MSc Geomatics for the Built Environment

A 3D data modeling approach for integrated management of below and above ground utility network features

Xander den Duijn

2018

A 3D DATA MODELING APPROACH FOR INTEGRATED MANAGEMENT OF BELOW AND ABOVE GROUND UTILITY NETWORK FEATURES

A thesis submitted to the Delft University of Technology in partial fulfillment
of the requirements for the degree of

Master of Science in Geomatics for the Built Environment

by

Xander den Duijn

January 2018

ISBN 999–99–9999–999–9

The work in this thesis was made in the:

Faculty of Architecture & the Built Environment
Delft University of Technology

Municipality of Rotterdam

Supervisors: Prof Dr. Sisi Zlatanova
Drs. Wilko Quak
Edo Roldan Sanchez (Gemeente Rotterdam)
Co-reader: Alexander Wandl

ABSTRACT

Precise and comprehensive knowledge about 3D urban space, critical infrastructures, and below ground features is required for simulation and analysis in the fields of urban and environmental planning, city administration and disaster management. In order to facilitate these applications, geo-information about functional, semantic, and topographic aspects of urban features, their mutual dependencies and relations are needed. Substantial work has been done in the modeling and representation of above ground features in the context of 3D city modelling by means of City Geography Markup Language (CityGML). However, the below ground part of the real world, of which utility networks form a big part, is often neglected in 3D city models. At the same time, several existing utility network data models exist. These are, however, commonly tailored to a specific domain and not suitable for the integrated modeling and representation of utility networks and city objects in 3D urban space

This research proposes a 3D data modeling approach for integrated management of below ground utility networks features (viz. electricity and sewer) and related above ground city objects (viz. streetlights and manhole covers). The data modeling approach is successfully examined by implementing relationships between 1) the below ground electricity network and above ground streetlights and 2) between the sewer network and the above ground manhole covers.

Having existing utility network data and city objects as input, a file in CityGML Utility Network Application Domain Extension (ADE) format is created. The manipulation of the data structure and content, according the proposed data modeling approach, is completed in Feature Manipulation Engine (FME). The output CityGML dataset allows interoperability but may become very large and objects may be arbitrarily nested leading to complex data structures. Therefore carefully optimized database schemas are required that enables efficient storage, management and data access of the CityGML data. The object-oriented CityGML data model, including the Utility Network ADE, is mapped to a relational database by means of the 3D City Database (3DCityDB). Subsequently, the CityGML data is inserted into the derived relational database. Several relevant (network) analyses are performed by querying the designed relational database. It shows the possibility to simulate what network features are affected by e.g. a utility strike by means of pgRouting and visualization in a Geographical Information System (GIS)

This research made one of the first attempts to thoroughly model existing utility network data and city objects according the CityGML Utility Network ADE. Following are further research that could optimize the proposed data modeling approach for better decision making in the field of asset management:

- Modeling multiple different utility networks and city objects
- Modeling in a higher Level of Detail (LoD)
- Detailing the CityGML Utility Network classes and use
- Better investigating on more types of analyses
- Implementing larger datasets
- Implementing datasets with a different accuracy
- Exporting a CityGML file from the relational database
- Better investigating on visualization of the data
- Investigating on how to model different types of relationships

Keywords: utility networks, electricity network, sewer network, data model, CityGML, Utility Network ADE, relationships, FME, PostgreSQL, PostGIS, relational database, 3DCityDB

ACKNOWLEDGEMENTS

I would like to thank everyone directly or indirectly involved throughout this research. All of you contributed to this research, in different ways.

A special gratitude goes out to my main supervisor from the Delft University of Technology, Dr Sisi Zlatanova. After having Sisi as a teacher in the first year of the Geomatics programme, we became colleagues in the TU.Bouw LightHouse project in which I was involved as a research assistant. This was my very first experience with utility network data, not knowing I would do my thesis research in this domain and still be very interested. With the help of Sisi, I had the chance to attend several workshops abroad that got me in contact with the right people for my thesis research. Besides the contribution to my thesis research, attending these workshops contributed to my public speaking skills and personal development. The feedback of Sisi helped me taking the next steps in my research and made me feel confident about my work. Also, Sisi encouraged me to do my thesis research at the municipality of Rotterdam which turned out to be a great choice. I believe Sisi and I got to know each other a little better on personal level too which I appreciate and made communicating easy.

I also would like to thank my other supervisor from the Delft University of Technology, Drs Wilko Quak. He has mainly been of great value helping my in the technical implementation. Discussing with Wilko has always been pleasant.

Furthermore, I would like to thank my supervisors from the municipality of Rotterdam, Edo Roldan Sanchez and Wil Kovacs, for giving access to their data, providing feedback from a different perspective, helping me getting in contact with experts and welcoming me in a comfortable working environment. Rotterdam has become my second home. Also, I would like to show my gratitude to the municipality of Rotterdam for the help in financing my trip to the Utility Networks ADE workshop in Karlsruhe.

My gratitude also goes out to Tatjana Kutzner. She helped me getting started with the processing of the data during the Utility Network ADE workshop in Munich and always extensively answered my questions.

I would like to thank Giorgio Agugiaro for his code that strongly contributed to the results of my research. On top of that, it was nice to meet Giorgio in person during the Utility Network ADE workshop in Karlsruhe. He always had a quick and clear answer to all of my questions.

Together with Sisi Zlatanova, Leon olde Scholtenhuis supervised the LightHouse project. His knowledge and experience in the utility networks domain helped me a lot. Also, I want to thank Leon for letting me join on a trip to the IWCCE conference in Seattle and giving the opportunity to present our work at the Pioneering event in Hengelo.

Finally, I would like to thank my friends, family and fellow Geomatics students for their support during my student life.

The thanked people contributed to my continuous interest in the research topic and work ethic. I am sure that I could not have done this and would not have been this proud of the work I delivered without their help.

Thank you all.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

1 INTRODUCTION

1.1 BACKGROUND

Precise and comprehensive knowledge about 3D urban space, critical infrastructures, and below ground features is required for simulation and analysis in the fields of urban and environmental planning, city administration, and disaster management. In order to facilitate these applications, geo-information about functional, semantic, and topographic aspects of urban features, their mutual dependencies and relations are needed. In recent years, substantial work has been done in the modeling and representation of above ground features in the context of 3D city and building models which has led to new approaches [Biljecki et al., 2015a]. The developed 3D city models provide a representation of cities that goes beyond visualization in terms of the application that they can support.

One well-known standard for representing semantic 3D city models is the international OGC standard CityGML [Gröger et al., 2012]. CityGML is an open data model and XML-based format for the storage, analysis, representation and exchange of semantic 3D city models. The common information model behind CityGML defines classes and relations for the most relevant topographic objects in cities and regional models with respect to their geometrical, topological, semantic and appearance properties. The CityGML information model is becoming the de-facto standard for storing and managing urban geographical information. By means of a so-called ADE the core model of CityGML can be extended systematically by application-specific attributes and object types. 3D city models have been neglecting utility networks in built environments for a long time [Hijazi et al., 2011]. With the CityGML Utility Network ADE a common data model is defined for representing different types of infrastructure networks, such as electricity, freshwater, wastewater, gas, oil, district heating and telecommunication.

