Report

Modelling Road Networks in the Netherlands

Geonovum

Date
Friday 29 October 2021

Author
Özge Tufan

Table of Contents

Chapter 1

Introduction

The aim of this report is to explain how a road network model is created for the Netherlands. Firstly, the functional requirements are identified from the Coherent Object Registration (SOR) conceptual model and compared with the INSPIRE Road Transport Network Model to investigate its suitability to fulfill the SOR requirements. Secondly, the model is created in UML which is then translated into a database to check certain functionalities with test data.

1.1 DisGeo: Coherent Object Registration (SOR)

In the Netherlands, various types of data that is used for numerous applications by many stakeholders are kept in separate basic registers with the aim of not having to collect these data more than once (Besemer et al., 2006). Examples to these basic registers are the Register of Addresses and Buildings (BAG) which includes the building footprints and the associated address information, and the Register of Topography (BRT) which includes topographic base maps for various scales. Even though the idea of basic registers successfully decreases the time and financial resources spent with the idea of not collecting the same data several times by different authorities, there is still a remaining problem when it comes to the interoperability of these basic registers. Even though the basic registers focus on different objects/concepts in the built environment, they may still include a certain level of overlap. Moreover, since all these basic registers have their own information models (Leijten et al., 2021), the integration between distinct basic registers can be challenging since the same real-world objects may have been defined differently in each of these registers. To solve this problem, the Ministry of the Interior and Kingdom Relations has started a project called DisGeo with the aim of bringing together the functionalities of four basic registers, which are the Register of Addresses and Buildings (BAG), the Register of Topography (BRT), the Register of Large-Scale Topography (BGT), and the Register of Land Registry (BRK) (Digitale Overheid, 2019). An important outcome of the DisGeo project is the development of the Coherent Object Registration (SOR) that aims to create a standard way of defining real-world objects and their properties so that the integration between distinct basic registers is ensured (Leijten et al., 2021). With this aim, the object types and properties within the BAG; BGT; BRT, parts of the Real Estate Valuation (WOZ) Administration; National Roads Database (NWB) and the BRK have been defined in a coherent way in the SOR conceptual model (Leijten et al., 2021).

1.2 Network Models

Network models are often used to represent the relationship between linear features on Earth such as roads, railways, waterways, and pipes. In the case of road network models, this information can be used for various purposes ranging from navigation to route determination; therefore, many initiatives have been taken over the years to create common road network models that can be used by many countries (INSPIRE, 2014; Geographic Data Files, 2020). In the Netherlands, the SOR includes a network model to define road networks; however, road network data is not yet included as one of the basic registers; therefore, it is not yet represented in a systematic way. Certain Dutch government agencies has created their own road network models to be able to represent roads as part of their datasets. For instance, National Roads Database (NWB) of Rijkswaterstaat includes almost all roads in the Netherlands while TOP10NL dataset in the Basic Register of Topography (BRT) includes road sections as well as many other topographical elements. However, the lack of a basic register about road network data creates differences in the way these road elements are defined. As a solution, the SOR conceptual model includes networks, elements in these networks and their relationships; however, since this is a conceptual model, there is no implementations yet. Therefore, this report first investigates the functional requirements for a road network model in the Netherlands by analysing the requirements for the SOR and the different road network datasets. Then, these requirements are compared with INSPIRE Road Transport Network Model to see if it is suitable to use in the case of the Netherlands.

1.3 Reading Guide

Following this chapter of Introduction, Chapter 2 analyses NWB – Wegen and TOP10NL datasets to get information about certain requirements of a network model. In Chapter 3, functional requirements for a road network model in the Netherlands are determined by investigating the network-related requirements in Coherent Object Registration (SOR), and in Chapter 4, these requirements are compared with the INSPIRE Road Transport Network Model to see if it is suitable to use this model in the case of the Netherlands. Chapter 5 explains how the UML modelling is done and how the model is translated into a database and tested with data. Finally, Chapter 6 concludes the report with key principles and findings.

Chapter 2

Road Network Data in the Netherlands

There are three main sources in the Netherlands where roads/road sections and their corresponding attributes can be found, namely the National Roads Database (NWB), TOP10NL dataset of the Basic Register of Topography (BRT) and the Basic Register of Large-Scale Topography (BGT). This chapter analyses the NWB and TOP10NL datasets to see how the road network data is included, and how the features and properties are added within the network model.

2.1 National Roads Database (NWB)

NWB – Wegen is the most comprehensive road network dataset in the Netherlands considering that it contains all roads with a street name and/or number, which corresponds to at least 98% of the Dutch road network (RWS et al., 2021). The two linear features used in this model are *road section* (wegvak) and *junction (junctie)* while the area representation of the road network is not a part of NWB - Wegen. These features have more than 50 attributes to add additional information to the model. In Table 1, the attributes in NWB – Wegen are brought together in logical groups instead of explaining each attribute individually since some attributes are highly related to one another.

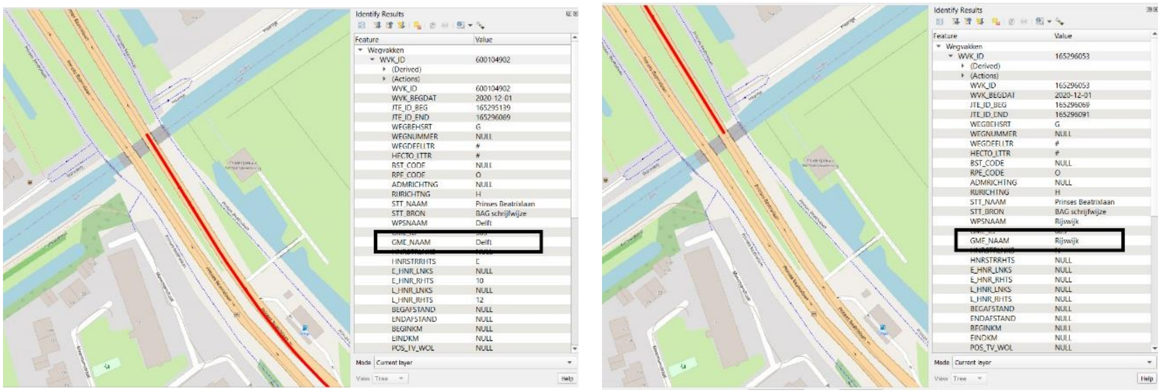
Table 1 Attribute groups in NWB – Wegen (Adapted from RWS et al, 2021)

Attribute group	Description
Identification number (ID)	Uniquely describes each road section and junction.
X and Y coordinates	Exact geographical position of road sections and junctions in RD New – EPSG: 28992.
Road management	The name and code of the authority that is responsible from a road.
Road location	Road number, street name, place, municipality name/code, district name/code.
Route information	Route letter and route number.
Road function	Differentiation between entrance, roundabout, parking, carriageway, etc.
Road position	Traffic flow direction and relative position of the road to determine left/right side.
Hectometer information	Information about road part letter, hecto letter, hectometer position and interval
House number*	Indicates the house numbers on the left/right side of a road. * For integration with the BAG dataset.

It is important to note that these attributes are attached to each road section element; however, most of them are optional in the sense that their values may be left NULL. This may result in the unnecessary storage of certain attributes for certain elements. For instance, hectometers are only located along provincial roads and national roads; therefore, the attributes about this are filled only for the road sections that are part of

a provincial/national road. As a result, these attributes will have NULL values for other roads such as municipal roads.
Another important aspect of NWB – Wegen dataset is that a change in one of the attributes of a road section results in a split into two or more road sections (RWS et al., 2021). For instance, a change in the street name or municipal boundry results in this split. This can be observed in Figure 1 where a change in the municipality name causes a split in the road section. This is not an efficient way to keep track of property changes of a road section since this leads to the unnecessary storage of unchanged attributes in smaller road sections.

Figure 1 Split in road section in NWB – Wegen dataset



2.2. Basic Register of Topography (BRT) TOP10NL

The Basic Register of Topography (BRT) is one of the basic registers in the Netherlands that contains topographic base maps for various scales, and TOP10NL dataset is one of these base maps that is suitable for the scales of 1:5,000 to 1:25,000 (Kadaster, 2020). This dataset includes many object classes such as buildings, terrain and relief, and road sections *(wegdeel)* are included as well. Point, line, and area geometries are used in this model and more than 30 attributes are attached to each of these geometries. Similar to NWB – Wegen, certain attributes relate to each other; therefore, these attributes are brought together in logical groups and shown in Table 2.

Table 2 Attribute groups in TOP10NL (Kadaster, 2020)

Attribute group	Description
Identification number (ID)	Uniquely describes each geometry.
Geometry	Geometry of the feature.
Temporal aspects	Start and end date of each element, and the registration date to the dataset.
Infrastructure type	Whether it is a connection, crossing, other traffic area, etc.
Road type	Differentiation between motorway, highway, regional road, street, etc.
Physical appearance	Whether it is on the fixed part of a bridge, in tunnel etc.

Pavement information	Width of the pavement, whether it is paved or not.
Road features	Number of lanes, road name, road number, status (in use, in progress, etc).
Bridge/tunnel information	Name of the bridge/tunnel.
Altitude level	Altitude level of the road.
Node information	Name of the node, if applicable.

It can be seen that some of these attributes are the same as NWB – Wegen attributes, which give basic information about a road section such as the ID, name and road type. On top of these attributes, TOP10NL adds more detailed information about the temporal aspects, number of lanes, name of tunnels/bridges, altitude level of a road section, and so on. The addition of these attributes is valuable in the sense that they allow for the specialization of road sections which can be used for further analysis. Moreover, the inclusion of area representation may be useful in certain applications where the use of line geometry is not sufficient. The connection between the linear representation and area representation is provided with the use of a *localid*. While all objects in TOP10NL have unique identification, *localid* stay the same for objects that are different representations of the same real-world object. For example, a road may be represented with both line and area geometries. In this case, the line object and the area object have unique IDs while their *localid* is the same to show their connection. However, this type of a connection is not made between distinct objects such as roads and buildings. Therefore, it is not possible to show how different kinds of objects in real-world relate to one another in TOP10NL.

Chapter 3

Functional Requirements from the DisGeo Coherent Object Registration (SOR)

This chapter focuses on identifying the functional requirements of a road network model in the Netherlands by considering the requirements for networks in the coherent object registration (SOR). The identified requirements are also compared with the NWB and TOP10NL datasets to see their availability in these network models.

