features all have a Top10ID. If an IMGeo object appears to have more than one Top10ID, there has to be made a choice between these Top10ID's.

The method to be used here is the maximum area method. The full code is inserted in Appendix F. The area of the objects belonging to a certain IMGeo-object (with its GML_ID) is collected in a set. After the last value has been collected, the maximum of this set is computed. With this value the correct set of GML-ID and Top10ID can be selected and finally be appended to a new file called IMGeoTop10NLFinal1, which is the first input for the constrained tGAP structure. The method is shown in algorithm 5.1. The inputs for this method are the IMGeoSpatialJoin table from subsection 5.5.1 and the AppendTop10ID table from subsection 5.4. The resulting table is called 'MaximumAreaDataset'.

```
For every object i from IMGeo_Merge:
        Select from AppendTop10ID where GML_ID = i
        For every selected object:
                Select the area of the object in 'set'
        Max(set)
        Append object from IMGeoSpatialJoin to MaximumAreaDataset with area=max(Set) and GML_ID = i
```

Algorithm 5.1: Iteration of the maximum area method

After applying this method a plot has been made of the test area, in which the IMGeo-objects are coloured according to their region class, which gives an indication of what region

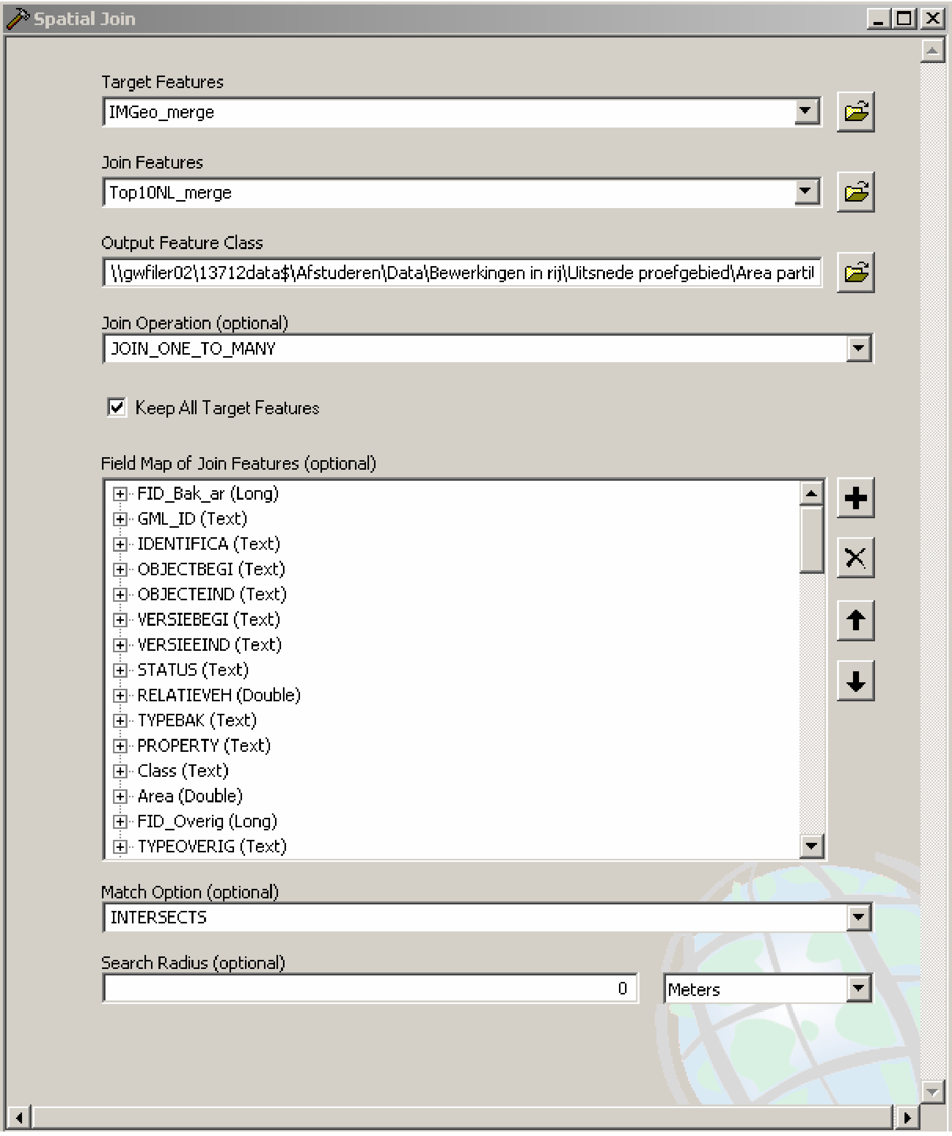


Figure 5.7: The Spatial Join operation in ArcGIS

object they are assigned to. For this maximum area code this gives the result shown in figure 5.8.

Comparing figure 5.8 to figure 5.9, which is again the original Top10NL data, some expected and some unexpected effects can be seen. A large part of the road areas between houses are now classified as terrain objects as region class.

We also see that some residence objects at the corners of housings blocks are now classified as terrain objects in their assigned region. This is not according to what we want, but it is a consequence of the geometrical differences between the models described in chapter 4. The last unexpected failure we see in the right corner beneath where a large terrain part is now classified as residence object. This is also what we didn't expect to happen.

After these processes we end up with all IMGeo objects assigned to one region. The geometry of the IMGeo objects will not change during the merging inside the tGAP. This is something that needs attention in the tGAP structure, but which is currently subject of research at the group GISt as TU Delft.

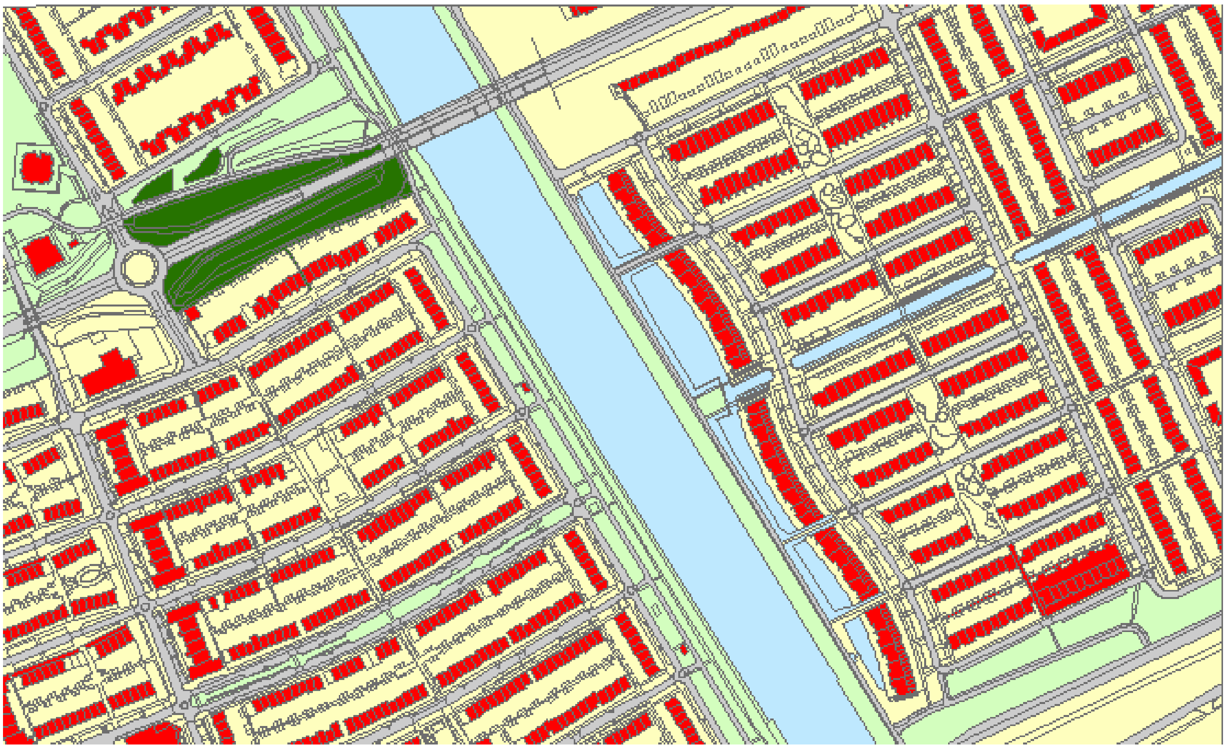


Figure 5.8: The classification according to the maximum area method.

Figure 5.9: The original Top10NL dataset

5.5.3 Testing the constrained tGAP structure for the maximum area method

As in section 5.4 the constrained tGAP has also been applied to this pre-processed dataset. The tests were again done with and without weights giving two different end results. These results are shown in figures 5.10 and 5.11 and can be compared to figure 5.8, which shows the Top10NL dataset as it should be. Again the intermediate results are provided in Appendix H.

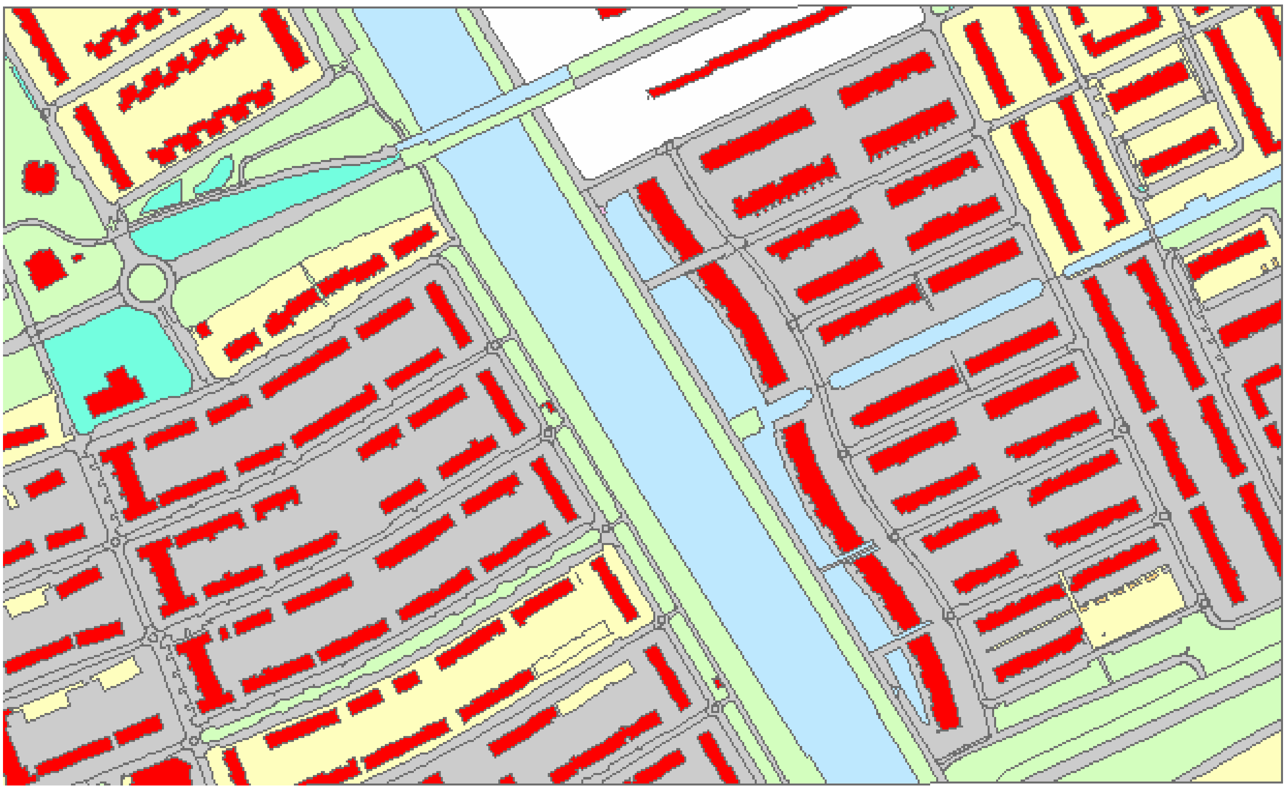


Figure 5.10: Visualisation of the end result using the maximum area method without weights.

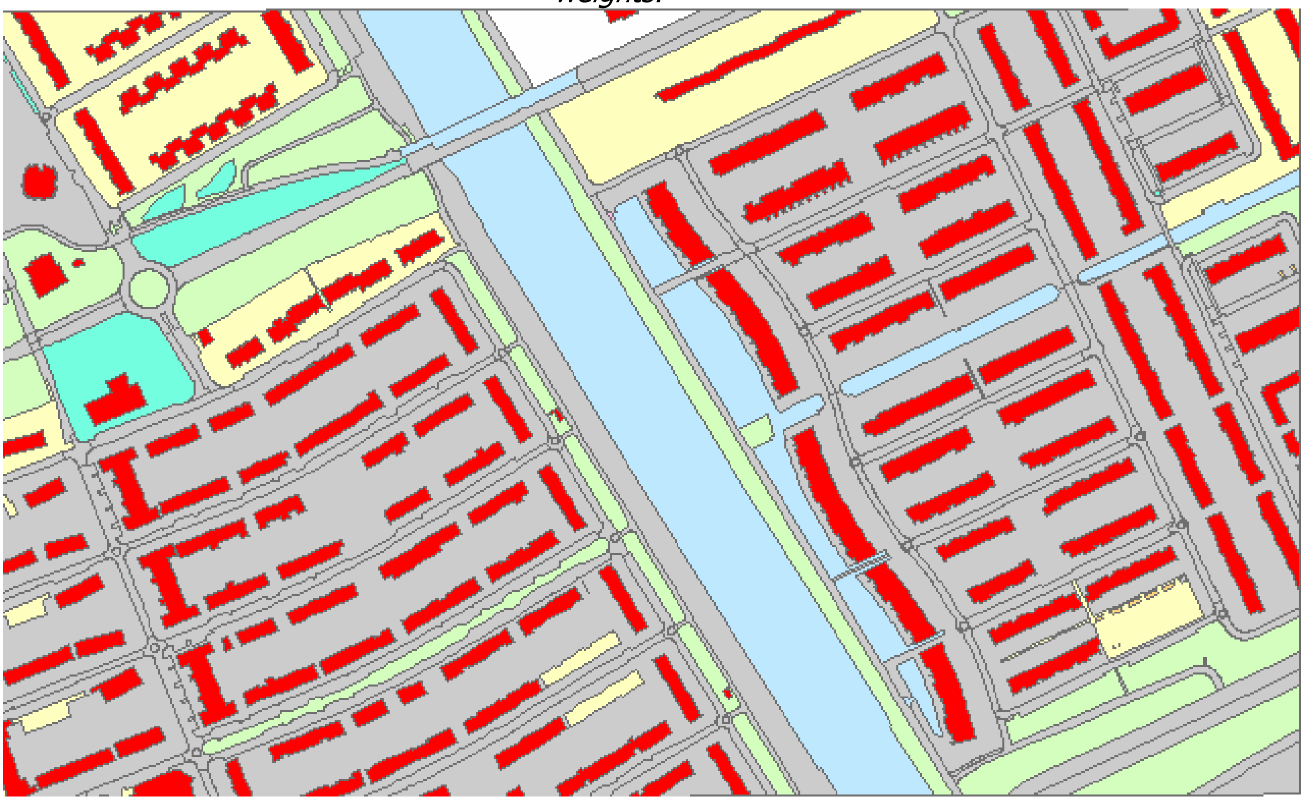


Figure 5.11: Visualisation of the end result using the maximum area method with weights.

5.6 The 35%-split method

In the 35%-split method the IMGeo objects are split if there is more than one Top10NL object overlapping for more than 35% with the IMGeo objects relative to the IMGeo object size. The reason this method is developed is mainly because the classification of buildings needs to be improved. The corner buildings in the MaximumAreaDataset were too often classified as terrain.

The number of Top10NL objects that overlap for more than 35% with an IMGeo object differs per IMGeo object. It can either be 2, 1 or 0. In case of two Top10NL objects satisfying the 35% rule the IMGeo object is split. This is done as a pre process before running the code of the maximum area method. With this code only those IMGeo objects are selected that have two Top10ID's satisfying this rule. One of these IMGeo objects is written to a separate shapefile. For the remaining objects the maximum area method again is applied. The tables IMGeoSpatialJoin and AppendTop10ID are again used in this method; the only part that differs is the code of subsection 5.5.2. Algorithm 5.2 can be seen as a pre-process to the maximum area method. The total number of objects that is split according to this 35%-split method is 236 (out of 4159 IMGeo objects).

Algorithm 5.2 shows how the code works. The resulting dataset is called '35%SplitDataset'. As with the previous codes, the full code is provided in Appendix F.

```
For every object i from IMGeo_Merge:
        Select from AppendTop10ID where GML_ID = i
        For every selected object:
                If (Shape area / Area of the original IMGeo object) > 0,35:
                        Add to 'set'
        If 'set' contains 2 objects:
                Append one of the objects to 35%SplitDataset
                Assign a new unique GML_ID to the other object
```

Algorithm 5.2: Iteration of the 35%-split method

The visualisation of the resulting final dataset following from this code can be compared to the Top10NL dataset as it should be. The Top10NL dataset is shown in figure 5.13. The result of the 35% split method is shown in figure 5.12 with the IMGeo objects classified according to their region class.

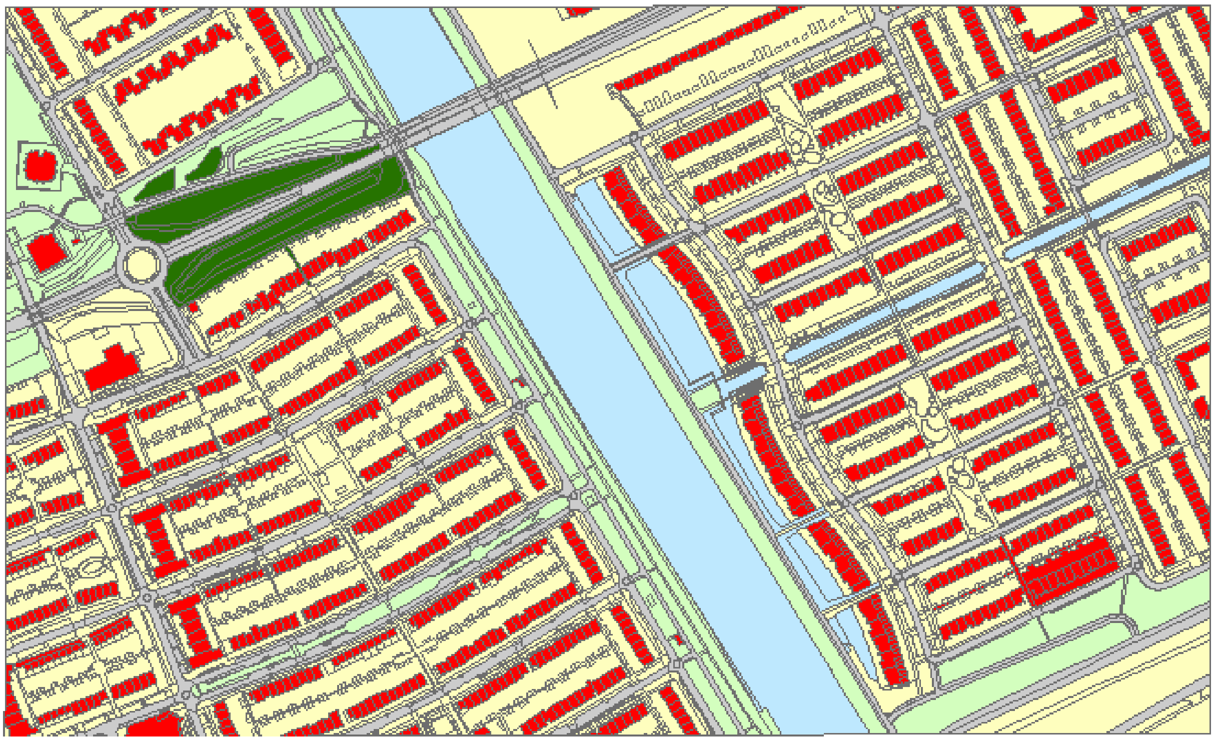


Figure 5.12: The classification according to the 35%-split method.

Figure 5.13: The original Top10NL dataset

Results for the constrained tGAP using the 35%-split method
Figures 5.14and 5.15 show the end results of the constrained tGAP generalisation using the 35%-split method. The code is processed with and without weights and all intermediate results are provided in Appendix H. The result can also be compared to the Top10NL dataset in figure 5.13.

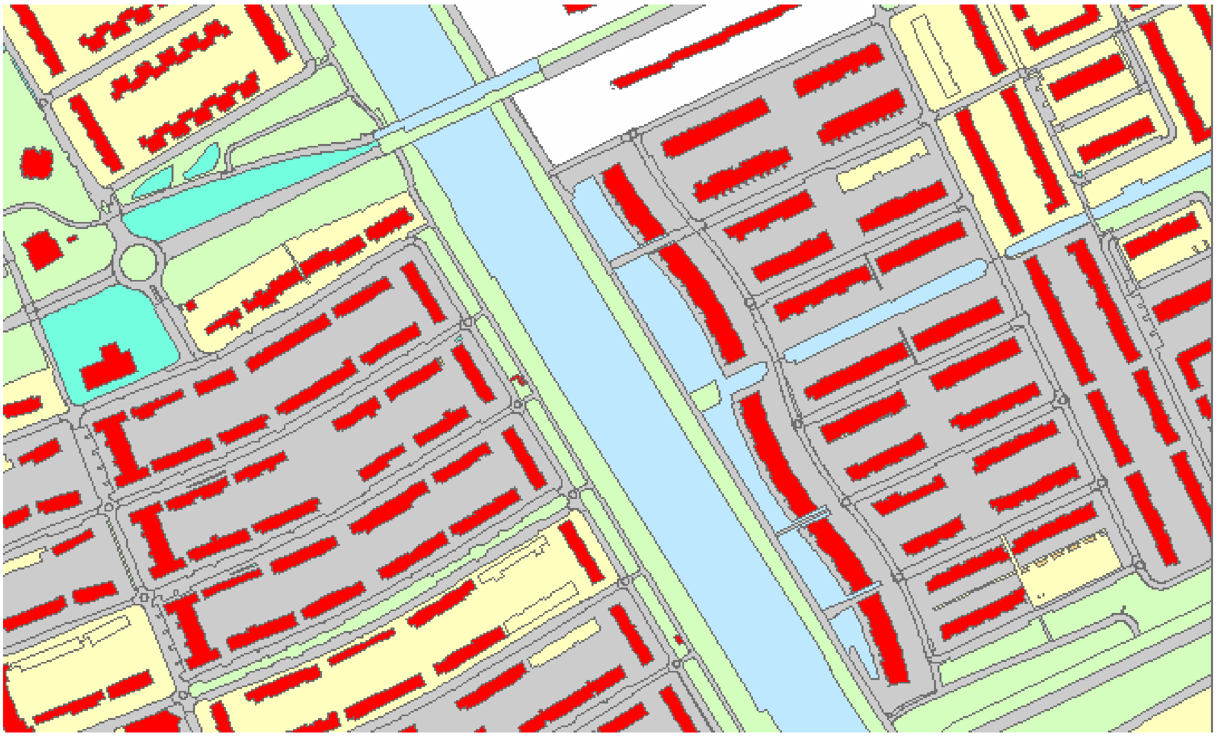


Figure 5.14: Visualisation of the end result using the 35%-split method without weights.

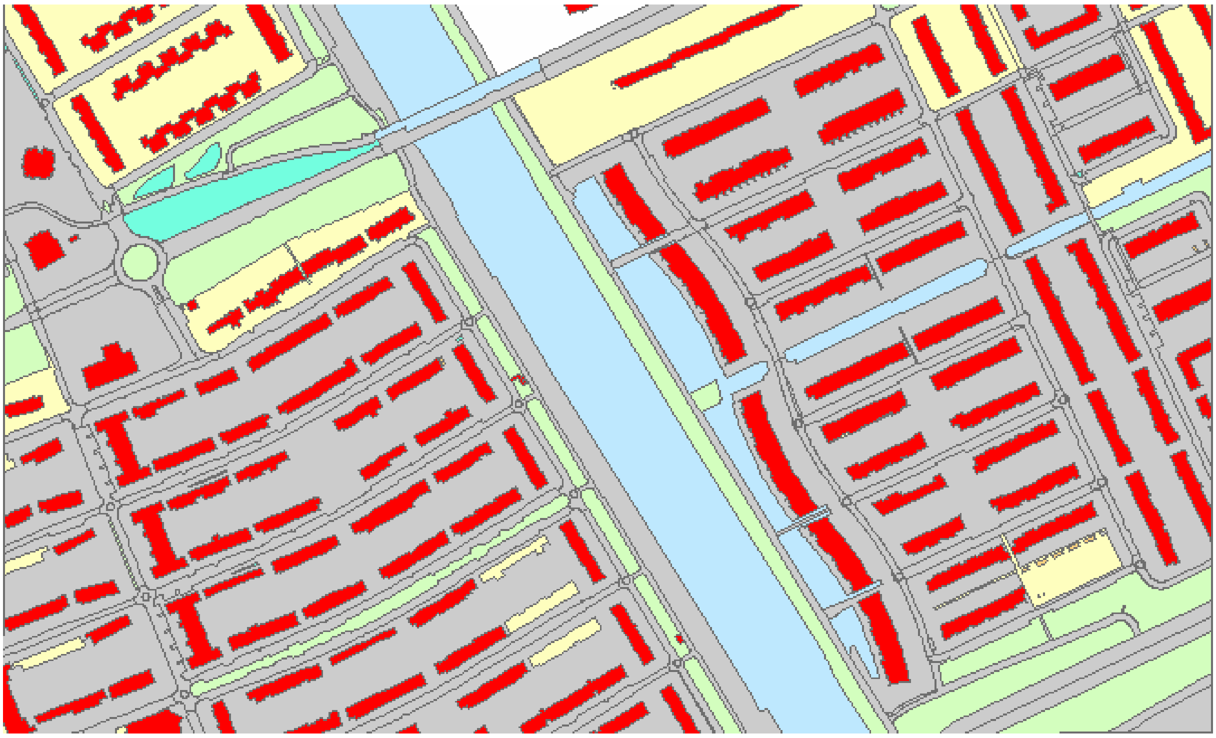


Figure 5.15: Visualisation of the end result using the 35%-split method with weights.

5.7 The building first method

The building first method is, like the 35%-split method, a pre-process before running the maximum area code. The method is developed, because the 35%-split method also didn't give the results we hoped for.

The principle of this method is that buildings in IMGeo preferably have to be assigned to a building region. The method to build the input dataset for this method is very simple. The dataset with the joined features (IMGeoSpatialJoin) from section 5.5.1 contains all possible overlaps between IMGeo objects and Top10NL objects. If an IMGeo building object would have to be assigned to a Top10NL building region, it would have to have at least some overlap. The criterion to assign IMGeo objects to a region therefore has been to select all those features from the Spatial Join table where the class and the region class are equal to 1001, the code for buildings. For all the other objects again the maximum area was applied.

The visualisation of the end result of the building first method is shown in figure 5.16. This can be compared with the original visualisation of the Top10NL model in figure 5.17.

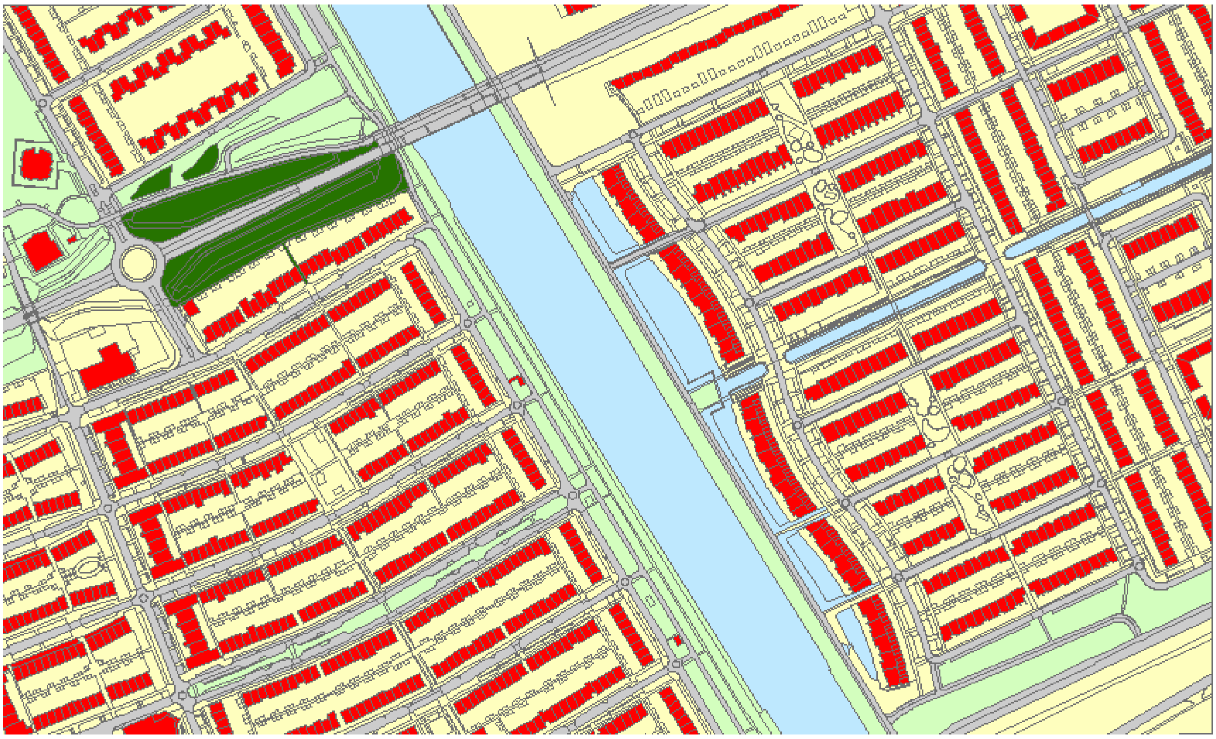


Figure 5.16: Visualisation of the classification according to the building first method

Figure 5.17: The original Top10NL dataset

Results for the constrained tGAP using the building first method

Figures 5.18 and 5.19 show the end results of the constrained tGAP generalisation using the building first method. The code is processed with and without weights and all intermediate results are provided in Appendix H. The result can also be compared to the Top10NL dataset in figure 5.17.

Figure 5.18: Visualisation of the end result using the building first method without weights.

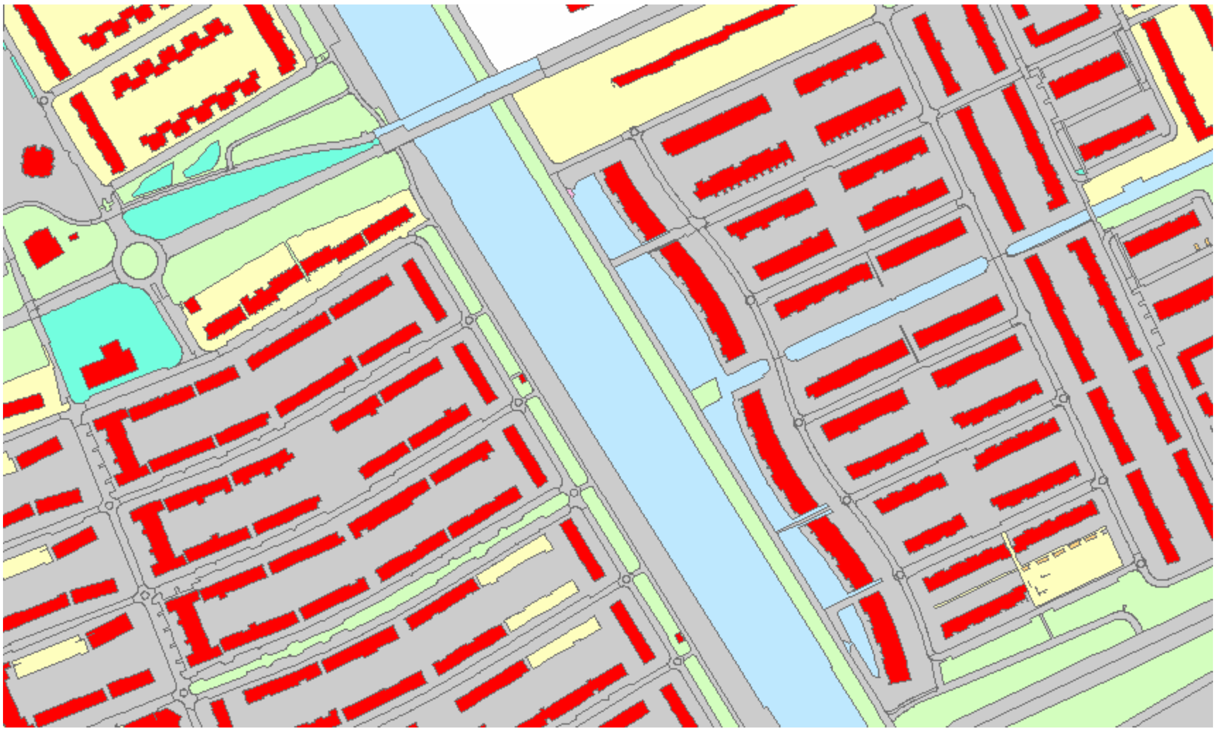


Figure 5.19: Visualisation of the end result using the building first method with weights.