To support risk analysis and planning of emergency responce actions, modeling the critical infrastructures and their mutual dependencies in 3D space is essential [Becker et al., 2011]. Existing utility network data models however lack a rich information model for multiple and different underground structures. Moreover, the mutual relations between networks as well as embedding into 3D urban space are not supported yet. In contrast, existing utility network data models are commonly tailored to a specific type of commodity, dedicated to serve as as-built documentation and thus are currently not suitable for the integrated modeling and representation of utility networks and city objects in 3D urban space [Becker et al., 2013].

1.2 SCIENTIFIC RELEVANCE

Several research papers are written and standards exist that deal with utility network within 3D city models. The below ground part of the real world, of which utility networks form a big part, is often neglected in the field of 3D city modeling. The existing data models for utility networks are focused on just below ground utility networks. In addition, they are tailored to a specific domain. Hence, it allows for modeling of below ground utility network features to a certain extent but modeling of related above ground city objects is not sufficiently supported.

1.3 PROBLEM STATEMENT AND MOTIVATION

In today's technologically advanced society the dependency of every citizen and company on having a working infrastructure is extremely high [Semm et al., 2012]. Utility networks are complex. A 3D data model for utility networks and related city objects can be used for management purposes and grasp the complexity of existing underground utility networks. The existing models used for managing utility networks however often only support representation in 2D, and are associated with several drawbacks. Therefore, interest in 3D representation of utility networks is rapidly increasing. Having the data modeled in 3D has several benefits. For example, overlapping lines representing different networks and different vertical elements can be missing in 2D but not in 3D [Du et al., 2006]. Moreover, topological analyses are greatly hampered by using a 2D representation of the utility networks which would not be the case in 3D. In addition, representation of utility networks within 3D city models support the maintenance and incident management of network infrastructure [olde Scholtenhuis et al., 2017]. Lastly, it should be mentioned that the municipality of Rotterdam registers their utility network data in 3D. Thus, neglecting the third dimension would mean information loss.

Different utility network data models were developed by different industries to provide means to represent, exchange and store utility networks [Hijazi et al., 2017]. However a comprehensive 3D standard data model, which provides a common basis for the integration of the different utility networks within a 3D city model in order to facilitate analyses, visualization and management tasks, lacks. As a result, the following problems can be identified:

- Lack of analytic capabilities due missing topological network structure
- Lack of relationships e.g. between network features, or between network features and other city objects, and their embedding into 3D urban space
- Exchanging data (interoperability) is greatly hampered due incompatible and incomplete data

As an example of incomplete data and lack of relationships; the line between the above ground feature of interest, e.g. a streetlight, and the main electricity line is often not registered. As a result, detailed network analyses are greatly hampered.

The municipality of Rotterdam has created a 3D city model in CityGML format which is publicly available. The most recent version does not just contain buildings but also other so-called themes like trees, ground level, design, building information as well as underground pipes and cables. A viewer is made for visualization of the Rotterdam 3D model (Figure 1.1). The pipes and cables are modeled as generic city objects. The concept of generic city objects and attributes allows for the storage and exchange of 3D objects which are not covered by any explicitly modelled thematic class within CityGML or which require attributes not represented in CityGML. In order to avoid problems concerning semantic interoperability, generic extensions are used if appropriate thematic classes or attributes are not provided by any other CityGML module [Gröger et al., 2012]. Generic extensions, however, do not allow for hierarchies and topological modeling.

(a) (b) (c)

Figure 1.1: Snapshots of the Rotterdam 3D model visualized in the web viewer

With the CityGML standard the municipality intents to move forward in tackling the presented problems. The municipality of Rotterdam, as manager of numerous assets in the city, could benefit from a comprehensive integrated data model that supports the following business and societal activities:

- Utility services operation and maintenance
- Emergency management and disaster response
- Construction planning and management
- Medium and long term planning for development, utilities and transport
- Information model foundations of smart cities

Moreover, there is a rising interest due to the increase of the urban and technical world.

Having a comprehensive integrated data model for below ground utility network features and city objects could be interesting for utility owners, managers, contractors, surveyors and other stakeholders. The municipality of Rotterdam is such a stakeholder that shows interest. An holistic understanding of the network is needed in many situations. In such cases, it should be possible to perform geospatial analyses in order to determine the implicit dependencies between network features, or between network features and other city objects. Following are some practical cases that motivate the use of a integrated topological data model.

- Finding answers to questions concerning reachability/connectivity e.g. is it possible to get from asset A to asset B through the same network.
- For managing the storm drainage system the municipality wants to know the amount of water that will be discharged into the system. The to be proposed data model could support topological analyses, e.g. finding the area of buildings and non permeable surfaces that are connected to specific parts of the network.
- For disaster management the municipality of Rotterdam shows interest in what buildings and city objects will be out of service after e.g. a utility strike. A data model for integrated management of utility networks and related city objects could facilitate in finding which parts of the network itself and, in particular, which city objects are affected by this failure.
- In order to warn residents about a scheduled or unscheduled maintenance operation, the municipality of Rotterdam and grid operators need to know what buildings are connected to the utility network the maintenance operation is performed on. It should be possible to query the database in which the relations between the buildings and the network are stored.
- The responsibility of a facility manager includes monitoring and maintenance of building infrastructure, such as water, gas or electricity. These tasks used to be completed using paper maps, which make integrated analysis of networks challenging [Hijazi et al., 2012]. There is a need for an integrated modeling of the utilities with other city features. For example, an easy access point can be found by navigating through the network, which allows performing the maintenance operations and minimizing disturbances.
- Risk analyses form an important part of asset management for the municipality of Rotterdam. To support risk analysis and planning the integrated modeling in 3D space is required. This will allow the joint visualizations of 3D city models and 3D utility networks, which would be very helpful to understand the locations and spatial relations of infrastructures in the context of

city objects. An holistic understanding facilitated by a joint 3D visualization would result in a better decision making process. Moreover, visualization and analysis of dependencies between city objects and network components, such as which city objects depends on which infrastructure, will be facilitated.

This research aims at a framework for storing utility network data within 3d city models that facilitates asset management at the municipality of Rotterdam.

1.4 RESEARCH QUESTION

The thesis will be guided by the following main research question:

How to efficiently model below ground utility networks and related above ground city objects in order to facilitate integrated asset management?

The objective of the research is to propose a suitable data modeling approach that will allow for integrated management of underground utility networks and related above ground city objects. In order to achieve this, the following sub-research questions are defined:

1. *What dependencies and relations between below and above ground utility network features are of importance for the municipality of Rotterdam in order to facilitate asset management?*
2. *To what extent can the current state of an existing utility network data model such as the CityGML Utility Network ADE fulfil the needs of the municipality of Rotterdam?*
3. *Is the proposed data modeling approach suitable for implementation of existing utility network data and city objects in 3D city models?*
4. *Which mapping methods are required to derive a relational database from the designed data model?*
5. *Can the designed relational database be used to perform essential (spatial) operations?*
6. *How to visualize the modeled data and (spatial) operations on the designed relational database?*

1.5 RESEARCH SCOPE

This research includes studying several existing data models for utility networks. The data model with the most potential will be used as starting point and be elaborated. The findings, in addition to the ones gathered from experts at the municipality of Rotterdam, can be used to test and, if needed, complement the data model and support in fulfilling the needs of the municipality of Rotterdam.

The research scope will be limited by only looking at the Low-Voltage electricity and conventional gravity sewer networks. These utility networks differ in many ways and therefore expect to require a different modeling approach.