3.1 Overview of the Requirements

Functional requirements of a road network model might differ from country to country depending on the existing systems and needs. The conceptual model for coherent object registration (SOR) in the Netherlands provides requirements for different types of networks such as roads, railways and waterways (Leijten et al., 2021). By focusing on the requirements for a road network in the SOR, Table 3 gives an overview of the functional requirements that have been identified for creating a road network model in the Netherlands. These are compared with NWB – Wegen and TOP10NL datasets and it is specified with *Yes* or *No* if similar elements are part of these datasets even if their names are not exactly the same. Then, each requirement is described in detail later in this chapter.

Table 3 Functional requirements of road network data

Requirement	**Present in NWB**	**Present in TOP10NL**
Identification of All Elements	Yes	Yes
Temporal Attributes		
StartValidity	No	Yes
endValidity	No	Yes
timeRegistration	Yes	Yes
Separate Network Description	No	No
Network Elements		
Node – *linear*	Yes	Yes
Link – *linear*	Yes	Yes
Node – *area*	No	Yes
Link - *area*	No	Yes
Level of Detail (LoD)	No	No
Road level	-	-
Carriageway level	-	-
Lane level	-	-
Traffic Functional Zones		
Geometry	No	No

Functional zone types	Yes	Yes
Road Zones		
Geometry	No	No
Road zone types	No	No
Separation of Real and Functional Objects	No	No
Properties		
Road type	Yes	Yes
Route number	Yes	Yes
Exit number	No	Yes
Hectometer point	Yes	No
Driving direction	Yes	No
Public road indication	No	No
Status	No	Yes
Linear Referencing	No	No
Network Relationships		
Relationships between network elements	Yes	Yes
Relationships between network elements and functional objects	No	No

3.1.1 Identification of All Elements

All elements in a road network must be uniquely identified with an ID so that no duplicates are allowed, which would cause errors in the dataset.

3.1.2 Temporal Attributes

Each element in a road network must have temporal attributes showing the validity of information according to legislation with the *startValidity* and *endValidity* attributes. Moreover, *timeRegistration* attribute must be present to show the date when the data was entered into the database. It is important to note that these temporal concepts are implemented in the SOR from the NEN 3610 standard, and they will not be dealt with in detail in this report.

3.1.3 Separate Network Description

It is found more practical according to the SOR to be able to define each network separately so that it is possible to create connections between different networks by using their elements.

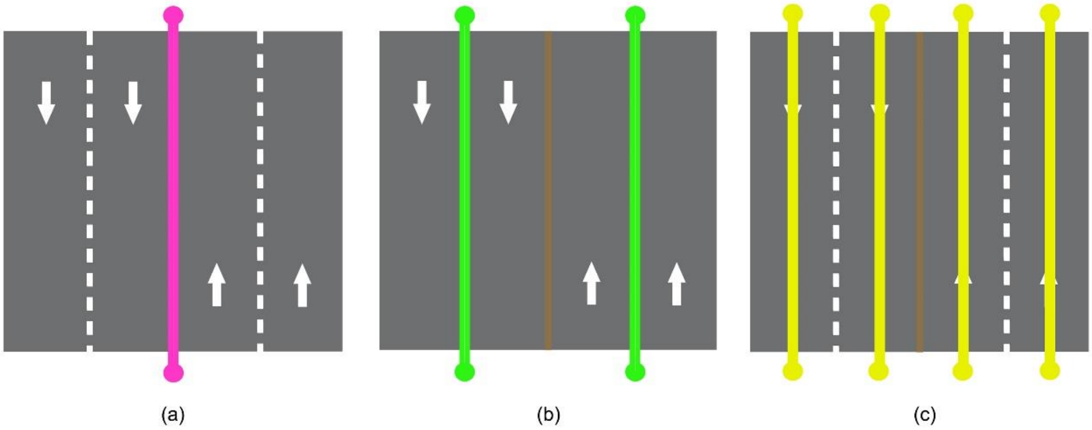
3.1.4 Network Elements

A network has two network elements: nodes and links. While nodes indicate a choice point where the user can decide between different directions, links are created by connecting two adjacent nodes. According to the SOR, the linear representation is mandatory for nodes and links with 2D *Point* and 2D *Line* geometries respectively. Moreover, an area representation may be added to these elements by using 2D planes; however, this representation is not mandatory in the SOR.

3.1.5 Level of Detail (LoD)

A road network may be represented in different levels of detail depending on the model requirements and different applications. According to the SOR, 3 distinct LoDs can be described: road level, carriageway level and lane level. While the road level and carriageway level are required to define a road network, lane level is left optional; therefore, it may be added if it is necessary to describe the network (Leijten et al., 2021). Figure 2 illustrates how different levels of detail are defined along a road according to the SOR. While the road level (a) represents the road with a single centerline geometry, the carriageway level (b) represents the same road with two centerline geometries for the two carriageways, and the lanel level (c) represents it with centerline geometries for each lane.

Figure 2 Levels of Detail: (a) Road level, (b) Carriageway level, (c) Lane level



3.1.6 Traffic Functional Zones

The coherent object registration defines traffic functional zones as functional spaces related to the traffic that are located along a network. Examples to such spaces are parking areas, gas stations, service areas, and so on. These functional zones are not part of a road network, but they have a (hyper)connection to the nodes and links along which they lie. Moreover, they are represented as 2D plane geometries.

3.1.7 Road Zones

Road zones are defined as functional spaces used to separate various road layout (Leijten et al., 2021). These zones are represented as 2D plane areas, and examples to such road layout areas are speed bumps, traffic islands and cattle grids. However, even though a road zone is a subclass of functional spaces, it is not determined in the requirements for SOR how these road zone objects will relate to the network elements such as nodes and links. It can be said that this creates confusion about the modelling of these objects since there is no indication about the desired relationship between road zones and the network.

3.1.8 Separation of Real and Functional Objects

The coherent object registration aims to separate functional objects from real objects so that the real objects can be represented with distinct geometries and be linked to their corresponding functional objects. In the case of road networks, this consists of separating network elements (nodes and links) from the information regarding road surface. In this model, while pavement object is modelled as a real object, network elements are modelled separately so that these two concepts can be linked afterwards.

3.1.9 Properties

Many properties may be attached to objects in a road network. According to the SOR, these properties include *Road type*, *Route number*, *Exit number*, *Hectometer point*, *Driving direction*, *Public road indication* and *Status* of the element. While the status is available for both nodes and links, all the other properties are defined only for links.

3.1.10 Linear Referencing

Linear referencing can be used to attach properties in certain distances of a link. The SOR uses linear referencing for the properties that may change along a link. For instance, the hectometer point property is used to define the hectometer points along a link. Since there may be more than one hectometer point, this property is attached to links by using linear referencing. According to this, the distances from the start of a link are specified to record the exact location of a hectometer point. However, other properties that do not change along a link, such as Road type and Driving direction, are added as attributes to each link element instead of using linear referencing.

3.1.11 Network Relationships

The SOR defines two main relationship types in a network model. Firstly, a relationship between network elements is defined. For instance, a relationship between nodes and links qualifies as this type of a relationship since both nodes and links are defined in the road network model. Secondly, a concept of *hyperconnection* is defined to explain the relationship between network elements and functional objects that are not part of the network. For instance, a relationship between a link and a Traffic Functional Zone object is modelled as a hyperconnection. Since DisGeo models not only road networks but also many other concepts and objects in real world, this second type of relationship is needed to be able to connect objects from different models within DisGeo.

Chapter 4

INSPIRE Road Transport Network Model

This chapter compares the functional requirements according to the SOR with the elements in INSPIRE's network model for road transport to decide if this model could be used to meet all the requirements.

4.1 INSPIRE Road Transport Network Model

In this section, the functional requirements that have been determined in the previous chapter are compared with the INSPIRE Road Transport Network model to see if it can be directly used for the case of the Netherlands, and to identify the deficiencies and needed improvements to this model. INSPIRE's Road Transport Network Model depends on two other network models of INSPIRE that are more generic. Firstly, Generic Network Model defines the basic features such as Nodes and Links that are shared by any kind of network (INSPIRE, 2014). Then, an application schema for Common Transport Elements is defined with elements that are shared by various networks and common properties that may be used by these networks. Finally, a separate application schema is created for Road Transport that makes use of the two previous models and adds on them by including features and properties that are specific to Roads. In Table 4, the functional requirements are compared with the features and attributes in INSPIRE's model.

Table 4 Functional requirements and INSPIRE elements

Functional Requirements	INSPIRE Network Model Elements	Element type
Identification of All Elements	inspireId	Attribute
Temporal Attributes		
startValidity	validFrom	Attribute
endValidity	validTo	Attribute
timeRegistration	beginLifespanVersion	Attribute
Separate Network Description	Network	<<featureType>>
Network Elements		
Node - *linear*	Node	<<featureType>>
Link - *linear*	Link	<<featureType>>
Node - *area*	NetworkArea	<<featureType>>
Link - *area*	NetworkArea	<<featureType>>
Level of Detail (LoD)		
Road level	Not present	-
Carriageway level	Not present	-
Lane level	Not present	-
Traffic Functional Zones		
Geometry	Not present	-

Functional zone type	FormofWay	Property
Road Zones		
Geometry	Not present	-
Road zone type	Not present	-
Separation of Real and Functional Objects	Not present	-
Properties		
Road type	FormofWay	Property
Route number	Not present	-
Exit number	Not present	-
Hectometer point	Not present	-
Driving direction	TrafficFlowDirection	Property
Public road indication	AccessRestriction	Property
Status	ConditionofFacility	Property
Linear Referencing	SimpleLinearReference	<<dataType>>
Network Relationships		
Relationships between network elements	Simple associations	
Relationships between network elements and functional objects	NetworkReference	<<datatype>>

It can be seen that while many requirements can be modelled with similar INSPIRE objects, there are still certain elements that are not present in the INSPIRE model. Moreover, it is more appropriate for some requirements to define new elements instead of using the similar concepts in INSPIRE considering the explanations in the SOR. Therefore, each element is explained below and the suitability of using the indicated INSPIRE element is assessed.