5.8 Comparison of the methods and conclusions

This section compares the 4 methods presented in this chapter. The results are judged and finally conclusions are drawn.

All methods have to do with the same problem. It is shown that far too much objects are classified as roads. Large terrain parts are merged to roads. The reason for this is that lots of small road objects in IMGeo form the pavement. They are merged in the beginning of the process and these larger polygons in the end are of more importance than the terrain objects next to it. This can be prevented by assigning a higher weight to the class terrain than to the class roads. Of course roads are normally of high importance, but as shown in the analysis of the models in chapter 4 in the large scale topographical map lots more objects are classified as road objects than in the Top10NL map, for this reason the importance of road objects must be set considerably lower than expected.

Taking into account that this problem will be dealt with in chapter 6, we will now compare the 4 methods presented and choose the best. The end result of the simple overlay method may seem quite good, but the disadvantage of it is that the geometry of IMGeo, which is obviously better than the geometry of Top10NL, is not used to build the end result. The geometry of Top10NL is more and more taken over in this method. The other three methods use the geometry of IMGeo throughout the process.

The main focus comparing these three methods will be on the buildings. Figures 5.20 – 5.23 show a part of the original IMGeo dataset and the way it is classified in the three methods mentioned.

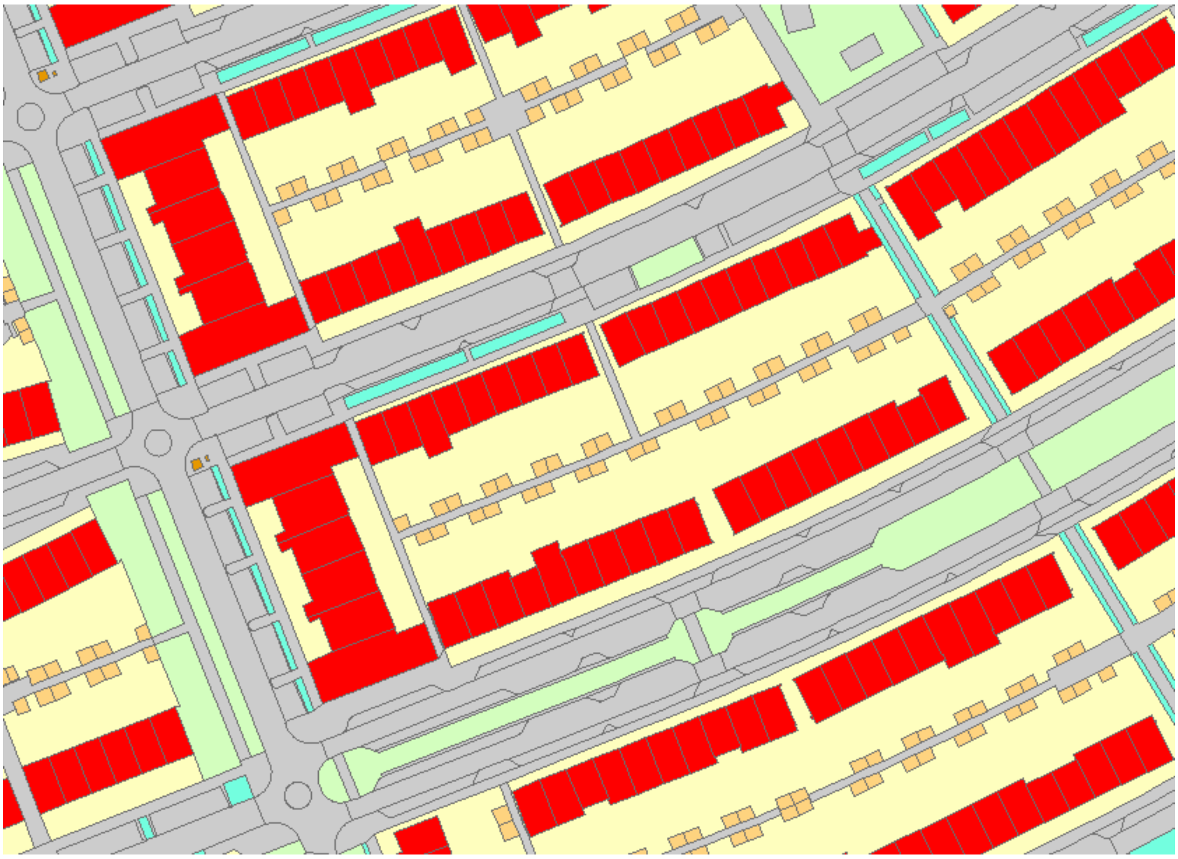


Figure 5.20: Part of the original IMGeo dataset

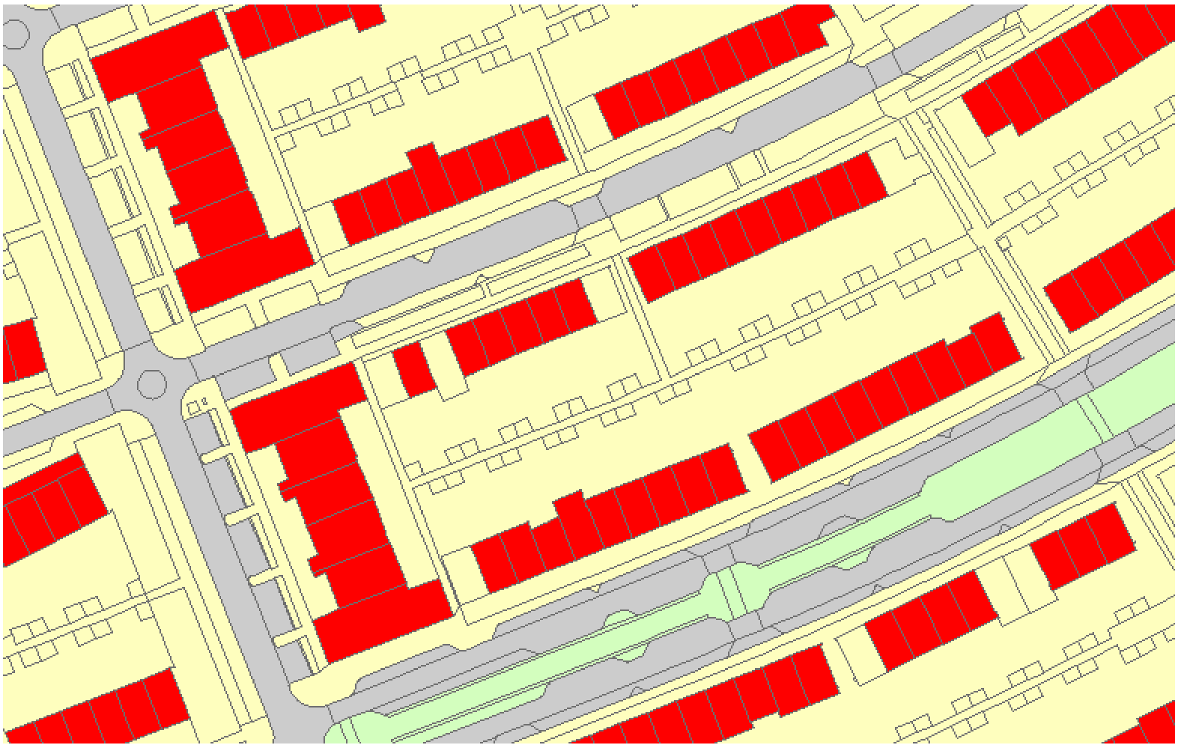


Figure 5.21: Part of the IMGeo dataset according to the maximum area method

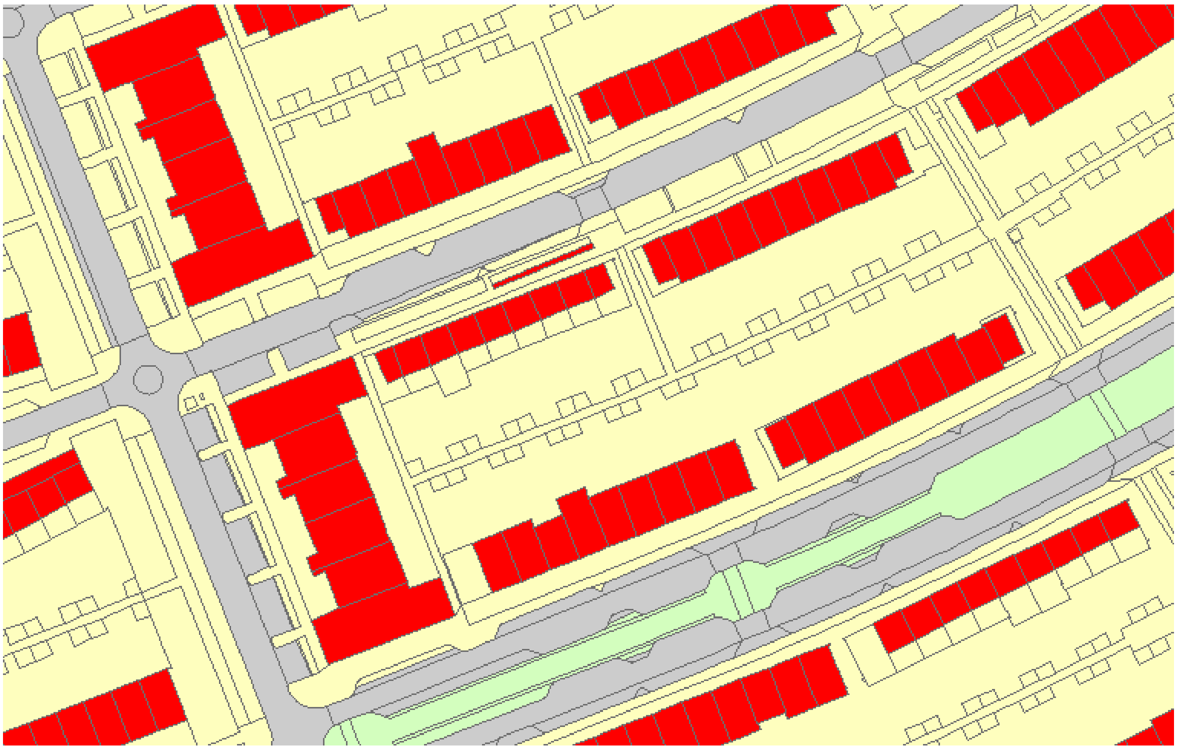


Figure 5.22: Part of the IMGeo dataset according to the 35%-split method

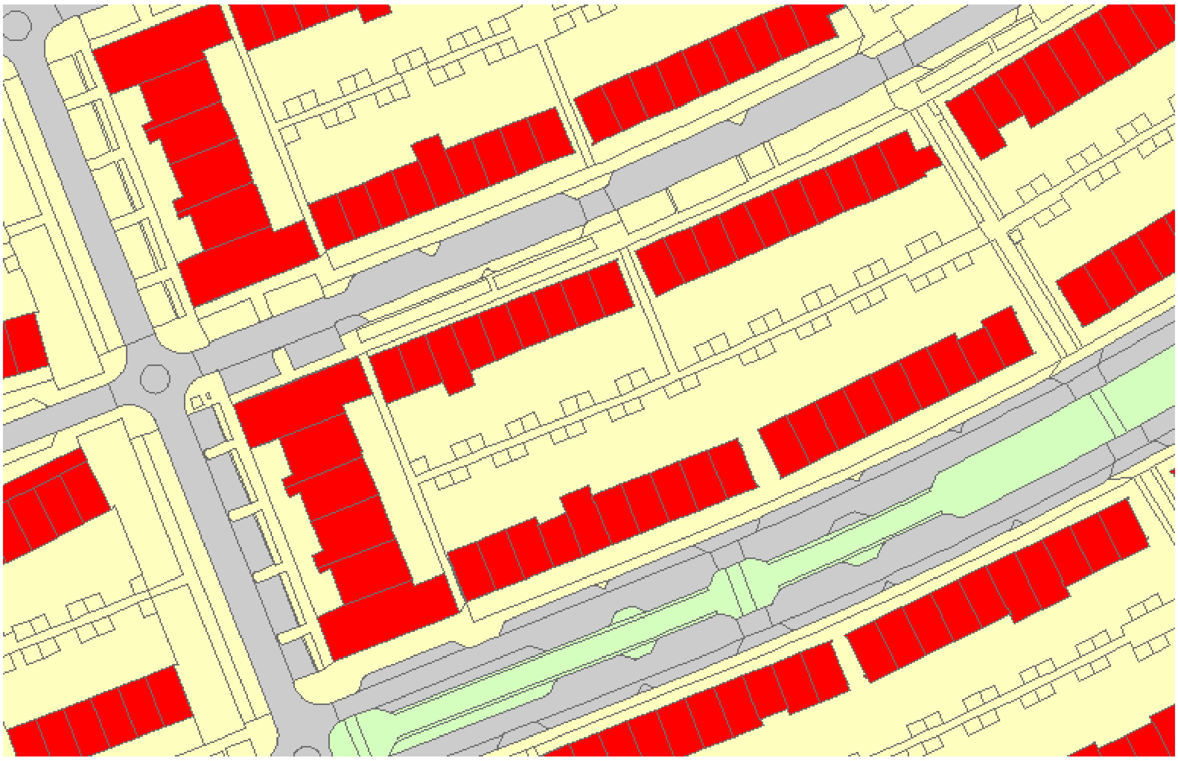


Figure 5.23: Part of the IMGeo dataset according to the building first method

The differences between the classifications of the buildings in figures 5.20 – 5.23 are remarkable. In figure 5.21 we see some buildings classified as terrain because the building in Top10NL is placed in such a way that the maximum overlap the IMGeo building has is with the terrain. In figure 5.22 we see also an unwanted effect. The buildings become very small due to the split action. Because these results weren’t satisfying the building first method was developed. The result in figure 5.23 shows that all buildings are classified as buildings, the result we expected.

The necessity to split objects in the case that more than one Top10NL object overlaps the IMGeo object for 35 % (or any other percentage) or more wasn’t proven. Only 236 out of 4159 objects needed to be split; this is only 5,6% of the total amount of objects. The splitting of buildings according to the 35%-split rule is not what we wanted and it appeared that most of the other split objects involved buildings (143). Also some roads were split (68); the cause of this is that the splitting of road objects is done differently in both models. This is something that needs attention and will be discussed in chapter 6.

The buildings first method will be the method on which further tests will be done. Taking this method as point of departure is also good with respect to the project GBKN buildings in Top10NL, which was described in section 2.7.3. If the GBKN buildings are to be placed in Top10NL it is good to take them all as a whole to our next research phase in which especially the roads will need special attention. The improvements are described in chapter 6.

6. Improvements for the constrained tGAP structure

Now the building first method has been chosen in chapter 5 as the best method to continue the research with, we continue with improvements for the results of chapter 5. First, section 6.1 describes the modifications that have been done to the weights and compatibilities to get to a proper end result. In section 6.2 the tGAP procedures are executed without the constraint and finally in section 6.3 the methods of this chapter are applied on the whole dataset of Almere.

6.1 Improvements of the weights and compatibilities

The improvement of the weights- and compatibility values is mainly based on 'trial and error'. Tables 6.1 and 6.2 show the comparison between the weights table used in chapter 5 and the weights table used by (Van Putten and Van Oosterom, 2000). Because the research by Van Putten also focussed on a polygon GBKN the weights used by her can be seen as best practice.

Class	Weight
Terrain (TRN)	5
Road (WEG)	80
Water (WTR)	10
Crossing (KNP)	2
Building (GBW)	400
Railroad (SBN)	20

Table 6.1: Class weights used by Van Putten (2000).

Class	Code	Weight
Residence object / Building	1001	0,9
Other Building	5003	0,4
Road	2001	0,6
Water	3001	0,5
Lot	4001	0,3
Fallow land	4002	0,1
Plants	4003	0,3
Terrain (to be determined)	4004	0,1
Grass / Grassland	4005	0,3
Bin	5001	0,1

Table 6.2: Initial class weights for this research.

A remarkable fact is that Van Putten used other classes; the classes railroad and crossing are not realised in this IMGeo test dataset. The IMGeo dataset contains some topographical elements (Bin and Other Building) and has the objects of the class 'terrain' subdivided into 5 subtypes of terrain.

The weights Van Putten assigns to the objects differ from the weights used in chapter 5 as well. As we set Van Putten's work as best practice and we know that the weights used in chapter 5 need to be changed, we first need to know what can be learned of the weights of Van Putten.

The value Van Putten puts for buildings is very high compared to the other objects, this seems to be a good suggestion, because in our output also some building blocks were generalised to terrain, which is unwanted. The second most important object Van Putten mentions is road. This will not work for our dataset, since we have far more road objects in the IMGeo model (due to side walks) than we have in the Top10NL model which causes a classification problem in the end. The weight for terrain is set very low in Van Putten's method, but the value for e.g. 'Lot' needs to be higher than road to compensate the number of road objects.

Tables 6.3 and 6.4 show the respectively the compatibility matrix of Van Putten and the compatibility matrix used in chapter 5.

	TRN	WEG	WTR	KNP	GBW	SBN
TRN	0,1	0,2	0,	0,5	0,9	0,5
WEG	0,1	1	0,005	0,9	0,005	0,005
WTR	0,4	0,1	1	0,1	0,1	0,1
KNP	0,1	0,9	0,005	1	0,005	0,8
GBW	0,7	0,2	0,005	0,005	1	0,005
SBN	0,1	0,005	0,005	0,8	0,005	1

Table 6.3: Compatibility values used by Van Putten (2000).

	Class 1→	**1001**	**5003**	**2001**	**3001**	**4001**	**4002**	**4003**	**4004**	**4005**	**5001**
Class 2 ↓	***Cost***										
1001		0	1	50	100	1	10	50	50	100	100
5003		1	0	50	100	50	10	50	50	100	100
2001		50	50	0	100	50	50	5	10	50	5
3001		100	100	100	0	100	100	100	100	100	100
4001		10	5	5	100	0	10	10	10	20	10
4002		10	5	10	100	5	0	10	10	50	20
4003		50	50	50	100	20	50	0	5	10	10
4004		10	5	50	100	20	10	100	0	50	20
4005		50	50	10	100	20	50	5	5	0	5
5001		100	100	100	100	100	100	100	100	100	0

Table 6.4: Initial compatibility values for this research

The way Van Putten defines compatibility is different from the one used in this research. Van Putten uses the length of the common boundary together with the compatibility value to determine the most compatible neighbour; the maximum value is taken. In this research we use the area of the neighbour and the transition cost to determine this most compatible neighbour; we take the neighbour that costs least for transition. For this reason the compatibility values are actually 'class incompatibilities'. This method was proposed in the work of Haunert (Haunert et al., 2007) and is adopted in this research without further consideration.

What can be learned from the method of Van Putten is the fact that it shouldn't be that objects of the same sort should be automatically merged. If all small road objects in a terrain region would be merged, a road object will be created with such area that the whole terrain area will be classified as road in the end. This is exactly what happened in chapter 5 when all the terrain objects were in the end classified as road objects.

Results
With this information a 'trial and error' process was started up. The steps taken in this phase are visualised in Appendix I. With ever changing values for the weights and compatibilities a

proper end result was reached in the end. The final result of the last trial in this process is visualised in figure 6.1. This can be compared to the original Top10NL dataset, which is provided in figure 6.2 and the original IMGeo dataset, provided in figure 6.3. The final weight- and compatibility values for this test dataset in Almere are presented in tables 6.5 and 6.6.

Class	**Code**	**Weight**
Residence object / Building	1001	13
Other Building	5003	1
Road	2001	1,2
Water	3001	1,3
Lot	4001	9
Fallow land	4002	1
Plants	4003	0,9
Terrain (to be determined)	4004	0,1
Grass / Grassland	4005	1
Bin	5001	0,1

Table 6.5: Final weights for the IMGeo test dataset.

	Class 1→	**1001**	**5003**	**2001**	**3001**	**4001**	**4002**	**4003**	**4004**	**4005**	**5001**
Class 2 ↓	***Cost***										
1001		0	1	50	100	1	10	50	50	100	100
5003		1	0	50	100	50	10	50	50	100	100
2001		50	50	1	100	50	50	5	10	100	5
3001		100	100	100	0	100	100	100	100	20	100
4001		50	5	5	100	0	10	10	10	20	10
4002		10	5	10	100	5	0	10	10	50	20
4003		50	50	50	100	20	50	1	5	1	10
4004		10	5	50	100	20	10	100	0	50	20
4005		50	50	5	10	20	50	1	5	1	5
5001		100	100	100	100	100	100	100	100	100	0

Table 6.6: Final compatibility values for the IMGeo test dataset.

Figure 6.1: Final result of weights and compatibility improvements

Figure 6.2: The original Top10NL dataset

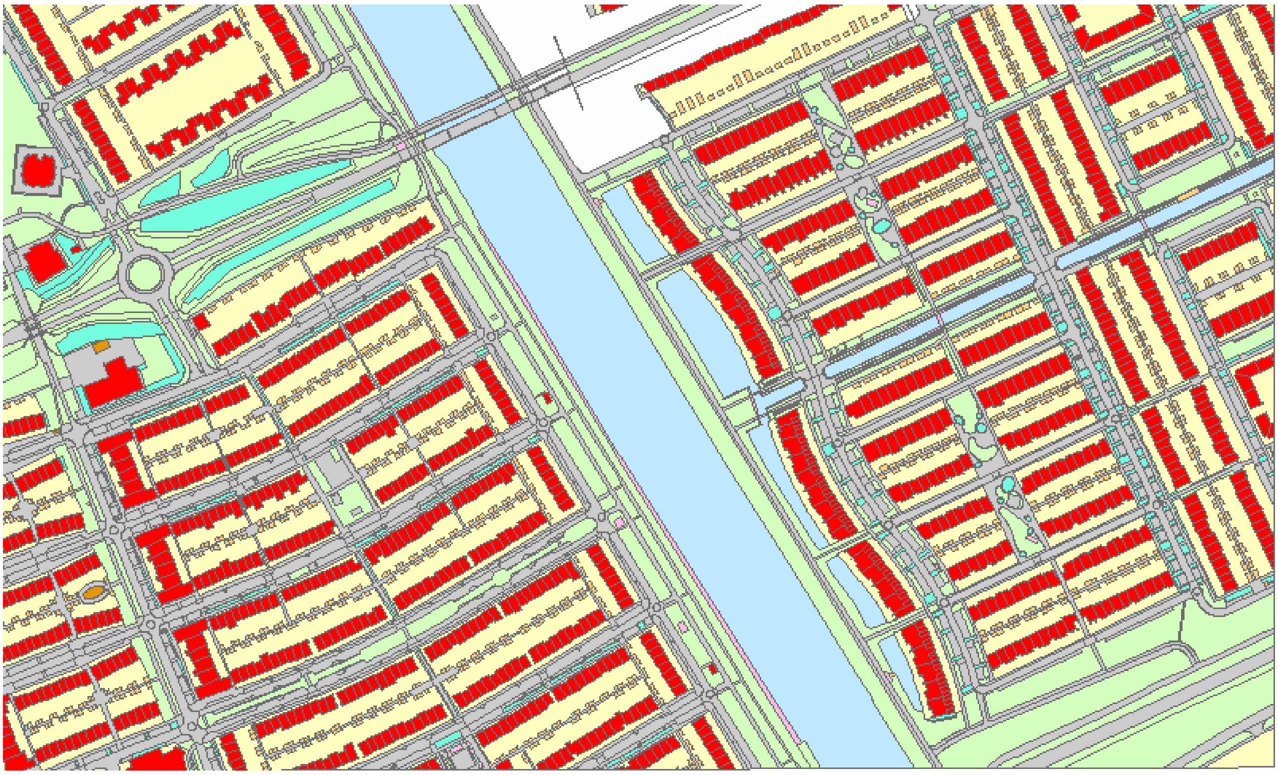


Figure 6.3: The original IMGeo dataset

The end result following from the constrained tGAP generalisation is not exactly like the Top10NL dataset, there is a number of causes for this, which will be discussed:

- The geometry of IMGeo was maintained during generalisation
- The classification in both models differs
- The class of the final object is not forced to become the region class
- Large road objects cause strange phenomena

The reason why the geometry of IMGeo was maintained during generalisation has been described before. We trust the geometrical accuracy of IMGeo more than Top10NL. Moreover, the tGAP algorithms executed for this dataset will be extended with line simplification algorithms in the near future, probably giving better end results.

The second and the third cause are related to each other. The classification of some objects in the IMGeo dataset can never be the same as the objects in the Top10NL dataset, because they don't exist in that dataset. For example, there are no wood objects in the IMGeo test dataset, whereas some plant- and grass objects are classified as wood in the Top10NL dataset. The object hierarchy could be that plants and grass have to be generalised in the end to wood, but this is not the case, because also in Top10NL grass objects appear.

The only way to solve this problem is to force the object to take over the region class as class in the end of the tGAP process. To do this some proposals can be done, which were not executed during this research.
First, in the last merge of every region a check can be executed which checks whether the class of the neighbour absorbing the object is equal to the region class, if not, then change it to the region class.
Second, the region class can be a weighing factor in determining what the most compatible neighbour of an object is by for example inserting such a weighing factor in the compatibility matrix. This would make the compatibility matrix a 3D matrix, because the value will then depend on the class of the least important face, the class of the neighbour and the region class. This of course has the advantage of a smooth transition towards the end, but has as disadvantage that the class of the end object will never become the region class if there is no such object in the region. For this the forcing will be needed.

An important issue is the large road objects that appear in both datasets, especially in the IMGeo dataset. Due to the fact that every IMGeo object is assigned to just one region, some strange end results can be generated in which an object has a strange shape. The main explanation for this is then that a former large road object in IMGeo caused this strange boundary of the object. In figure 6.4 and 6.5 a comparison is made of the end result and the original IMGeo file showing that a smarter cutting of road objects leads to better results.

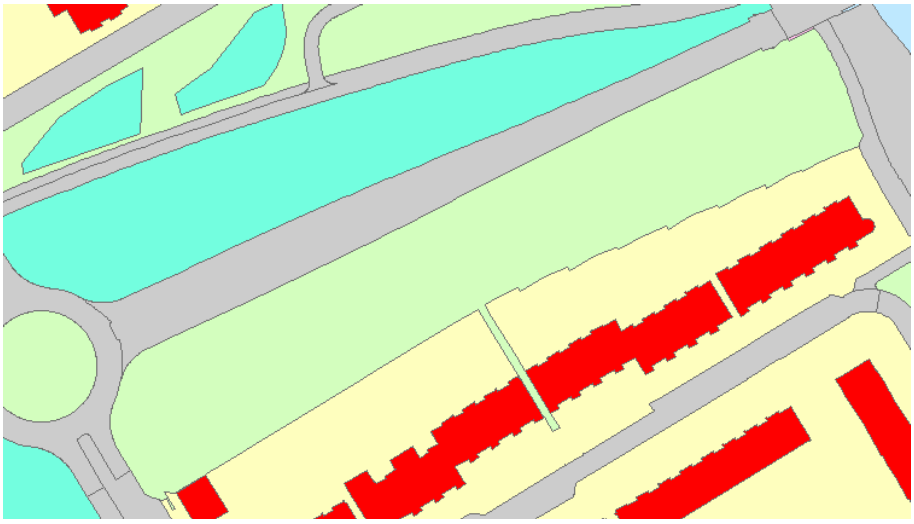


Figure 6.4: A strange looking grass object in the end result

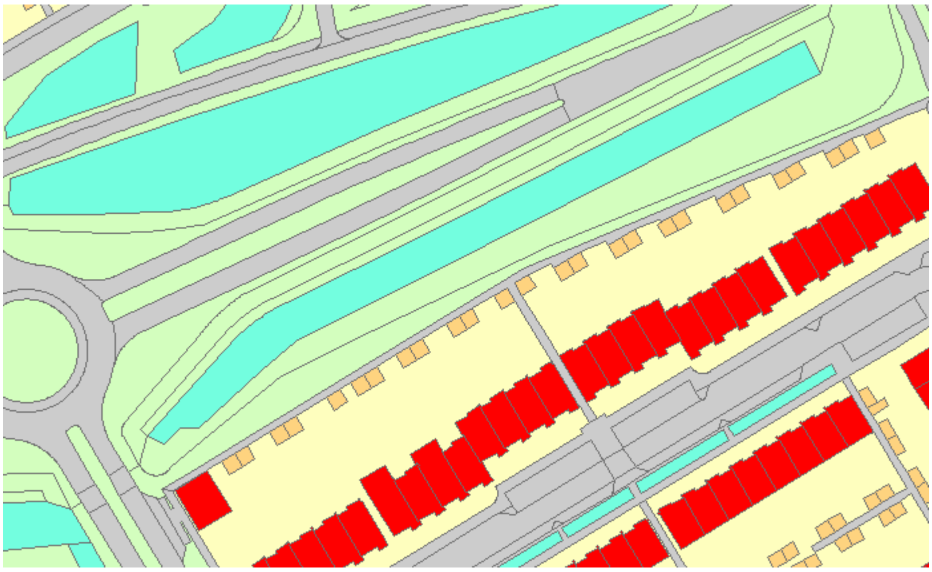


Figure 6.5: The road object causing the strange shape in figure 6.4

The example shown in figures 6.4 and 6.5 shows that it would be wise for a generalisation structure to cut the road objects at all crossings.

The second example shows that road objects are too large compared to other objects to be able to make a fair generalisation structure based on the least important area. Figure 6.6 shows small grass objects which are generalised to a road object creating such a large road object that in the end also the larger and more important grass object is absorbed in it. The road object which is made up along the canal originates from the very narrow (bicycle) road next to the canal. This object obviously needed to be generalised to a grass object, but it didn't, because the road object was not neatly cut into several pieces.