Many above ground city objects exist that relate to one or more below ground utility network features. All of these objects are different, but the relation with the below ground utility network is what they have in common. Examples of above ground city objects that relate to the below ground electricity network are: streetlights, traffic lights and electrical cabinets. Above ground city objects that relate to the underground waste water network are e.g. manholes and pumping stations. For this research, only streetlights, as part of the electricity network, and manholes (covers), as part of the sewer network, will be used for testing purposes. Also, only actual physical relationships will be considered. Other types of relationships, e.g. spatial relationships like proximity or the fact they are managed by the same company, are not part of this research. Moreover, the utility networks and related above ground city objects will be presented in a rather low LoD since this research is initially not focused on how to optimally depict a real world situation,

Furthermore, other types of data, e.g. financial data, will not be considered. The same goes for data that relate to time and planning.

(a) a streetlight in Munich

(b) a manhole

Figure 1.2: Two real world examples of above ground features that relate to the below ground utility network

1.6 THESIS OUTLINE

The following chapter presents the related work (Chapter 2). Chapter 3 will provide the reader with some background information that is essential for understanding the thesis research. Then, Chapter 4 will present the conceptual design and workflow developed and used to fulfil the objectives of this research. Subsequently the implementation will be explained in Chapter 5 before testing and validation follow in Chapter 6. Finally, conclusions will be drawn and the research questions will be answered in Chapter 7 followed by points of improvement and future research.

2 RELATED WORK

This section discusses the work done and papers written by others in the field of 3D modeling, with the focus on below ground utility network features.

2.1 3D CITY MODELING

Significant research has been conducted in the field of semantic 3D city modeling. 3D city models are a virtual abstraction of the real world and have existed for several decades [Emgard and Zlatanova, 2007]. Semantic 3D city models represent city objects such as buildings, bridges, tunnels, roads and vegetation. These city objects mainly constitute the visible part of a city, e.g. that part striking ones eye immediately when looking around a city.

Cities are increasingly adopting 3D city models. These days 3D city models are not only used for pure visualization purposes but are increasingly employed in a number of domains. Local governments use 3D city models for e.g. urban planning and environmental simulations in order to assess whether it is economically beneficial to make certain decisions [Biljecki et al., 2015b; Benner et al., 2005].

One well-known standard for representing 3D city models is the international OGC standard CityGML [Gröger et al., 2012]. In [Kolbe, 2009] an overview is given about CityGML, its modeling aspects and design decisions, recent applications, and its relation to other 3D standards like Industry Foundation Classes (IFC), X3D, and Keyhole Markup Language (KML).

CityGML is an open data model and Extensible Markup Language (XML)-based format for the representation and exchange of virtual 3D city models. CityGML not only represents the shape and graphical appearance of city models but specifically addresses the object semantics and the representation of the thematic properties, taxonomies and aggregations. In CityGML data is given high flexibility to grow with respect to their spatial and semantic structuring as well as their topological correctness in different stages of data acquisition or along a city model processing chain.

2.2 3D MODELING OF BELOW GROUND UTILITY NETWORKS

Although substantial work has been done in the modeling and representation of above ground features in the context of 3D city models, below ground features are often neglected both in theory and practice. This section discusses related work done by others in the field of 3D modeling below ground utility networks.

In [Emgard and Zlatanova, 2007] a framework for integrated modeling of geographic 3D data, combining below and above ground features, into CityGML is proposed. The framework is based on the subdivision of features into:

- Earth surface features
- Above earth surface features
- Below earth surface features

Although some objects exist both above and below the surface, e.g. streetlights, a clear distinction between features above and below the surface is defendable due to the fundamental difference (Figure 2.1).

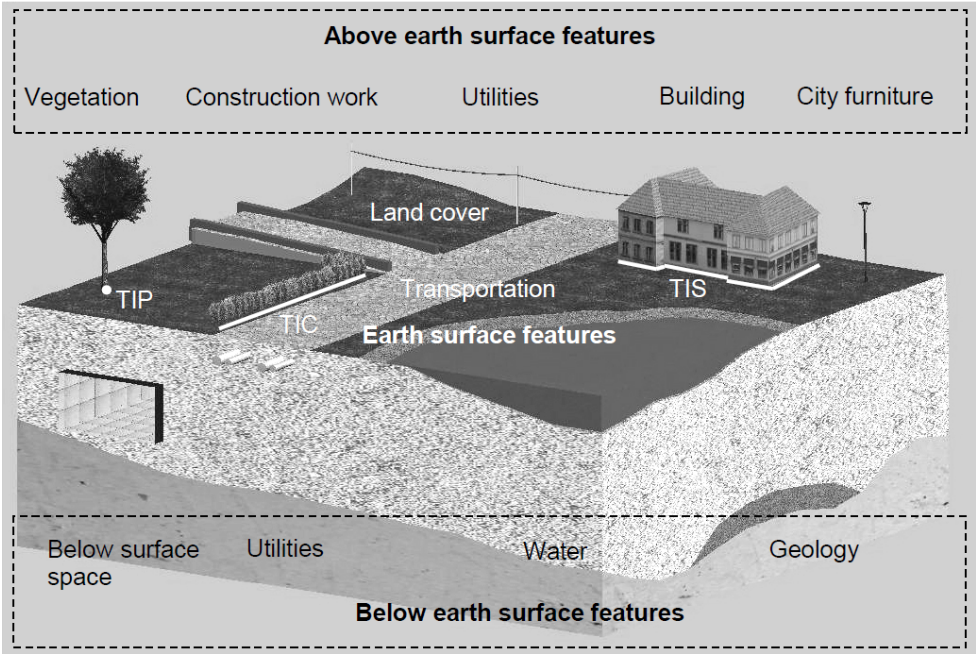


Figure 2.1: Subdivision of features. Image from [Emgard and Zlatanova, 2007]

Below ground utility networks are critical infrastructures and form a significant part of the real world. [Becker et al., 2011] presents the requirements and a novel framework for the integrated 3D modeling of such critical infrastructures within 3d city models. The proposed framework, named Utility Network ADE, extends the core CityGML model in order to integrate the utility networks into 3D city models. A key prerequisite for corresponding network models is the possibility to link utility networks in order to denote dependencies explicitly. Furthermore, the model must support geospatial analyses in order to determine the implicit dependencies based on spatial relations like proximity between network components within the same or different networks. Such geospatial analyses require the embedding of network structures into 3D space as well as the 3D topographic representation of network components in addition to a graph-based model. Moreover, the topographic representation allows for the integration of network components into their 3D urban context in order to capture the mutual influence of urban objects. Up till now, none of the existing utility network data models support all of these aspects.

The presented Utility Network ADE core module in [Becker et al., 2011] covers only the topological and topographic representation of network entities. [Becker et al., 2013] extends the Utility Network ADE, by introducing the functional and semantic classification of network objects. It shows how concepts and classes can be defined to fulfill the requirements of complex analyses and simulation, and how properties of specific networks can be defined with respect to 3D topography but also network connectivity and functional aspects.

Finally, [Hijazi et al., 2017] presents three methods complementing the CityGML Utility Network ADE by functional aspects regarding the modeling of supply areas, the characterisation of city objects and network features according to functional roles and the representation of the potential and current supply of commodities to city objects. These functional aspects allow for representing supply and disposal tasks in cases when no detailed modelling of supply networks is available. In the paper it is concluded that the CityGML Utility Network ADE represents a suitable data model for modelling heterogeneous networks in the context of 3D city models.