4.1.1 Identification of All Elements

This requirement is met with the "inspireId" attribute which has a unique identifier as the attribute value so that each element can be uniquely described.

4.1.2 Temporal Attributes

It can be said that the temporal attributes required according to the SOR have a direct translation to INSPIRE elements. While *validFrom* and *validTo* attributes are used to specify the date and time that the object became valid/invalid in real-world, *beginLifespanVersion* specifies the date and time that the object was added to the dataset (INSPIRE, 2014). It is important to note that INSPIRE also has an *endLifespanVersion* attribute to specify the date and time that the object was removed from the dataset; however, since this is not stated as a requirement in the SOR, it is not included in the table.

4.1.3 Separate Network Description

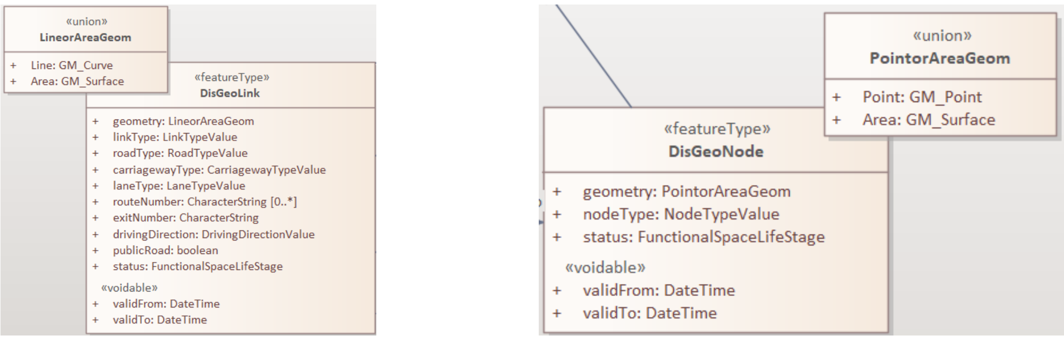
This requirement has a direct translation to INSPIRE with the *Network* element. Each Network element may represent a different network such as Road network, Cable network, Railway network, and so on. This way, elements of each network are separated from each other and kept as distinct models.

4.1.4 Network Elements

INSPIRE uses the primitives defined in the ISO 19107 (2019) standard. For linear representation, INSPIRE has Node and Link elements that represent the choice points and the connections between them with

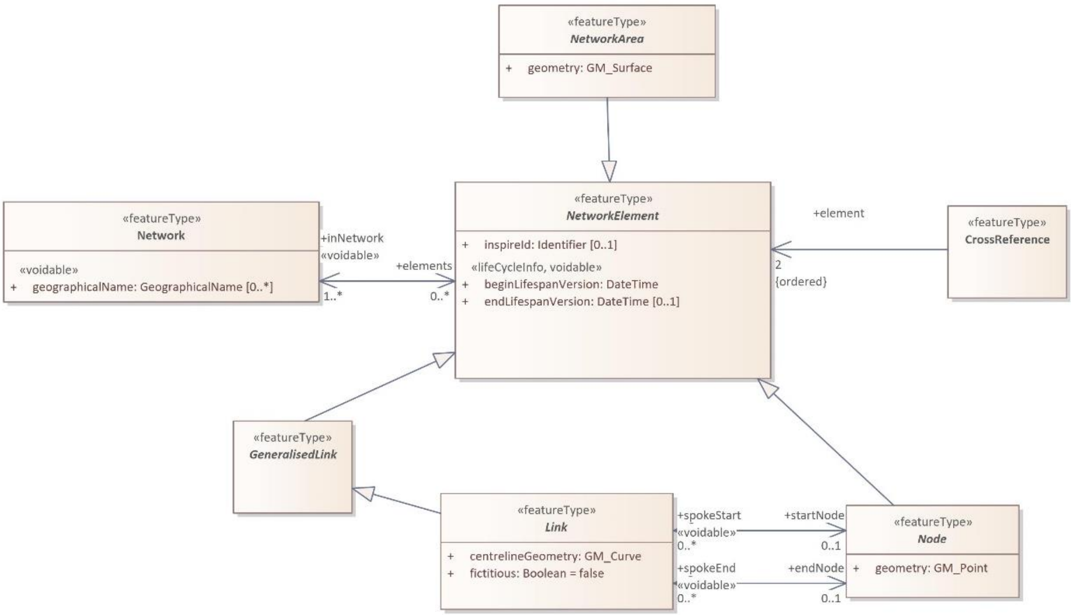
GM_Point and GM_Curve geometries respectively. This way, the connection between Nodes and Links can be provided since the topological relationship between them is explicit. However, area representation is formulated differently in the SOR than in INSPIRE. In the SOR, Node and Link elements are created such that the geometry can be chosen as linear representation or area representation. Figure 3 shows how Nodes and Links would be modelled according to the SOR with this method. While a Node object can have a geometry of GM_Point or GM_Surface, the geometry of a Link object can be chosen as GM_Curve or GM_Surface. This method might create a more compact model with only two network elements (Node and Link) with the possibility of choosing distinct geometries; however, it might also create problems when it comes to implementation. For instance, storing different geometries together as the same type of element might complicate the model and make it harder to implement.

Figure 3 Geometry options of Links and Nodes according to the SOR



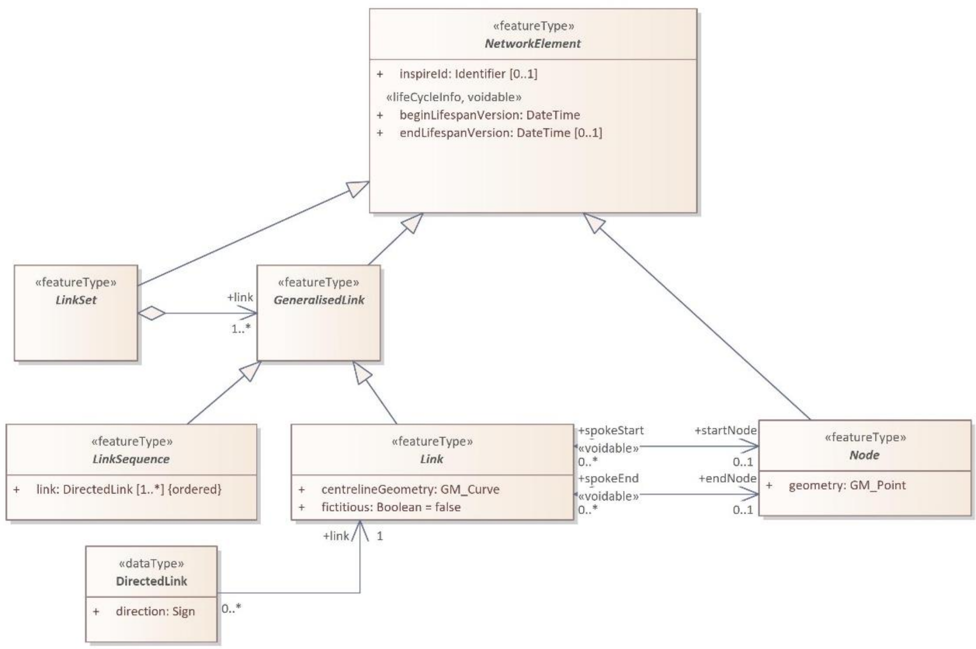
In contrast, the INSPIRE model does not provide Node and Link elements that can be modelled with a GM_Surface geometry. Instead, a NetworkArea element is created with a GM_Surface geometry to be able to include an area representation for both nodes and links (Figure 4) while keeping this as a separate element from the linear representation.

Figure 4 Area representation (INSPIRE, 2013)



Since a real-world object may be modelled with both linear and area representations, INSPIRE provides the *CrossReference* element to be able to connect distinct representations of the same spatial object. For instance, CrossReference might take a *Network_Area* and a *Link* element to specify that these two elements are different representations of the same real-world object. It is important to note that there is no specification in the SOR about how the linear and area representations would be connected. Therefore, the *CrossReference* concept of INSPIRE might be beneficial for this purpose in the implementation part. Furthermore, the INSPIRE model has two more feature types for linear representation besides Node and Link. While a *LinkSequence* is used to group links provided that they are given a direction, a *LinkSet* can be used to group links and link sequences (INSPIRE, 2014). How these features are modelled in INSPIRE is shown in Figure 5. It can be seen that these extra features can be beneficial since they allow for the grouping of links to represent distinct routes. Moreover, they can be used to use linear referencing more efficiently for the cases where the reference continues from one link to the next one. Since the SOR only defines nodes and links as network elements, these extra concepts are not included in the model. However, they can be used in the future to develop the network model in the SOR.

Figure 5 LinkSequence and LinkSet elements in the INSPIRE model (INSPIRE, 2013)



4.1.5 Level of Detail (LoD)

The SOR allows a network to be defined in three levels of detail: road level, carriageway level and lane level. This is ensured with the *linktype* attribute for Links where the value can be road, carriageway or lane to describe the LoD of the element. For Nodes, on the other hand, even though it is desired to specify the LoD, it is not mentioned in the SOR how to achieve this. Similarly, INSPIRE does not have such a mechanism to represent a network in various LoDs. Since it is not possible to meet this requirement with the INSPIRE model, new Node and Link elements will have to be defined by taking into account these three LoDs so that it can be specified which level each element belongs to.

4.1.6 Traffic Functional Zones

A traffic functional zone represents an area that is important for the road network, but is not a part of it such as a gas station or a parking area. It must be modelled with a 2D plane geometry outside the network model. The reason why this element is desired to be modelled outside the network is that Traffic Functional Zones are defined as a subclass of Functional Spaces, which includes other elements that are not related to the road network model such as Railway Zones, Aviation Zones, and so on. Therefore, it is desired to model each of these objects separately, and to have a mechanism to connect them when necessary. INSPIRE has a FormofWay property that includes in a code list a classification of roads, which includes the values for traffic functional zones. For the geometric representation, INSPIRE only has a NetworkArea element to define area objects in real-world; however, this element is a subclass of *NetworkElement*, which means that it is modelled as part of the network. Therefore, a new TrafficFunctionalZone element must be created outside the network model with a GM_Surface geometry and related attributes.

4.1.7 Road Zones

Similar to Traffic Functional Zones, Road Zones must be modelled outside the road network. Therefore, a new RoadZone element should be added with a GM_Surface geometry and needed attributes to meet this requirement.