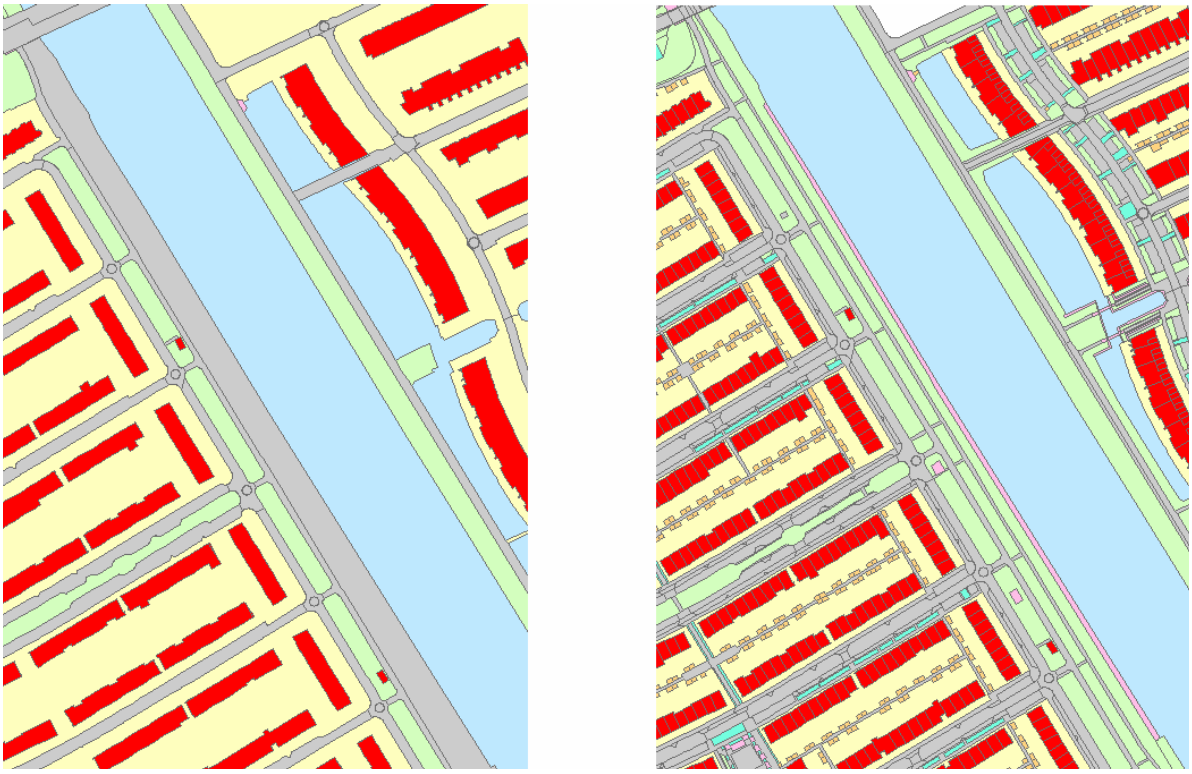


Figure 6.6: The end result (left) showing a large and unexpected road object along the canal and the original IMGeo dataset (right).

6.2 tGAP without constraint

After having determined the right weights for the constrained tGAP we also would like to see how the objects react on the case where there is no constraint. This means that objects can also be merged with objects outside their own region. If this gives good results, this is a sign that the constraint might not even be necessary. Without constraints the whole test dataset is in the end generalised to only one object. This appeared to be a road object. In appendix J the whole generalisation process to come to an end result is shown. At this place we only show the result after 3805 merging operations, which is exactly the amount of merges also done in the test dataset. Figure 6.7 shows the result.

Figure 6.7: Normal tGAP after 3805 merges

In figure 6.7 it is shown that there are some differences between the results of the normal tGAP and the constrained tGAP. The remarkable cases are mainly roads, buildings and water.

As stated in chapter 4 the total area of road objects in IMGeo is far larger than in Top10NL. This is mainly caused by pavements, which are not taken into account by the Top10NL for cartographic reasons. Because of the constraint and the large terrain objects the extension of the roads in IMGeo are in the end merged to terrain objects, whereas in the normal tGAP the roads can expand freely. Especially in the north-west corner of the test dataset this gives unwanted results.

Buildings are of course of very much importance. They are most of the time closed in by terrain objects. In figure 6.7 it is shown that also the building objects are taking more space than they should have. Especially in the south-west corner of figure 6.7 some examples are shown.

The only two water objects in figure 6.7 form the large canal, whereas in the original Top10NL dataset also some smaller water objects exist east of the canal. In figure 6.7 these water objects are absorbed by other objects, because the constraint was the element that preserved the water objects from being merged.

The weights used in this test are the weights from table 6.5 and therefore not optimalised for the normal tGAP. It is assumed that the optimalised weights for the constrained tGAP should also hold for the normal tGAP.

To guide the generalisation of datasets like this we can conclude from this test that the constraint is really a useful concept which can improve the quality of the results from the tGAP structure.

6.3 Testing the constrained tGAP on a larger dataset

The methods used until now have been tested on just a fraction of the data. In this section the results are shown of the test of the methods developed for the whole test area indicated in figure 4.2. To give an impression of the larger test area figure 6.8 is inserted, showing the Top10NL representation of the area. The Top10NL is shown here instead of the IMGeo, because the scale of the map wouldn't be appropriate for the IMGeo representation. The black rectangle indicates the smaller test area used before.

Of course the number of objects in this test dataset is a lot higher. There are over 27,000 IMGeo objects and over 3500 Top10NL objects in this test area. But the main difference between the previous test data and this larger dataset is the fact that some objects appear in this larger dataset that didn't appear before.

The new objects in the IMGeo dataset are (between brackets their code):

- Wood (4006)
- Arable land (4010)
- Separation (5002)
- Unknown (6001)

The new objects in the Top10NL dataset are:

- Built area (4009)
- Arable land (4010)
- Unknown (6001)

At least one of these new object types is remarkable. The object class 'unknown' is inserted, because the intsersection method of section 5.4 didn't assign a Top10ID to all IMGeo objects properly. The disadvantage of this 'Union-operator' was that not all objects received an ID, because they seemed to ArcGIS not to be belonging to any object in the other model. The fact that these objects were really small and that only a very small part of the objects didn't receive an ID (< 0,1 %) led to the conclusion that this operator could be used. The objects that didn't receive an ID and class in the first place were given an ID manually and they got the class 6001, which means unknown.

Figure 6.8: Visualisation of the Top10NL large test area

The weights table and compatibility matrix have been extended for the new object classes. The values for the classes that already were determined are the same as in table 6.5 and 6.6. The new tables are shown in tables 6.7 and 6.8.

Class	Code	Weight
Residence object / Building	1001	13
Road	2001	1,2
Water	3001	1,3
Lot	4001	9
Fallow land	4002	1
Plants	4003	0,9
Terrain (to be determined)	4004	0,1
Grass / Grassland	4005	1
Wood	4006	1
Arable land	4010	0,5
Bin	5001	0,1
Separation	5002	0,1
Other Building	5003	1
Unknown	6001	0,0000000001

Table 6.7: Class weights for the large test dataset

	Class 1→	**1001**	**2001**	**3001**	**4001**	**4002**	**4003**	**4004**	**4005**	**4006**	**4010**	**5001**	**5002**	**5003**	**6001**
Class 2 ↓	***Cost***														
1001		0	50	100	1	10	50	50	100	100	100	100	100	1	1
2001		50	1	100	50	50	5	10	100	100	50	5	5	50	1
3001		100	100	0	100	100	100	100	20	100	100	100	100	100	1
4001		50	5	100	0	10	10	10	20	100	50	10	10	5	1
4002		10	10	100	5	0	10	10	50	100	20	20	20	5	1
4003		50	50	100	20	50	1	5	1	10	50	10	10	50	1
4004		10	50	100	20	10	100	0	50	100	50	20	20	5	1
4005		50	5	10	20	50	1	5	1	10	20	5	5	50	1
4006		50	50	100	50	50	1	10	10	0	50	20	20	50	1
4010		50	100	100	20	20	50	10	20	10	0	50	50	50	1
5001		100	100	100	100	100	100	100	100	100	100	0	5	100	1
5002		100	100	100	20	100	100	100	100	100	100	5	0	100	1
5003		1	50	100	50	10	50	50	100	100	100	100	100	0	1
6001		100	100	100	100	100	100	100	100	100	100	100	100	100	1

Table 6.8: Compatibility matrix for the large test dataset

Figure 6.9 shows the end result of the constrained tGAP for this large dataset. Appendix K also shows the intermediate results. In spite of the iteration for the previous test dataset road objects remain a problem when combining both datasets. For this reason some iterationsteps were also applied for this larger test dataset to come to an optimal result. The result of this is shown in figure 6.10. The weights that were changed to come to this result are provided in table 6.9.

Figure 6.9: End result of the constrained tGAP tree with weights of table 6.7.

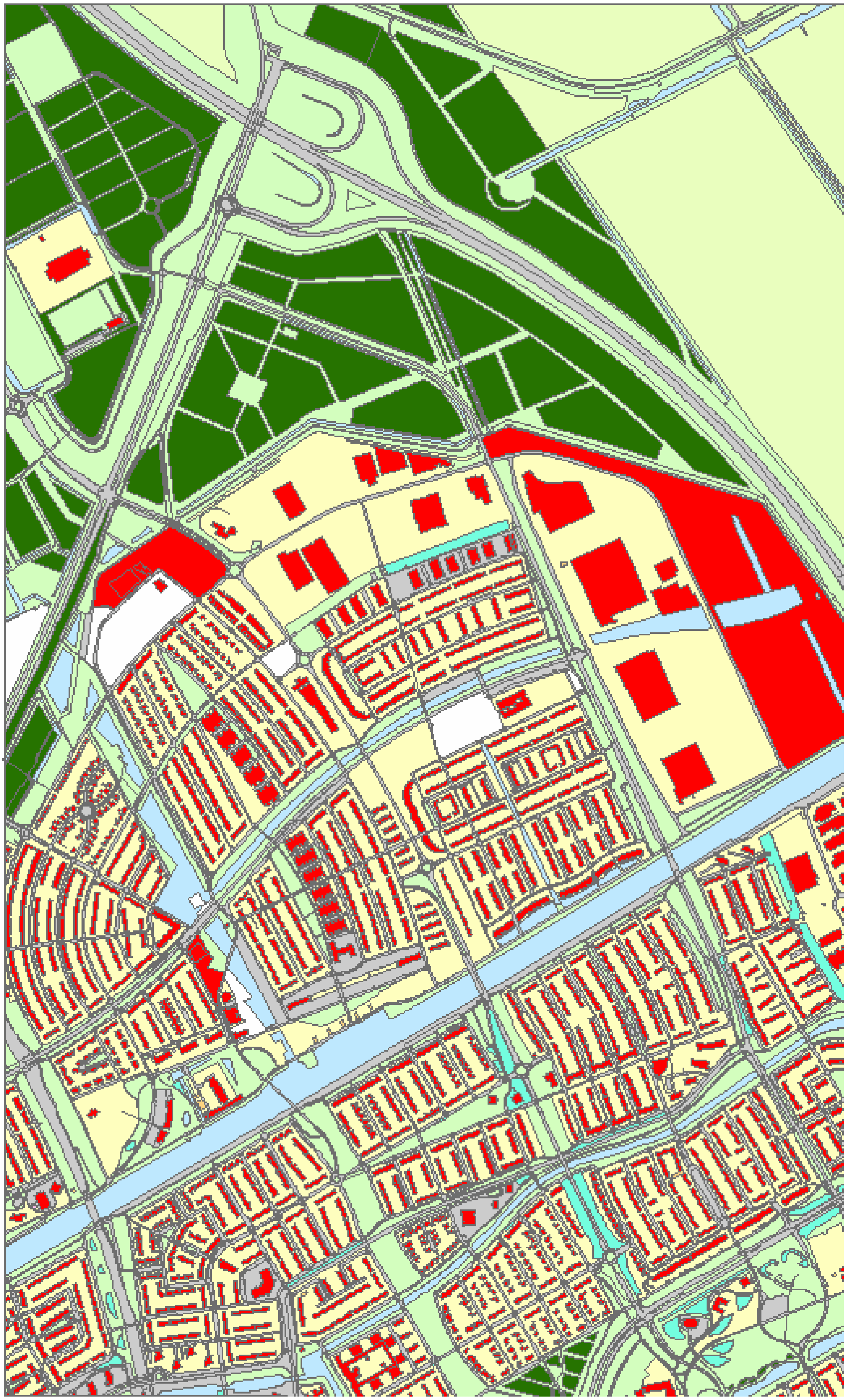


Figure 6.10: End result of the constrained tGAP tree with weights of table 6.9

Class	Code	Weight
Residence object / Building	1001	17
Road	2001	1,05
Water	3001	1,5
Lot	4001	14
Fallow land	4002	1
Plants	4003	0,8
Terrain (to be determined)	4004	0,1
Grass / Grassland	4005	0,9
Wood	4006	1
Arable land	4010	0,5
Bin	5001	0,1
Separation	5002	0,1
Other Building	5003	0,3
Unknown	6001	0,0000000001

Table 6.9: Changed weights for the iterated end result of figure 6.10

The values currently used give almost the same results for our previous test area. During the trials the values for 'building' and 'lot' were continuously decreased to get to a weight as proportional as possible to the other object. It appeared these values needed some adjustment upwards. The results in figure 6.10 now are very satisfying.

There is one remarkable fact that arises in figure 6.10 and that is the point of the difference in actuality between both maps. Looking at figure 6.10 we see a large building object in the south-most part of the map. Comparing this to figure 6.8 we see that there is no building at this place in Top10NL. Because of this the building on this fallow land couldn't be classified in the building region and therefore now takes the whole object around it as well.

6.4 Conclusions

The adjustment of weights and compatibility values gives a good and balanced end result for the constrained tGAP structure for the test area used in this research. It is well worth researching whether there can be found a connection between the distribution of overlapping areas between two datasets and the weights for the constrained tGAP structure to be derived from that. If this is the case no iteration will be needed anymore. For this research more test datasets need to be inspected.

The road objects caused most problems in this test dataset. It is therefore wise for the administrators of road objects to cut these objects into well defined smaller pieces to be able to control them, not only for generalisation purposes. Lots of municipalities in The Netherlands are currently working on the conversion from a line GBKN to a polygon GBKN and this is one of the issues they have to solve.

The constraint in the constrained tGAP structure appeared to be quite necessary. If we apply the tGAP without constraint we see immediately the enormous expansion of road objects whereas we want them to be canalised and reduced in the end result. This canalisation is hard to create with compatibility values, but also worth more research.

7. Conclusions, recommendations and future research

The aim of this report has been to answer the following question:

How can a vario-scale IMGeo be designed and developed by applying the constrained tGAP structure with Top10NL as initial constraint?

This chapter will first answer this question based on the findings of the master thesis in section 7.1. After this recommendations will be given in 7.2 and suggestions for future research will be done in section 7.3.

7.1 Conclusions

The municipality of Rotterdam wanted to find a way to be able to generate vario-scale maps out of one data source through this research. From the beginning it has been stated that this research is far ahead of the current situation in Rotterdam. During this research it showed that Rotterdam is really working to cooperate with other municipal services to get to a situation where large scale topographical data is really produced at one place and used by all other parties within the municipality. Whether this dataset will be used in the future as a source for the medium- and small scale map, was the main question of this research. The conclusion can be drawn that the intention to get there certainly is present. Gemeentewerken Rotterdam has been one of the parties that initiated IMGeo. For now IMGeo is defined as a large scale topographical model, but the vision is to have a vario-scale topographical model in the future.

The research issues in section 1.3 addressed questions about the organisation of the municipality of Rotterdam and the way a generalisation structure would fit in this. The organisation was studied in chapter 2 and it can be concluded that generating the medium scale base map from the large scale topographical base map is still some steps ahead of the current processes within the municipality of Rotterdam.

IMGeo and Top10NL are the datasets that have been investigated during this research. Both are based on NEN3610, but both models are developed independently. If the developers of IMGeo should have looked closer at the definitions of Top10NL, differences in details between the models could have been avoided. Since these differences do exist, this is something more care could have been taken of.

Combining both datasets in this research it has been found that the differences between the models are geometrical as well as semantical. The geometrical differences are there because the collection rules differ and because the accuracy in the IMGeo model is higher than in Top10NL. Also a small amount of the geometrical differences can be attributed to generalisation operations like displacement. These geometrical differences lead to the fact that a lot of pre-processing needs to be done to really make a good connection between the models. The semantical differences in the models, or the errors in the classification, lead to the conclusion that no hierarchy between the object classes in both models can be created.

The test data of both models showed also that a lot of pre-processing is necessary to enable it as a source for the constrained tGAP. Both models need to be an area partition. IMGeo is supposed to be an area partition at relative height level 0; however, the test data, originating from a pilot with IMGeo in Almere, was not an area partition. This wasn't implemented yet in the conversion. Top10NL is not yet an area partition. By doing geo-processing operations in ArcGIS it was possible to make both models an area partition, but those operations can't be done backwards. For this reason it will never be possible to extract a Top10NL file according to the current product specifications using the constrained tGAP tree. If the Kadaster would change the product specification to an area partition, the possibility to produce Top10NL using the constrained tGAP structure would increase.

Although the conclusion might be drawn that current Top10NL data can't be extracted from IMGeo data, the conclusion can be drawn that IMGeo and Top10NL are very well combinable. The fact that most of the objects in both datasets are polygons makes them suitable for a combination in the tGAP structure with IMGeo objects as the basis and Top10NL objects as the regions.

From the four methods that have been investigated to assign an IMGeo object to a Top10NL-region in the pre-processing phase the 'building first'-method is the best. This method assigns all IMGeo buildings to a building region even if they only overlap a little. All the other objects are processed according to the 'maximum area'-method; the Top10NL object overlapping most with the IMGeo object is the region to which the IMGeo object is assigned to.

Repeated trial of class weight- and compatibility values gave the right values for the weights and the compatibility matrix. These values appeared to be about correct for the larger dataset as well, which was tested with the same method and values, only small adjustments were needed. A conclusion that can be drawn here is that the position of road objects in the generalisation process is very crucial; a better division of road objects into smaller pieces would absolutely lead to better generalisation results.

The final conclusion is that the constrained tGAP certainly offers possibilities to extract medium scale topography from large scale topography, but that this method will probably have a long way towards an implementation in a commercial product. Whether the tGAP can be used in Rotterdam will therefore depend on when the municipality of Rotterdam wants to realize their vision to maintain topographical data only at the largest scale and how fast the progress of the tGAP towards a commercial product will be; both the organisation of Gemeentewerken Rotterdam and the TU Delft with its tGAP structure have a long term vision.

7.2 Recommendations

The recommendations are treated in two blocks. First of all recommendations to the municipality of Rotterdam and other municipalities are given, after this recommendations to the Kadaster about cooperation and the specifications of Top10NL are given. Recommendations for future research is treated separately.

Rotterdam
A recommendation can be made to municipalities that want to convert their GBKN to IMGeo data. The pilot data of Almere was in some cases not according to the standard. If a municipality chooses to turn to this standard, all things have to fit. If for example the data at relative height level 0 is not an area partition, it is still hard to use as input for the tGAP structure. The conversion of data can also lead to inconsistencies in the dataset. In the pilot data of Almere doubled objects with unique ID's showed up. These errors don't need to be

the consequence of conversions, the errors can also have been in the source data, but it is something to take into account when doing the conversion.

To road objects should be paid extra attention in the future, especially in IMGeo. To keep objects controlable, not only with respect to generalisation, they need to be cut into smaller pieces, most likely at junctions (Uitermark et al., 1999). Municipalities that are currently making a polygon GBKN, such as IMGeo, should be especially aware of this problem.

The overall recommendation to the municipality of Rotterdam is that the tGAP structure offers possibilities for the extraction of vario-scale data out of large scale topographical data, but it is not the only way. By the time the municipality wants to implement this, the tGAP must not be seen as the only way to generalise. Walking in front of this Gemeentewerken Rotterdam could look for partners in their search for ways to solve this issue. Within the projects DURP Ondergronden and MobiMaps parties like ESRI are also interested in generalising large scale topographical data. TU Delft and ITC are always interested in renewing initiatives. Rotterdam could play the role of the perfect playground for researching parties to test their methods on, assumed that Rotterdam stays a renewing municipality in the field of geo-information.

Kadaster
The cooperation between IMGeo and Top10NL will hopefully improve. Both models are created apart from each other and are not related to each other, besides through NEN3610. To come to an automatic generalisation structure in which the constraint plays a role it is necessary to get to a class hierarchy, which only can be made if Top10NL is derived from IMGeo or if IMGeo is the detailed version of Top10NL.

To come to a situation in which medium scale topographical data and further scales can be derived from large scale topographical data, there must be agreement on what should be the content of the derived scales. The project IMTop is a good example of a process to come to content derived from generalised data. The recommendation can be made to include also large scale topographical data in this research.

The product specification of Top10NL is currently that it is not an area partition. This has led to a lot of pre-processing in this thesis work. Since the real world actually is an area partition, the recommendation can be made to the Kadaster to consider to change the specification of Top10NL in order to make it an area partition.

During this research continuously the translation between a polygon structure and a topological structure had to be made. A recommendation to both IMGeo and Top10NL is to consider the use of a topological structure for the data model instead of a polygon structure. A topological structure will save storage space and avoid topological errors in models.

7.3 Future research

The concept of a constrained tGAP hasn't been totally developed after this research. In the summer of 2007 it was only a concept developed by the universities of Delft and Hannover. There is plenty of room for improvement and further research after this research.

In this research the pre-processing of the data is done with polygons; this gave problems, also due to the fact that a standard deviation is used in these polygon computation in ArcGIS. In a topological structure there is no room for these standard deviations; it is more strict. Probably other problems will arise when going to a topological structure, but this is something to be researched. A topographical model with a topological structure also requires antoher way of thinking; Delft University of Technology can play a large role in creating support for this idea for instance in convincing GeoNovum of the necessity of a topological structure.

To improve the final results of this master thesis a line generalisation algorithm is really necessary. The presentation of the final results is not optimal, because of the lack of line simplification. This is currently subject of research at the group GISt at Delft University of Technology.

During this research the region class of a constraint object hasn't been taken into account. It is recommended to do this in the future to prevent errors with e.g. road objects filling up large areas. The region class can be a weighing factor in determining the most compatible neighbour, making the compatibility matrix a 3D matrix; if this is not enough to get the desired end result (e.g. because of semantical mismatches), the region class can also be forced in the last merging step of the region.

A possible drawback of the tGAP structure is the fact that it has to be an area partition. With a reality which is more and more expanding in the third dimension the need for a 3D solution for the tGAP will also be larger and larger. Most topographical registrations nowadays at least have an attribute 'relative heigth', which enables us to put more than 1 object at the same place. This can't be done in the tGAP right now, because of the strict rule of the area partition. More research will be needed here to come to a solution.

Another drawback of the tGAP is the fact that importance values can't be translated to a scale. With the constrained tGAP this can possibly change. The scale of the end result is known and the scale of the starting point is known. By looking at the size of the merged objects scales in between possibly can be derived.

The weights in the constrained tGAP can possibly be determined by studying the amount of overlap between the models. This can't be determined on the basis of only one test dataset. To determine whether this relation exists studying more datasets is needed.

References

Anders, K.-H., Bobrich, J., *MRDB approach for automatic incremental update.* ICA Workshop on Generalisation and Multiple Representation, Leicester 2004, 2004.

Basic model Geo-information (in Dutch), NEN 3610, Nederlands Normalisatie Instituut, 2005.

Bakker, N., Bruns, B., Storm, M., *Data model Top10NL (in Dutch),* version 2.3, February 2005, (Topografische Dienst) Kadaster and International Institute For Geo-Information and Earth Observation (ITC), 2005.

Bobrich, J., *Cartographic Map Generalisation in Urban Districts.* In Proceedings of GIS Research UK, 9th Annual Conference, pp. 513-515, 2001.

Boelhouwer, M., *The future of small-scale topography in Rotterdam (in Dutch),* Gemeentewerken Rotterdam, project code SAE30306, 2006.

Campschroer, J., Krijgsman, W., Weerd, L. van der, *Architectuur van het stelsel; deel I:Het wezen van het stelsel (in Dutch),* november 2006, versie 1.3, Stroomlijning Basisgegevens, 2006.

CHANGE Kurzdokumentation. Hannover: Institut für Kartographie und Geoinformatik, (on-line). Universität Hannover, 2003.

Damen, J., *Building block generalisation (to be published),* Master Thesis, Utrecht University, 2008.

Downey, A., *How to think like a (Python) programmer,* Green Tea Press, Needham, Massachusets, 2007.

Galanda, M., *Automated Polygon Generalization in a Multi Agent System.* PhD thesis, University of Zürich, 2003.

Google Earth, earth.google.com, 2008.

Product informaton GBKN (in Dutch), www.gbkn.nl, LSV GBKN (on-line), 2006.

Hampe, M., Anders, K.-H., Sester, M., *MRDB applications for data revision and real-time generalisation.* In: Proceedings of the 21st International Cartographic Conference. Durban, 10.–16. August 2003, pp.192–201, 2003.

Hampe, M., Sester, M., Harrie, L., *Multiple representation databases to support visualisation on mobile devices.* In: Proceedings of the 20th ISPRS Congress, July12-23, 2004, Istanbul, Turkey: *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 135-140, 2004.

Haunert, J.-H., *Efficient area aggregation by combination of different techniques.* In: Proc. of 10th ICA Workshop on Generalisation and Multiple Representation, 2–3 August 2007, Moscow, Russia, 2007.

Haunert, J. H., Dilo, A., Oosterom, P. van, *Using the constrained tGAP for progressive vector data transfer,* 2007.

Hidding, J., Uitermark, H.T., *GBKN-buildings in Top10NL (in Dutch),* In: Geo-Info 2006-1, pp.20-22, 2006.

Hofman, A. M., *Application of generalisation in Rotterdam; a literature study.* TU Delft, www.gdmc.nl, 2007.

ICA, *Multilingual Dictionary of Technical Terms in Cartography.* Franz Steiner Verlag, Wiesbaden, 1973.

ISO, *ISO 19109:2005 Geographic Information - Rules for Application Schema.* International Organisation for Standardization, 2005.

IMGeo version 1.0 (in Dutch), 2007.

Kresse, W., Fadaie, K., *ISO Standards for Geographic Information*. Springer, Heidelberg, New York, Tokyo, 2004.

Landelijk Samenwerkingsverband GBKN, *GBKN towards an authentic registration? (in Dutch),* Documentnr. 02.05/053, 2002.

Landelijk Samenwerkingsverband GBKN, *GBKN specificaties in NEN 1878; LSV-specificaties, versie 1.1 (in Dutch),* Documentnr. 04.05/070, 2004.

Lentjes, P., *Structure of Top10NL (in Dutch),* Powerpoint presentation by Alterra, Wageningen UR, April 3rd 2007, 2007.

Mc Master, R.B., Shea, K.S., *Generalization in digital cartography,* Association of American Geographers, 1992.

Meijers, M., *Implementation and testing of variable scale topological data structures,* Master Thesis TU Delft, 2006.

Müller, J., Lagrange, J., and Weibel, R., *GIS and Generalization: Methodology and Practice.* Number 1 in GISDATA. Taylor & Francis, 1995.

Neuffer, D., Hopewell, T., Woodsford, P., *Integration of Agent-based Generalisation with Mainstream Technologies and other System Components,* ICA Workshop on Generalisation and Multiple Representation, Leicester 2004, 2004.

Ormeling, F.J., Kraak, M.J., *Cartography: Visualisation of Spatial Data (in Dutch),* Delft: Delftse Universitaire Pers, 1993.

Oosterom, P.J.M. van, *The GAP-tree, an approach to "on-the-fly" map generalisation of an area partitioning,* In: volume GIS and Generalization: Methodology and Practice of GISDATA, chapter 9, pages 120–132. Taylor & Francis, London, 1993.

Oosterom, P.J.M. van, *Variable-scale topological data structures suitable for progressive data transfer: the GAP-face tree and GAP-edge forest*, In: Cartography and geographic information science, 32, pp. 331-346, 2005.

Oosterom, P.J.M. van, Vries M. de, Meijers, M., *Vario-scale data server in a web service context,* In: Proceedings of the ICA Commission Workshop on Map Generalisation and Multiple Representation, June 25th 2006, 14 p., 2006.

Putten, J.D. van, Oosterom, P.J.M. van, *Generaliseren van vlakkenpartities (1); GAP-trees, theorie en implementatie (in Dutch)*, In: Geodesia, Volume 42, 10, pp. 443-448, 2000.

Putten, J.D. van, Oosterom, P.J.M. van, *Generaliseren van vlakkenpartities (2); GAP-trees, testresultaten en verbeteringen (in Dutch)*, In: Geodesia, Volume 42, 11, pp. 499-505, 2000.

Rietdijk, M., Verhoef, J.J., *Address unknown (in Dutch),* Uitgever Programma Stroomlijning Basisgegevens, Den Haag, 2002.

Schravendeel, D., Steen, K. van der, Rij, K. van, Rietdijk, M., Heemskerk-van Holtz, P., *Studyreport on Systematic deepening (in Dutch),* Ministerie van VROM, documentnr. 050228 Stelselrapport versie 03 (concept), 2005.

Sester, M., *Optimization approaches for generalization and data abstraction.* International Journal of Geographical Information Science 19(8-9), pp. 871-897, 2005.

Spaccapietra, S., Vangenot, C., Parent, C., Zimányi, E*., MurMur: A Research Agenda on Multiple Representations.* In: International Symposium on Database Applications in Non-Traditional Environments, DANTE 1999. 28-30 November 1999, Kyoto, Japan. IEEE Computer Society 1999, ISBN 0-7695-0496-5, pp. 373-384, 1999.

Stoter, J., Quak, W., Van Oosterom, P.J.M., Meijers, M., Lemmens, R., Uitermark, H., *Considerations for the design of a semantic data model for a multi-representation topographical database.* In: Proceedings, ISGI 2007, International Symposium on Generalisation of Information, 2007.

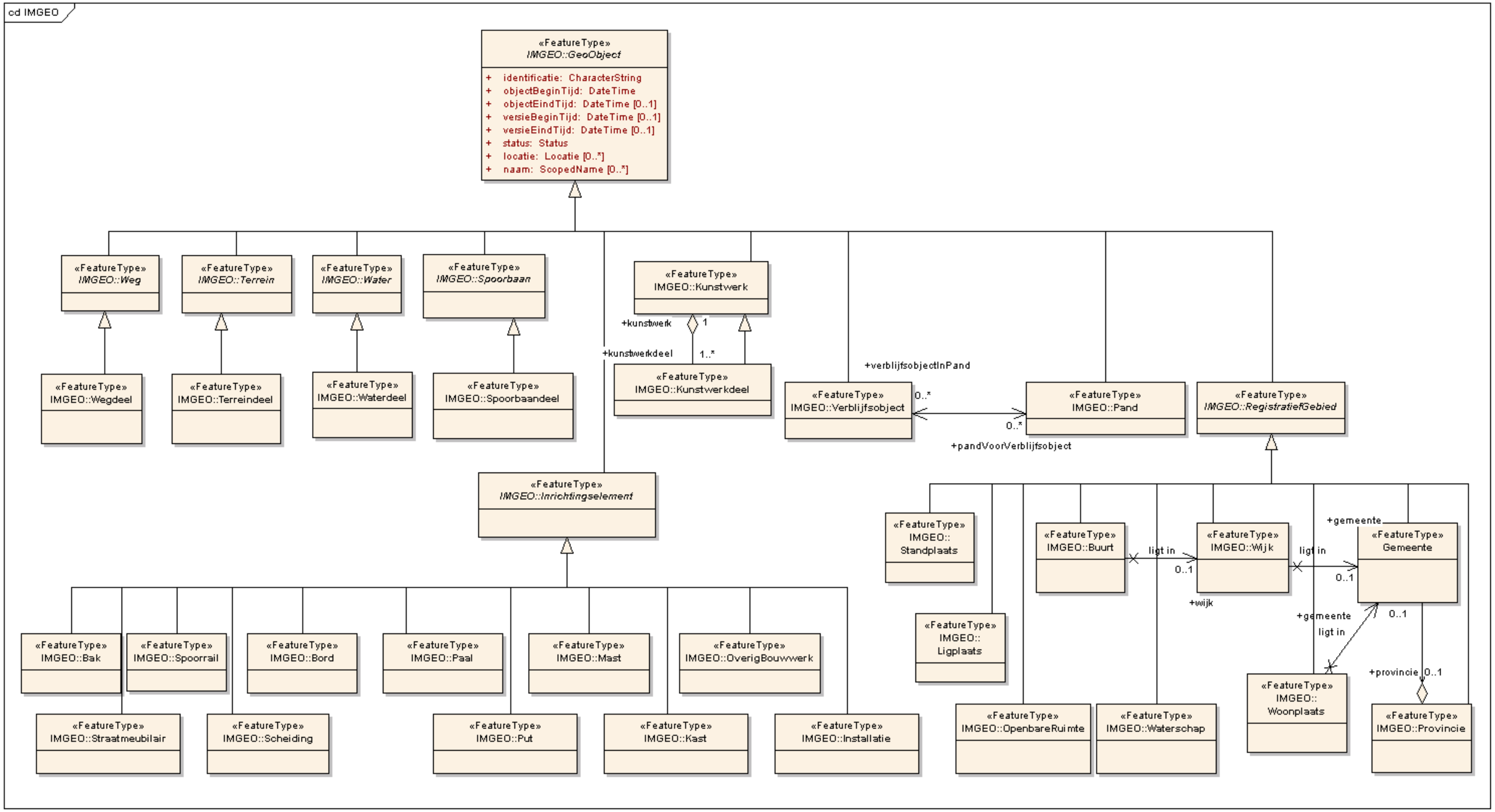
Stoter, J., Lemmens, R., Köbben, B., *Preliminary investigation TopNL (in Dutch),* International Institute for Geo-information Science and Earth Observation (ITC), 2007.