Numerous dependencies and relationships exist between below ground network features as well as between below ground network features and above ground city objects. As nearly all crises affect geospatial entities, large interruptions might affect neighbouring networks as well and therefore lead to a citywide crisis. Therefore, geo-information about functional, semantic, and topographic aspects of urban features, their mutual dependencies and their relationships are needed, particularly throughout crises [Becker et al., 2012a].

The project discussed in [Becker et al., 2012a] firstly identified this information and secondly integrated it into the Utility Network ADE. The integration with the CityGML standard facilitates the integration of multi-utility networks into 3D city models for joint visualization and analysis tasks. Furthermore, interoperable exchange of, and access to 3D multi-utility networks is enabled. The developed data model represents 3D topography, 3D topology, and functional properties and dependencies of the networks and their components. Hierarchical representations for both networks and components are supported as well. These characteristics allow performing geospatial analyses in order to determine the implicit dependencies between network components within the same or different infrastructures, or between network features and other city objects based on spatial relations such as proximity.

[Becker et al., 2013, 2011; Hijazi et al., 2017] has shown the many past developments of suitable data model for utility networks; the CityGML Utility Network ADE. However, no thorough study is conducted on how to integrate existing utility network data into 3D city models with the proposed framework. Furthermore, the focus was mostly just on the modeling of below ground utility networks and neglected dependencies and relationships between below and above ground objects.

So far, the discussed type of utility networks are the ones located below ground and outdoor. However, a significant part of the utility network exist indoor, withing buildings. Interior utilities are usually created and maintained using Building Information Modeling (BIM) models, while exterior utilities are stored, managed and analyzed using GIS. Following a background literature review, [Hijazi et al., 2009] and the follow up research presented in [Hijazi et al., 2011] presents an approach for the integration of interior building utility networks in city models by means of mapping the BIM/IFC data model to the CityGML Utility Network ADE, as presented in [Becker et al., 2011].

2.3 A DATABASE FOR 3D CITY MODELS

Traditionally Database Management System (DBMS) have been used for storage and management of large volumes of data and to ensure the logical consistency and integrity of data, which is also one major requirements in handling spatial data. In [Zlatanova, 2006] the importance and challenges of 3D geometries in a spatial DBMS are discussed, and points out the large step DBMS have made toward maintenance of 3D geometries. From a system dealing with management of administrative data an increasingly number of DBMS nowadays are providing spatial data types, spatial indexing and extended spatial functionality, like PostgreSQL [Zlatanova and Stoter, 2006]. Generally, only basic 3D functions are supported by the DBMS. Rather complex analyses are often not supported and have to be performed in front-end GIS applications [Zlatanova, 2006].

[Zlatanova and Stoter, 2006] is devoted to the role of DBMS in new generation GIS architecture and focuses on the manner spatial data can be managed, e.g. stored and analysed, in DBMS. Two important aspects of DBMS functionality are addressed in detail: spatial models and spatial analysis. Special attention is put on the third dimension because of the increasing demand for 3D modeling, analysis and presentations in many applications. Moreover, the operations that can be performed on DBMS as well as the benefits of having the data topologically structured over geometrically structured are described. A more detailed argumentation for the need to organize topology in DBMS is given in [Van Oosterom et al., 2002].

The support of 3D geometries allows for storage and management of 3D city models in DBMS. A DBMS for 3D city models stores its components in a hierarchically structured, multi-scale way, which allows for a stable and reliable data management and facilitates GIS modeling and analysis tasks. A database is required if 3D city models have to be continuously managed. 3D city model databases form a key element in 3D spatial data infrastructures that require support for storing, managing, maintenance, and distribution of 3D city model contents. Moreover, a database can provide an easy access method for e.g. visualization of 3D city models. Following are work and papers that particularly relate to storage and management of 3D city models in a database.

In [Stadler et al., 2009] the development of 3DCityDB is described. This database schema implements the CityGML standard with semantically rich and multi-scale urban objects facilitating complex analysis tasks, far beyond visualization. 3DCityDB can be used to store, represent, and manage virtual 3D city models on top of a standard spatial relational database. The main objective was to achieve both the efficient storage and fast processing of CityGML.

The following steps, also depicted in Figure 2.2, can be distinguished in the development of the 3DCityDB:

a. Simplification of the CityGML data model
b. Derivation of the relational database schema
c. Creation of an import and export tool

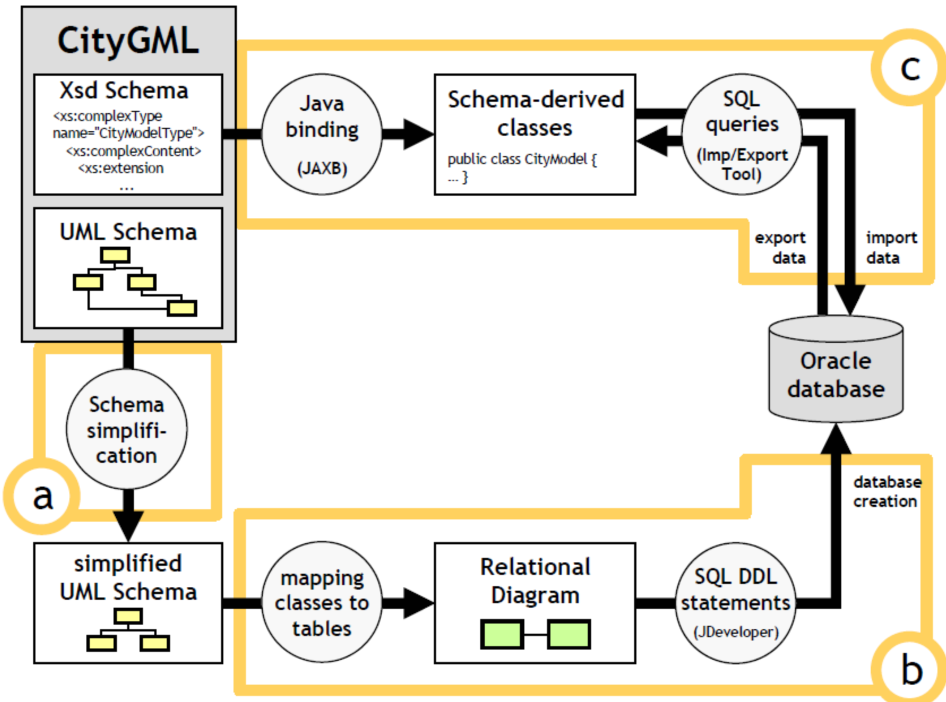


Figure 2.2: Tasks in the development of 3D city database. Image from [Stadler et al., 2009]

In item b the object-oriented CityGML data model is mapped to a relational database. The object-oriented application schemas of e.g. CityGML contain specialization and aggregation hierarchies, nested objects, and complex attributes. In order to map these structures onto a relational database, different rules have been proposed and discussed in the past. The mapping of specialization hierarchies onto database tables may follow one of three different mapping methods [Rumbaugh et al., 1991]:

- mapping all classes to a single table
- mapping non-abstract classes to their own tables
- mapping each class to a single table

Figure 2.3 illustrates the methods for the mapping of the CityGML abstract root class _BRepGeometry.