4.1.8 Separation of Real and Functional Objects

In INSPIRE, there is a RoadSurfaceCategory property that specifies if a surface is paved or not. However, this does not meet the requirement of separating real and functional objects since it is only added as a property without a geometry. In the SOR, it is required that the Pavement object has a 2D plane geometry so that the exact surface areas can be determined and modelled. Moreover, this object must be modelled outside the network as well.

4.1.9 Properties

The way the properties of a Link is modelled in the SOR is highly different than INSPIRE. In the former, all properties are added as attributes unless they are location specific. Only in this case, linear referencing is used to attach these properties in certain positions along a Link. In INSPIRE, on the other hand, all properties are defined as separate elements that can be attached to network elements with or without linear referencing depending on the nature of the property. This has the benefit of avoiding NULL values in attributes since only the related properties are attached. However, this can result in a complex system harder to form connections between properties and network elements. Therefore, each property defined in the SOR is added as an attribute to the corresponding network element, and only the attributes requiring linear referencing are added as properties with INSPIRE's method. This way, the complexity of the model can be decreased; however, it does not solve the problem of having NULL values since certain attributes might stay empty for some elements. Moreover, it is important to note that the INSPIRE model has more than 15 attributes such as Speed limit, OwnerAuthority, Number of Lanes, and so on while the SOR defines only 7 attributes for Links. Therefore, the INSPIRE properties can be analysed in the future to see if it is beneficial to add some of them to the SOR for including more detailed information.

4.1.10 Linear Referencing

In INSPIRE, linear referencing is only used to attach properties to network elements. However, elements like Traffic Functional Zones and Road Zones in the SOR require the use of linear referencing as well; therefore, these elements will be provided with a linear referencing mechanism by using the INSPIRE data types.

4.1.11 Network Relationships

While the relationships between network elements are modelled as simple associations in the INSPIRE model, there is no other type of network relationship that connects network elements with the elements from different models. Therefore, elements outside the network such as Traffic functional zones and Road zones are defined in a separate package so that the differentiation between network elements and the others is ensured. To connect these with the network elements, the SOR uses the term *hyperconnection* to differentiate this type of relationship from the simple relationships between network elements. In INSPIRE, a similar relationship can be formed by using the NetworkReference concept that is originally used to attach properties to network elements. This concept will be explained in detail in the next chapter.

Chapter 5

UML Modelling and Testing

This chapter explains how a UML model is created for road network data by considering the functional requirements identified earlier in this report. Then, it is described how the elements in the UML diagram are translated into a database to create some test data to check certain features. Finally, an alternative option is provided to make the relationship between linear and area representations more explicit.

5.1 Road Network Model in UML

5.1.1 Road Network

All UML modelling is done using Enterprise Architect, and a separate root node called *Model* is created to store all individual packages. Figure 6 shows how the road network elements are modelled with UML, and the enumerations are shown in Figure 7. The features with the *Network::* namespace belong to the Generic Network Model of INSPIRE (2013), while the other features are new additions that are required to fulfil some requirements. Accordingly, the Network feature is used to describe each network separately. This is necessary when there can be various networks within a road network such as a road network for cars or a road network for bicycles. A road Network may have 0 or more Network Elements. Instead of using the INSPIRE’s Node and Link features, two new features are created as DisGeoNode and DisGeoLink to differentiate from the INSPIRE model. Since the SOR enables the representation of a Network Element either with a linear or area geometry, these options are provided in the UML model as well. While the geometry of a DisGeoNode can be chosen from the PointorAreaGeom union (GM_Point or GM_Surface), the geometry of a DisGeoLink can be chosen from the LineorAreaGeom union (GM_Curve or GM_Surface). Similar to INSPIRE, the geometric primitives defined in ISO 19107 standard (2019) are used in this model. Moreover, attributes of Network Elements are added like they are described in the SOR. To connect DisGeoLink elements from different LoDs, the *relatesTo* relationship is formed with DisGeoLink itself so that, for instance, one road can be described in different LoDs, and all links can relate to one another. Finally, the CrossReference feature is used from INSPIRE to make the connection between linear and area geometries. CrossReference takes two Network Elements to state that they are different representations of the same real-world object. This is crucial to be able to switch between different representations.

Figure 6 UML Model for Road Networks

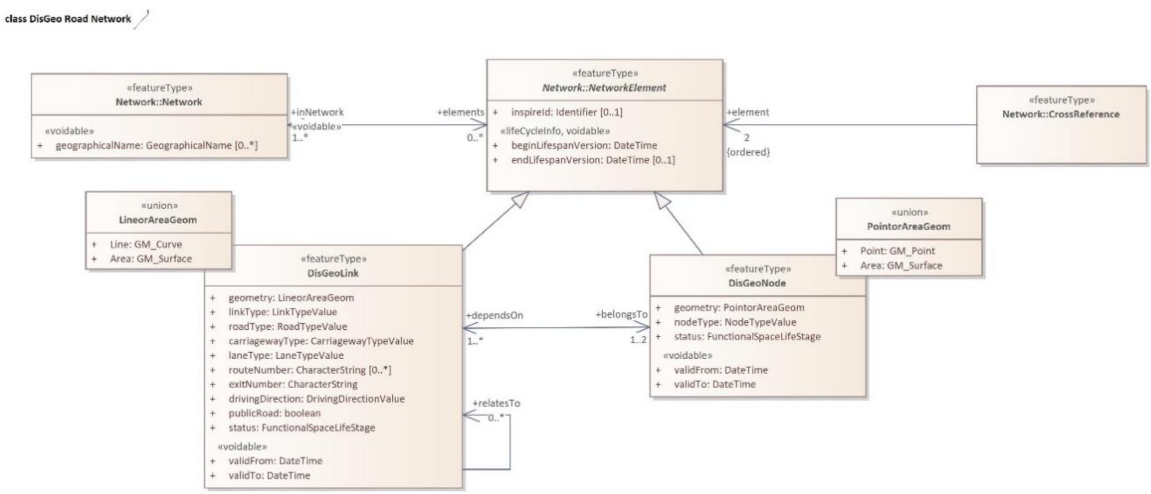
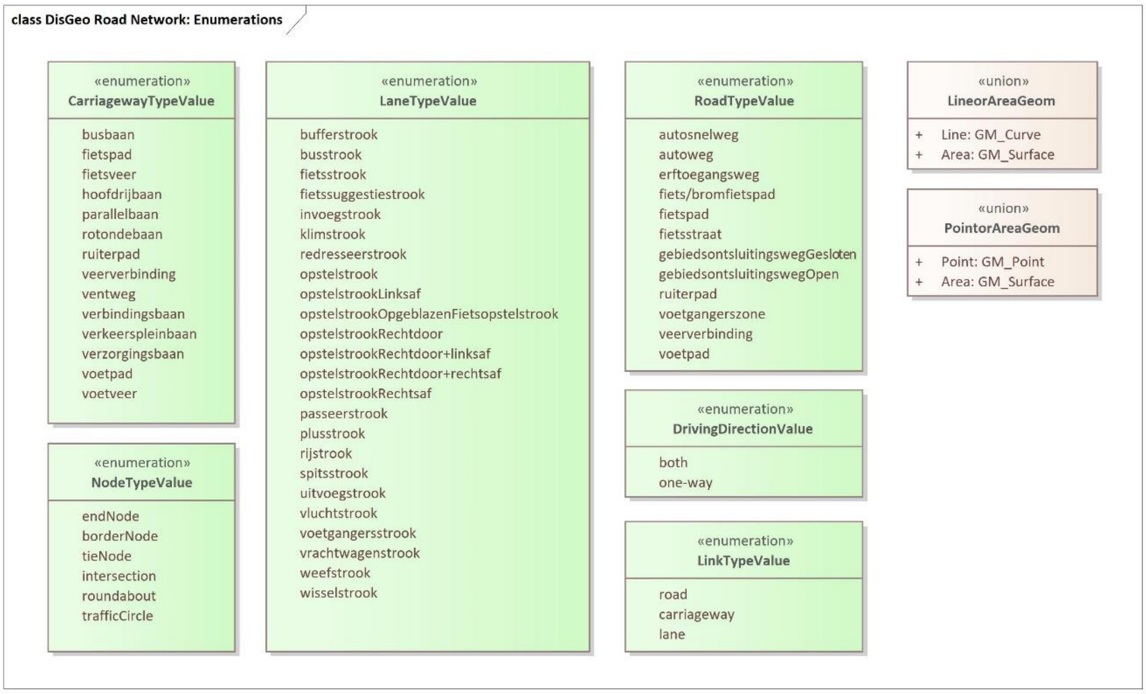


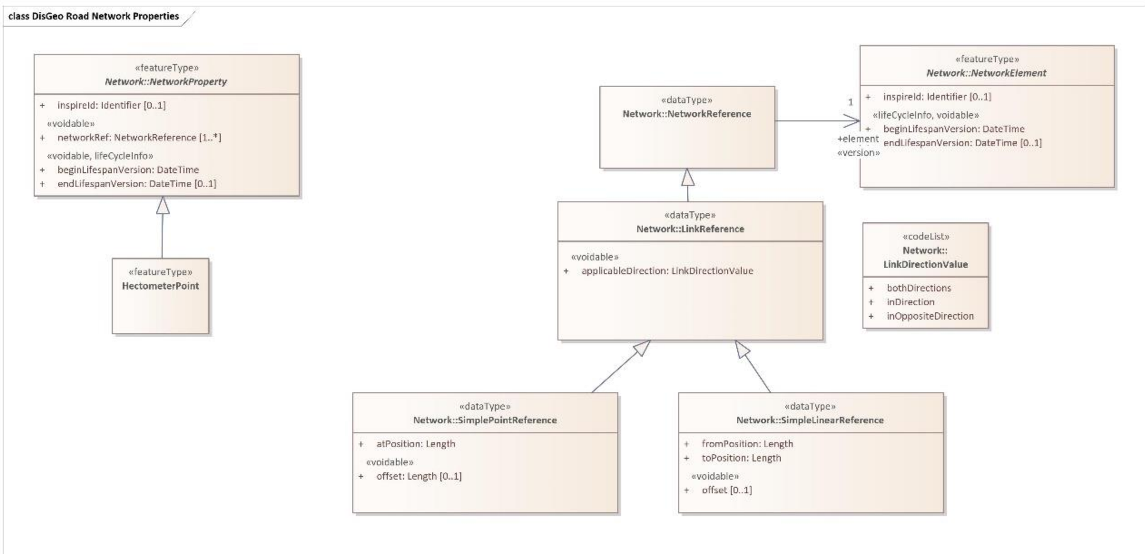
Figure 7 Enumerations for the Road Network Package



5.1.2 Road Network Properties

As the SOR states, location-specific properties are attached to links with linear referencing (Leijten, 2021). Since the INSPIRE model already includes linear referencing, this part of the model is added directly (Figure 8). Properties in INSPIRE may be connected to a Network Element with the *networkRef* attribute. The data type for the value of this attribute changes according to the nature of the property. If NetworkReference data type is chosen, the property applies to the whole of the Network Element. On the other hand, SimplePointReference and SimpleLinearReference data types are used to attach the property on a certain position along the link and on an interval along the link, respectively. The only location-specific property in the network model of the SOR is *HectometerPoint*s, which are located every 100 meters along provincial/national roads; therefore, this is added separately as a property instead of an attribute of links.