Uitermark, H., Vogels, A., Oosterom, P. van, *Semantic and Geometric Aspects of Integrating Road Networks.* In proceedings of the 2nd International conference on Interoperating Geographic Information Systems, 10-12 March'99, Zurich, LNCS 1580, Springer-Verlag, pages 177-188, 1999.

Ware, M.J., Jones, C.B., Thomas, N., *Automated map generalization with multiple operators: a simulated annealing approach.* International Journal of Geographical Information Science, vol 17(no. 8): pages 743-769. December 2003, 2003.

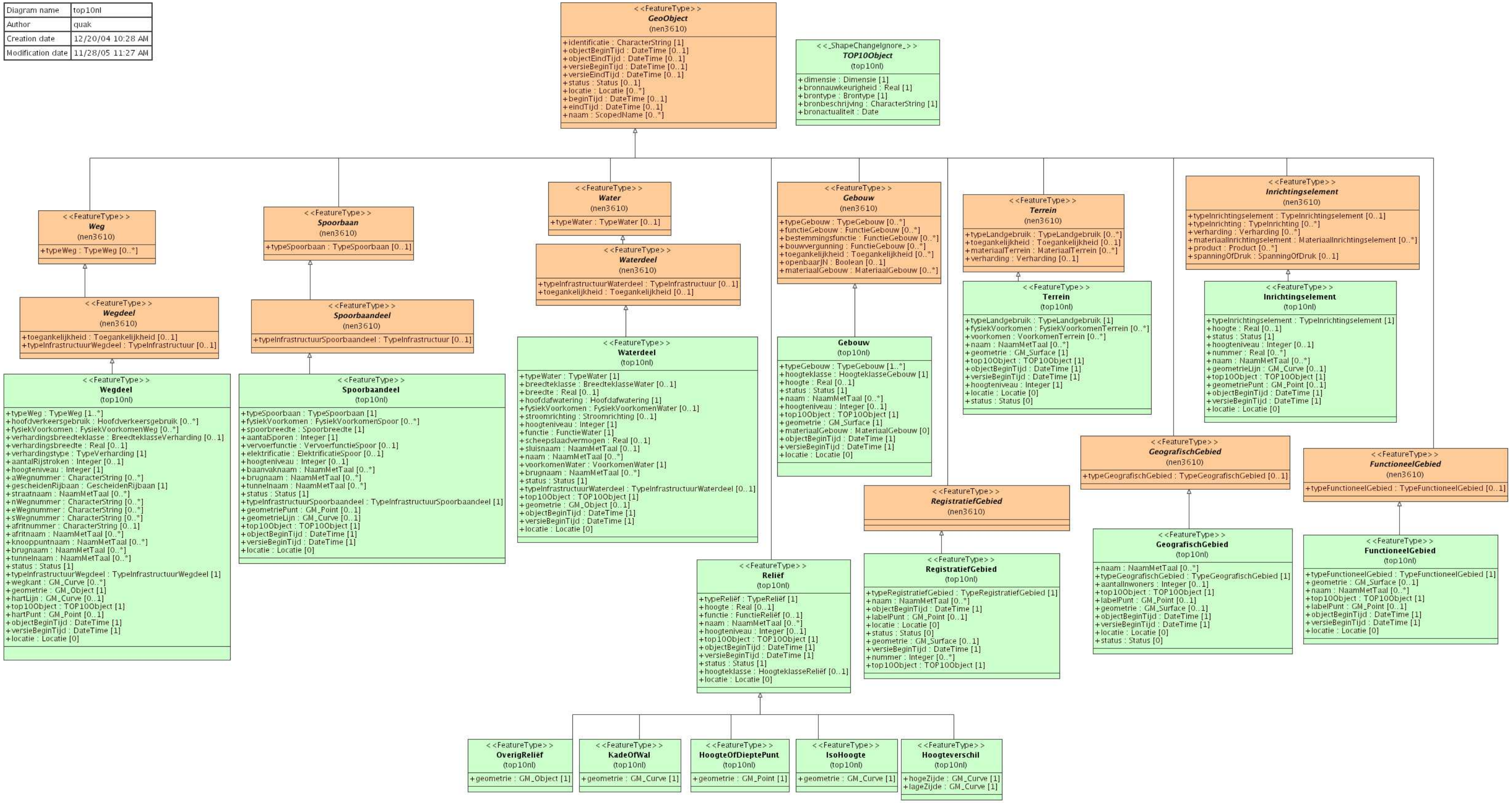
List of appendices

Appendix A: UML class diagram of IMGeo



Appendix B: UML class diagram of Top10NL

Diagram name	top10nl
Author	quak
Creation date	12/20/04 10:28 AM
Modification date	11/28/05 11:27 AM



Appendix C: Type of geometry of spatial attributes in IMGeo

Objectklasse	Attribuut		Attribuutwaarde	punt	lijn	vlak
geo-object	identificatie		< identificerende waarde >			
geo-object	objectBeginTijd		< systeemtijd >			
geo-object	objectEindTijd		< systeemtijd >			
geo-object	versieBeginTijd		< systeemtijd >			
geo-object	versieEindTijd		< systeemtijd >			
geo-object	status		plan			
geo-object	status		bestaand			
geo-object	status		historie			
geo-object	locatie		Aanduiding van locatie door adresgegevens			
geo-object	naam		Benaming van het geo-object			
wegdeel	type infrastructuur		verbinding			x
wegdeel	type infrastructuur		kruising			x
wegdeel	type infrastructuur		vlakte			x
wegdeel	type weg		OV-baan			x
wegdeel	type weg		overweg			x
wegdeel	type weg		pad			x
wegdeel	type weg		parkeervlak			x
wegdeel	type weg		perron (voor tramverkeer)			x
wegdeel	type weg		rijbaan			x
wegdeel	type weg		rijwielpad			x
wegdeel	type weg		vluchtheuvel			x
wegdeel	type weg		voetgangersgebied			x
wegdeel	type weg		voetpad			x
wegdeel	type weg		wegberm			x
wegdeel	type weg		woonerf			x
wegdeel	type weg		onbekend (IMGeo:nader te bepalen)			x
wegdeel	verhardingstype		gesloten verharding			x
wegdeel	verhardingstype		open verharding			x
wegdeel	verhardingstype		onverhard			x
wegdeel	hoogteniveau		< getal >			x
spoorbaandeel	type infrastructuur		verbinding			x
spoorbaandeel	type infrastructuur		kruising			x
spoorbaandeel	type infrastructuur		vlakte			x
spoorbaandeel	type spoorbaan		trein			x
spoorbaandeel	type spoorbaan		tram			x
spoorbaandeel	type spoorbaan		metro			x
spoorbaandeel	type spoorbaan		(haven)kraan			x
spoorbaandeel	type spoorbaan		sneltram, lightrail			x
spoorbaandeel	type spoorbaan		onbekend (IMGeo:nader te bepalen)			x
spoorbaandeel	hoogteniveau		< getal >			x
waterdeel	type infrastructuur		verbinding			x
waterdeel	type infrastructuur		kruising			x
waterdeel	type infrastructuur		vlakte			x

Objectklasse	Attribuut		Attribuutwaarde	punt	lijn	vlak
waterdeel	type water		waterloop			x
waterdeel	type water		meer, plas, ven, vijver			x
waterdeel	type water		greppel, droge sloot			x
waterdeel	type water		zee			x
waterdeel	type water		droogvallend			x
waterdeel	type water		bron, wel			x
waterdeel	type water		onbekend (IMGeo:nader te bepalen)			x
waterdeel	hoogteniveau		< getal >			x
gebouw	pand					x
gebouw	hoogteniveau		< getal >			x
terrein	type landgebruik		bos			x
terrein	type landgebruik		bedrijfsterrein			x
terrein	type landgebruik		braakliggend terrein			x
terrein	type landgebruik		cultuurgrond			x
terrein	type landgebruik		erf			x
terrein	type landgebruik		gras			x
terrein	type landgebruik		natuur en landscahp			x
terrein	type landgebruik		overig groenobject			x
terrein	type landgebruik		plantvak			x
terrein	type landgebruik		recreatieterrein			x
terrein	type landgebruik		sportterrein			x
terrein	type landgebruik		talud			x
terrein	hoogteniveau		< getal >			x
terrein	type verharding		gesloten verharding			x
terrein	type verharding		open verharding			x
terrein	type verharding		onverhard			x
kunstwerk	type kunstwerk		bassin		x	x
kunstwerk	type kunstwerk		brug		x	x
kunstwerk	type kunstwerk		damwand		x	x
kunstwerk	type kunstwerk		duiker		x	x
kunstwerk	type kunstwerk		fly-over		x	x
kunstwerk	type kunstwerk		loopbrug		x	x
kunstwerk	type kunstwerk		perron (voor treinverkeer)		x	x
kunstwerk	type kunstwerk		sluis		x	x
kunstwerk	type kunstwerk		strekdam		x	x
kunstwerk	type kunstwerk		tunnel		x	x
kunstwerk	type kunstwerk		viaduct		x	x
kunstwerk	type kunstwerk		waterkering		x	x
kunstwerk	type kunstwerk		nader te bepalen		x	x
kunstwerk	hoogteniveau		<getal>		x	x
inrichtingselement	type inrichtingselement	type bak	afval apart plaats	x		x
inrichtingselement	type inrichtingselement	type bak	afvalbak	x		x
inrichtingselement	type inrichtingselement	type bak	drinkbak	x		x
inrichtingselement	type inrichtingselement	type bak	plantenbak	x		x
inrichtingselement	type inrichtingselement	type bak	zandbak	x		x
inrichtingselement	type inrichtingselement	type bord	informatiebord	x	x	x
inrichtingselement	type inrichtingselement	type bord	plaatsnaambord	x	x	x
inrichtingselement	type inrichtingselement	type bord	straatnaambord	x	x	x
inrichtingselement	type inrichtingselement	type bord	verkeersbord	x	x	x

Objectklasse	Attribuut		Attribuutwaarde	punt	lijn	vlak
inrichtingselement	type inrichtingselement	type bord	verklikker transportleiding	x	x	x
inrichtingselement	type inrichtingselement	type installatie	boorgat	x		x
inrichtingselement	type inrichtingselement	type installatie	brandstofpomp	x		x
inrichtingselement	type inrichtingselement	type installatie	peilbuis	x		x
inrichtingselement	type inrichtingselement	type installatie	windturbine	x		x
inrichtingselement	type inrichtingselement	type kast	CAI-kast	x		x
inrichtingselement	type inrichtingselement	type kast	elektrakast	x		x
inrichtingselement	type inrichtingselement	type kast	gaskast	x		x
inrichtingselement	type inrichtingselement	type kast	KPN kast	x		x
inrichtingselement	type inrichtingselement	type kast	rioolkast	x		x
inrichtingselement	type inrichtingselement	type kast	schakelkast openbare verlichting	x		x
inrichtingselement	type inrichtingselement	type kast	trafo	x		x
inrichtingselement	type inrichtingselement	type kast	verkeersinstallatiekast	x		x
inrichtingselement	type inrichtingselement	type mast	bovenleidingmast	x	x	x
inrichtingselement	type inrichtingselement	type mast	hoogspanningsmast	x	x	x
inrichtingselement	type inrichtingselement	type mast	laagspanningsmast	x	x	x
inrichtingselement	type inrichtingselement	type mast	straalzender	x	x	x
inrichtingselement	type inrichtingselement	type mast	zendmast	x	x	x
inrichtingselement	type inrichtingselement	type overig bouwwerk	bordes			x
inrichtingselement	type inrichtingselement	type overig bouwwerk	luifel			x
inrichtingselement	type inrichtingselement	type overig bouwwerk	overigBouwwerk			x
inrichtingselement	type inrichtingselement	type overig bouwwerk	steiger			x
inrichtingselement	type inrichtingselement	type paal	afsluitpaal	x		
inrichtingselement	type inrichtingselement	type paal	bolder	x		
inrichtingselement	type inrichtingselement	type paal	haltepaal	x		
inrichtingselement	type inrichtingselement	type paal	hectometerpaal	x		
inrichtingselement	type inrichtingselement	type paal	lantaarnpaal	x		
inrichtingselement	type inrichtingselement	type paal	meerpaal	x		
inrichtingselement	type inrichtingselement	type paal	paal/steen	x		
inrichtingselement	type inrichtingselement	type paal	parkeerautomaat	x		
inrichtingselement	type inrichtingselement	type paal	praatpaal	x		
inrichtingselement	type inrichtingselement	type paal	recalmezuil	x		
inrichtingselement	type inrichtingselement	type paal	remmingswerk	x		
inrichtingselement	type inrichtingselement	type paal	seinpaal	x		
inrichtingselement	type inrichtingselement	type paal	verkeerslicht	x		
inrichtingselement	type inrichtingselement	type paal	verkeerszuil	x		
inrichtingselement	type inrichtingselement	type paal	vlaggenmast	x		
inrichtingselement	type inrichtingselement	type paal	wegwijzer	x		
inrichtingselement	type inrichtingselement	type put	benzine- /olieput	x		
inrichtingselement	type inrichtingselement	type put	brandkraan /-put	x		
inrichtingselement	type inrichtingselement	type put	drainageput	x		
inrichtingselement	type inrichtingselement	type put	gasput	x		
inrichtingselement	type inrichtingselement	type put	inspectieput	x		
inrichtingselement	type inrichtingselement	type put	kolk	x		
inrichtingselement	type inrichtingselement	type put	waterleidingput	x		
inrichtingselement	type inrichtingselement	type scheiding	geluidsscherm		x	x
inrichtingselement	type inrichtingselement	type scheiding	heg		x	x
inrichtingselement	type inrichtingselement	type scheiding	hek		x	x
inrichtingselement	type inrichtingselement	type scheiding	kademuur		x	x

Objectklasse	Attribuut		Attribuutwaarde	punt	lijn	vlak
inrichtingselement	type inrichtingselement	type scheiding	muur		x	x
inrichtingselement	type inrichtingselement	type scheiding	terreinscheiding		x	x
inrichtingselement	type inrichtingselement	type scheiding	walbescherming		x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	abil	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	boom	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	brievenbus	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	fietsenrek	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	kunstobject	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	openbaar toilet	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	oprit	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	slagboom	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	speelwerktuig	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	straatmeubilair (overig)	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	telefooncel	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	trap	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	verkeersdrempel	x	x	x
inrichtingselement	type inrichtingselement	type straatmeubilair	zitbank	x	x	x
inrichtingselement	type inrichtingselement	type spoorrail	(haven)kraan		x	
inrichtingselement	type inrichtingselement	type spoorrail	metro		x	
inrichtingselement	type inrichtingselement	type spoorrail	tram		x	
inrichtingselement	type inrichtingselement	type spoorrail	trein		x	
inrichtingselement	type inrichtingselement	type spoorrail	sneltram, lightrail		x	
inrichtingselement	hoogteniveau		< getal >	x	x	x
registratief gebied	type registratief gebied		standplaats			x
registratief gebied	type registratief gebied		ligplaats			x
registratief gebied	type registratief gebied		openbare ruimte			x
registratief gebied	type registratief gebied		buurt			x
registratief gebied	type registratief gebied		waterschap			x
registratief gebied	type registratief gebied		wijk			x
registratief gebied	type registratief gebied		woonplaats			x
registratief gebied	type registratief gebied		gemeente			x
registratief gebied	type registratief gebied		provincie			x

Note: This table has also been used and is made in cooperation with the project group IMTop.

Appendix D: Type of geometry of sptial attributes in Top10NL

Objectklasse	Attribuut	Attribuutwaarde	punt	lijn	vlak
wegdeel	identificatie	< identificerende waarde >			
wegdeel	objectBeginTijd	< systeemtijd >			
wegdeel	objectEindTijd	< systeemtijd >			
wegdeel	versieBeginTijd	< systeemtijd >			
wegdeel	versieEindTijd	< systeemtijd >			
wegdeel	brontype	luchtfoto			
wegdeel	brontype	kaart			
wegdeel	brontype	RD			
wegdeel	brontype	GBKN			
wegdeel	brontype	top10vector			
wegdeel	brontype	overig			
wegdeel	bronbeschrijving	< tekst >			
wegdeel	bronactualiteit	< datum >			
wegdeel	bronnauwkeurigheid	< getal >			
wegdeel	dimensie	2D			
wegdeel	dimensie	3D			
wegdeel	type infrastructuur	verbinding		x	x
wegdeel	type infrastructuur	kruising	x		x
wegdeel	type infrastructuur	overig verkeersgebied			x
wegdeel	type weg	autosnelweg	x	x	x
wegdeel	type weg	hoofdweg	x	x	x
wegdeel	type weg	regionale weg	x	x	x
wegdeel	type weg	lokale weg	x	x	x
wegdeel	type weg	straat	x	x	x
wegdeel	type weg	startbaan, landingsbaan	x	x	x
wegdeel	type weg	rolbaan, platform	x	x	x
wegdeel	type weg	overig	x	x	x
wegdeel	type weg	onbekend	x	x	x
wegdeel	hoofdverkeersgebruik	snelverkeer	x	x	x
wegdeel	hoofdverkeersgebruik	gemengd verkeer	x	x	x
wegdeel	hoofdverkeersgebruik	busverkeer	x	x	x
wegdeel	hoofdverkeersgebruik	fietsers, bromfietsers	x	x	x
wegdeel	hoofdverkeersgebruik	voetgangers	x	x	x
wegdeel	hoofdverkeersgebruik	ruiters	x	x	x
wegdeel	hoofdverkeersgebruik	vliegverkeer	x	x	x
wegdeel	hoofdverkeersgebruik	parkeren			x
wegdeel	hoofdverkeersgebruik	parkeren: carpoolplaats			x
wegdeel	hoofdverkeersgebruik	parkeren: P+R parkeerplaats			x
wegdeel	hoofdverkeersgebruik	overig	x	x	x
wegdeel	hoofdverkeersgebruik	onbekend	x	x	x
wegdeel	fysiek voorkomen	op vast deel van brug	x	x	x
wegdeel	fysiek voorkomen	op beweegbaar deel van brug	x	x	x
wegdeel	fysiek voorkomen	overkluisd	x	x	x
wegdeel	fysiek voorkomen	in tunnel	x	x	x
wegdeel	fysiek voorkomen	als veer/pont		x	
wegdeel	verhardingsbreedteklasse	> 7 meter	x	x	x
wegdeel	verhardingsbreedteklasse	4 - 7 meter	x	x	x
wegdeel	verhardingsbreedteklasse	2 -4 meter	x	x	x
wegdeel	verhardingsbreedteklasse	< 2 meter	x	x	

Objectklasse	Attribuut	Attribuutwaarde	punt	lijn	vlak
wegdeel	verhardingsbreedte	< werkelijke breedte in meters met 1 decimaal >	x	x	x
wegdeel	gescheiden rijbaan	ja	x	x	x
wegdeel	gescheiden rijbaan	nee			
wegdeel	verhardingstype	verhard	x	x	x
wegdeel	verhardingstype	half verhard	x	x	x
wegdeel	verhardingstype	onverhard	x	x	x
wegdeel	verhardingstype	onbekend	x	x	x
wegdeel	aantal rijstroken	< aantal >	x	x	x
wegdeel	status	realisatie: nog niet in uitvoering	x	x	x
wegdeel	status	realisatie: in uitvoering	x	x	x
wegdeel	status	in gebruik	x	x	x
wegdeel	status	buiten gebruik	x	x	x
wegdeel	status	onbekend	x	x	x
wegdeel	straatnaam (Nl)	< Nederlandse eigennaam straat >	x	x	x
wegdeel	straatnaam (Fr)	< Friese eigennaam straat >	x	x	x
wegdeel	A-wegnummer	< A-nummer weg >	x	x	x
wegdeel	N-wegnummer	< N-nummer weg >	x	x	x
wegdeel	E-wegnummer	< E-nummer weg >	x	x	x
wegdeel	S-wegnummer	< S-nummer weg >	x	x	x
wegdeel	afritnummer	< nummer afrit >	x	x	x
wegdeel	afritnaam	< naam afrit >	x	x	x
wegdeel	knooppuntnaam	< naam knooppunt >	x	x	x
wegdeel	brugnaam	< naam brug >	x	x	x
wegdeel	tunnelnaam	< naam tunnel >	x	x	x
wegdeel	hoogteniveau	< getal >	x	x	x
spoorbaandeel	identificatie	< identificerende waarde >			
spoorbaandeel	objectBeginTijd	< systeemtijd >			
spoorbaandeel	objectEindTijd	< systeemtijd >			
spoorbaandeel	versieBeginTijd	< systeemtijd >			
spoorbaandeel	versieEindTijd	< systeemtijd >			
spoorbaandeel	brontype	luchtfoto			
spoorbaandeel	brontype	kaart			
spoorbaandeel	brontype	RD			
spoorbaandeel	brontype	GBKN			
spoorbaandeel	brontype	top10vector			
spoorbaandeel	brontype	overig			
spoorbaandeel	bronbeschrijving	< tekst >			
spoorbaandeel	bronactualiteit	< datum >			
spoorbaandeel	bronnauwkeurigheid	< getal >			
spoorbaandeel	dimensie	2D			
spoorbaandeel	dimensie	3D			
spoorbaandeel	type infrastructuur	verbinding		x	
spoorbaandeel	type infrastructuur	kruising	x		
spoorbaandeel	type spoorbaan	trein	x	x	
spoorbaandeel	type spoorbaan	tram	x	x	
spoorbaandeel	type spoorbaan	metro	x	x	
spoorbaandeel	type spoorbaan	gemengd	x	x	
spoorbaandeel	fysiek voorkomen	op vast deel van brug	x	x	
spoorbaandeel	fysiek voorkomen	op beweegbaar deel van brug	x	x	
spoorbaandeel	fysiek voorkomen	overkluisd	x	x	
spoorbaandeel	fysiek voorkomen	in tunnel	x	x	
spoorbaandeel	spoorbreedte	normaalspoor	x	x	

Objectklasse	Attribuut	Attribuutwaarde	punt	lijn	vlak
spoorbaandeel	spoorbreedte	smalspoor	x	x	
spoorbaandeel	spoorbreedte	gemengd	x	x	
spoorbaandeel	aantal sporen	< nummer >	x	x	
spoorbaandeel	vervoerfunctie	gemengd gebruik	x	x	
spoorbaandeel	vervoerfunctie	personenvervoer	x	x	
spoorbaandeel	vervoerfunctie	goederenvervoer	x	x	
spoorbaandeel	vervoerfunctie	museumlijn	x	x	
spoorbaandeel	elektrificatie	geëlektrificeerd	x	x	
spoorbaandeel	elektrificatie	niet geëlektrificeerd	x	x	
spoorbaandeel	elektrificatie	gemengd	x	x	
spoorbaandeel	status	realisatie: nog niet in uitvoering	x	x	
spoorbaandeel	status	realisatie: in uitvoering	x	x	
spoorbaandeel	status	in gebruik	x	x	
spoorbaandeel	status	buiten gebruik	x	x	
spoorbaandeel	status	onbekend	x	x	
spoorbaandeel	brugnaam	< naam brug >	x	x	
spoorbaandeel	tunnelnaam	< naam tunnel >	x	x	
spoorbaandeel	baanvaknaam	< naam baanvak >	x	x	
spoorbaandeel	hoogteniveau	< getal >	x	x	
waterdeel	identificatie	< identificerende waarde >			
waterdeel	objectBeginTijd	< systeemtijd >			
waterdeel	objectEindTijd	< systeemtijd >			
waterdeel	versieBeginTijd	< systeemtijd >			
waterdeel	versieEindTijd	< systeemtijd >			
waterdeel	brontype	luchtfoto			
waterdeel	brontype	kaart			
waterdeel	brontype	RD			
waterdeel	brontype	GBKN			
waterdeel	brontype	top10vector			
waterdeel	brontype	overig			
waterdeel	bronbeschrijving	< tekst >			
waterdeel	bronactualiteit	< datum >			
waterdeel	bronnauwkeurigheid	< getal >			
waterdeel	dimensie	2D			
waterdeel	dimensie	3D			
waterdeel	type infrastructuur	verbinding		x	x
waterdeel	type infrastructuur	kruising	x		x
waterdeel	type infrastructuur	overig watergebied	x	x	x
waterdeel	type water	waterloop		x	x
waterdeel	type water	meer, plas, ven, vijver			x
waterdeel	type water	greppel, droge sloot		x	
waterdeel	type water	zee			x
waterdeel	type water	droogvallend			x
waterdeel	type water	bron, wel	x		
waterdeel	type water	onbekend	x	x	x
waterdeel	breedteklasse	0,5 - 3 meter	x	x	
waterdeel	breedteklasse	3 - 6 meter	x	x	
waterdeel	breedteklasse	> 6 meter			x
waterdeel	breedte	< werkelijke breedte in meters met 1 decimaal >		x	x
waterdeel	hoofdafwatering	ja	x	x	x
waterdeel	hoofdafwatering	nee	x	x	x
waterdeel	fysiek voorkomen	in sluis		x	x
waterdeel	fysiek voorkomen	op brug		x	x

Objectklasse	Attribuut	Attribuutwaarde	punt	lijn	vlak
waterdeel	fysiek voorkomen	in duiker		x	x
waterdeel	fysiek voorkomen	in afsluitbare duiker		x	x
waterdeel	fysiek voorkomen	in grondduiker		x	x
waterdeel	fysiek voorkomen	in afsluitbare grondduiker		x	x
waterdeel	fysiek voorkomen	overkluisd	x	x	x
waterdeel	functie	drinkwaterbekken		x	x
waterdeel	functie	haven		x	x
waterdeel	functie	natuurbad		x	x
waterdeel	functie	viskwekerij		x	x
waterdeel	functie	vistrap	x	x	x
waterdeel	functie	vloeiveld		x	x
waterdeel	functie	waterval	x	x	x
waterdeel	functie	waterzuivering		x	x
waterdeel	functie	zwembad		x	x
waterdeel	functie	overig		x	x
waterdeel	functie	onbekend		x	x
waterdeel	voorkomen	met riet			x
waterdeel	voorkomen	overig	x	x	x
waterdeel	stroomrichting	eenrichting	x	x	x
waterdeel	stroomrichting	twee richtingen (getijde invloed)		x	x
waterdeel	stroomrichting	stilstaand	x	x	x
waterdeel	scheepslaadvermogen	< laadvermogen in ton >	x	x	x
waterdeel	status	realisatie: nog niet in uitvoering	x	x	x
waterdeel	status	realisatie: in uitvoering	x	x	x
waterdeel	status	in gebruik	x	x	x
waterdeel	status	buiten gebruik	x	x	x
waterdeel	status	onbekend	x	x	x
waterdeel	naam (Nl)	< Nederlandse naam water >	x	x	x
waterdeel	naam (Fr)	< Friese naam water >	x	x	x
waterdeel	sluisnaam	< naam sluis >	x	x	x
waterdeel	brugnaam	< naam brug >	x	x	x
waterdeel	hoogteniveau	< getal >	x	x	x
gebouw	identificatie	< identificerende waarde >			
gebouw	objectBeginTijd	< systeemtijd >			
gebouw	objectEindTijd	< systeemtijd >			
gebouw	versieBeginTijd	< systeemtijd >			
gebouw	versieEindTijd	< systeemtijd >			
gebouw	brontype	luchtfoto			
gebouw	brontype	kaart			
gebouw	brontype	RD			
gebouw	brontype	GBKN			
gebouw	brontype	top10vector			
gebouw	brontype	overig			
gebouw	bronbeschrijving	< tekst >			
gebouw	bronactualiteit	< datum >			
gebouw	bronnauwkeurigheid	< getal >			
gebouw	dimensie	2D			
gebouw	dimensie	3D			
gebouw	type gebouw	brandtoren			x
gebouw	type gebouw	bezoekerscentrum			x
gebouw	type gebouw	bunker			x
gebouw	type gebouw	crematorium			x
gebouw	type gebouw	deelraadsecretarie			x

Objectklasse	Attribuut	Attribuutwaarde	punt	lijn	vlak
gebouw	type gebouw	dok			x
gebouw	type gebouw	elektriciteitscentrale			x
gebouw	type gebouw	fabriek			x
gebouw	type gebouw	fort			x
gebouw	type gebouw	gascompressiestation			x
gebouw	type gebouw	gemaal			x
gebouw	type gebouw	gemeentehuis			x
gebouw	type gebouw	gevangenis			x
gebouw	type gebouw	grenskantoor			x
gebouw	type gebouw	hotel			x
gebouw	type gebouw	huizenblok			x
gebouw	type gebouw	hulpsecretarie			x
gebouw	type gebouw	kapel			x
gebouw	type gebouw	kas, warenhuis			x
gebouw	type gebouw	kasteel			x
gebouw	type gebouw	kerk			x
gebouw	type gebouw	kerncentrale, kernreactor			x
gebouw	type gebouw	kliniek, inrichting, sanatorium			x
gebouw	type gebouw	klokkentoren			x
gebouw	type gebouw	klooster, abdij			x
gebouw	type gebouw	koeltoren			x
gebouw	type gebouw	koepel			x
gebouw	type gebouw	kunstijsbaan			x
gebouw	type gebouw	lichttoren			x
gebouw	type gebouw	luchtwachttoren			x
gebouw	type gebouw	manege			x
gebouw	type gebouw	metrostation			x
gebouw	type gebouw	militair gebouw			x
gebouw	type gebouw	motel			x
gebouw	type gebouw	museum			x
gebouw	type gebouw	parkeerdak, parkeerdek, parkeergarage			x
gebouw	type gebouw	peilmeetstation			x
gebouw	type gebouw	politiebureau			x
gebouw	type gebouw	pompstation			x
gebouw	type gebouw	postkantoor			x
gebouw	type gebouw	psychiatrisch ziekenhuis, psychiatrisch centrum			x
gebouw	type gebouw	radarpost			x
gebouw	type gebouw	radartoren			x
gebouw	type gebouw	radiotoren, televisietoren			x
gebouw	type gebouw	recreatiecentrum			x
gebouw	type gebouw	reddingboothuisje			x
gebouw	type gebouw	reddinghuisje, schuilhut			x
gebouw	type gebouw	religieus gebouw			x
gebouw	type gebouw	remise			x
gebouw	type gebouw	ruïne			x
gebouw	type gebouw	schaapskooi			x
gebouw	type gebouw	school			x
gebouw	type gebouw	schoorsteen			x
gebouw	type gebouw	sporthal			x
gebouw	type gebouw	stadion			x
gebouw	type gebouw	stadskantoor			x
gebouw	type gebouw	tank			x
gebouw	type gebouw	tankstation			x