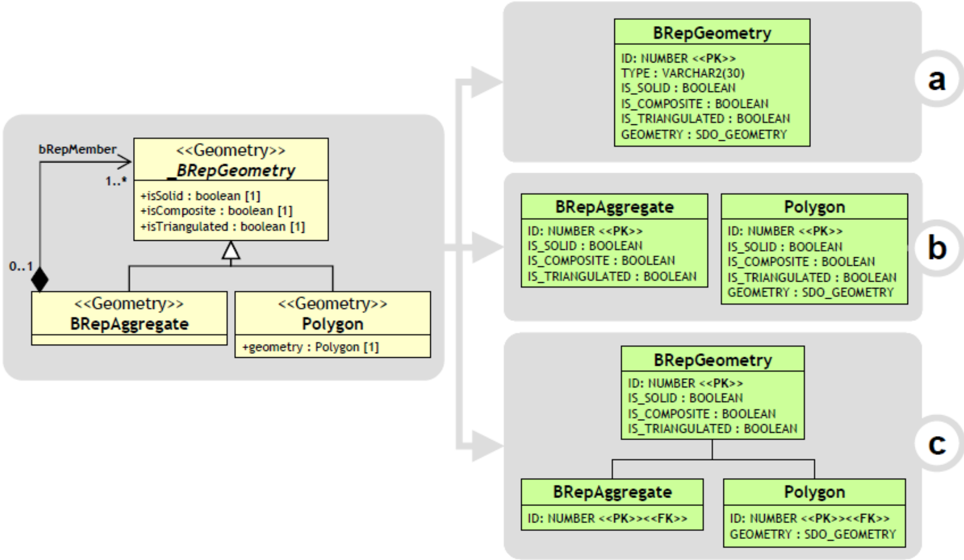


Figure 2.3: Methods for mapping a class hierarchy to database tables. Image from [Stadler et al., 2009]

In case of 3DCityDB, the desired mapping method for each CityGML class was chosen and carried out individually. The criteria for the most suitable mapping method are the expected number of tuples within the tables, the number of joins to reconstruct the objects, and the overall complexity of the most important queries (with respect to the CityGML object structure). By avoiding multiple tables the number of joins is reduced, resulting in higher database performance.

Note that the derived relational database schema only implements the core CityGML data model, without consideration of ADE's e.g. the Utility Network ADE. Hence, it is not possible to store utility network CityGML data in the core 3DCityDB. A possible solution for storing utility network data in 3DCityDB is presented with a recently published first version of a 3DCityDB extension for ADE's Energy and Utility Network ADE is published [Yao and Kolbe, 2017].

In [Emgard and Zlatanova, 2007] a 3D Information Model (3DIM) is presented which intends to integrate geographical features on the earth surface as well as above and below the earth surface. In addition, a suggestion for a possible database implementation of the extended information model is addressed. [Emgård and Zlatanova, 2008] details the mapping of the 3D information model, presented in [Emgard and Zlatanova, 2007], to a relational database. Two alternative mapping methods are discussed and evaluated that are optimized for spatial or semantic queries respectively. A distinction is made between implementation with the semantics separated from the geometry and an implementation in which the tables combine both semantics and geometry.

3 BACKGROUND

This chapter will provide the reader with some background information that is of importance for understanding the thesis research. Since the data used for experimenting in Chapter 5 is a product of several used methods to register and store subsurface utility information, possible ways of gathering utility network data are presented. Subsequently, a brief explanation of the electricity as well as the sewer network, and their functioning, will be given. This is essential for understanding the model decisions presented in the Chapter 4. Then, existing utility network data models are briefly described. The pro's and cons are summarized where after the data model with most potential will be used for further development in this research.

3.1 GATHERING UTILITY NETWORK DATA

In many western countries, most utilities are buried beneath the surface level which makes it challenging to detect utilities. Between the myriad of utility owners, there is little consistency about how information about subsurface is registered and stored in utility plans [Bitenc et al., 2008]. In addition, the way in which location data is measured and registered differs from country to country. Oldest hand drawn plans often use distances between utilities and other infrastructure to register the relative utility location. More recent utility plans are mostly stored in a 2D GIS and use Global Positioning System (GPS) coordinates obtained from land surveys. Over the years, solutions are proposed to convert existing 2D plans into 3D visualizations of the underground [Du et al., 2006; Döner et al., 2011]. Network assetmanagement systems nowadays integrate both these old and new types of utility plans in one geospatial database. As a result, the source data contains varying location information.

In the Netherlands the cadastre facilitates the dissemination of utility network location data with the Kabels en Leidingen Informatie Centrum (KLIC). The main purpose of KLIC is to prevent damage to below ground pipes and cables [Kadaster, 2017]. Most municipalities in the Netherlands are not responsible for the registration and storage of utility network data. In contrast, the municipality of Rotterdam registers and stores all below ground pipes and cables in the so-called Leidingen verzamelkaart (LVZK). The LVZK is mainly used for licensing purposes by the municipality of Rotterdam. This digital map contains all pipes and cables and gets updated every time a new one is installed or a change has been made.

Since ca. 2010 the municipality of Rotterdam registers the depth of the utility, in addition to the 2D location, using GPS. An accurate GPS measurement of the utility is only possible before burying the utilities. Though non-destructive technologies, e.g. Ground Penetrating Radar (GPR), can be used to detect and locate underground utilities after burying [Jaw and Hashim, 2013]. A standard or estimated depth is used in case a GPS measurement is conducted after burying the utility. The LVZK is, as a result, a mixture of accurately and inaccurately measured utility location data.

The municipality of Rotterdam is however not responsible for maintenance of most of the pipes and cables. Only the sewer system is managed by the municipality of Rotterdam. The electricity and gas network in Rotterdam is managed by Stedin, grid operator in Friesland, Noord-Holland, Zuid-Holland, Utrecht and Limburg

[Stedin, 2017]. Hence, utility network data of Rotterdam exists in different systems, owned and managed by different organisations. On top of that, the utility network data of Stedin is different from the utility network data in the LVZK of Rotterdam.

Furthermore, the connection line between the above ground feature and the main electricity line is often not registered. As a result, it is unknown to what electricity line e.g. a streetlight is connected to.

3.2 UTILITY NETWORKS

In this section a brief explanation of the functioning of the electricity and sewer network will be given.

3.2.1 Electricity network

A distinction between electricity cables exist, based on the voltage of the electric power it transmits:

- Extra High-Voltage cable
- High-Voltage cable or HV cable
- Medium-Voltage cable or MV cable
- Low-Voltage cable or LV cable

The different networks are connected through transformer substations. For example, a MV/LV transformer electrical substation transforms the voltage supplied by the distribution network at Medium-Voltage into voltage values suitable for the power supply of the Low-Voltage lines (Figure 3.1). A transformer substation is a good example of an above ground feature that strongly relates to below ground utility network features.