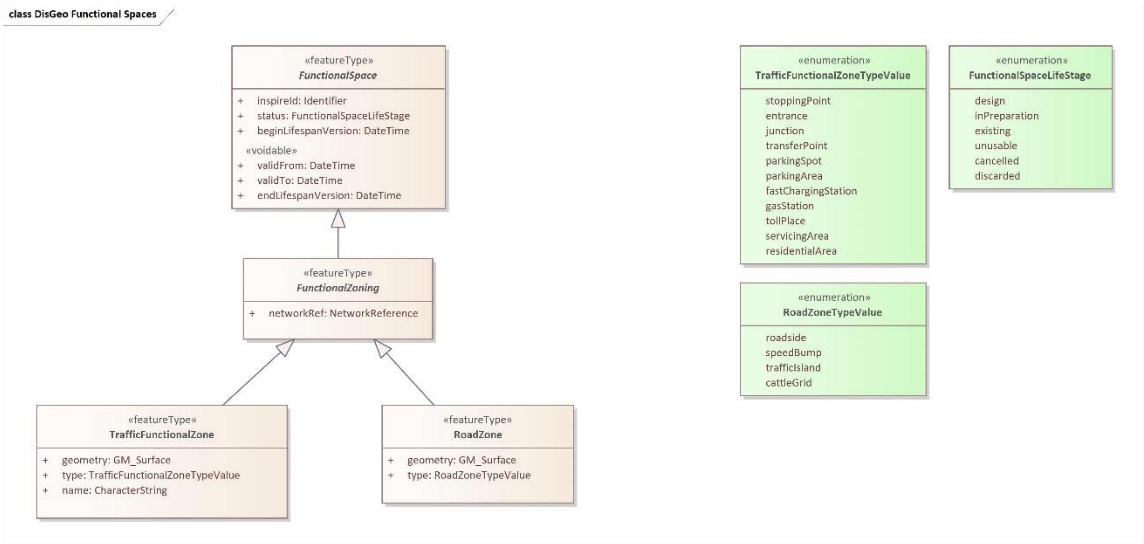
Figure 8 UML Model for Road Network Properties



5.1.3 Functional Spaces

The two features that have a relation to the road network model in Functional Spaces are Traffic Functional Zones and Road Zones. These features are not modelled as Network Elements since the SOR states that Functional Spaces must be modelled outside the network (Leijten, 2021). Figure 9 shows how these features are modelled as a separate package.

Figure 9 UML Model for related Functional Spaces

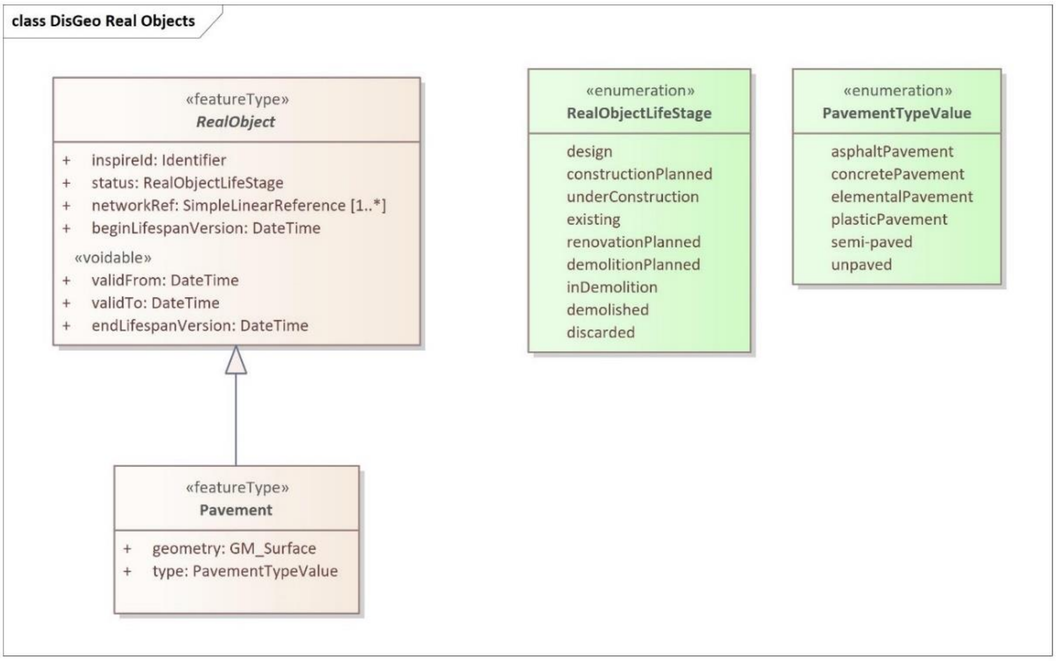


Both Traffic Functional Zones and Road Zones have a GM_Surface geometry since they are defined as area features in the SOR. Moreover, the connection between Traffic Functional Zones and the network is defined as a *hyperconnection*, and the NetworkReference concept of INSPIRE is used to fulfil this. Similar to what is described in the previous subsection about properties, NetworkReference data type can be used to link a Traffic Functional Zone to a Network Element. The connection between a Road Zone and the network, on the other hand, is not defined in the SOR; therefore, it is assumed here that the same relationship can be established for Road Zones as well. As a result, both Traffic Functional Zones and Road Zones inherit the *networkRef* attribute from FunctionalZoning, and the NetworkReference data type is chosen since it is not desired in the SOR to link these features with linear referencing.

5.1.4 Real Objects

The SOR differentiates functional objects, such as nodes and links, from real objects which include vegetation, buildings, pavement, bridges, and so on. The connection between the road network and real objects is defined such that a Link is on 1 or more real objects, and linear referencing is used to specify the exact location of pavement objects along the Link (Leijten, 2021). Figure 10 demonstrates how the Pavement object is modelled separately from the road network model, and as a subclass of Real Objects. Each Pavement object has a GM_Surface geometry and a *type* attribute to specify the type of pavement like asphalt, concrete and plastic. This way, the functional object, Link, is modelled separately from the real object, Pavement, instead of adding the type of pavement as an attribute to links. Moreover, *networkRef* attribute is used here with the SimpleLinearReference data type so that each Pavement object can be attached to the corresponding DisGeoLink on a certain location along the link.

Figure 10 UML Model for Pavement in Real Objects



5.1.5 Registration Spaces

The final connection of the road network model is with Public Spaces, which is defined under Registration Spaces in the SOR (Leijten, 2021). Like Pavement objects, Public Spaces are defined with a GM_Surface geometry (Figure 11). Both DisGeoNode and DisGeoLink elements can have a connection with Public Spaces. While DisGeoNode objects will use the NetworkReference data type for this connection (since the reference will apply to the whole of the node), DisGeoLink objects will use the SimpleLinearReference data type for this purpose. Since SimpleLinearReference is a subclass of the NetworkReference data type (see Figure 8), the more general NetworkReference is used in the UML model so that the appropriate data type can be chosen considering the type of Network Element.

Figure 11 UML Model for Public Spaces

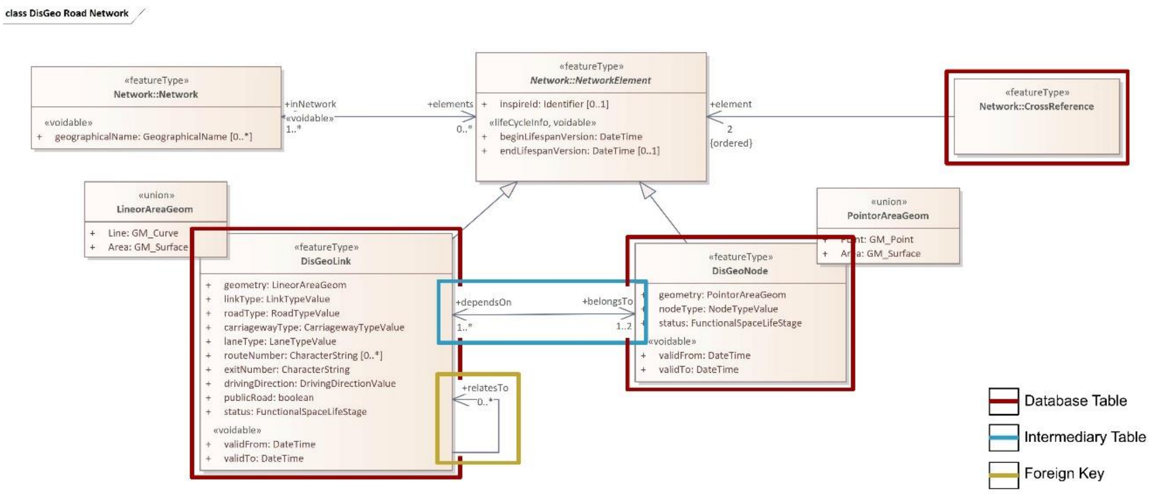
5.2 Creating the Model in a Database

PostgreSQL is used to create a database for the road network model, and certain rules are followed to map the UML diagrams into the database:

1. Elements that are defined as *featureType*s are translated into database tables. However, if there is an inheritance relationship between two tables, a database table is created only for the subclass while attributes inherited from the superclass are also added to this table.
2. 1-1 or 1-many relationships are formed with foreign keys while many-to-many relationships are formed by adding intermediary tables.
3. All enumerations and extra data types (such as SimplePointReference and SimpleLinearReference) are added as new types.