Objectklasse	**Attribuut**	**Attribuutwaarde**	**punt**	**lijn**	**vlak**
gebouw	type gebouw	telecommunicatietoren			x
gebouw	type gebouw	toren			x
gebouw	type gebouw	transformatorstation			x
gebouw	type gebouw	treinstation			x
gebouw	type gebouw	uitzichttoren			x
gebouw	type gebouw	universiteit			x
gebouw	type gebouw	veiling			x
gebouw	type gebouw	verkeerstoren			x
gebouw	type gebouw	vuurtoren			x
gebouw	type gebouw	waterradmolen			x
gebouw	type gebouw	watertoren			x
gebouw	type gebouw	wegenwachtstation			x
gebouw	type gebouw	wegrestaurant			x
gebouw	type gebouw	werf			x
gebouw	type gebouw	windmolen			x
gebouw	type gebouw	windmolen: korenmolen			x
gebouw	type gebouw	windmolen: watermolen			x
gebouw	type gebouw	windturbine			x
gebouw	type gebouw	zendtoren			x
gebouw	type gebouw	ziekenhuis			x
gebouw	type gebouw	zwembad			x
gebouw	type gebouw	overig			x
gebouw	hoogteklasse	laagbouw			x
gebouw	hoogteklasse	hoogbouw			x
gebouw	hoogteklasse	onbekend			x
gebouw	hoogte	< hoogte boven maaiveld in meters >			x
gebouw	status	realisatie: nog niet in uitvoering			x
gebouw	status	realisatie: in uitvoering			x
gebouw	status	in gebruik			x
gebouw	status	buiten gebruik			x
gebouw	status	onbekend			x
gebouw	naam (Nl)	< Nederlandse naam gebouw >			x
gebouw	naam (Fr)	< Friese naam gebouw >			x
gebouw	hoogteniveau	< getal >			x
terrein	identificatie	< identificerende waarde >			
terrein	objectBeginTijd	< systeemtijd >			
terrein	objectEindTijd	< systeemtijd >			
terrein	versieBeginTijd	< systeemtijd >			
terrein	versieEindTijd	< systeemtijd >			
terrein	brontype	luchtfoto			
terrein	brontype	kaart			
terrein	brontype	RD			
terrein	brontype	GBKN			
terrein	brontype	top10vector			
terrein	brontype	overig			
terrein	bronbeschrijving	< tekst >			
terrein	bronactualiteit	< datum >			
terrein	bronnauwkeurigheid	< getal >			
terrein	dimensie	2D			
terrein	dimensie	3D			
terrein	type landgebruik	aanlegsteiger			x
terrein	type landgebruik	akkerland			x
terrein	type landgebruik	basaltblokken, steenglooiing			x

Objectklasse	Attribuut	Attribuutwaarde	punt	lijn	vlak
terrein	type landgebruik	bebouwd gebied			x
terrein	type landgebruik	boomgaard			x
terrein	type landgebruik	boomkwekerij			x
terrein	type landgebruik	bos: gemengd bos			x
terrein	type landgebruik	bos: griend			x
terrein	type landgebruik	bos: loofbos			x
terrein	type landgebruik	bos: naaldbos			x
terrein	type landgebruik	dodenakker			x
terrein	type landgebruik	dodenakker met bos			x
terrein	type landgebruik	fruitkwekerij			x
terrein	type landgebruik	grasland			x
terrein	type landgebruik	heide			x
terrein	type landgebruik	laadperron			x
terrein	type landgebruik	populieren			x
terrein	type landgebruik	spoorbaanlichaam			x
terrein	type landgebruik	zand			x
terrein	type landgebruik	overig			x
terrein	type landgebruik	onbekend			x
terrein	fysiek voorkomen	overkluisd			x
terrein	fysiek voorkomen	in tunnel			x
terrein	fysiek voorkomen	op brug			x
terrein	voorkomen	met riet			x
terrein	voorkomen	dras, moerassig			x
terrein	naam (Nl)	< Nederlandse naam terrein >			x
terrein	naam (Fr)	< Friese naam terrein >			x
terrein	hoogteniveau	< getal >			x
inrichtingselement	identificatie	< identificerende waarde >			
inrichtingselement	objectBeginTijd	< systeemtijd >			
inrichtingselement	objectEindTijd	< systeemtijd >			
inrichtingselement	versieBeginTijd	< systeemtijd >			
inrichtingselement	versieEindTijd	< systeemtijd >			
inrichtingselement	brontype	luchtfoto			
inrichtingselement	brontype	kaart			
inrichtingselement	brontype	RD			
inrichtingselement	brontype	GBKN			
inrichtingselement	brontype	top10vector			
inrichtingselement	brontype	overig			
inrichtingselement	bronbeschrijving	< tekst >			
inrichtingselement	bronactualiteit	< datum >			
inrichtingselement	bronnauwkeurigheid	< getal >			
inrichtingselement	dimensie	2D			
inrichtingselement	dimensie	3D			
inrichtingselement	type inrichtingselement	aanlegsteiger		x	
inrichtingselement	type inrichtingselement	baak	x		
inrichtingselement	type inrichtingselement	bomenrij		x	
inrichtingselement	type inrichtingselement	boom	x		
inrichtingselement	type inrichtingselement	boorput	x		
inrichtingselement	type inrichtingselement	boortoren	x		
inrichtingselement	type inrichtingselement	BOS-pomp	x		
inrichtingselement	type inrichtingselement	brandtoren	x		
inrichtingselement	type inrichtingselement	dam, koedam	x	x	
inrichtingselement	type inrichtingselement	dukdalf	x		
inrichtingselement	type inrichtingselement	gaswinning	x		

Objectklasse	Attribuut	Attribuutwaarde	punt	lijn	vlak
inrichtingselement	type inrichtingselement	gedenkteken, monument	x		
inrichtingselement	type inrichtingselement	geluidswering		x	
inrichtingselement	type inrichtingselement	gemaal	x		
inrichtingselement	type inrichtingselement	golfmeetpaal	x		
inrichtingselement	type inrichtingselement	GPS kernnetpunt	x		
inrichtingselement	type inrichtingselement	grenspunt	x		
inrichtingselement	type inrichtingselement	heg, haag		x	
inrichtingselement	type inrichtingselement	hekwerk		x	
inrichtingselement	type inrichtingselement	helikopterlandingsplatform	x		
inrichtingselement	type inrichtingselement	hoogspanningsleiding		x	
inrichtingselement	type inrichtingselement	hoogspanningsmast	x		
inrichtingselement	type inrichtingselement	hunebed	x		
inrichtingselement	type inrichtingselement	kaap	x		
inrichtingselement	type inrichtingselement	kabelbaan		x	
inrichtingselement	type inrichtingselement	kabelbaanmast	x		
inrichtingselement	type inrichtingselement	kapel	x		
inrichtingselement	type inrichtingselement	kilometerpaal	x		
inrichtingselement	type inrichtingselement	kilometerpaal spoorweg	x		
inrichtingselement	type inrichtingselement	kilometerpaal water	x		
inrichtingselement	type inrichtingselement	kilometerraaibord	x		
inrichtingselement	type inrichtingselement	kilometerraaipaal	x		
inrichtingselement	type inrichtingselement	koeltoren	x		
inrichtingselement	type inrichtingselement	koepel	x		
inrichtingselement	type inrichtingselement	kogelvanger schietbaan	x	x	
inrichtingselement	type inrichtingselement	kraan	x		
inrichtingselement	type inrichtingselement	kruis	x		
inrichtingselement	type inrichtingselement	laadperron	x	x	
inrichtingselement	type inrichtingselement	leiding		x	
inrichtingselement	type inrichtingselement	licht, lichtopstand	x		
inrichtingselement	type inrichtingselement	lichttoren	x		
inrichtingselement	type inrichtingselement	luchtvaartlicht	x		
inrichtingselement	type inrichtingselement	markant object	x		
inrichtingselement	type inrichtingselement	muur		x	
inrichtingselement	type inrichtingselement	oliepompinstallatie	x		
inrichtingselement	type inrichtingselement	paal	x		
inrichtingselement	type inrichtingselement	paalwerk		x	
inrichtingselement	type inrichtingselement	peilmeetstation	x		
inrichtingselement	type inrichtingselement	peilschaal	x		
inrichtingselement	type inrichtingselement	pijler	x		
inrichtingselement	type inrichtingselement	radarpost	x		
inrichtingselement	type inrichtingselement	radiobaken	x		
inrichtingselement	type inrichtingselement	radiotelescoop	x		
inrichtingselement	type inrichtingselement	RD punt	x		
inrichtingselement	type inrichtingselement	schietbaan		x	
inrichtingselement	type inrichtingselement	schoorsteen	x		
inrichtingselement	type inrichtingselement	seinmast	x		
inrichtingselement	type inrichtingselement	sluisdeur	x	x	
inrichtingselement	type inrichtingselement	station		x	
inrichtingselement	type inrichtingselement	stormvloedkering	x		
inrichtingselement	type inrichtingselement	strandpaal	x		
inrichtingselement	type inrichtingselement	strekdam, krib, golfbreker		x	
inrichtingselement	type inrichtingselement	stuw	x	x	
inrichtingselement	type inrichtingselement	tol	x	x	

Objectklasse	**Attribuut**	**Attribuutwaarde**	**punt**	**lijn**	**vlak**
inrichtingselement	type inrichtingselement	toren	x		
inrichtingselement	type inrichtingselement	uitzichttoren	x		
inrichtingselement	type inrichtingselement	verkeersgeleider		x	
inrichtingselement	type inrichtingselement	visplaats	x		
inrichtingselement	type inrichtingselement	vlampijp	x		
inrichtingselement	type inrichtingselement	wegafsluiting	x	x	
inrichtingselement	type inrichtingselement	wegwijzer	x		
inrichtingselement	type inrichtingselement	windmolen	x		
inrichtingselement	type inrichtingselement	windmolen: korenmolen	x		
inrichtingselement	type inrichtingselement	windmolen: watermolen	x		
inrichtingselement	type inrichtingselement	windmolentje	x		
inrichtingselement	type inrichtingselement	windturbine	x		
inrichtingselement	type inrichtingselement	zeevaartlicht	x		
inrichtingselement	type inrichtingselement	zendmast	x		
inrichtingselement	type inrichtingselement	zichtbaar wrak	x		
inrichtingselement	type inrichtingselement	overig	x	x	
inrichtingselement	type inrichtingselement	onbekend	x	x	
inrichtingselement	hoogte	< hoogte boven maaiveld in meters >	x	x	
inrichtingselement	naam (Nl)	< Nederlandse naam inrichtingselement >	x	x	
inrichtingselement	naam (Fr)	< Friese naam inrichtingselement >	x	x	
inrichtingselement	nummer	< nummer inrichtingselement >	x	x	
inrichtingselement	status	realisatie: nog niet in uitvoering	x	x	
inrichtingselement	status	realisatie: in uitvoering	x	x	
inrichtingselement	status	in gebruik	x	x	
inrichtingselement	status	buiten gebruik	x	x	
inrichtingselement	status	onbekend	x	x	
inrichtingselement	hoogteniveau	< getal >	x	x	
reliëf	identificatie	< identificerende waarde >			
reliëf	objectBeginTijd	< systeemtijd >			
reliëf	objectEindTijd	< systeemtijd >			
reliëf	versieBeginTijd	< systeemtijd >			
reliëf	versieEindTijd	< systeemtijd >			
reliëf	brontype	luchtfoto			
reliëf	brontype	kaart			
reliëf	brontype	RD			
reliëf	brontype	GBKN			
reliëf	brontype	top10vector			
reliëf	brontype	overig			
reliëf	bronbeschrijving	< tekst >			
reliëf	bronactualiteit	< datum >			
reliëf	bronnauwkeurigheid	< getal >			
reliëf	dimensie	2D			
reliëf	dimensie	3D			
reliëf	type reliëf	dieptelijn		x	
reliëf	type reliëf	dieptepunt	x		
reliëf	type reliëf	hoogtelijn		x	
reliëf	type reliëf	hoogtepunt		x	
reliëf	type reliëf	kade, wal	x		
reliëf	type reliëf	laagwaterlijn	x		
reliëf	type reliëf	peil	x		
reliëf	type reliëf	peil: winterpeil	x		
reliëf	type reliëf	peil: zomerpeil		x	
reliëf	type reliëf	talud, hoogteverschil		x	

Objectklasse	Attribuut	Attribuutwaarde	punt	lijn	vlak
reliëf	type reliëf	steile rand, aardrand		x	
reliëf	type reliëf	onbekend	x	x	
reliëf	hoogte	< hoogte t.o.v. NAP in meters met 1 decimaal >	x	x	
reliëf	hoogteklasse	1 - 2,5 meter		x	
reliëf	hoogteklasse	> 2,5 meter		x	
reliëf	hoogteklasse	> 1 meter		x	
reliëf	functie	geluid weren		x	
reliëf	naam (Nl)	< Nederlandse naam reliëf >	x	x	
reliëf	naam (Fr)	< Friese naam reliëf >	x	x	
reliëf	status	realisatie: nog niet in uitvoering	x	x	
reliëf	status	realisatie: in uitvoering	x	x	
reliëf	status	in gebruik	x	x	
reliëf	status	buiten gebruik	x	x	
reliëf	status	onbekend	x	x	
reliëf	hoogteniveau	< getal >	x	x	
registratief gebied	identificatie	< identificerende waarde >			
registratief gebied	objectBeginTijd	< systeemtijd >			
registratief gebied	objectEindTijd	< systeemtijd >			
registratief gebied	versieBeginTijd	< systeemtijd >			
registratief gebied	versieEindTijd	< systeemtijd >			
registratief gebied	brontype	luchtfoto			
registratief gebied	brontype	kaart			
registratief gebied	brontype	RD			
registratief gebied	brontype	GBKN			
registratief gebied	brontype	top10vector			
registratief gebied	brontype	overig			
registratief gebied	bronbeschrijving	< tekst >			
registratief gebied	bronactualiteit	< datum >			
registratief gebied	bronnauwkeurigheid	< getal >			
registratief gebied	dimensie	2D			
registratief gebied	dimensie	3D			
registratief gebied	type registratief gebied	land	x		x
registratief gebied	type registratief gebied	provincie	x		x
registratief gebied	type registratief gebied	gemeente	x		x
registratief gebied	type registratief gebied	stadsdeel	x		x
registratief gebied	type registratief gebied	wijk	x		x
registratief gebied	type registratief gebied	buurt	x		x
registratief gebied	type registratief gebied	waterschap	x		x
registratief gebied	type registratief gebied	nationaal park	x		x
registratief gebied	type registratief gebied	Bundesland	x		x
registratief gebied	type registratief gebied	Regierungsbezirk	x		x
registratief gebied	type registratief gebied	Kreis	x		x
registratief gebied	naam (Nl)	< Nederlandse naam registratief gebied >	x		x
registratief gebied	naam (Fr)	< Friese naam registratief gebied >	x		x
registratief gebied	nummer	< registratief nummer >	x		x
geografisch gebied	identificatie	< identificerende waarde >			
geografisch gebied	objectBeginTijd	< systeemtijd >			
geografisch gebied	objectEindTijd	< systeemtijd >			
geografisch gebied	versieBeginTijd	< systeemtijd >			
geografisch gebied	versieEindTijd	< systeemtijd >			
geografisch gebied	brontype	luchtfoto			
geografisch gebied	brontype	kaart			
geografisch gebied	brontype	RD			

Objectklasse	**Attribuut**	**Attribuutwaarde**	**punt**	**lijn**	**vlak**
geografisch gebied	brontype	GBKN			
geografisch gebied	brontype	top10vector			
geografisch gebied	brontype	overig			
geografisch gebied	bronbeschrijving	< tekst >			
geografisch gebied	bronactualiteit	< datum >			
geografisch gebied	bronnauwkeurigheid	< getal >			
geografisch gebied	dimensie	2D			
geografisch gebied	dimensie	3D			
geografisch gebied	type geografisch gebied	bank, ondiepte, plaat	x		x
geografisch gebied	type geografisch gebied	bosgebied	x		x
geografisch gebied	type geografisch gebied	buurtschap	x		x
geografisch gebied	type geografisch gebied	duingebied	x		x
geografisch gebied	type geografisch gebied	eiland	x		x
geografisch gebied	type geografisch gebied	geul, vaargeul	x		x
geografisch gebied	type geografisch gebied	heidegebied	x		x
geografisch gebied	type geografisch gebied	heuvel, berg	x		x
geografisch gebied	type geografisch gebied	huizengroep	x		x
geografisch gebied	type geografisch gebied	kaap, hoek	x		x
geografisch gebied	type geografisch gebied	meer, plas, ven, vijver	x		x
geografisch gebied	type geografisch gebied	plaats, bewoond oord	x		x
geografisch gebied	type geografisch gebied	polder	x		x
geografisch gebied	type geografisch gebied	streek, veld	x		x
geografisch gebied	type geografisch gebied	terp	x		x
geografisch gebied	type geografisch gebied	vliedberg	x		x
geografisch gebied	type geografisch gebied	wad	x		x
geografisch gebied	type geografisch gebied	woonwijk	x		x
geografisch gebied	type geografisch gebied	zee	x		x
geografisch gebied	type geografisch gebied	zeegat, zeearm	x		x
geografisch gebied	type geografisch gebied	overig	x		x
geografisch gebied	type geografisch gebied	onbekend	x		x
geografisch gebied	aantal inwoners	< nummer >	x		x
geografisch gebied	naam (Nl)	< Nederlandse naam geografisch gebied >	x		x
geografisch gebied	naam (Fr)	< Friese naam geografisch gebied >	x		x
functioneel gebied	identificatie	< identificerende waarde >			
functioneel gebied	objectBeginTijd	< systeemtijd >			
functioneel gebied	objectEindTijd	< systeemtijd >			
functioneel gebied	versieBeginTijd	< systeemtijd >			
functioneel gebied	versieEindTijd	< systeemtijd >			
functioneel gebied	brontype	luchtfoto			
functioneel gebied	brontype	kaart			
functioneel gebied	brontype	RD			
functioneel gebied	brontype	GBKN			
functioneel gebied	brontype	top10vector			
functioneel gebied	brontype	overig			
functioneel gebied	bronbeschrijving	< tekst >			
functioneel gebied	bronactualiteit	< datum >			
functioneel gebied	bronnauwkeurigheid	< getal >			
functioneel gebied	dimensie	2D			
functioneel gebied	dimensie	3D			
functioneel gebied	type functioneel gebied	arboretum	x		x
functioneel gebied	type functioneel gebied	bedrijventerrein	x		x
functioneel gebied	type functioneel gebied	begraafplaats	x		x
functioneel gebied	type functioneel gebied	boswachterij	x		x

Objectklasse	Attribuut	Attribuutwaarde	punt	lijn	vlak
functioneel gebied	type functioneel gebied	bungalowpark	x		x
functioneel gebied	type functioneel gebied	camping, kampeerterrein	x		x
functioneel gebied	type functioneel gebied	caravanpark	x		x
functioneel gebied	type functioneel gebied	circuit	x		x
functioneel gebied	type functioneel gebied	crossbaan	x		x
functioneel gebied	type functioneel gebied	dierentuin, safaripark	x		x
functioneel gebied	type functioneel gebied	eendenkooi	x		x
functioneel gebied	type functioneel gebied	emplacement	x		x
functioneel gebied	type functioneel gebied	erebegraafplaats	x		x
functioneel gebied	type functioneel gebied	gaswinning	x		x
functioneel gebied	type functioneel gebied	gebied met hoge objecten	x		x
functioneel gebied	type functioneel gebied	gebouwencomplex	x		x
functioneel gebied	type functioneel gebied	golfterrein	x		x
functioneel gebied	type functioneel gebied	grafheuvel	x		x
functioneel gebied	type functioneel gebied	grindwinning	x		x
functioneel gebied	type functioneel gebied	groeve	x		x
functioneel gebied	type functioneel gebied	haven	x		x
functioneel gebied	type functioneel gebied	heemtuin	x		x
functioneel gebied	type functioneel gebied	helikopterlandingsterrein	x		x
functioneel gebied	type functioneel gebied	infiltratiegebied	x		x
functioneel gebied	type functioneel gebied	jachthaven	x		x
functioneel gebied	type functioneel gebied	kartingbaan	x		x
functioneel gebied	type functioneel gebied	kazerne, legerplaats	x		x
functioneel gebied	type functioneel gebied	landgoed	x		x
functioneel gebied	type functioneel gebied	mijn	x		x
functioneel gebied	type functioneel gebied	mijnsteenberg	x		x
functioneel gebied	type functioneel gebied	militair oefengebied, schietterrein	x		x
functioneel gebied	type functioneel gebied	mosselbank	x		x
functioneel gebied	type functioneel gebied	natuurgebied, natuurreservaat	x		x
functioneel gebied	type functioneel gebied	oliewinning	x		x
functioneel gebied	type functioneel gebied	openluchtmuseum	x		x
functioneel gebied	type functioneel gebied	openluchttheater	x		x
functioneel gebied	type functioneel gebied	park	x		x
functioneel gebied	type functioneel gebied	pinetum	x		x
functioneel gebied	type functioneel gebied	plantsoen	x		x
functioneel gebied	type functioneel gebied	productie-installatie	x		x
functioneel gebied	type functioneel gebied	recreatiegebied	x		x
functioneel gebied	type functioneel gebied	renbaan	x		x
functioneel gebied	type functioneel gebied	skibaan	x		x
functioneel gebied	type functioneel gebied	slipschool	x		x
functioneel gebied	type functioneel gebied	sluizencomplex	x		x
functioneel gebied	type functioneel gebied	sportterrein, sportcomplex	x		x
functioneel gebied	type functioneel gebied	stortplaats	x		x
functioneel gebied	type functioneel gebied	tankbaan	x		x
functioneel gebied	type functioneel gebied	tennispark	x		x
functioneel gebied	type functioneel gebied	transformatorstation	x		x
functioneel gebied	type functioneel gebied	tuincentrum	x		x
functioneel gebied	type functioneel gebied	verzorgingsplaats	x		x
functioneel gebied	type functioneel gebied	viskwekerij	x		x
functioneel gebied	type functioneel gebied	vliegveld, luchthaven	x		x
functioneel gebied	type functioneel gebied	volkstuinen	x		x
functioneel gebied	type functioneel gebied	werf	x		x
functioneel gebied	type functioneel gebied	wildwissel	x		x

Objectklasse	Attribuut	Attribuutwaarde	punt	lijn	vlak
functioneel gebied	type functioneel gebied	windturbinepark	x		x
functioneel gebied	type functioneel gebied	woonwagencentrum	x		x
functioneel gebied	type functioneel gebied	ijsbaan	x		x
functioneel gebied	type functioneel gebied	zandwinning	x		x
functioneel gebied	type functioneel gebied	zenderpark	x		x
functioneel gebied	type functioneel gebied	zoutwinning	x		x
functioneel gebied	type functioneel gebied	zuiveringsinstallatie	x		x
functioneel gebied	type functioneel gebied	zweefvliegveldterrein	x		x
functioneel gebied	type functioneel gebied	zwembad complex	x		x
functioneel gebied	type functioneel gebied	onbekend	x		x
functioneel gebied	naam (Nl)	< Nederlandse naam functioneel gebied >	x		x
functioneel gebied	naam (Fr)	< Friese naam functioneel gebied >	x		x

Note: This table has also been used and is made in cooperation with the project group IMTop.

Appendix E: Statistics of the test area

Statistics IMGeo-Top10NL Almere test area

Object class IMGeo	Area IMGeo (in m^2)	Associated object class Top10NL	Area Top10NL (in m^2)		
weg	103230,8233	weg	63053,58583		
water	39460,32347	water	42251,57954		
verblijfsobject	68469,64143	gebouw	65761,17976		
overig bouwwerk	8705,497494	gebouw / terrein			
terrein	193220,3692	terrein	242115,1067		
bak	147,06358	terrein			
				Diff:	
Total	413233,7185		413181,4519		52,266595
Intersected area:	412906				

Differences compared to Top10NL

With an overlay of IMGeo objects it is computed how many percent of the IMGeo objects lies in other Top10NL objects

Top10NL object	IMGeo object	Difference (in m^2)	Origin in Top10NL	Quantity	in %
Weg	weg	51240,02362	Terrein	49777,06287	97%
			Gebouw	1287,082458	3%
			Water	175,878327	0%
				51240,02366	100%
Gebouw	verblijfobject	12180,94307	Terrein	12168,19444	100%
			Weg	12,748632	0%
			Water	0	0%
				12180,94308	100%

Water	Water	976,458563	Terrein	692,695285	71%
			Weg	283,763256	29%
			Gebouw	0	0%
				976,458541	100%
Terrein	terrein	22576,09849	Weg	10799,28792	48%
	bak		Gebouw	8184,184136	36%
	overig bouwwerk		Water	3606,396973	16%
				22589,86903	100%

Percentage of the object class that is classified differently in the other model

Top10NL	**Area in IMGeo**	**Area in Top10NL**	**Overlapping area**	**Percentage w.r.t. IMGeo**	**Top10NL**
weg	103230,8233	63053,58583	52438,23929	51%	83%
water	39460,32347	42251,57954	38483,86494	98%	91%
gebouw	68469,64143	65761,17976	56288,69844	82%	86%
terrein	202072,9303	242115,1067	179496,8319	89%	74%
Total	413233,7185	413181,4519	326707,6345	79%	79%

Appendix F: Python scripts

The next script is used for the 'maximum area method' as described in subsection 5.5.2. The '#' indicates comments. The comments in the code tell what happens at what stage in the code.

```
# ---------------------------------------------------------------------------
# Script name: AssignIMGeotoTop10ID.py
# Description: This script follows the script 'AppendTop10ID'.
# This script resulted in a shapefile with all the IMGeo objects which (partially)
# overlap Top10NL objects. In this script a choice is made to which of these Top10NL
# objects the IMGeo objects are to be assigned. In this first case just the Top10NL
# object which has the largest overlap with the IMGeo object will become the Region Class
# of this IMGeo object.
# Created on: fr dec 6 2007 12:29:05
#   (generated by ArcGIS/ModelBuilder)
# ---------------------------------------------------------------------------

# Import system modules
import sys, string, os, arcgisscripting
import math
import time
from sets import Set

# Create the Geoprocessor object
gp = arcgisscripting.create()

# Set the necessary product code
gp.SetProduct("ArcView")

# Load required toolboxes...
gp.AddToolbox("C:/Program Files/ESR_ArcGisDesktop92EN/ArcGIS/ArcToolbox/Toolboxes/Conversion Tools.tbx")
gp.AddToolbox("C:/Program Files/ESR_ArcGisDesktop92EN/ArcGIS/ArcToolbox/Toolboxes/Data Management
Tools.tbx")
gp.AddToolbox("C:/Program Files/ESR_ArcGisDesktop92EN/ArcGIS/ArcToolbox/Toolboxes/Analysis Tools.tbx")

# Input features:
IMGeo_Merge = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal
Geodatabase.mdb\\Dataset\\IMGeo_Merge_new2"
AppendTop10ID_2 = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal
Geodatabase.mdb\\Dataset\\AppendTop10ID_2"
IMGeo_SpatialJoin = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal
Geodatabase.mdb\\Dataset\\IMGeo_merge_SpatialJoin_new2"
IMGeoTop10NLFinal1b = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal
Geodatabase.mdb\\Dataset\\IMGeoTop10NLFinal1b"


# Search for the next GML_ID in IMGeo_Merge
a=1
for i in range(4159):
# Loop over all the values GML_ID can have. These records are selected from the
# 'AppendTop10ID' shapefile, which contains the areas of the clipped IMGeo objects.
    b=str(a)
    x = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal Geodatabase.mdb\\Troep\\" + b
    gp.Select(AppendTop10ID_2, x, "[GML_ID] = %s*%s" %(a, 1))
# The area of the parts of the GML-ID shapes are compared here. The object with the
# largest area is the object to which the GML-ID finally is assigned to.
    rows = gp.SearchCursor(x)
    row = rows.next()
    values = Set()
    while row:
```

```
        fields = gp.ListFields(x)
        field = fields.next()
        while field:
            if field.name == "SHAPE_Area":
                values.add(row.GetValue(field.name))
                SHAPE_Area = field.name
            field = fields.next()
        row = rows.next()
# Now the maximum of the areas is computed.
    c = max(values)
    y = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal Geodatabase.mdb\\Troep\\" + b
+ "a"
    gp.select(AppendTop10ID_2, y, "[%s] < %s+0.000001 AND [%s] > %s-0.000001 AND [GML_ID] = %s*%s"
%(SHAPE_Area, c, SHAPE_Area, c, a, 1))
# Now the Top10ID of the 'winning' area is selected in order to be able to extract the right value from
# the IMGeo_SpatialJoin shape.
    rows = gp.SearchCursor(y)
    row2 = rows.next()
    d = 0
    while row2:
        fields = gp.ListFields(y)
        field2 = fields.next()
        while field2:
            if field2.name == "Top10ID":
                d = row2.GetValue(field2.name)
            field2 = fields.next()
        row2 = rows.next()
    z = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal Geodatabase.mdb\\Troep\\" + b +
"b"
# The selection that is made has two parameters: b for the right GML-ID and d for the right Top10NL-ID.
    gp.select(IMGeo_SpatialJoin, z, "[GML_ID] = %s*%s AND [Top10ID] = %s*%s" %(b, 1, d, 1))
# This selection is appended to the final result.
    gp.Append_management(z, IMGeoTop10NLFinal1b, "TEST", IMGeoTop10NLFinal1b)
    time.sleep(0.2)
# The database lock is removed with the statement 'Refresh Catalog'. This enables us to delete the files we don't
# need anymore after having processed this one GML-ID.
    gp.RefreshCatalog("\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal Geodatabase.mdb")
    try:
        gp.delete(x)
        gp.delete(y)
        gp.delete(z)
    except:
        print 'Jammer dan'
    a=a+1
```

The next script is used to select all objects that are to be split according to the 35%-split method.

```
# Import system modules
import sys, string, os, arcgisscripting
import math
import time
from sets import Set

# Create the Geoprocessor object
gp = arcgisscripting.create()

# Set the necessary product code
gp.SetProduct("ArcView")

# Load required toolboxes...
gp.AddToolbox("C:/Program Files/ESR_ArcGisDesktop92EN/ArcGIS/ArcToolbox/Toolboxes/Conversion Tools.tbx")
gp.AddToolbox("C:/Program Files/ESR_ArcGisDesktop92EN/ArcGIS/ArcToolbox/Toolboxes/Data Management
Tools.tbx")
gp.AddToolbox("C:/Program Files/ESR_ArcGisDesktop92EN/ArcGIS/ArcToolbox/Toolboxes/Analysis Tools.tbx")

# Input features:
IMGeo_Merge = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal
Geodatabase.mdb\\Dataset\\IMGeo_Merge_new3"
AppendTop10ID_2 = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal
Geodatabase.mdb\\Dataset\\AppendTop10ID_3_Erase2"
IMGeo_SpatialJoin = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal
Geodatabase.mdb\\Dataset\\IMGeo_merge_SpatialJoin_new3"
IMGeoTop10NLFinal1b = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal
Geodatabase.mdb\\Dataset\\IMGeoTop10NLFinal2b"
split35 = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal
Geodatabase.mdb\\Dataset\\splitfeatures35"

maxGML = 4159
a=1
for i in range(4159):
    b=str(a)
    try:
      x = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal Geodatabase.mdb\\Troep\\" + b
        gp.Select(AppendTop10ID_2, x, "[GML_ID] = %s*%s" %(a, 1))
        rows = gp.SearchCursor(x)
        row = rows.next()
        values = []
        e = 0
        f = 0
        h = 0
        m = 0
        while row:
            fields = gp.ListFields(x)
            field = fields.next()
            while field:
                if field.name == "Top10ID":
                    h = row.GetValue(field.name)
                if field.name == "SHAPE_Area":
                    e = row.GetValue(field.name)
                    SHAPE_Area = field.name
                if field.name == "Area":
                    f = row.GetValue(field.name)
                    Area = field.name
                field = fields.next()
            g = e/f
            if g > 0.35:
                values.append(h)
```

```
            row = rows.next()
        if len(values) == 2:
            maxGML = maxGML + 1
            k = values[0]
            l = values[1]
            w = "\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal Geodatabase.mdb\\Troep\\"
+ b + "c"
            gp.select(AppendTop10ID_2, w, "[GML_ID] = %s*%s AND [Top10ID] = %s*%s" %(b, 1, k, 1))
            gp.calculatefield(w, "GML_ID", maxGML)
            gp.append(w, split35, "TEST", split35)
            gp.RefreshCatalog("\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal
Geodatabase.mdb")
            try:
                gp.delete(w)
            except:
                print 'Jammer dan'
        gp.RefreshCatalog("\\\\GWfiler02\\13712data$\\Afstuderen\\Data\\Geodatabase\\Personal
Geodatabase.mdb")
        try:
            gp.delete(x)
        except:
            print 'Jammer dan'
    except:
        print 'Deze heb ik eruit gehaald'
    a=a+1
```

Appendix G: tGAP code in PL/SQL

The code of the constrained tGAP has been produced by Jan Haunert and Arta Dilo, only small modifications are made to this by the author.