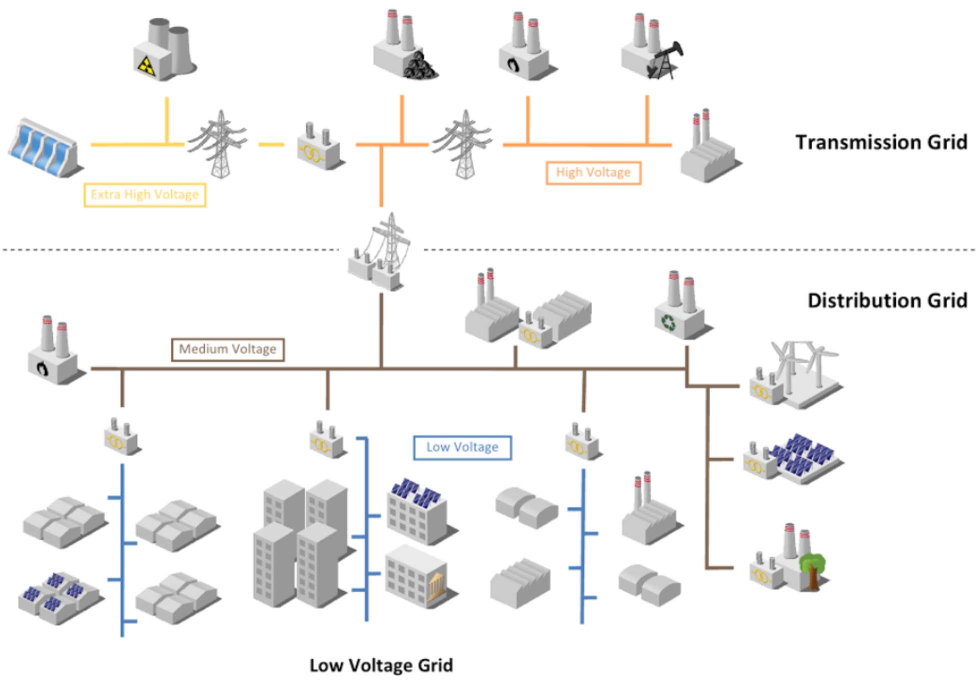


Figure 3.1: Hierarchy in electricity network. Image from [Kremers, 2013]

In addition, electricity cables exist that are only used for street lightning. The voltage of the electric power of these so-called 'OV' (In Dutch: Openbare Verlichting) cables is similar to the LV cables. However, OV cables are only carrying electric current in case street lights need to be turned on. Streetlights can be fed by electric current coming from LV or OV cables. Hence, only these types of electricity network features will be taken into account.

Not every streetlight is directly connected to the main electricity line (OV or LS). Multiple streetlights can be connected to each other. In this case only one of these streetlights is connected to the main electricity line. This streetlight is responsible for the functioning of the other streetlights (in series connection). Hence, in case this streetlight is broken, the other streetlights are not functioning either.

The type of connection of the streetlight depends whether it can be directly connected to the LV cable or to the OV cable. In case a fuse connection is used, it can be connected to the main OV cable or to another streetlight, but not to a LV cable. In contrast, the streetlight must be directly connected to a OV cable or to a LS cable in case a connection through a assisting wire is used. A streetlight with such a connection cannot be connected to another streetlight.

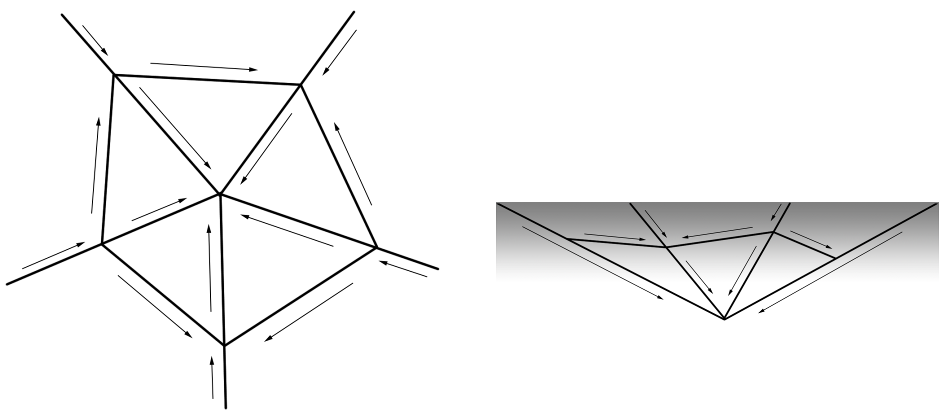
3.2.2 Sewer network

A sewer system comprises a network of pipes that collect and transport waste and storm water to a waste water treatment plant or the receiving waters. Sewer pipes exist in different shapes, sizes and material. The majority of the sewer pipes in developed worlds, however, has a round shape and is made of concrete. Other possible shapes are rectangular, egg shaped or ellipsoidal. Besides the physical differences, sewer pipes can have different functions. Based on their function the following sewer pipes (or shorter: sewers) can be distinguished:

- Standard sewer or a so-called gravity sewer that convey water using gravity
- Drainage sewer to run off the excess water to sea, reservoirs or any other suitable place
- Pressure sewer that use pumps instead of gravity to transport waste water

Pumping stations are used to move waste water to higher elevations in order to allow transport by gravity flow. The functioning of the sewer network depends on the functioning of the pumping station. A pumping station is good example of a above ground feature that strongly relates to below ground utility network features.

Figure 3.2 graphically depicts the functioning of a standard sewer network in which water flows from higher to lower elevations (illustrated by the arrows). Generally, the pumping station is located at the lowest point in the network.



(a) top view (b) side view

Figure 3.2: Graphical representation of the functioning of a standard sewer network

Another typical example of an above ground feature that relates to the below ground sewer network, is a manhole. A manhole is a vertical pipe, usually made of concrete, that connects the below ground sewer network to the surface. Manholes are mostly used for inspection of the sewer network. In addition, manholes are used in case of change in direction or material of the sewer pipe. Typically, only the top ring and manhole cover would be visible at the surface. A manhole cover is a removable plate forming the lid over the opening of a manhole. Often people confuse a manhole with a storm drain. However, a storm drain is not meant for inspection but is purely designed to drain excess rain and ground water.

3.3 EXISTING UTILITY NETWORK DATA MODELS

Different utility network data models are developed by different industries to provide means to represent, exchange and store utility networks [Hijazi et al., 2017; Kutzner and Kolbe, 2016]. In addition, utility network data models are also proposed in scientific literature [Halfawy, 2010].

Most of these utility network data models are developed to meet the needs of specific domains. Therefore they are used by different actors in different situations. This section gives a brief description of the existing utility network data models.

INSPIRE Utility Networks model

The INSPIRE Utility Networks model includes application schemas for electricity, oil & gas, sewer, telecommunication, and water networks [INSPIRE Thematic Working Group Utility and Governmental Services, 2013]. In addition to generic network information, each element can be detailed within its domain specific application schema through various attributes. The focus of the model is on describing the topography of network elements and devices and not on modeling their functionality and dependencies. The INSPIRE Utility Networks model defines a 2D topological relationship between network features. Other spatial relations such as network features to above ground city objects is not supported. Also, it does not allow for modeling hierarchies due the clear distinction between network features and the lack of a feature aggregation schema. Furthermore, the model does not support embedding of network features in 3D urban space. Hence, the utility network data model is not suitable for e.g. collision detection, impact simulation and 3D visualization [Kutzner and Kolbe, 2016].

IFC Utility model

IFC is an ISO standard which is predominantly used in BIM. It provides a 2D and 3D representation of network objects. Relationships between network objects are described using a connectivity concept, which comprises both the physical and logical connectivity. Therefore, it is possible to establish a linkage between different network types. IFC provides a rich semantic categorization of network objects based on their role in the network. The IFC data model is however developed with the intention to provide a way to model utilities inside a building. The integration of city scale network on small scale (large areas) is not supported [ISO 16739, 2016]. Modeling the spatial relationships between the utility network and city objects is very limited since it is only supported on building level. Visualizations and analyses are hence restricted to the building scale. Visualization of city objects, in 2D as well as in 3D, is not supported at all.