Following these rules, 10 tables are created in the database to represent all the elements in the road network model. Firstly, Figure 12 shows which elements are translated into the database for the Road Network package. It is important to note that the Network element is not translated into the database since we create the network model only for roads. Therefore, there is no need to add a Network table to test in this model. For the main elements of the model, two tables called "disgeonode" and "disgeolink" are created, and the attributes inherited from the NetworkElement object are added as fields to these tables. The geometry option provided with the <<union>> type in the UML diagram is translated as a geometry constraint for both tables so that only the allowed types can be added. To represent the *relatesTo* relationship of "disgeolink" to itself, an extra relatesTo field is added to this table which is a foreign key to the link_id field of "disgeolink". For the association between "disgeonode" and "disgeolink" tables, an intermediary table called "dependson" is created with node_id and link_id fields that are foreign keys to "disgeonode.node_id" and "disgeolink.link_id" fields. Finally, to translate the CrossReference element in the UML diagram, two tables are created: "node_cross_ref" and "link_cross_ref" so that the area objects can be linked with the linear objects that represent the same real-world object. The former table has the area_id and node_id fields, both of which references "disgeonode.node_id" since both linear and area objects are stored in the "disgeonode" table. Similarly, the latter table has the area_id and link_id fields, both of which references "disgeolink.link_id".

Figure 12 Database Elements from the Road Network Package



Secondly, Traffic Functional Zones and Road Zones are added as two tables called "trafficfunczone" and "roadzone". In the UML model, their connection to the road network model is provided with the NetworkReference data type. This relationship is translated into the database with foreign keys to the "disgeonode" and "disgeolink" tables that reference the trafficfunczone_id and roadzone_id fields from the

corresponding tables. Moreover, the attributes inherited from FunctionalSpace and FunctionalZoning elements (see Figure 9) are added as well into the "trafficfunczone" and "roadzone" tables.
Thirdly, Pavement element in the UML model is added with a "pavement" table, and the SimpleLinearReference data type is created since the connection to a Network Element is provided with linear referencing. However, it is not possible to define foreign keys in data type declarations in PostgreSQL; therefore, node_id and link_id fields are added to the "pavement" table which reference "disgeonode.node_id" and "disgeolink.link_id" fields respectively to be able to show the connection between a Pavement object and a Network Element.
Finally, PublicSpace element is mapped to a new table called "publicspace" in the database. Since Nodes can be connected to 0 or 1 public spaces, this relationship is provided with a foreign key to the "disgeonode" table. However, Links can be connected to 0 or more public spaces; therefore, an intermediary table called "linkpublicspace" is created to keep track of the relationship between links and public spaces.

5.3 Testing the Model

Since the time was not enough to test the entire model, only the main part of the model is tested which consists of the DiSGeoNode and DisGeoLink elements as well as the CrossReference element used from the INSPIRE model. Moreover, the city center of Delft is chosen as the test area.
Firstly, a mapping is performed from TOP10NL for the fields that correspond to the fields in the database. id, geom, tijdstipregistratie, and objectbegintijd fields of TOP10NL are mapped to the node_id/link_id, geom, beginlifespanversion, and validfrom fields respectively in the "disgeonode" and "disgeolink" tables. It is important to note that the area representations are stored separately from the linear representations in TOP10NL; however, since they are desired to be stored together in the SOR, both the linear and area objects are mapped together to the tables. Moreover, remaining empty fields are filled by checking the real-world objects. A small subset from the "disgenode" and "disgeolink" tables can be found in Figures 13 and 14 respectively.

Figure 13 disgeonode Table

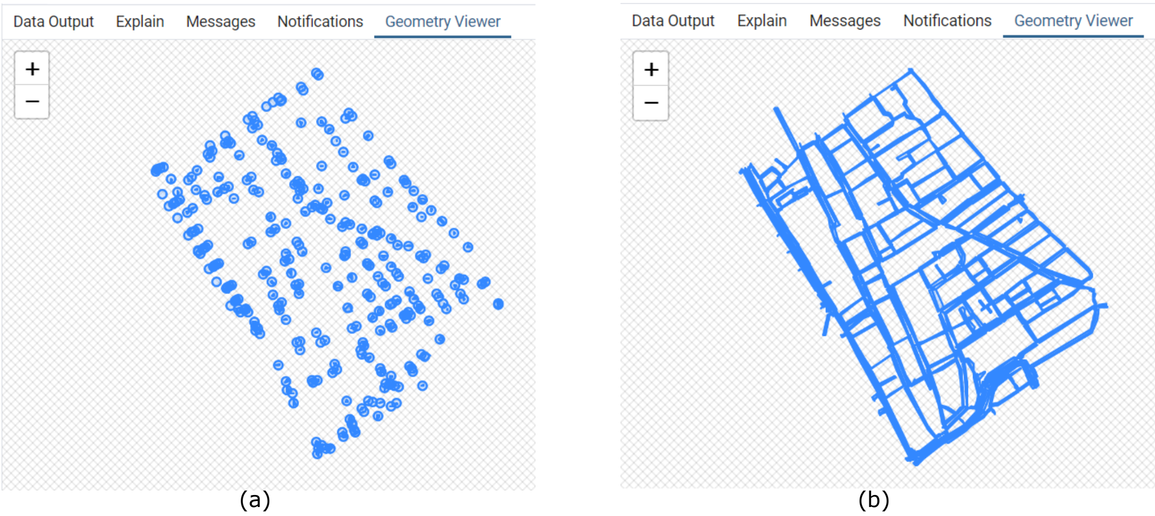
	node_id [PK] numeric	beginlifespanversion date	endlifespanversion date	geom geometry	nodetype nodetypevalue	status functionalspacelifestage	validfrom date	validto date	rzone_id numeric	pspace_id numeric	trafzone_id numeric
1	6182	2021-04-01	[null]	010100002040710...	tienode	existing	2008-11-24	[null]	[null]	[null]	[null]
2	6303	2012-08-31	[null]	010600002040710...	[null]	[null]	2008-11-24	[null]	[null]	[null]	[null]
3	47700	2018-09-01	[null]	010100002040710...	tienode	existing	2015-08-30	[null]	[null]	[null]	[null]
4	47703	2019-04-01	[null]	010100002040710...	tienode	existing	2008-11-24	[null]	[null]	[null]	[null]
5	47704	2019-11-01	[null]	010100002040710...	tienode	existing	2015-08-30	[null]	[null]	[null]	[null]
6	47706	2018-09-01	[null]	010100002040710...	tienode	existing	2015-08-30	[null]	[null]	[null]	[null]
7	50540	2019-09-01	[null]	010100002040710...	tienode	existing	2019-09-01	[null]	[null]	[null]	[null]
8	50542	2019-09-01	[null]	010100002040710...	tienode	existing	2019-09-01	[null]	[null]	[null]	[null]
9	56529	2019-02-01	[null]	010100002040710...	tienode	existing	2014-01-24	[null]	[null]	[null]	[null]
10	77590	2018-09-01	[null]	010100002040710...	tienode	existing	2015-08-30	[null]	[null]	[null]	[null]
11	178818	2020-02-01	[null]	010100002040710...	tienode	existing	2019-09-01	[null]	[null]	[null]	[null]
12	251453	2019-04-01	[null]	010100002040710...	tienode	existing	2008-11-24	[null]	[null]	[null]	[null]
13	253570	2019-02-01	[null]	010100002040710...	tienode	existing	2008-11-24	[null]	[null]	[null]	[null]
14	261649	2019-09-01	[null]	010600002040710...	[null]	[null]	2008-11-24	[null]	[null]	[null]	[null]
15	292129	2018-09-01	[null]	010100002040710...	tienode	existing	2008-11-24	[null]	[null]	[null]	[null]

Figure 14 disgeolink Table

	link_id [PK] numeric	beginlifespanversion date	endlifespanversion date	geom geometry	linktype linktypevalue	roadtype roadtypevalue	carriagewaytype carriagewaytypevalue	lanetype lanetypevalue	exitnumber integer	drivingdir drivingdirectionvalue	publicroad boolean	status functionalspacelifestage	validfrom date	vali date
370	1108052	2018-06-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2008-11-24	[null
371	1113336	2019-09-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2015-08-30	[null
372	1113337	2019-09-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2015-08-30	[null
373	1113339	2019-09-01	[null]	01050000204071...	carriageway	[null]	hoofdrijbaan	[null]	[null]	oneway	true	existing	2015-08-30	[null
374	1114953	2018-06-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2008-11-24	[null
375	1125273	2019-09-01	[null]	01050000204071...	carriageway	[null]	hoofdrijbaan	[null]	[null]	oneway	true	existing	2015-08-30	[null
376	1125274	2019-09-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2019-09-01	[null
377	1125279	2019-09-01	[null]	01050000204071...	carriageway	[null]	hoofdrijbaan	[null]	[null]	oneway	true	unusable	2015-08-30	[null
378	1132150	2020-04-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2008-11-24	[null
379	1134412	2019-09-01	[null]	01050000204071...	carriageway	[null]	hoofdrijbaan	[null]	[null]	oneway	true	existing	2015-08-30	[null
380	1134413	2019-11-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2015-08-30	[null
381	1134417	2019-09-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2015-08-30	[null
382	1136321	2018-06-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2008-11-24	[null
383	1140403	2018-06-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2008-11-24	[null
384	1147118	2018-06-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2008-11-24	[null
385	1147642	2018-06-01	[null]	01050000204071...	road	erftoegangsweg	[null]	[null]	[null]	oneway	true	existing	2008-11-24	[null

The Geometry Viewer can be used to visualize the objects in "disgeonode" and "disgeolink" tables (Figure 15).