The procedure build calls all other procedures; first the getNextImportantFace procedure to determine the least important face, after that the most compatible neighhour is determined in the function getMost CompatibleNeighbour, after that the faces are merged in the procedure mergeFace. In this procedure mergeFace the modifications are saved to the table. At the end of the iteration step the edges and faces that were removed receive the value of the iteration step as their imp_high value. Because of this the output can be generated at all possible importance steps, i.e. at all different scales.

The complete constrained tGAP code used in this research without weights is:

```
create or replace procedure build is
        newFaceID number;
        minImpFaceID number := 0;
        minImpFaceArea number := 0;
        mostCompNeighbourID number;
        importance pls_integer := 1;

    begin
        select max(face_id) into newFaceID from ARJEN_FACES;

        getNextImportantFace(minImpFaceArea, minImpFaceID);

        while minImpFaceID > 0

            loop
           mostCompNeighbourID := getMostCompatibleNeighbour( minImpFaceID );
               if mostCompNeighbourID  <> 0 then
            newFaceID := newFaceID + 1;
                 mergeFace(mostCompNeighbourID, minImpFaceID, newFaceID, importance);
            importance := importance + 1;
           end if;
               getNextImportantFace(minImpFaceArea, minImpFaceID);
        end loop;

        endRemainingEdges(importance);
        endRemainingFaces(importance);
    end;

=========================================================

create or replace procedure endEdges(faceID in number, importance in number) is
    begin
        execute immediate
          'update ARJEN_EDGES set IMP_HIGH = :1 where
           IMP_HIGH = 0 and (LEFT_FACE_ID = :2 or RIGHT_FACE_ID = :3)'
        using importance, faceID, faceID;
        commit;
    end;

=========================================================

create or replace procedure endFace(faceID in number, importance in number) is
    begin
        execute immediate
```

```
        'update ARJEN_FACES set IMP_HIGH = :1 where
         IMP_HIGH = 0 and FACE_ID = :2'
      using importance, faceID;
      commit;
   end;

==========================================================

create or replace procedure endRemainingEdges(importance in number) is
   begin
      execute immediate 'update ARJEN_EDGES set IMP_HIGH = :1 where IMP_HIGH = 0' using importance;
      commit;
   end;

create or replace procedure endRemainingFaces(importance in number) is
   begin
      execute immediate 'update ARJEN_FACES set IMP_HIGH = :1 where IMP_HIGH = 0' using importance;
      commit;
   end;

==========================================================

create or replace procedure getNextImportantFace(faceArea in out number, faceID in out number) is
   nextFaceArea number :=0;
   nextFaceID number :=0;
   begin
      select min(area) into nextFaceArea
       from ARJEN_FACES
      where
       imp_high = 0
       and (area > faceArea or (area = faceArea and face_id > faceID));

      select min(face_id) into nextFaceID
       from ARJEN_FACES
      where
       imp_high = 0
       and area = nextFaceArea
       and (area > faceArea or (area = faceArea and face_id > faceID));

       faceArea := nextFaceArea;
       faceID  := nextFaceID;
   end;

==========================================================
```

```
create or replace procedure mergeFace(faceA in number, faceB in number, newFaceID in number, importance in
number) is

        newArea number;
            newClass char;
            newRegion number;
        newRegionClass char;

        oldEdges id_list;

        oldEdgeID number;
        oldEdgeGeoID number;
        oldLeftFaceID number;
        oldRightFaceID number;

        newEdgeID number;

   begin
        select sum(AREA) into newArea from ARJEN_FACES where FACE_ID = faceA or FACE_ID = faceB;
        select CLASS, IDENT, REGION_CLA into newClass, newRegion, newRegionClass from ARJEN_FACES where
FACE_ID = faceA;

        storeNewFace(newFaceID, importance, newArea, newClass, newRegion, newRegionClass);

        select max(EDGE_ID) into newEdgeID from ARJEN_EDGES;
        select EDGE_ID bulk collect into oldEdges
        from ARJEN_EDGES where
         imp_high = 0 and
         ((LEFT_FACE_ID  = faceA and RIGHT_FACE_ID <> faceB) or
          (RIGHT_FACE_ID = faceA and LEFT_FACE_ID  <> faceB) or
          (LEFT_FACE_ID  = faceB and RIGHT_FACE_ID <> faceA) or
          (RIGHT_FACE_ID = faceB and LEFT_FACE_ID  <> faceA));

        for i in 1 .. oldEdges.count
        loop
         oldEdgeID := oldEdges(i);
         newEdgeID := newEdgeID + 1;

         select EDGE_GEO_ID, LEFT_FACE_ID, RIGHT_FACE_ID into oldEdgeGeoID, oldLeftFaceID, oldRightFaceID
from ARJENEDGE
         where EDGE_ID = oldEdgeID;

         if oldRightFaceID = faceA or oldRightFaceID = faceB then
          storeNewEdge(newEdgeID, importance, oldEdgeGeoID, oldLeftFaceID, newFaceID);
         else
          storeNewEdge(newEdgeID, importance, oldEdgeGeoID, newFaceID, oldRightFaceID);
         end if;

        end loop;

        endEdges(faceA, importance);
        endEdges(faceB, importance);
        endFace(faceA, importance);
        endFace(faceB, importance);

    end;

===========================================================
```

```
create or replace procedure storeNewEdge(edgeID in number, importance in number, edgeGeoID in number,
leftFaceID in number, rightFaceID in number) is
    begin
        execute immediate 'insert into ARJEN_EDGES (EDGE_ID, IMP_LOW, IMP_HIGH, EDGE_GEO_ID,
LEFT_FACE_ID, RIGHT_FACE_ID) values (:1, :2, 0, :3, :4, :5)' using edgeID, importance, edgeGeoID, leftFaceID,
rightFaceID;
        commit;
    end;

=================================================================

create or replace procedure storeNewFace(faceID in number, importance in number, area in number, class in
number, region in number, regionclass in number) is
    begin
        execute immediate 'insert into ARJEN_FACES (FACE_ID, IMP_LOW, IMP_HIGH, AREA, CLASS, REGION,
REGION_CLA) values (:1, :2, 0, :4, :5, :6, :7)' using faceID, importance, area, class, region, regionclass;
        commit;
    end;

=================================================================

create or replace function getMostCompatibleNeighbour (faceA in number) return number is

classA number;
regionA number;
areaA number;
regionclassA number;

faces id_list;

classTemp number;
regionTemp number;
costTemp number;

costBest number := -1;
faceBest number := 0;
n number;

begin

--query attributes of faceA

 select
  class, TOP10ID, area, REGION_CLASS
 into
  classA, regionA, areaA, regionclassA
 from
  ARJEN_FACES
 where
  imp_high = 0 and face_id = faceA;

--select all neighbours of faceA

 select distinct
  (left_face_id + right_face_id - faceA)
 bulk collect into
  faces
 from
  ARJEN_EDGES where
  imp_high = 0 and (right_face_id = faceA or left_face_id = faceA);

--iterate through neighbours

 for i in 1 .. faces.count
```

```
 loop

 --query attributes of neighbour
  select count(*) into n from ARJENFACE where imp_high = 0 and face_id = faces(i);
  if n >= 1 then
  select
   class, TOP10ID,
  into
   classTemp, regionTemp,
  from
   ARJEN_FACES
  where
   imp_high = 0 and face_id = faces(i);

 --if neighbour is in same region, then calculate cost

  if regionTemp = regionA then

   select cost into costTemp from SIMILARITIES
   where class1 = classA and class2 = classTemp;

   costTemp := costTemp * areaA;

  --keep the neighbour with lowest cost

   if costTemp < costBest or costBest = -1 then
        if n = 1 then
        select class, REGION_CLA into classtest, Regionclasstest from faces;
            if classtest <> Regionclasstest then
            Regionclasstest := classtest;
            end if;
        end if;
      faceBest := faces(i);
      costBest := costTemp;
   end if;

  end if;
  end if;
 end loop;

 return faceBest;
end;

===========================================================
```

For the implementation of the weights the procedure getNextImportantFace needed to be changed:

```
 create or replace procedure getNextImportantFace(faceArea in out number, faceID in out number) is
    nextFaceArea number :=0;
    nextFaceID number :=0;
    begin
        select min(f.area * w.weights) into nextFaceArea
         from ARJEN_FACES f, WEIGHTS w
        where
            f.class = w.class
             and imp_high = 0
         and (f.area * w.weights > faceArea or (f.area * w.weights = faceArea and face_id > faceID));

        select min(face_id) into nextFaceID
         from ARJEN_FACES f, WEIGHTS w
        where
            f.class = w.class and
```

```
          f.area * w.weights = nextFaceArea and
          imp_high = 0
       and (f.area * w.weights > faceArea or (f.area * w.weights = faceArea and face_id > faceID));

       faceArea := nextFaceArea;
       faceID  := nextFaceID;
   end;
```

==

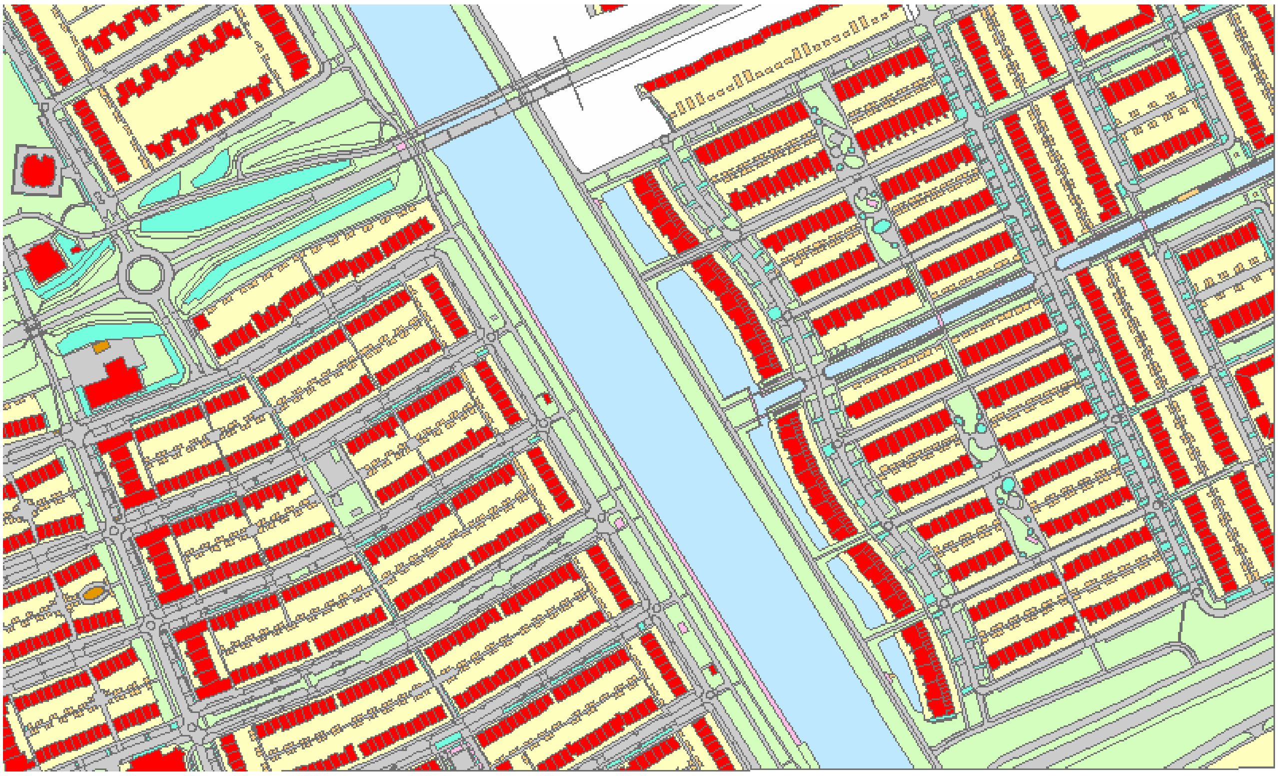
To run the tGAP without constraint the procedure getMostCompatibleNeighbour needed to be changed:

```
create or replace function getMostCompatibleNeighbour (faceA in number) return number is

classA number;
regionA number;
areaA number;

faces id_list;

classTemp number;
regionTemp number;
costTemp number;

costBest number := -1;
faceBest number := 0;
n number;

begin

--query attributes of faceA

 select
  class, TOP10ID, area
 into
  classA, regionA, areaA
 from
  ARJEN_FACES
 where
  imp_high = 0 and face_id = faceA;

--select all neighbours of faceA

 select distinct
  (left_face_id + right_face_id - faceA)
 bulk collect into
  faces
 from
  ARJEN_EDGES
 where
  imp_high = 0 and (right_face_id = faceA or left_face_id = faceA);

--iterate through neighbours

 for i in 1 .. faces.count
 loop

 --query attributes of neighbour
  select count(*) into n from ARJEN_FACES where imp_high = 0 and face_id = faces(i);
  if n > 0 then
  select
   class
```

```
  into
   classTemp
  from
   ARJEN_FACES
  where
   imp_high = 0 and face_id = faces(i);

   select cost into costTemp from SIMILARITIES
   where class1 = classA and class2 = classTemp;

   costTemp := costTemp * areaA;

  --keep the neighbour with lowest cost

   if costTemp < costBest or costBest = -1 then
    faceBest := faces(i);
    costBest := costTemp;
   end if;

  end if;
  end if;
 end loop;

 return faceBest;
end;

==========================================================
```

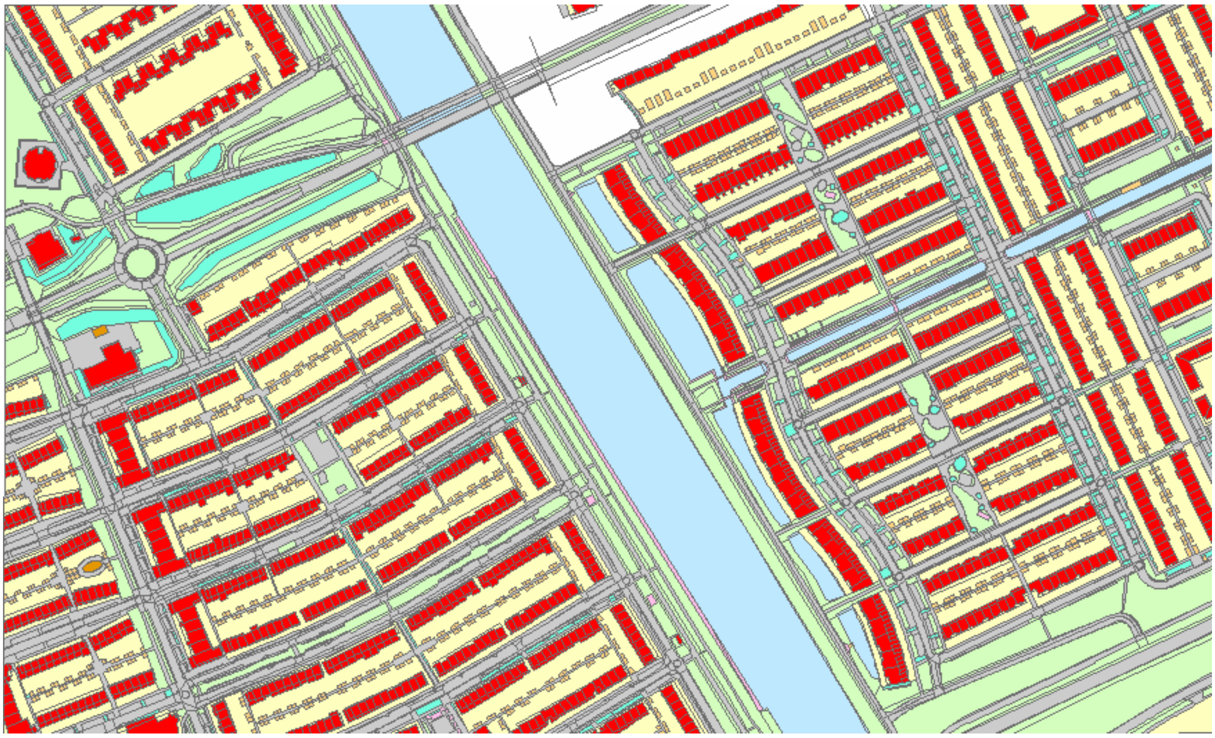
Appendix H: Visualised results of the constrained tGAP

This appendix shows all intermediate results of the generalisation of the test data used in chapter 5. In all cases 8 intermediate steps were recorded to give a good view of the progress in the generalisation. This appendix starts with the simple overlay method, both with and without weights, after that the maximum area method, the 35%-split method and the building first method. The starting point of all cases, for reference, is the same: the IMGeo as visualised beneath.



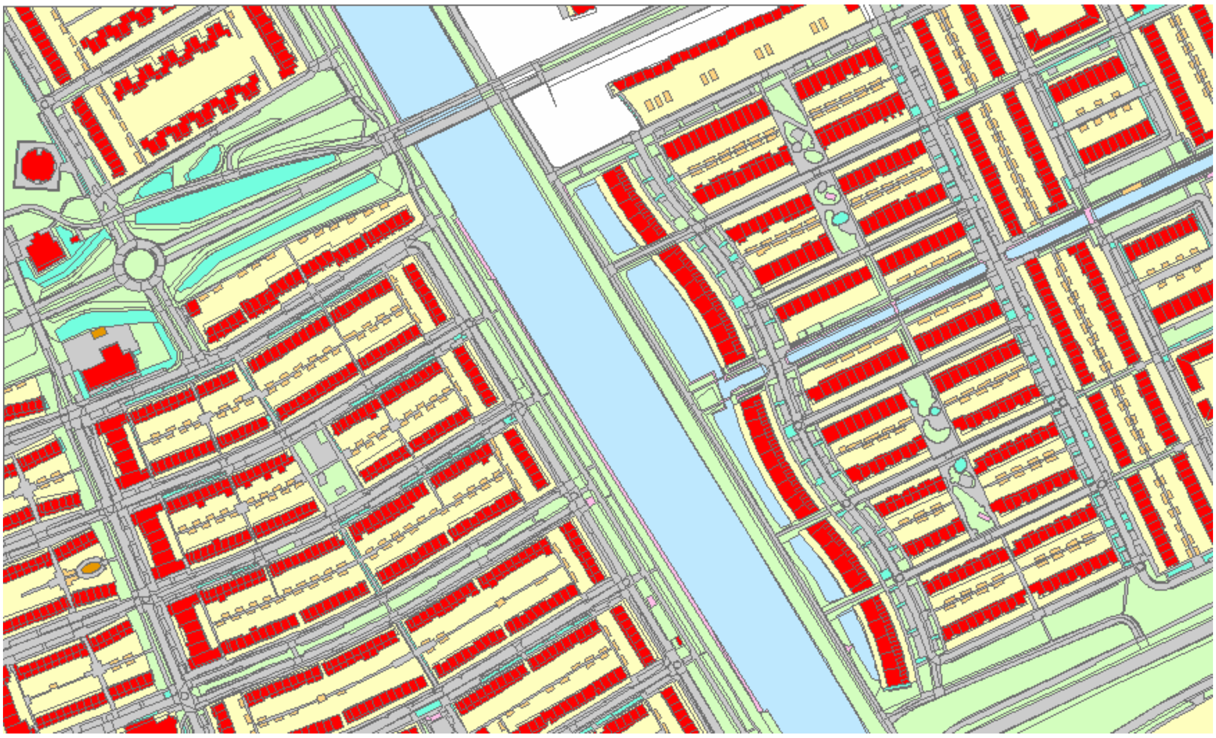
The original IMGeo test dataset

Simple overlay without weights; importance =1000



Simple overlay without weights; importance =2000

Simple overlay without weights; importance =3000



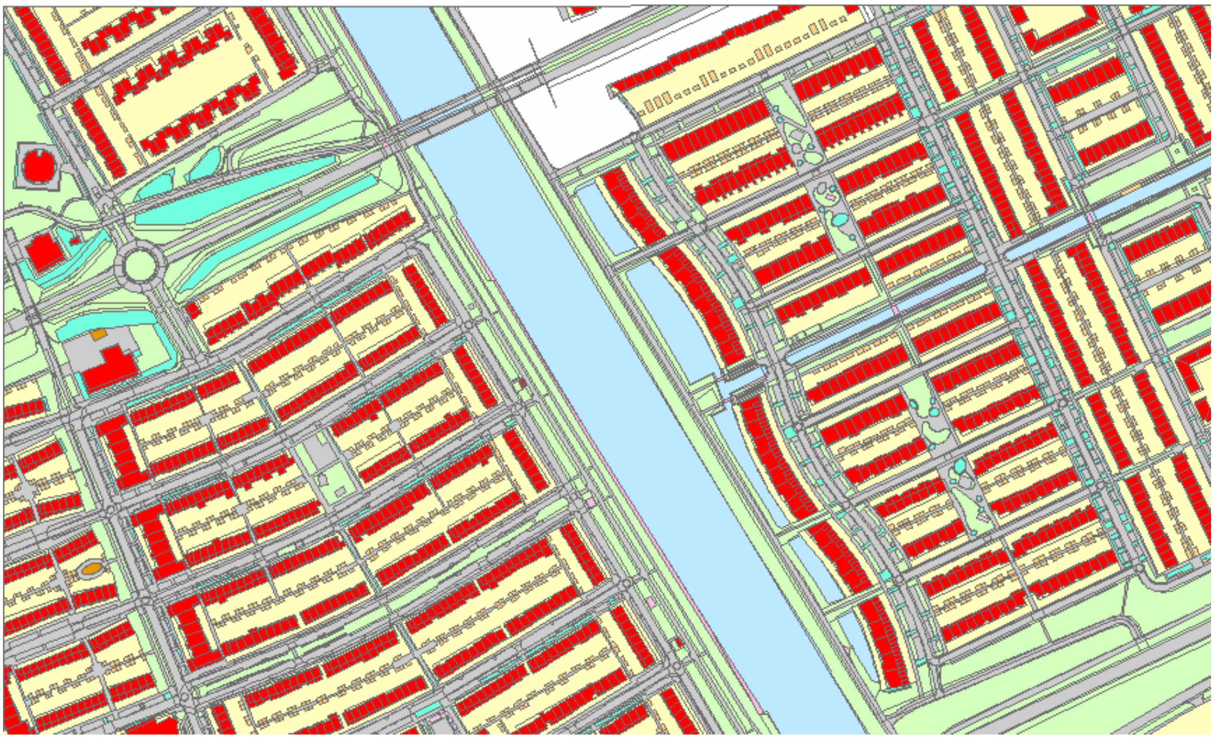
Simple overlay without weights; importance =4000

Simple overlay without weights; importance =5000

Simple overlay without weights; importance =6000

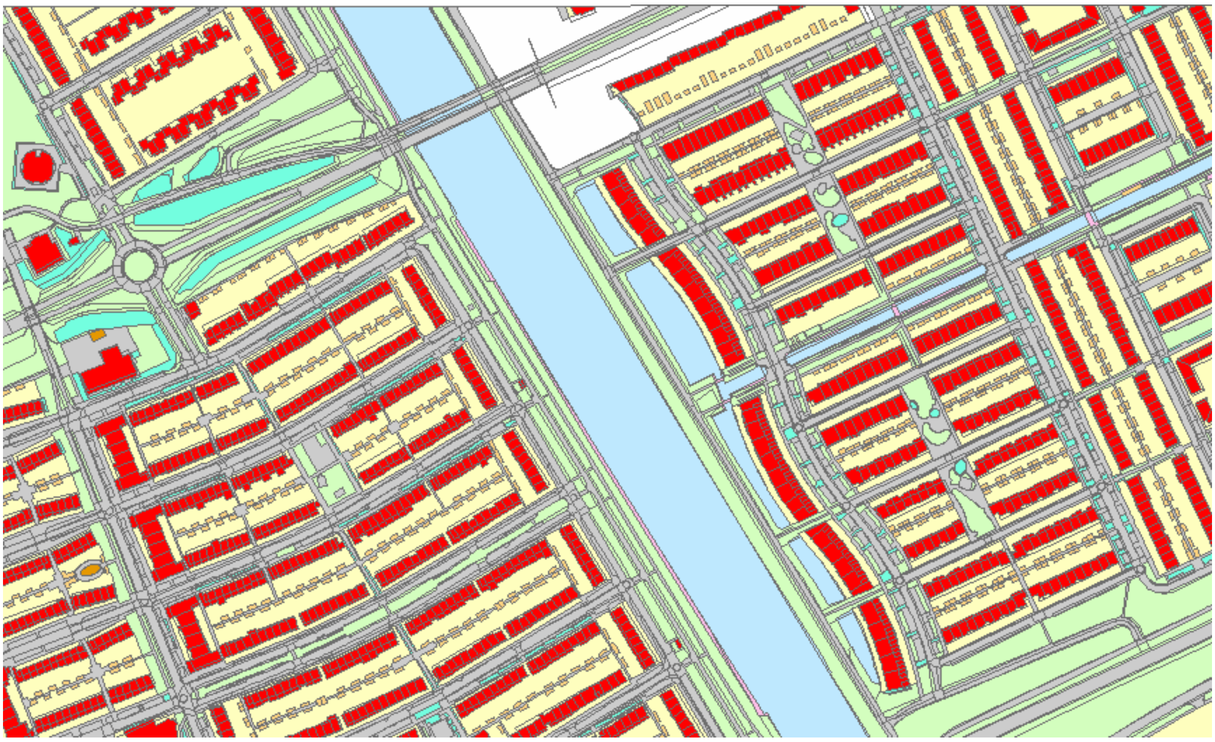
Simple overlay without weights; importance =7000

Simple overlay without weights; importance =7702

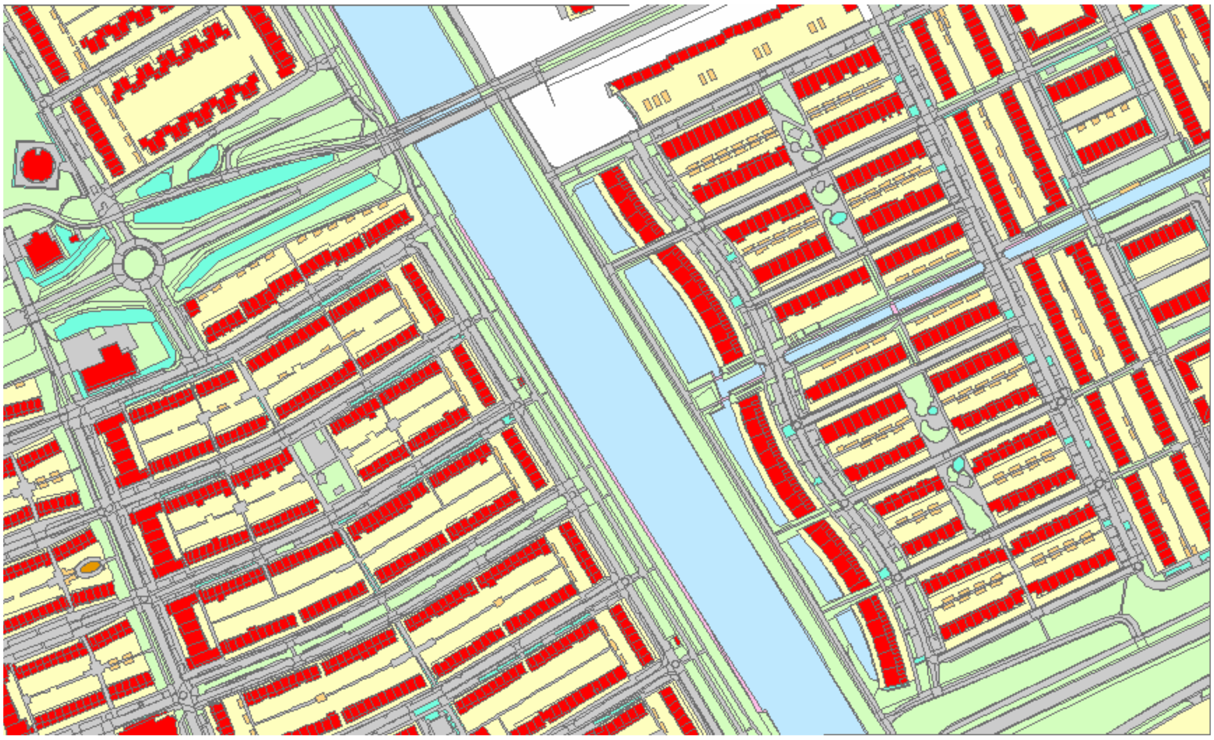


Simple overlay with weights; importance =1000

Simple overlay with weights; importance =2000



Simple overlay with weights; importance =3000



Simple overlay with weights; importance =4000

Simple overlay with weights; importance =5000

Simple overlay with weights; importance =6000

Simple overlay with weights; importance =7702

Simple overlay with weights; importance =7702

Maximum area method without weights; importance=500

Maximum area method without weights; importance=1000


Maximum area method without weights; importance=1500

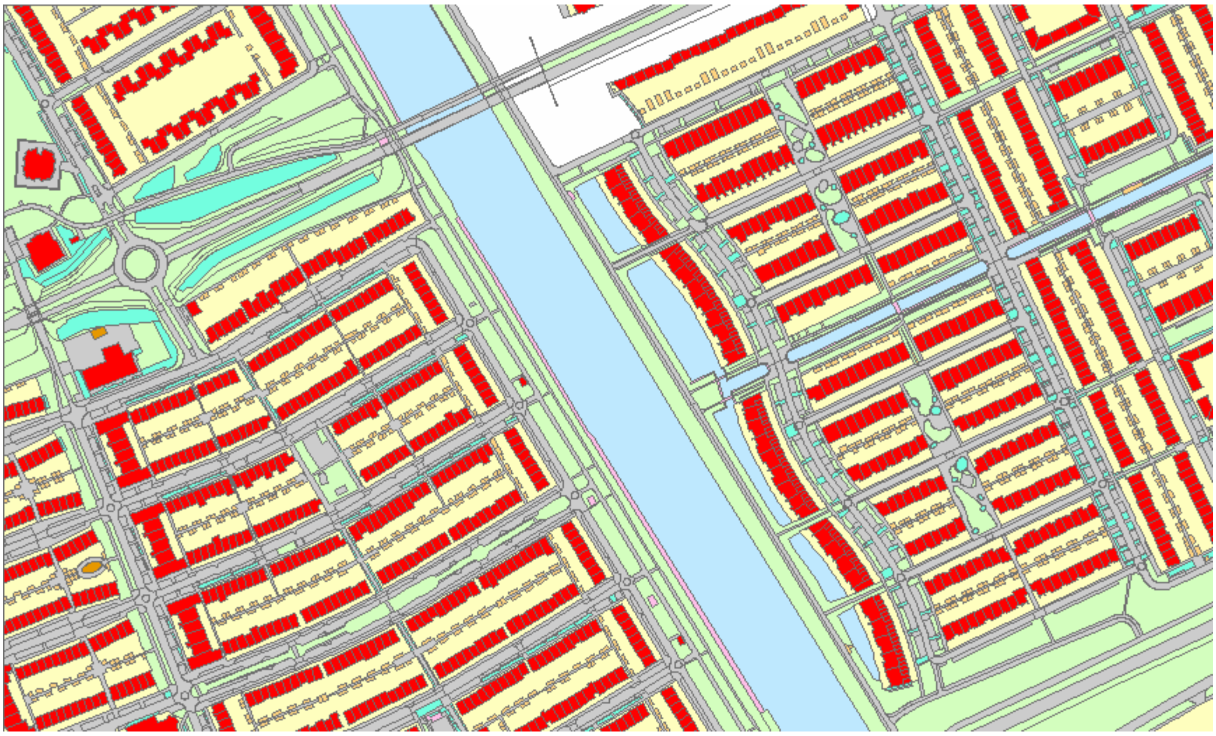

Maximum area method without weights; importance=2000

Maximum area method without weights; importance=2500

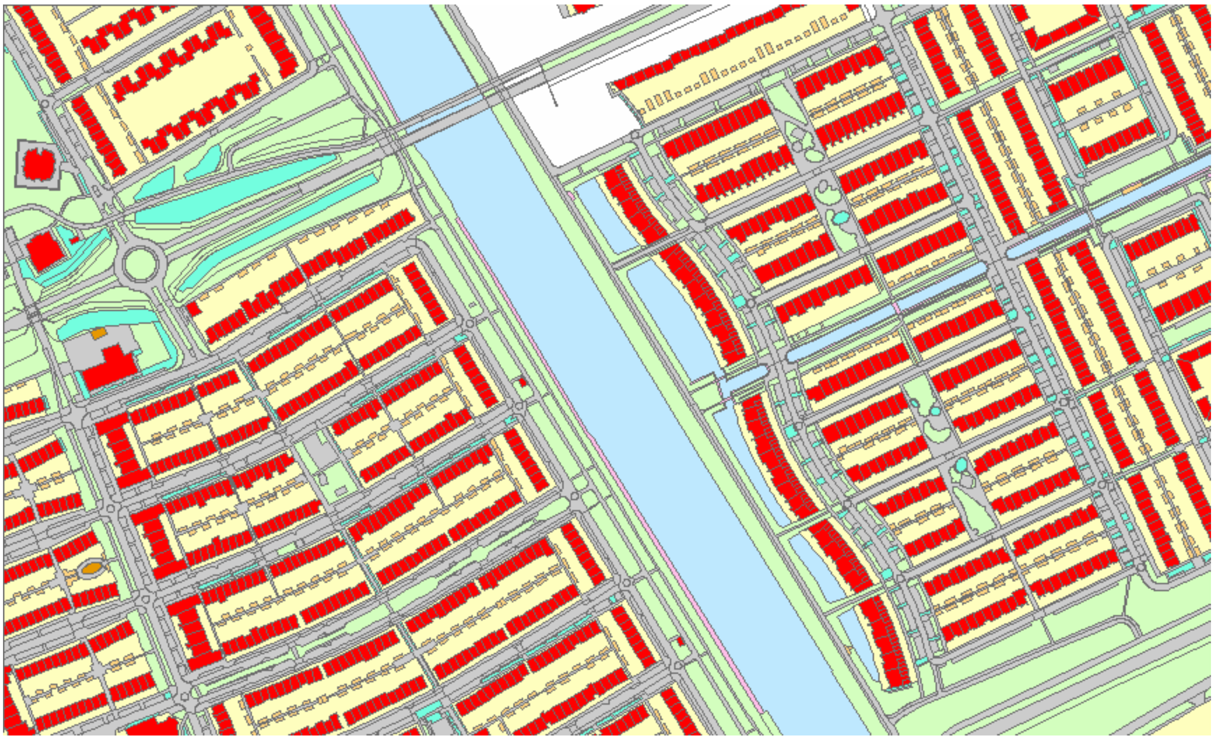
Maximum area method without weights; importance=3000