ESRI Utility Network

ArcGIS provides two sets of network data models to manage the logical and physical relations in a network [ESRI, 2016]. The ESRI Geometric Network model represents the basic structure for any utility network type. ArcGIS Schematics provides a mechanism to represent the relations between different features in a network. ArcGIS has a set of domain specific data models for gas, water and electricity that are customized based on the Geometric Network model. Each of the domain data models is representing a commodity-specific 2D GIS-based abstraction of the respective utility network in the real-world.

The data models are semantically rich and complex, but do only represent the 2D topography of the utility network besides the logical network connectivity information. Besides that, spatial relationships between different networks as well as between network features and city objects is not supported in the data model. The model is lacking the topographic representation of network objects in 3D urban space and also managing the logical relation between network objects is challenging. A 3D representation of the utility network is only possible for just pure 3D visualization purposes.

PipelineML

PipelineML is a Geographic Markup Language (GML)-based data interchange standard for the exchange of pipeline data focusing on the oil and gas industry [OGC, 2016]. In its current stage of development, the standard focuses on distribution components and 2D geometries only, terminal elements such as pump stations are not considered, neither is a topological representation of networks.

Due the fact PipelineML is a GML-based interchange standard, it is good for interoperability. Only the spatial relationships between network features in a network are modeled. Analytic capabilities between network features and city objects are completely missing. Moreover, PipelineML only focuses on oil and gas networks which limits the use for an integrated modeling of different utility networks.

Common Information Model (CIM)

The CIM is meant for power systems and has currently two primary uses: to facilitate the exchange of power system network data between companies; and to allow the exchange of data between applications within a company [McMorran, 2007]. The CIM comprises two standards; the IEC standard 61970-301 and the IEC 61968-11. The IEC standard 61970-301 is a semantic model that describes the components of a power system at an electrical level and the relationships between each component. The IEC 61968-11 extends this model to cover the other aspects of power system software data exchange such as asset tracking, work scheduling and customer billing.

The CIM extensively models the relationships and domain specific semantics, but completely neglects the geometry of each electrical components. Hence, spatial analyses or visualizations in 3D urban space are not supported.

Informatiemodel Kabels en Leidingen (IMKL)

IMKL, Information Model for Cables and Pipes, is a Dutch standard data model for all types of utility networks. A distinction is made between the different utility networks: electricity, gas, chemicals, drink water, waste water, telecommunication and district heating. Each network comprises network elements such as cables, pipes and junctions. All the network elements together form the network for a specific commodity. Every utility network is described by its location and topology of network elements. Moreover, the network is described by its theme, responsible organizations, commodity type, product type and other related properties [van den Brink et al., 2016].

IMKL is particularly meant for exchanging location information of utility network data and avoid damage to utilities. The data model does not consider the spatial relationships with above ground city objects. In [Zlatanova et al., 2011] an investigation is carried out what would be the best 3D standard for the Netherlands. Based on a comparison of several popular 3D GIS and CAD standards it is concluded that CityGML is the most promising. CityGML provides the best support regarding semantics, objects, attributes, georeferencing and use on the web. Several Dutch information models have been studied of which IMKL was one. IMKL has been further adapted to fit better to international standards, such as the OGC CityGML Utility Network ADE.

CityGML Utility Network ADE

CityGML is the international Open Geospatial Consortium (OGC) standard for representing and exchanging semantic 3D city models [CityGML, 2017]. CityGML has been developed by the members of the Special Interest Group (SIG) 3D since 2002. To support applications in context of urban planning and geo-design the core data model can be extended with so-called ADE's. An ADE can be used to extend the schema with new classes and attributes which are not explicitly modelled in CityGML.

The Utility Network ADE of CityGML represents a first approach to extend the abstract model of CityGML by integrating utility networks into the 3D urban space and to make their network topology and topography explicit. It aims to provide a common basis for the integration of the diverse models in order to facilitate joint analyses and visualization tasks. Furthermore, the Utility Network ADE allows for hierarchical modelling of networks and subnetworks as well as of components and subcomponents. Moreover, the application schema is designed for modeling dependencies and relationships between network features and city objects as well as between network features of different types of networks [Becker et al., 2012b].

The first draft version of the Utility Network ADE was presented in 2010 and limited to only the NetworkCore model. The last major ADE update was in 2012 by extending the data model. Besides the Core model, the Utilty Network data model is currently defined by the following models:

- NetworkComponents Model
- NetworkProperties Model
- FeatureMaterial Model
- HollowSpace Model

All CityGML Utility Network ADE Unified Modeling Language (UML) class diagrams are in Appendix A.

3.4 CONCLUSION; THE MOST PROMISING UTILITY NETWORK DATA MODEL

In Section 3.3 existing data models for utility networks are briefly described. In Table 3.1 a comparison is made between the different utility network data models. The prerequisites of the data model are put in the table as column names.

Data model	2D/3D	Scale	Topological relationships		Utility type
			network features	network features and city objects	
INSPIRE Utility Networks	2D	Urban	Yes	No	Any
IFC Utility model	2D+3D	Building	Yes	Yes*	Any
ESRI Utility Network	2D**	Urban	Yes	No	Any
PipelineML	2D	Urban	Yes	No	Oil and gas
CIM	-	Urban	Yes	No	Electricity
IMKL	2D+3D	Urban	Yes	No	Any
CityGML Utility Network ADE	2D+3D	Urban	Yes	Yes	Any***

Table 3.1: A comparison between the different utility network data models

*Only inside a building
**Only in 3D for just pure visualization purposes
***A single model is used for any type of utility network

The possibility to topologically relate below ground utility network features with above ground city objects is a key requirement. The comparison shows that only the CityGML Utility Network data model is capable of relating utility network features as well as utility network features and above ground city objects. Moreover, the application schema is designed for modeling dependencies and relationships between network features of different types of networks. With this comparison it is concluded that, for this particular research, the CityGML Utility Network ADE is the most promising data model. To summarize, characteristics of the CityGML Utility Network ADE are:

a. Simultaneous representation of heterogeneous utility networks, e.g. the ADE is not restricted to specific network types
b. Topographic 3D and topological representation of infrastructure networks (dual representation)
c. Hierarchical modeling on the feature and network level
d. Representation of functional aspects in the form of supply areas, the characterisation of city objects and network features according to functional roles and the representation of the potential and current supply of commodities to city objects
e. Linking utility networks with 3D city models; this is not supported by other existing standards
f. Modeling multi-utility scenarios; this is not covered by other existing utility modeling standards
g. The ADE provides a common data model which can serve as information hub for various stakeholders and various use cases. Data conforming to the ADE can be shared and integrated seamlessly with other utilities, without loss of information or functionality.
h. The ADE can not only be applied in urban areas, but also in rural and suburban areas to represent the transmission of resources from generation plants, wells and reservoirs located outside urban areas.

The items e and f in particular are most relevant for the research.

A possible drawback of the Utility Network ADE is the lack of documentation and the fact it is continuously under development. Moreover, no one ever thoroughly mapped existing utility network data and city objects to the CityGML Utility Network ADE which makes using the data model challenging from the start.

4 CONCEPTUAL DESIGN

This chapter provides an outline of the steps conducted in the thesis research from a conceptual point-of-view. Figure 4.1 gives a schematic overview of this process.