Figure 15 disgeonode Objects (a) and disgeolink Objects (b)



It is seen that the linear objects and area objects overlap since they are stored in the same tables (Figure 16). This also brings another problem about the storage of attributes. Since the only difference between the linear and area objects is their geometry, it needs to be decided whether to store all the other attributes twice in both representations, or only once in one representation. Storing the same information twice can be meaningless, and it may slow down the queries; therefore, all attributes are stored with the linear objects while the area objects have values only in the id and geom fields and the other fields are left with NULL values. At this point, the "node_cross_ref" and "link_cross_ref" tables are populated with the IDs of corresponding linear and area objects so that this table can be used to connect different representations, and their attributes. For instance, if we have an areal link object with the link_id = 1375963, we can use the query in Figure 17 to select all attributes of the same real-world object from its linear representation.

Figure 16 Overlap of Points and Polygons (a) and Overlap of Lines and Polygons (b)

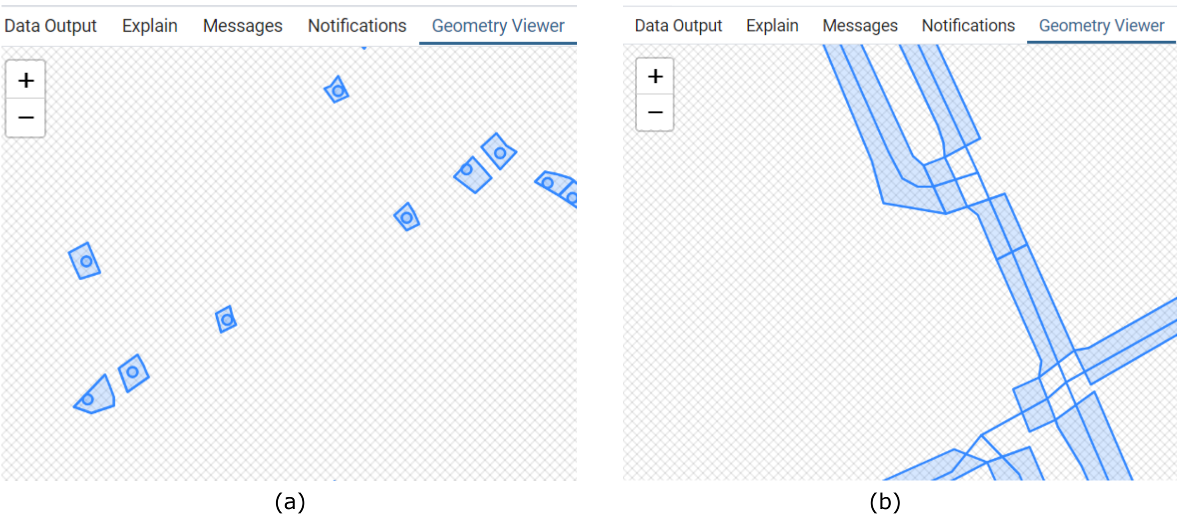


Figure 17 Use of link_cross_ref Table

Query Editor   Query History

```
SELECT *
FROM disgeolink, link_cross_ref
WHERE disgeolink.link_id = link_cross_ref.link_id and link_cross_ref.link_area_id = 1375963;
```

Data Output   Explain   Messages   Notifications

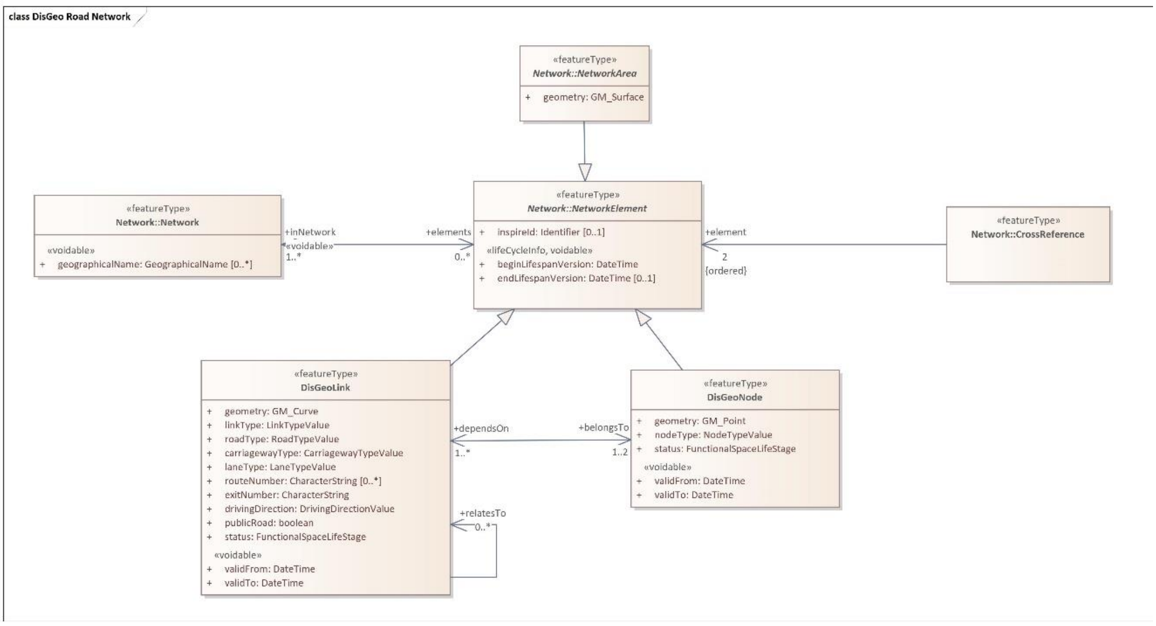
	link_id numeric	beginlifespanversion date	endlifespanversion date	geom geometry	linktype linktypevalue	roadtype roadtypevalue	carriagewaytype carriagewaytypevalue	lanetype lanetypevalue
1	1042900	2018-06-01	[null]	010500002040710...	road	erftoegangsweg	[null]	[null]

It can be concluded that the transition from the UML model to a database is not straightforward since certain features/relationships cannot be directly translated into the database. For instance, the geometry option of DisGeoNode and DisGeoLink elements is not very suitable in a database since this results in overlapping geometries within a table. While the relationship between different geometries can be ensured with the CrossReference element of INSPIRE, two different tables need to be created (node_cross_ref and link_cross_ref) since area objects are stored separately for nodes and links. Considering these problems, next section provides an alternative to the storage of area objects, which is expected to solve the aforementioned problems and to make the queries easier to understand and implement.

5.4 Alternative Model

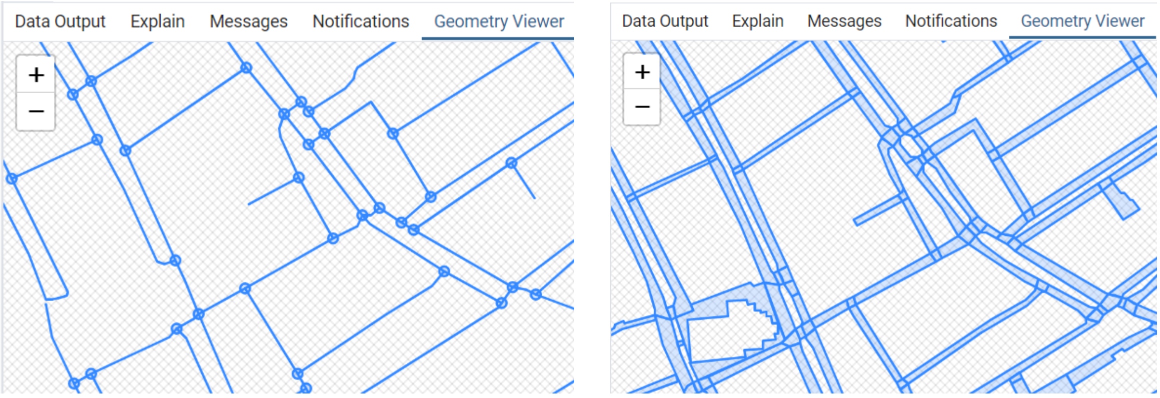
An alternative model is created in Enterprise Architect under the *Model-Area* root node. While the DisGeo Road Network package slightly changes in this model, all the other packages stay the same. To solve the problems with the area objects, this model uses the NetworkArea element from the INSPIRE model to keep all area objects separate from the linear objects. Figure 18 shows how this changes the UML diagram of the road network package. Instead of providing an option for the geometry attribute to choose a linear or area geometry, all area objects are stored in the NetworkArea element with a GM_Surface geometry. It is seen that the NetworkArea element does not have all the attributes of DisGeoNodes and DisGeoLinks since these attributes can be retrieved using the CrossReference concept.

Figure 18 Alternative Model in UML



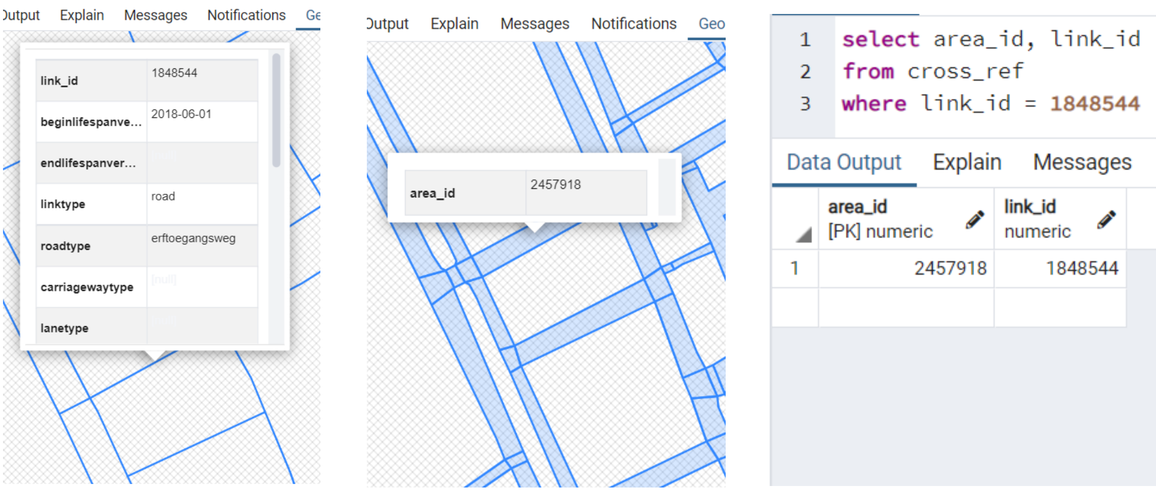
To test this alternative model, an extra “disgeoarea” table is created where all the area objects are stored with unique IDs. Moreover, since the area objects for both nodes and links are stored in one table, one single “cross_ref” table is created instead of having two separate “node_cross_ref” and “link_cross_ref” tables. Figure 19 shows how linear and area objects are stored separately while Figure 20 illustrates how the “cross_ref” table can be queried. This way, it is easier to form queries and select objects since the geometry type does not vary within a single table. While the use of the CrossReference element does not change in this alternative, it makes more sense to use the “cross_ref” table in this model to form a connection between linear and area objects because the “disgeoarea” table only has 2 fields: area_id and geom. To get information about all the other fields, a query like the one in Figure 17 need to be made.