Maximum area method without weights; importance=3500

Maximum area method without weights; importance=3805



Maximum area method with weights; importance=500



Maximum area method with weights; importance=1000

Maximum area method with weights; importance=1500

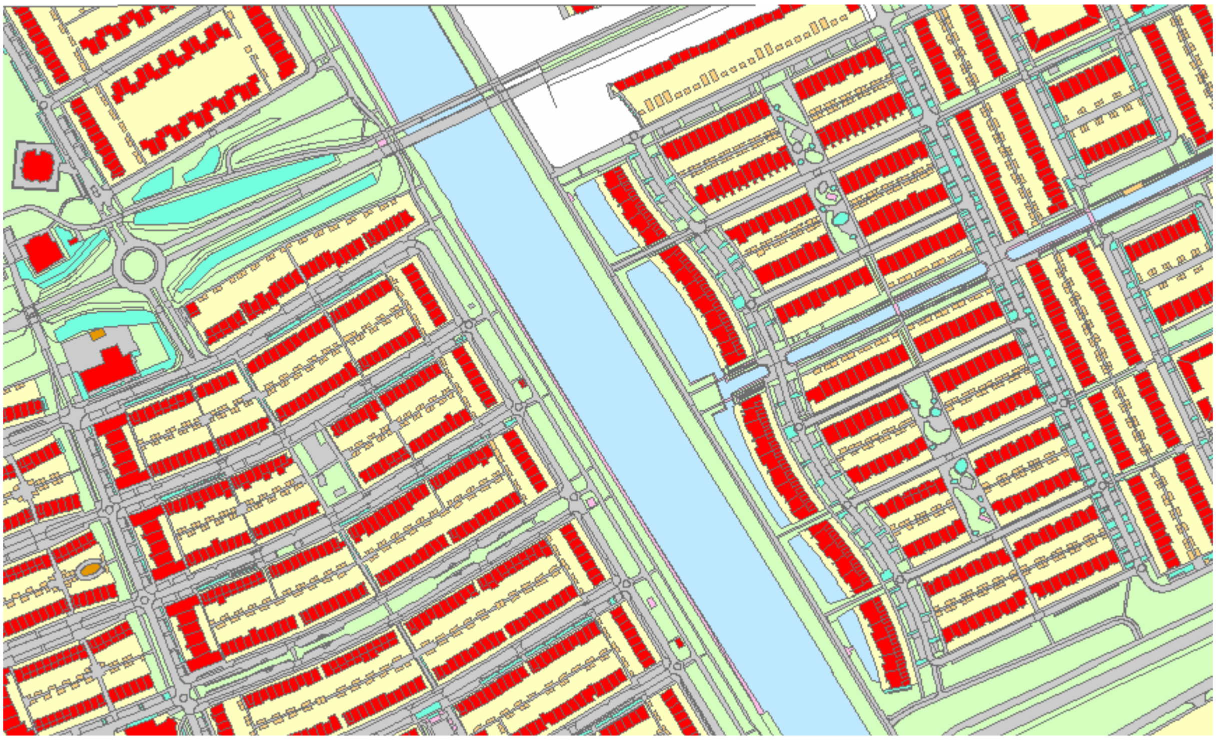
Maximum area method with weights; importance=2000

Maximum area method with weights; importance=2500

Maximum area method with weights; importance=3000

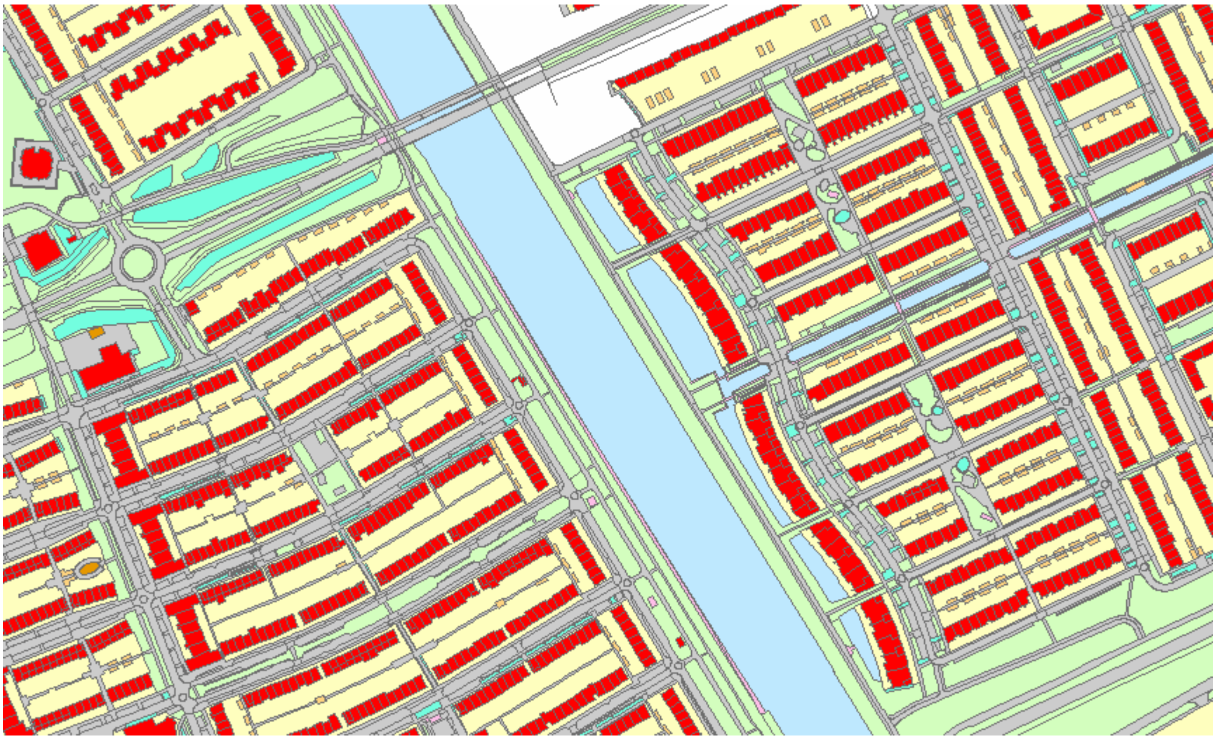

Maximum area method with weights; importance=3500


Maximum area method with weights; importance=3805



35%-split method without weights; importance=500

35%-split method without weights; importance=1000



35%-split method without weights; importance=1500

35%-split method without weights; importance=2000

35%-split method without weights; importance=2500

35%-split method without weights; importance=3000

35%-split method without weights; importance=3500

35%-split method without weights; importance=4041


35%-split method with weights; importance=500


35%-split method with weights; importance=1000

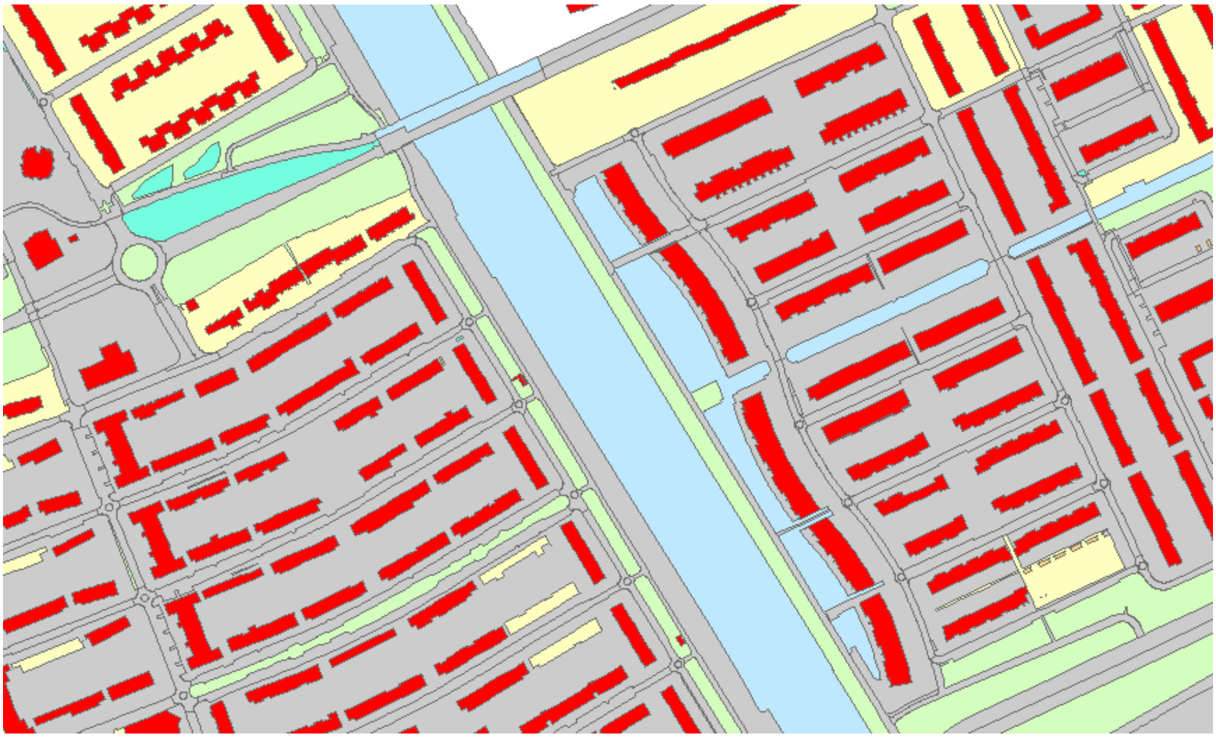
35%-split method with weights; importance=1500

35%-split method with weights; importance=2000

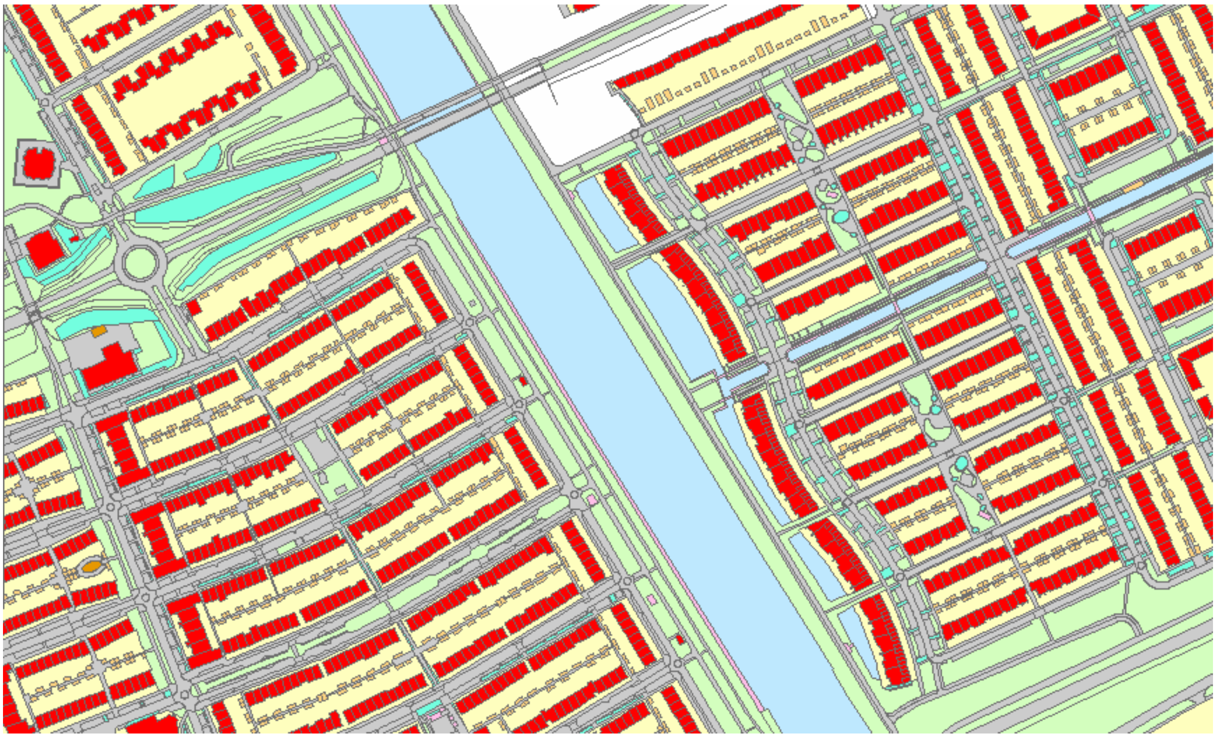
35%-split method with weights; importance=2500

35%-split method with weights; importance=3000

35%-split method with weights; importance=3500

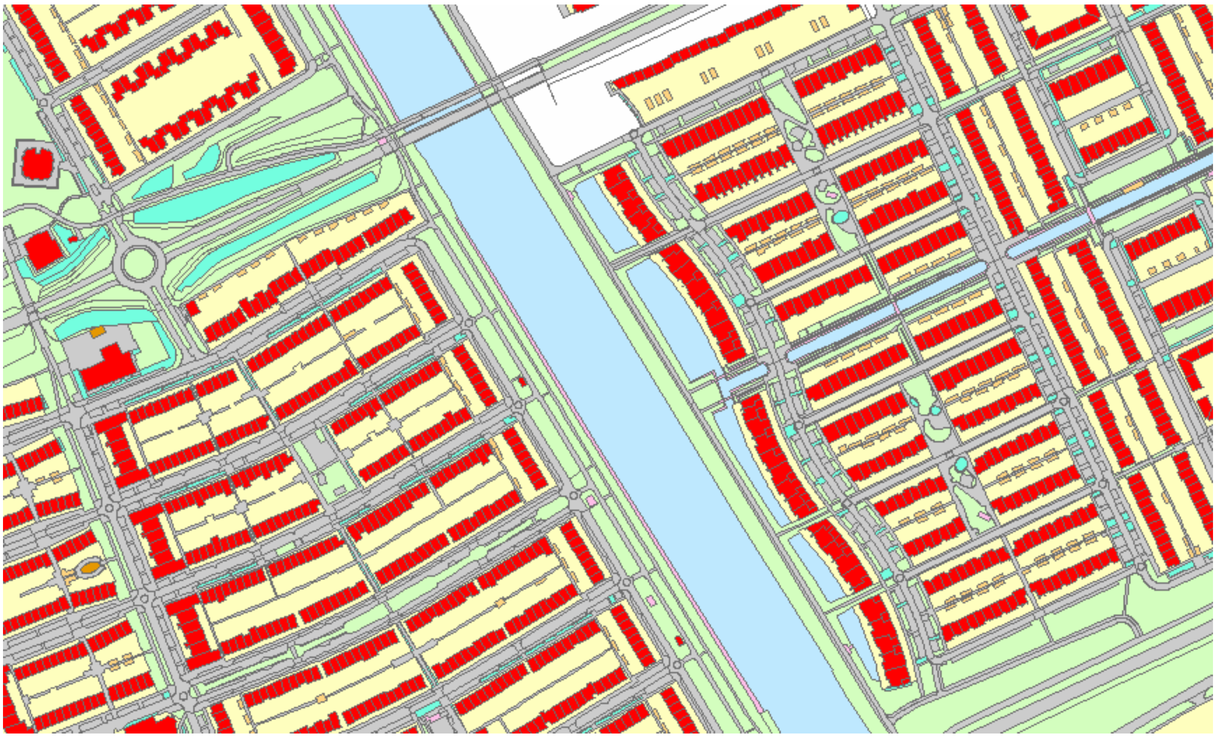


35%-split method with weights; importance=4041



Building first method without weights; importance=500

Building first method without weights; importance=1000



Building first method without weights; importance=1500

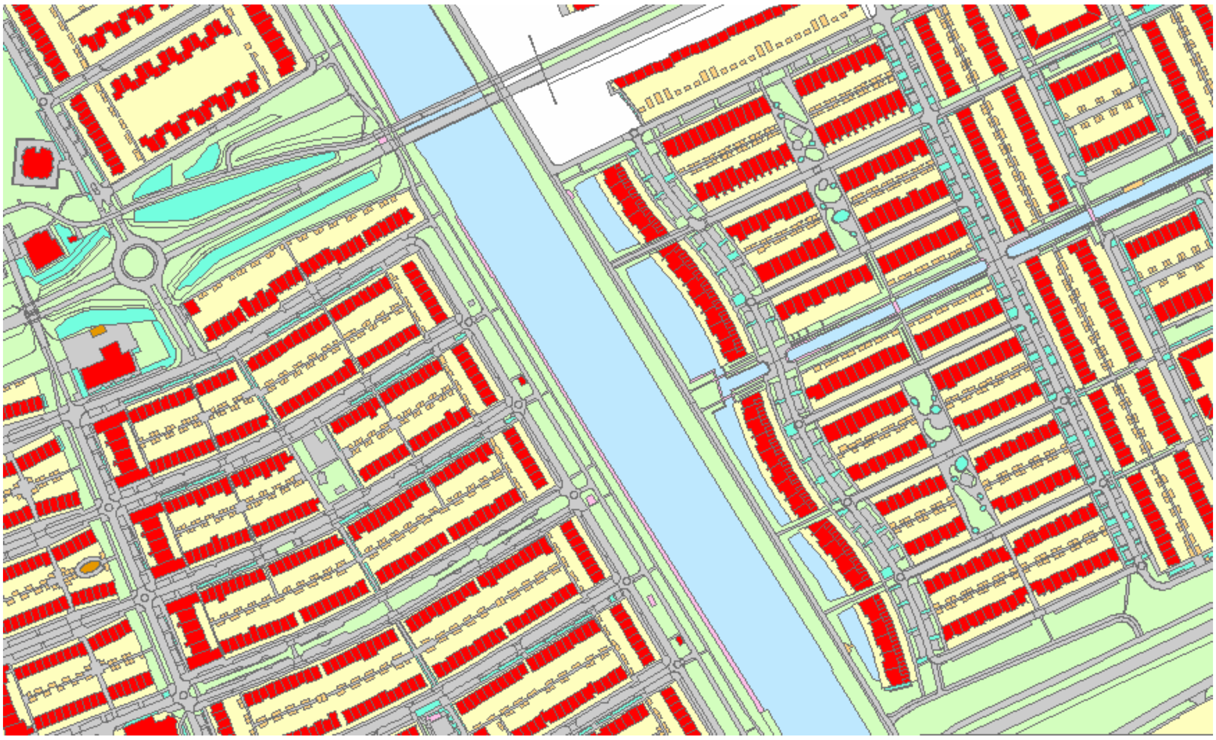
Building first method without weights; importance=2000

Building first method without weights; importance=2500

Building first method without weights; importance=3000

Building first method without weights; importance=3500

Building first method without weights; importance=3805



Building first method with weights; importance=500

Building first method with weights; importance=1000

Building first method with weights; importance=1500

Building first method with weights; importance=2000

Building first method with weights; importance=2500

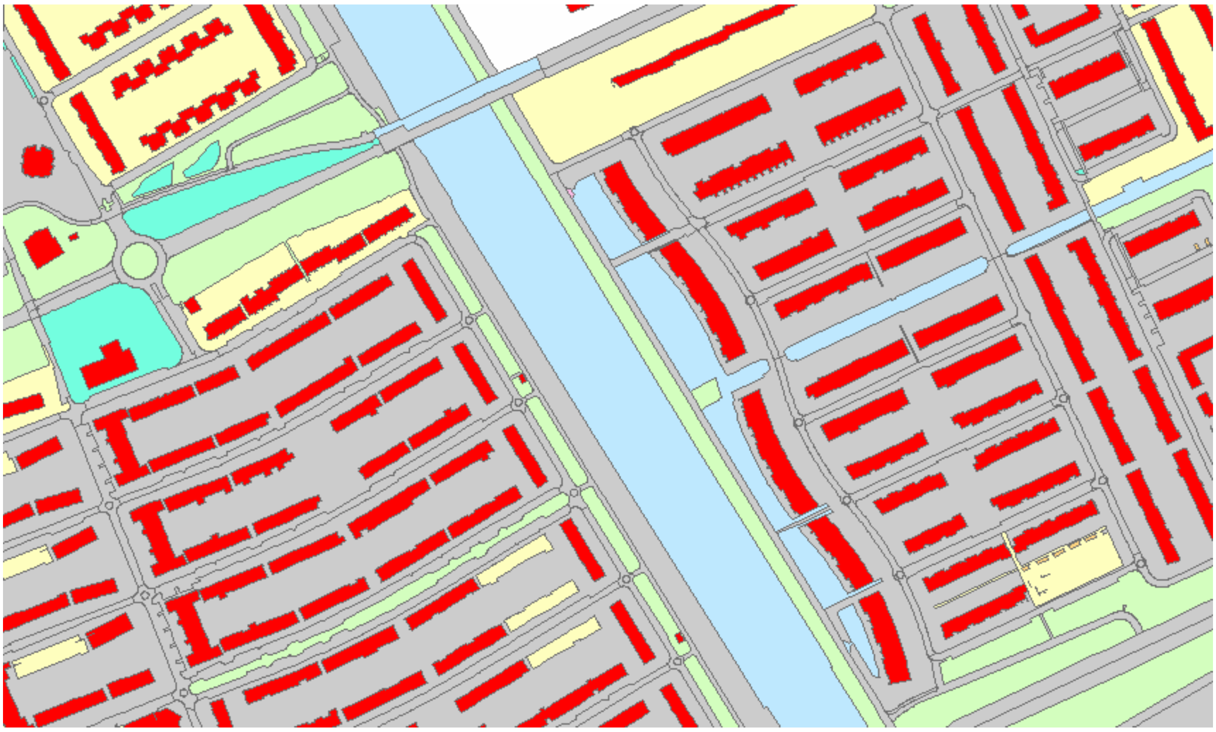
Building first method with weights; importance=3000

Building first method with weights; importance=3500

Building first method with weights; importance=3805

Appendix I: Visualisation of the trials to improve the values for weights and compatibilities

This appendix shows the iteration steps that were done to come to good values for the weights and comaptibilties for the test dataset as mentioned in section 6.1. Starting point of this iteration was the end result of chapter 5.



Visualisation of the end result using the building first method with weights.

Result after trial 1

Result after trial 2

Result after trial 3

Result after trial 4

Result after trial 5


Result after trial 6


Result after trial 7


Result after trial 8


Result after trial 9

Result after trial 10

Result after trial 11

Result after trial 12

Result after trial 13

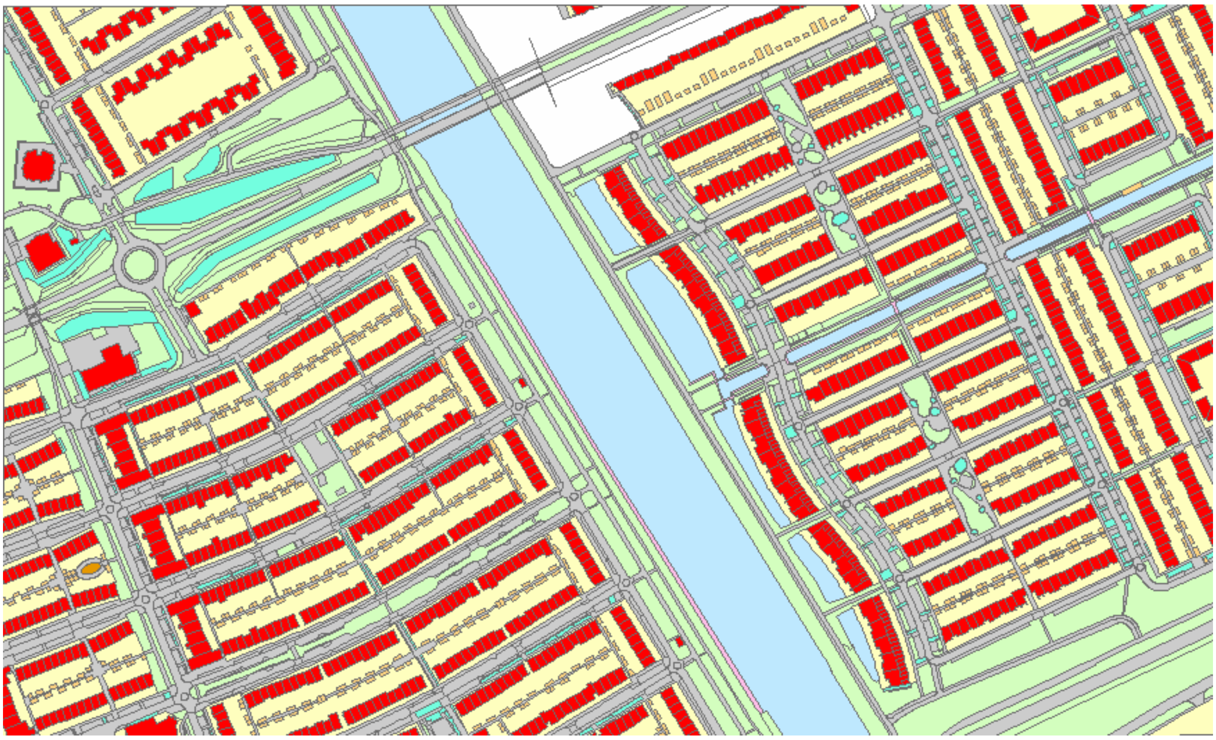
Result after trial 14

Result after trial 15

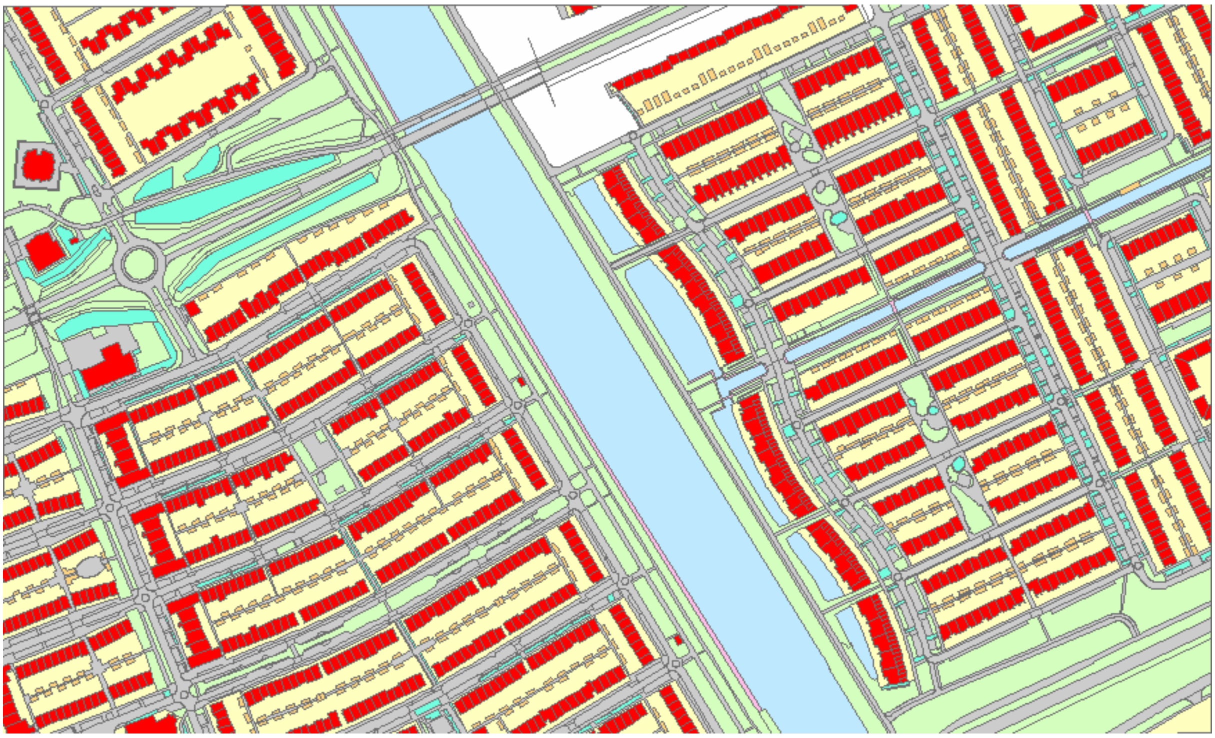
Final result after trial 16

Appendix J: Visualisation of the tGAP without constraint

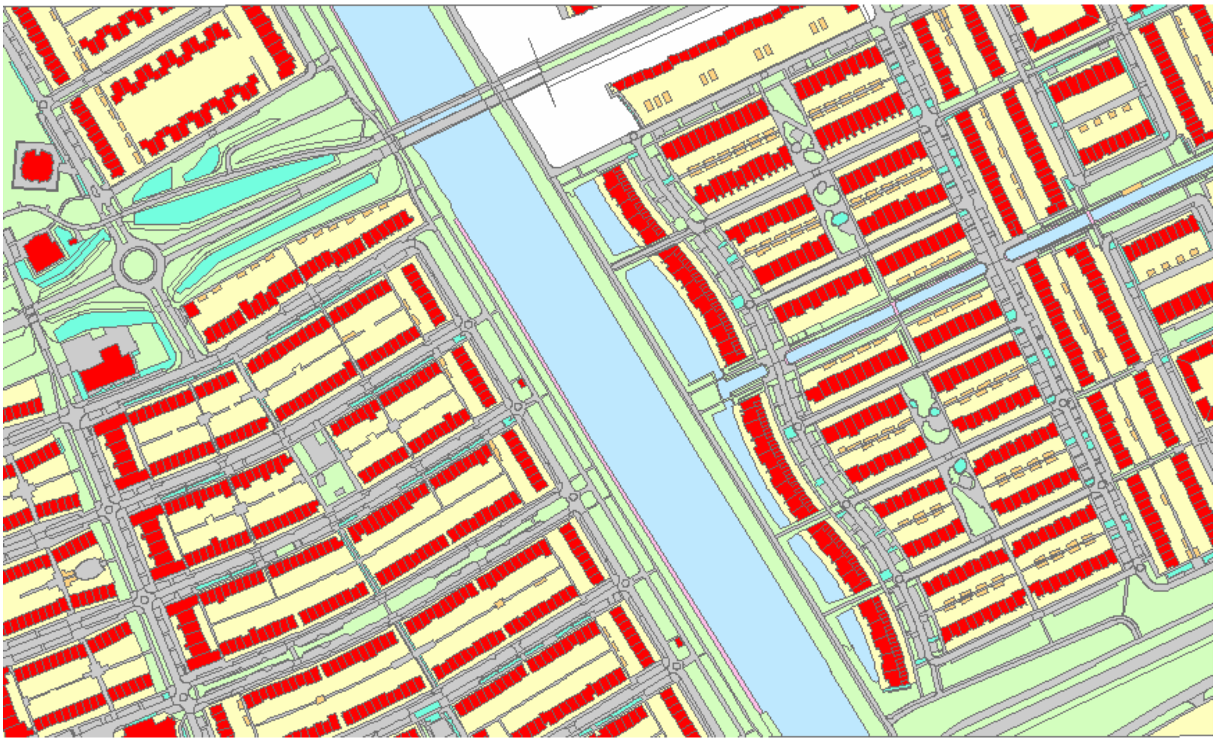
This appendix shows the visualisation of the steps of the test data in the tGAP structure without constraint. This means that IMGeo objects might even be merged with objects outside their preciously defined region. This is done to see whether a constraint is really necessary. The tGAP is visualised until there is only one object left, here we can see that the 'edge-remover' in the tGAP-code didn't work properly all the time.



tGAP without constraint; importance=500



tGAP without constraint; importance=1000



tGAP without constraint; importance=1500

tGAP without constraint; importance=2000

tGAP without constraint; importance=2500


tGAP without constraint; importance=3000

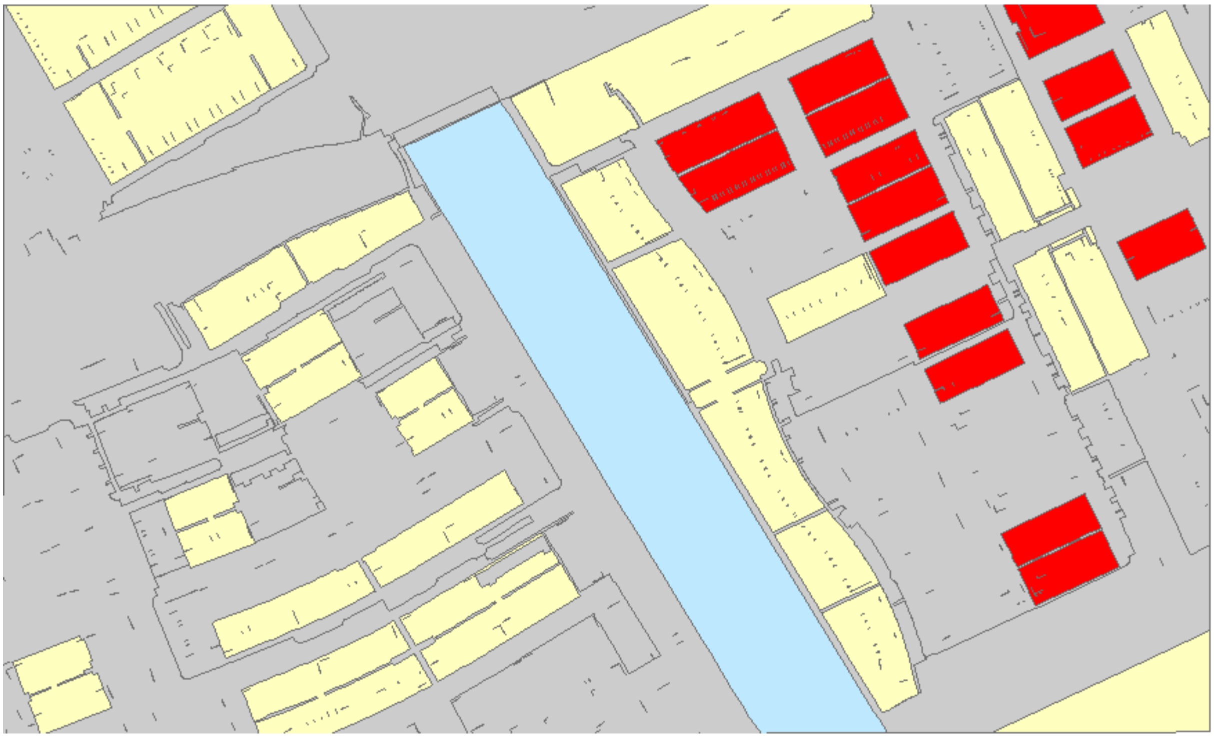

tGAP without constraint; importance=3500

tGAP without constraint; importance=3700

tGAP without constraint; importance=3805

tGAP without constraint; importance=3900

tGAP without constraint; importance=4000

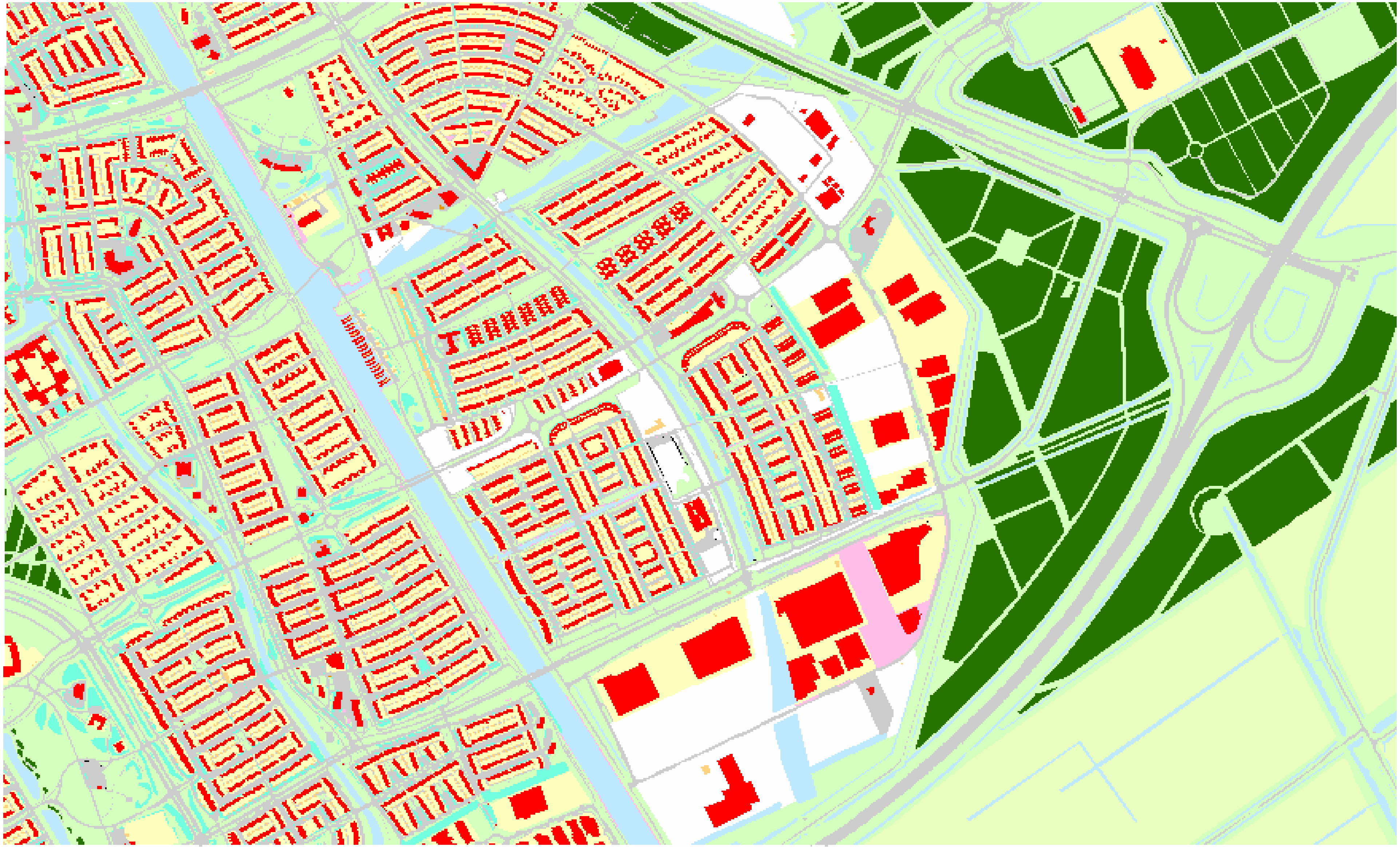


tGAP without constraint; importance=4100

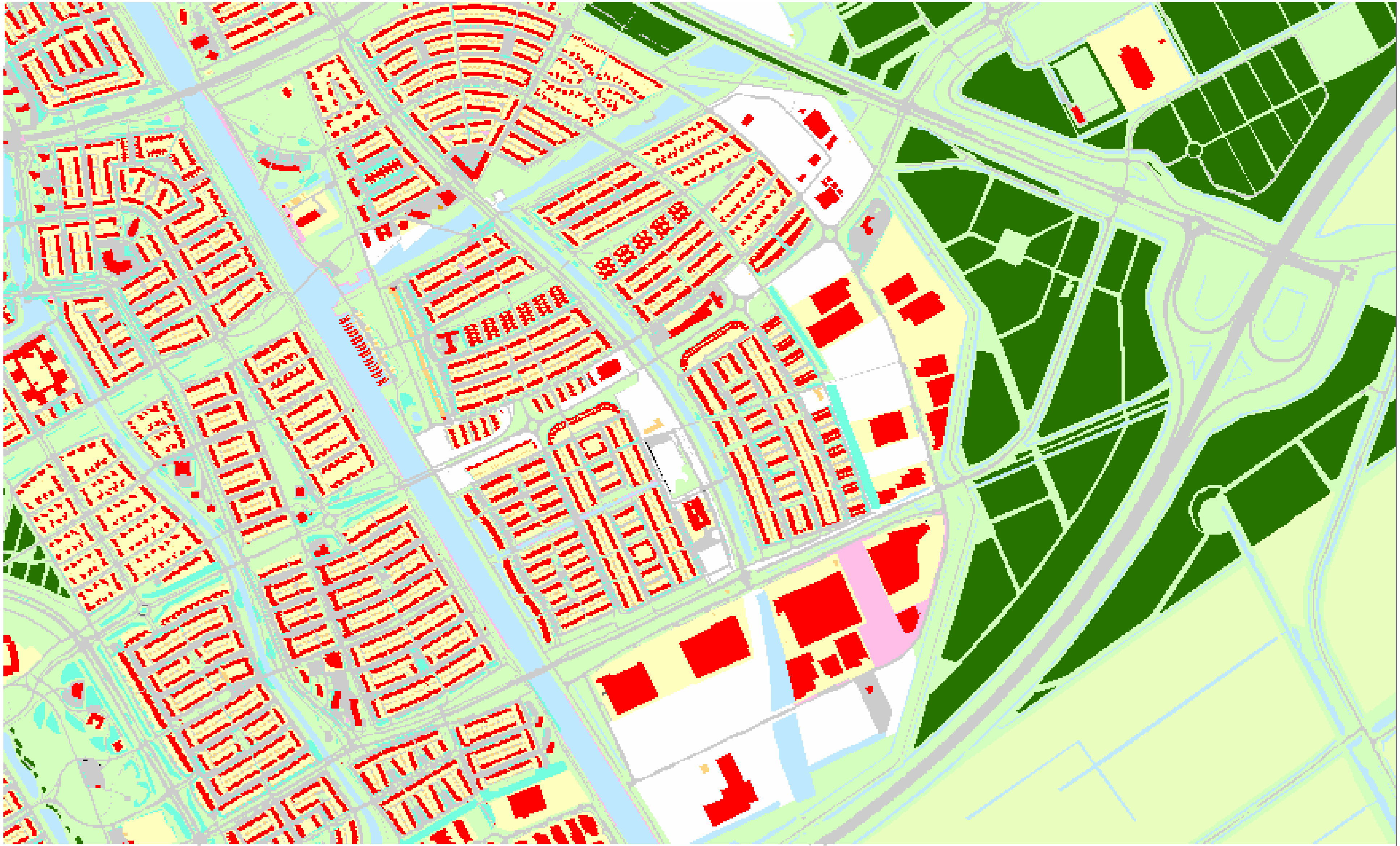
tGAP without constraint; importance=4148, only one (road) object left

Appendix K: Visualisation of the constrained tGAP methods for the large dataset

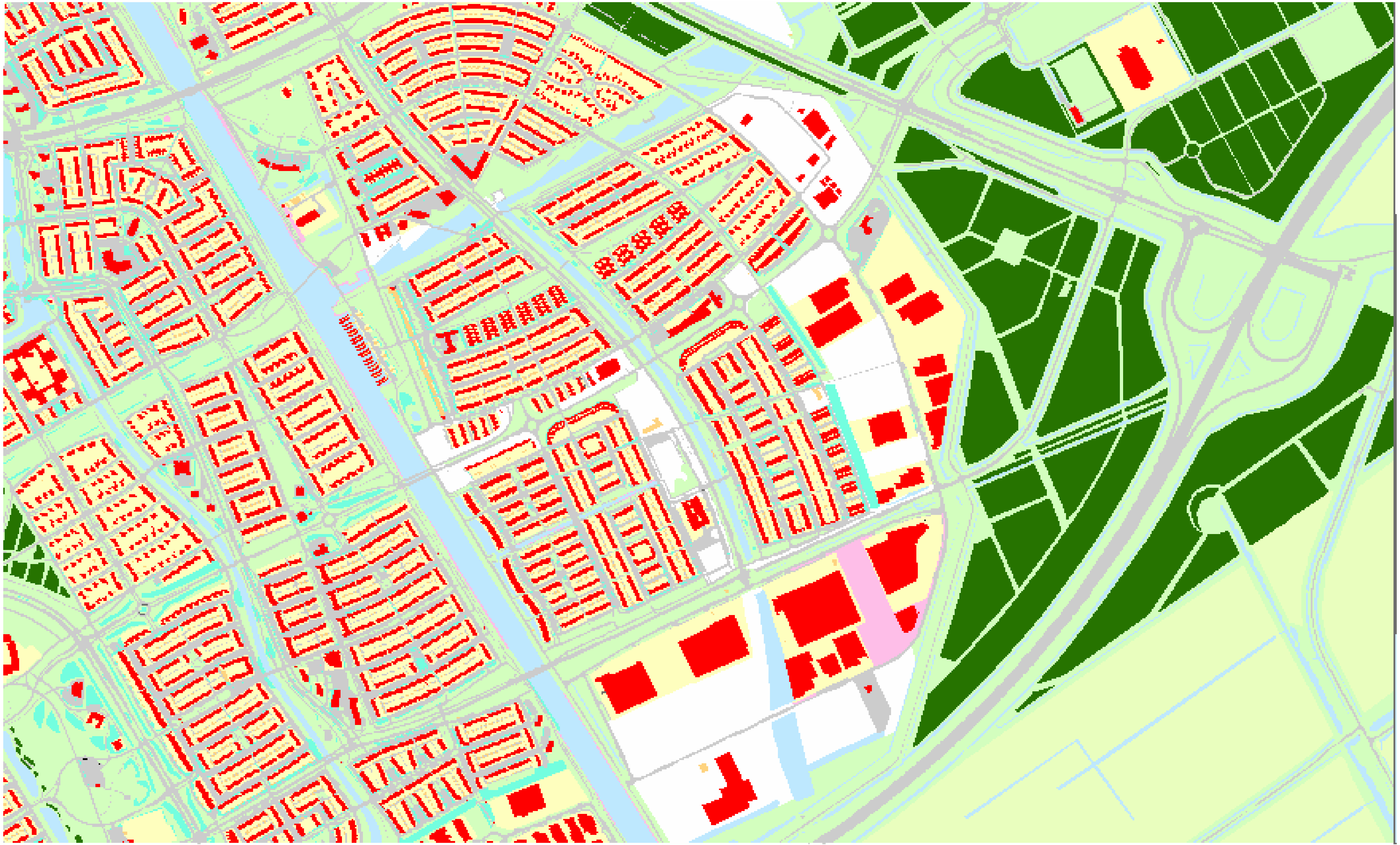
This appendix shows the visualisation in steps of the large dataset. Starting at importance level 0 (the original IMGeo file) to the end restult after 24482 merges.



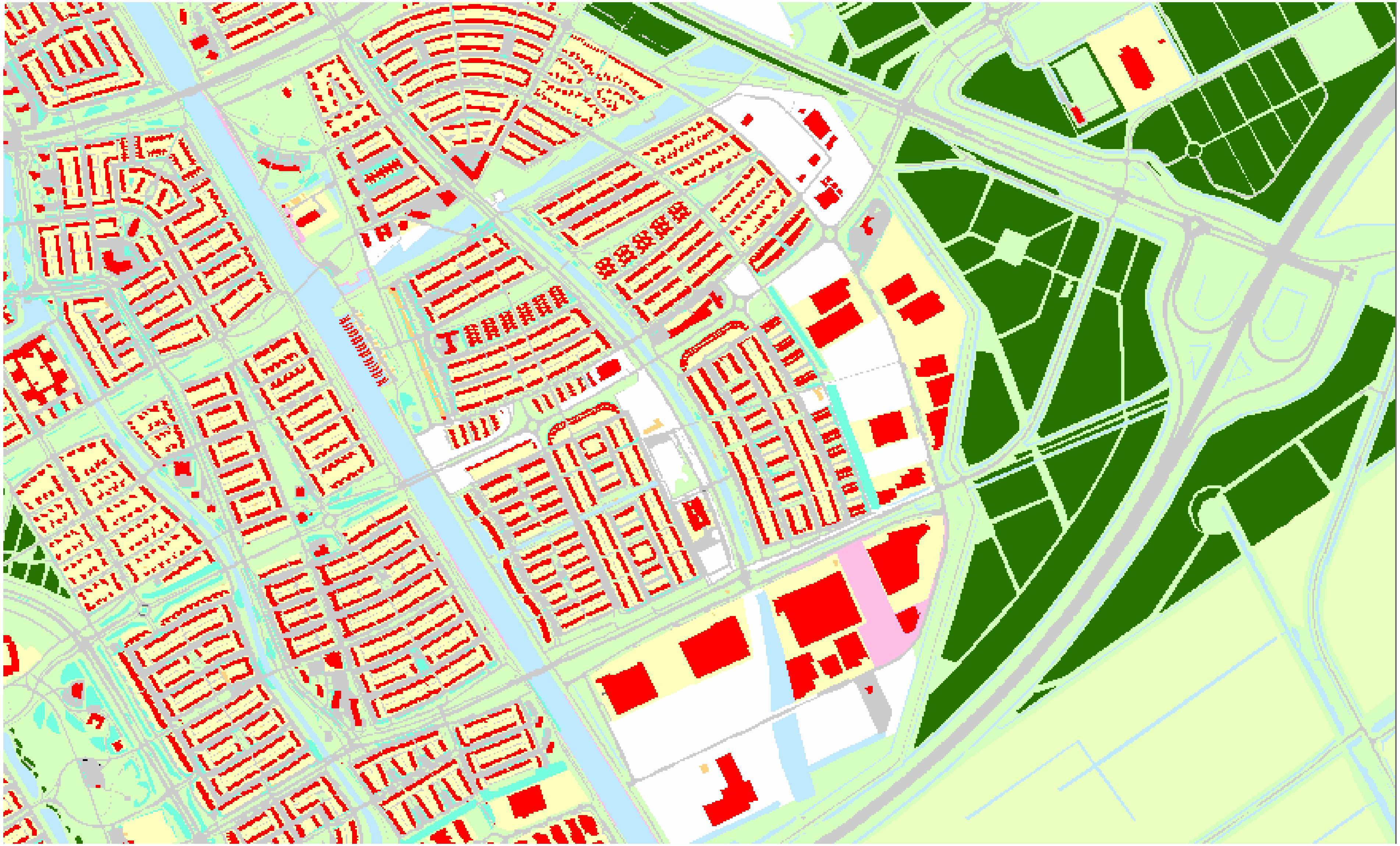
Visualisation of the large dataset; importance=0



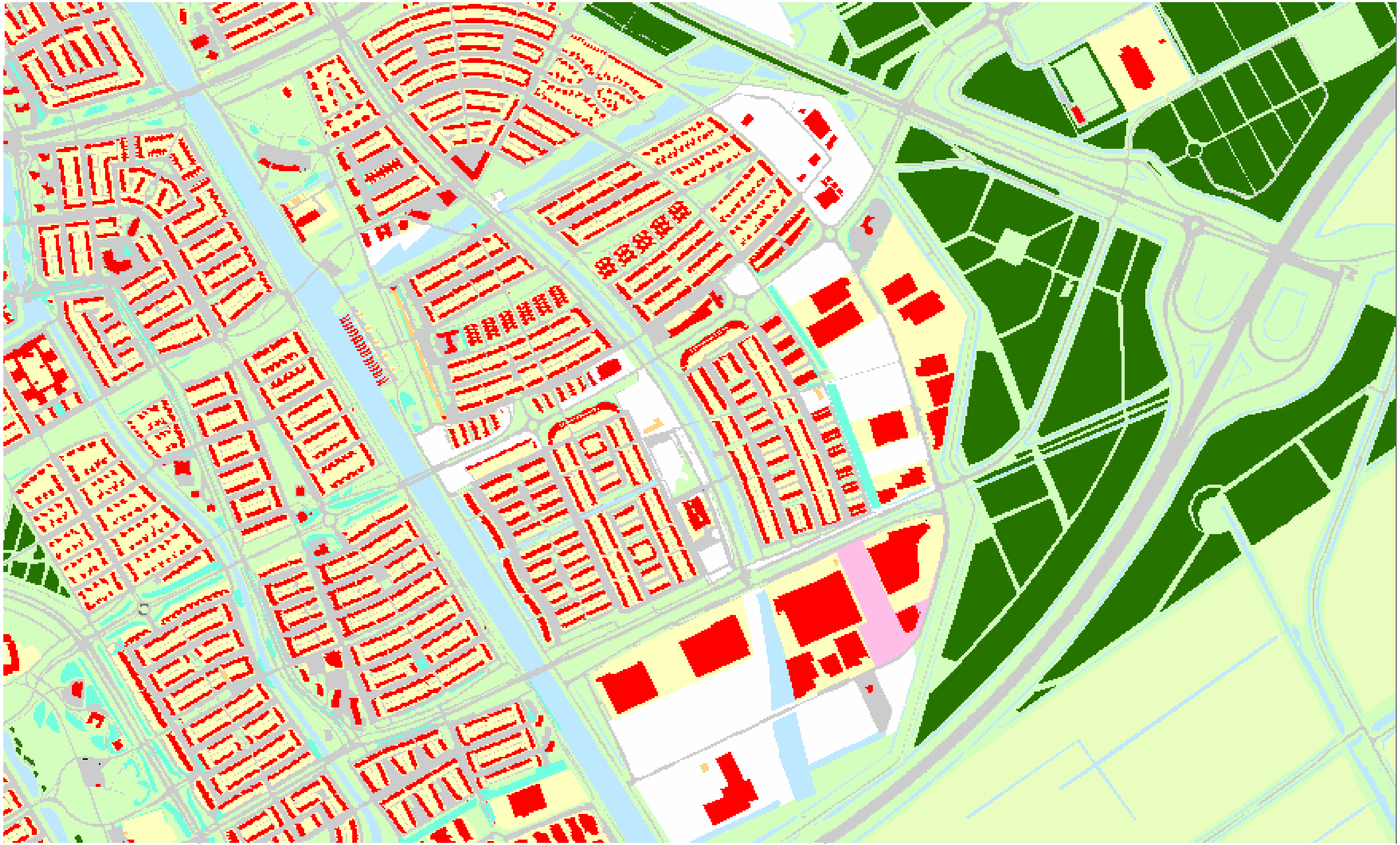
Visualisation of the large dataset; importance=4000



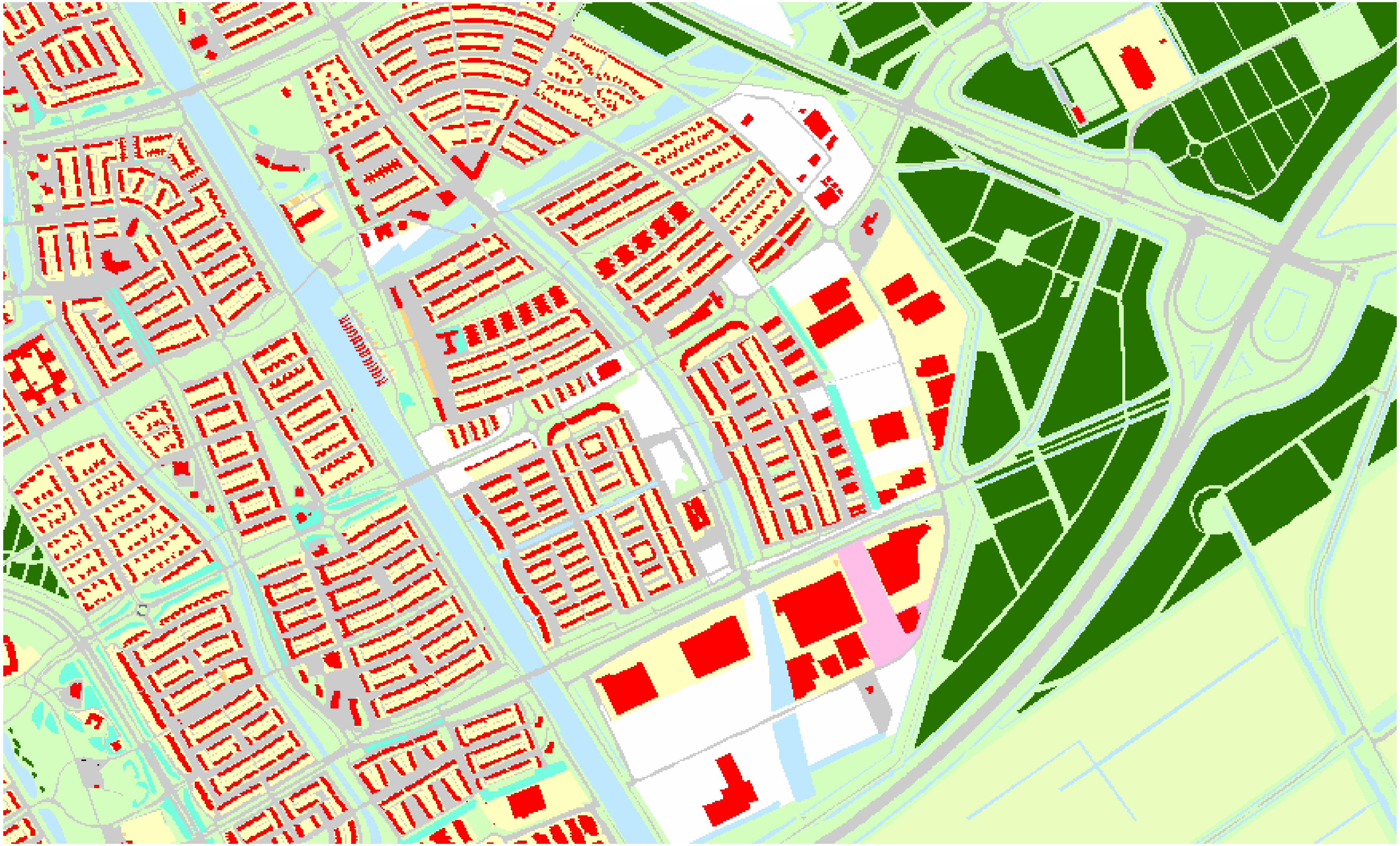
Visualisation of the large dataset; importance=8000



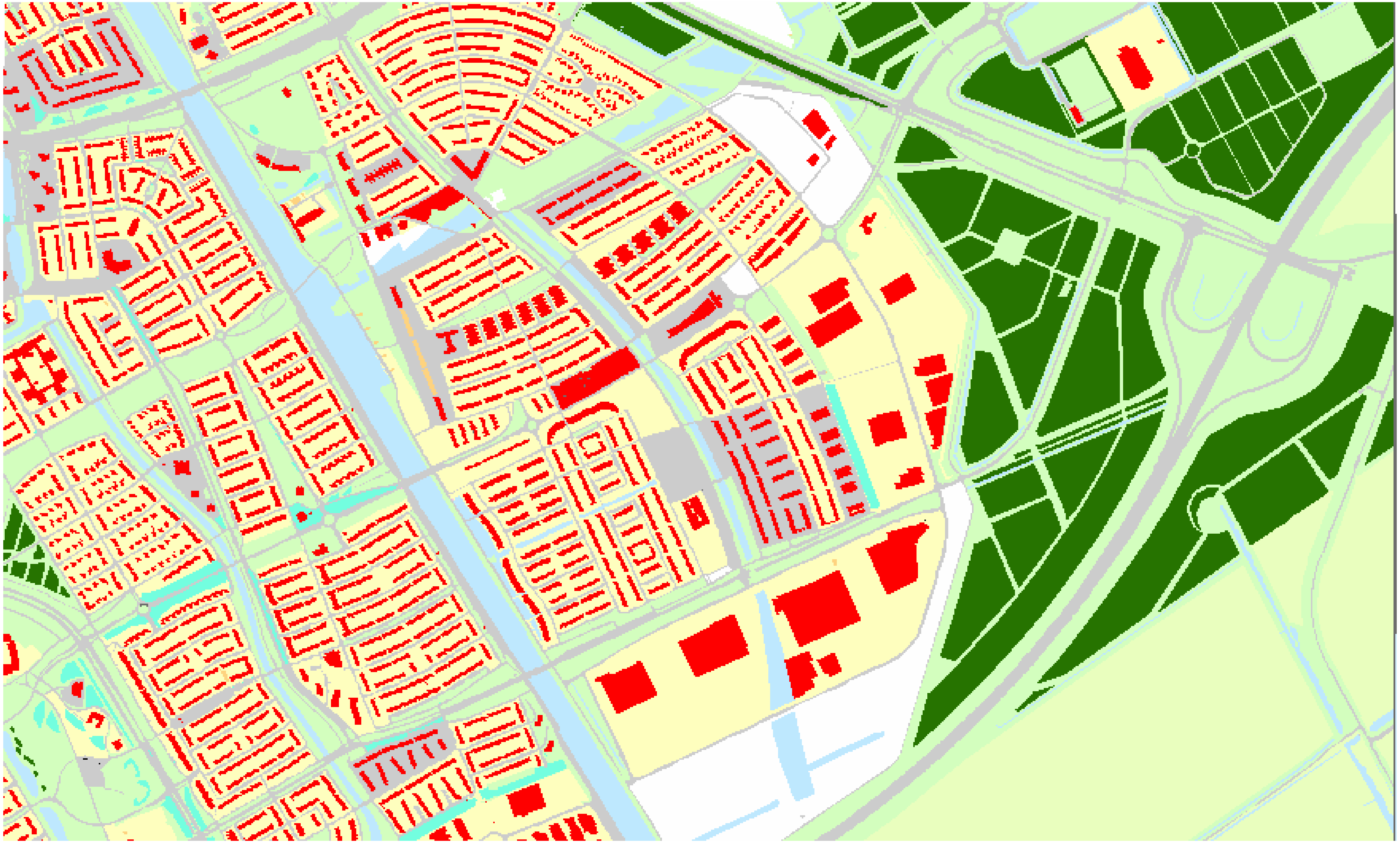
Visualisation of the large dataset; importance=12000



Visualisation of the large dataset; importance=16000



Visualisation of the large dataset; importance=20000



Visualisation of the large dataset; importance=24482