Figure 19 *geom* Field Selected From “disgeonode” and “disgeolink” Tables (a) and From “disgeoarea” Table (b)



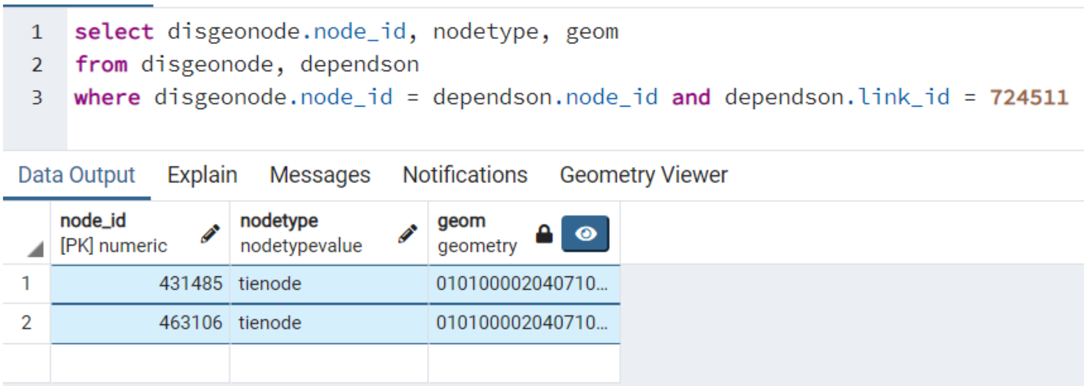
(a) (b)

Figure 20 Example query from “cross_ref” table with test data

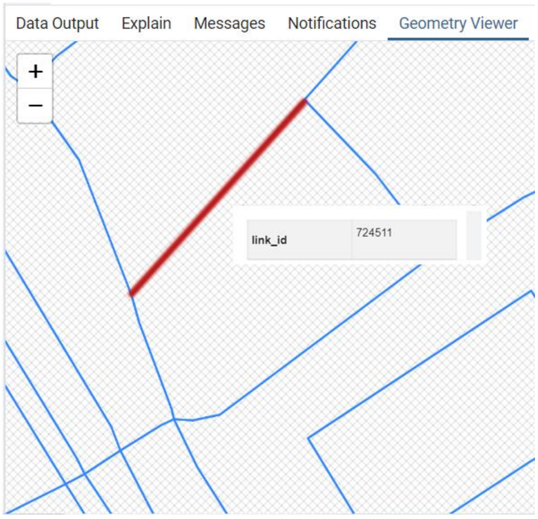


After testing the alternative model and the CrossReference concept, “dependson” table was also populated with test data to see if the connection between nodes and links can be queried. This table consists of two fields: node_id and link_id, which reference the corresponding fields from “disgeonode” and “disgeolink” tables. An example query is made in Figure 21 to find the start and end nodes of a link, and the resulting IDs and the node types are shown. It can be concluded that the “dependson” table can be used for many purposes like finding the nodes of a link, all links that a node is connected to, and so on. With these tests, we can say that the “cross_ref” and “dependson” tables work nicely for various purposes, which is highly important since these create the main part of the road network model.

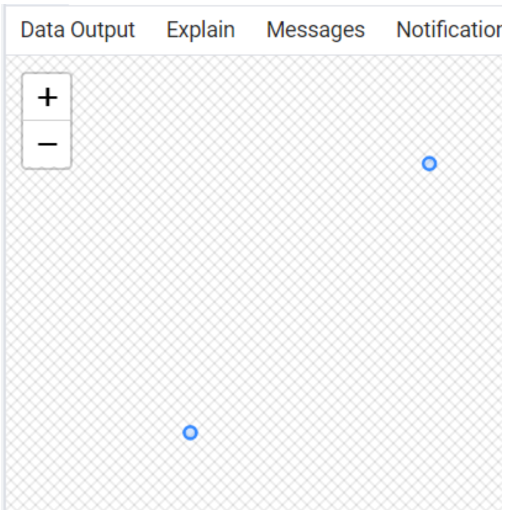
Figure 21 Example query with "dependson" table (a), selected link (b), geometries of the resulting nodes (c)



(a)



(b)



(c)

Chapter 6

Conclusion

The aim of this report was to investigate how to create a model for road network data by considering the requirements identified in the Coherent Object Registration (SOR) conceptual model, which aims to harmonize existing basic registers in the Netherlands. Existing datasets of National Roads Database (NWB) – Wegen and the Basic Register of Topography (BRT) TOP10NL were also considered, and the INSPIRE Road Transport Network Model was analysed to see if the SOR requirements can be met using this network model. Then, the road network model was created in UML by using Enterprise Architect, and the final step consisted of translating the UML model into a database to check certain functionalities with test data.

When the NWB – Wegen and TOP10NL datasets are analysed, it is seen that both models support a linear representation with nodes and road sections while only TOP10NL provides an area representation. However, this does not allow specifying different Levels of Detail (LoD) since there is only one area object that does not differentiate between various traffic areas. In addition, both NWB – Wegen and TOP10NL does not use linear referencing, but they add properties as attributes which causes the storage of unnecessary information for certain objects. When the network requirements in the SOR are analysed, it is seen that the only network elements are Nodes and Links which must have a linear representation while an area representation is optional. These network elements can be provided in road level, carriageway level and lane level so that a network can be modelled in different levels of detail. Moreover, additional areas that are not part of the network, but are important to have a connection with the network are defined such as Traffic functional zones, Road zones and Pavement objects. Similar to NWB and TOP10NL, properties are added to Links as attributes; however, linear referencing is also desired to be used so that location dependent attributes and elements can be added. Comparison of these requirements with the INSPIRE Road Transport Network model shows that some elements have a translation to INSPIRE's model while the others need to be modelled as new elements. For instance, INSPIRE does not support a level of detail in network elements, or it does not include a mechanism to define Traffic functional zones and Road zones outside the network. Moreover, although some elements have a translation to INSPIRE's model, how INSPIRE defines certain objects does not fully meet the SOR requirements; therefore, these elements must be reviewed. In general, the INSPIRE Road Transport Network Model is more comprehensive and more complex than the SOR requirements; therefore, more research is needed about how to simplify the INSPIRE's model while also making additions so that all SOR requirements can be met.

In the UML modelling part, certain elements are used from the INSPIRE's model which are Network, NetworkElement, CrossReference, NetworkProperty, and the NetworkReference data type; however, all other requirements are met by creating new elements since INSPIRE's model does not meet them. Then, the road network model is translated into a database by creating tables for each *featureType* and for many-to-many relationships, and the tables are populated with test data from TOP10NL and by doing visual checks, where the test area is the city center of Delft. To formulate area representation and its connection to the network, two options are created. The first one meets the SOR requirements, where the geometry of the two network elements (DisGeoNode and DisGeoLink) can be chosen as either linear or area, and these are stored together in the corresponding tables. However, this resulted in overlapping geometries and extra rows with many NULL values; therefore, the second option is created by using the NetworkArea element from the INSPIRE's model. In this case, all area geometries are stored as NetworkArea features and their connection to the corresponding linear elements are provided with the CrossReference element, which solved both problems of the first option.

Due to limited time, only the relationship between linear and area objects and the one between DisGeoNodes and DisGeoLinks are tested, which did not give any errors or problems. Future work may include testing the rest of the model which includes the elements outside the network such as Traffic functional zones, Road zones, and Pavement objects. Since some of these elements need linear referencing for the connection to the network, NetworkReference and SimpleLinearReference data types of INSPIRE can be tested to see if they work with surface geometries. Moreover, the concept of Level of Detail is provided with an association of DisGeoLink to itself to show which elements represent the same real-world object in different LoDs; however, this may not be the most efficient way to show this relationship. Transportation module of CityGML can be analysed to gather information about how different LoDs can be incorporated. Finally, it may be beneficial to test the model for a larger area in the Netherlands instead of focusing on a small area like the city center of Delft since the current datasets may have errors or incompatible elements with the created road network model.

References

Besemer, J., Laarakker, P., Murre, L., Kroode, R., & Zuiderveen Borgesuis, N. (2006, November). Basic registers for geo-information. In *XXIII International FIG Congress* (pp. 5-13).

Doorontwikkeling van basisregistraties in Samenhang (Dis-Geo). Digitale Overheid. (2019, February 28). Retrieved October 21, 2021, from https://www.digitaleoverheid.nl/nieuws/doorontwikkeling-van-basisregistraties-in-samenhang-dis-geo/.

INSPIRE. (2013). (tech.). *D2.10.1 INSPIRE Data Specifications – Base Models – Generic Network Model*

INSPIRE. (2014). (tech.). *D2.8.I.7 Data Specification on Transport Networks – Technical Guidelines*.

International Organization for Standardization. (2020). *Intelligent transport systems — Geographic Data Files (GDF) GDF5.1* (ISO 20524-1).
Retrieved from https://www.iso.org/standard/68244.html

International Organization for Standardization. (2019). *Geographic information — Spatial schema* (ISO 19107). Retrieved from https://www.iso.org/standard/66175.html

Kadaster. (2020). (rep.). *Basisregistratie Topografie: Catalogus en Productspecificaties*.

Leijten , S., Krijtenburg , D. & Rietdijk , M. (2021). (rep.). *DiS Geo: Eisen aan model samenhangende objectenregistratie*. Retrieved September 15, 2021, from https://docs.geostandaarden.nl/disgeo/emso/.

RWS – CIV - DOA. (2021). (rep.). *Handleiding Nationaal Wegenbestand (NWB-Wegen)*.