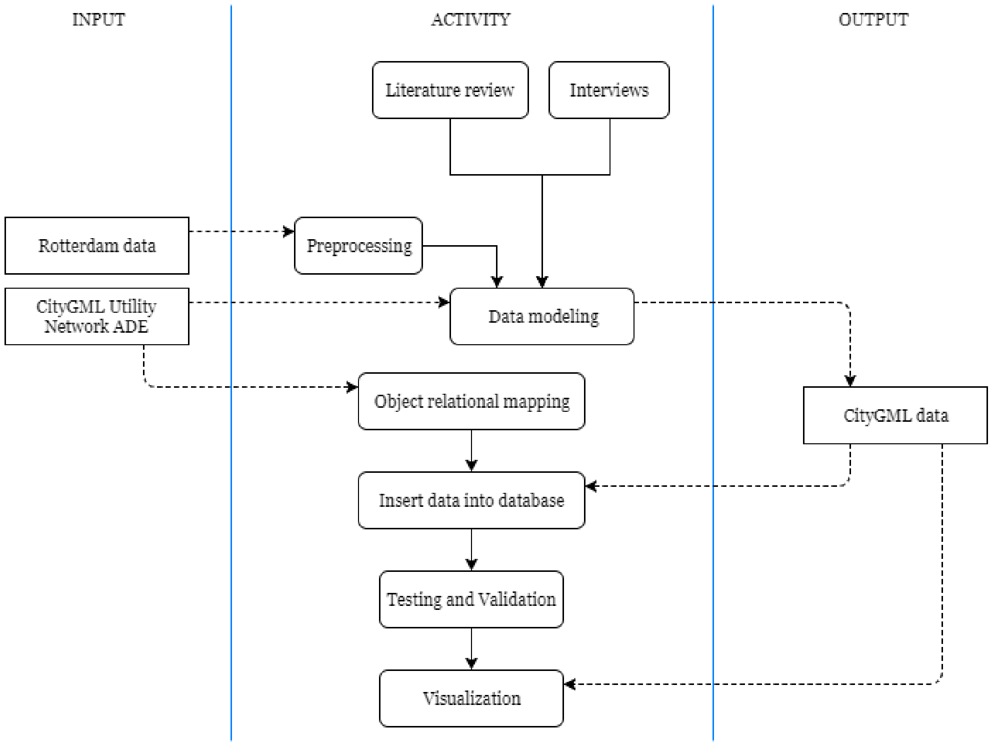


Figure 4.1: Workflow

Based on the comparison presented in Section 3.4, the CityGML Utility Network ADE turns out to be the most promising data model based on a comparison between the examined existing data models. The prerequisites of the data model are acquired from interviews with experts at the municipality of Rotterdam and the investigation of existing data models. Then, after some data pre-processing which will be discussed in detail in Section 5.5, both the electricity and sewer network data will be mapped to the CityGML Utility Network ADE. The focus will be on how to connect below and above ground features of interest. Subsequently, the data model will be mapped onto a relational database model. Then, the modeled data will be inserted into the database where after the database will be queried. By querying the database it will be validated whether the mapping is done correctly or not on the one hand. On the other hand the developed data model can be tested by performing some interesting geospatial analyses which can subsequently be visualized.

4.1 UTILITY NETWORK DATA MAPPING

This section describes how the utility network data is mapped to the CityGML Utility Network ADE. First the mapping of the electricity network will be discussed. Then, the mapping of the sewer network will be discussed. Focus is put on relating below ground utility network features and above ground city objects.

In order to understand the actions taken and decisions made, it is important to have a general understanding of the current CityGML Utility Network data model. Figure 4.2 desribes the CityGML Utility Network ADE core module at the abstract level of graphic UML. Except for the NetworkLink, all classes are used for the mapping of both the electricity and the sewer network data. The core model can be roughly divided into two parts, a topographical part and a topological graph part. The Network and its AbstractNetworkFeature instances are used for the topographic representation of the utility networks. The FeatureGraph, as member of the NetworkGraph, represents a separate graph structure for each utility element reflecting the functional, structural as well as the topological aspects of each element.

4.1.1 Electricity network data mapping

This section describes in more detail how the electricity network is mapped to the CityGML Utility Network ADE. Figure 4.3 is a UML class diagram with the classes used for the topographical representation of the electricity network. Some detail is added through e.g. the AbstractFeatureMaterial and AbstractCommodity classes. However, only the classes belonging to the core module are depicted for readability purposes. After discussing the topographical part, the graphical representation will be discussed.

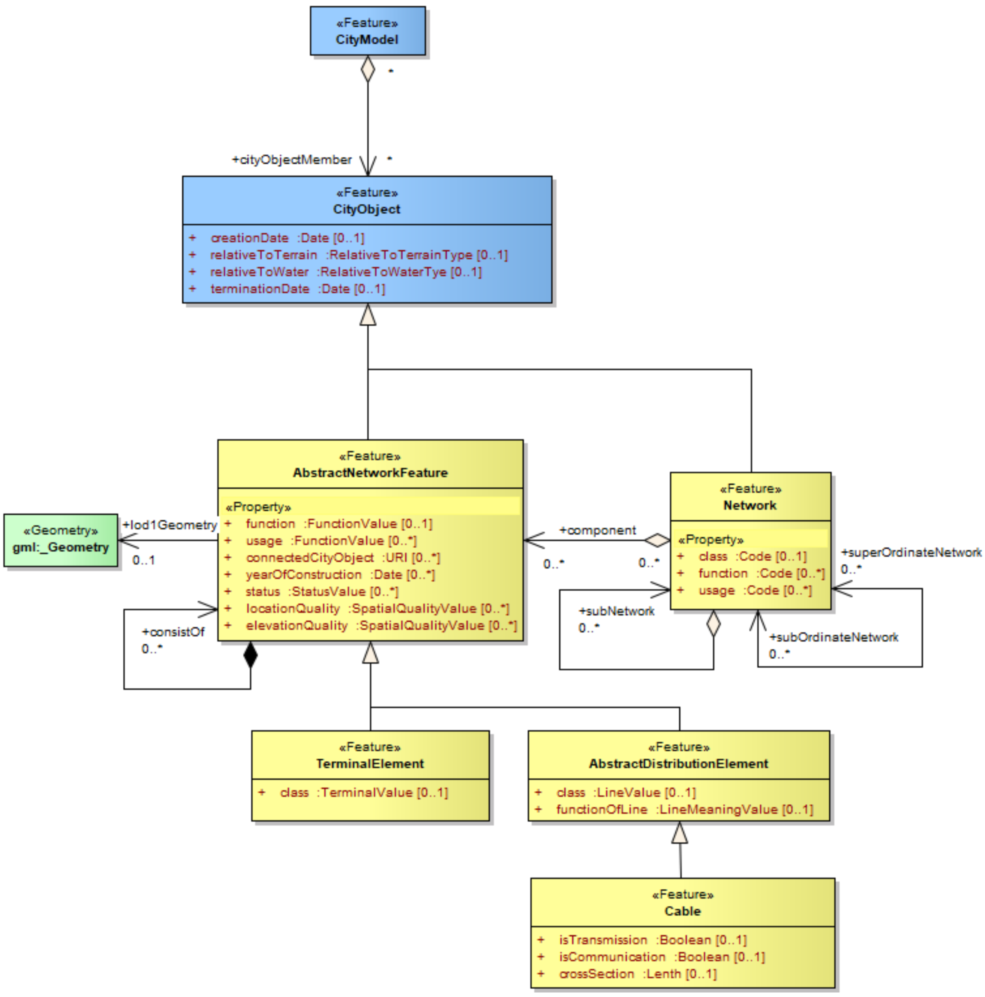


Figure 4.3: A UML diagram with all classes used for topographical representation of the electricity network

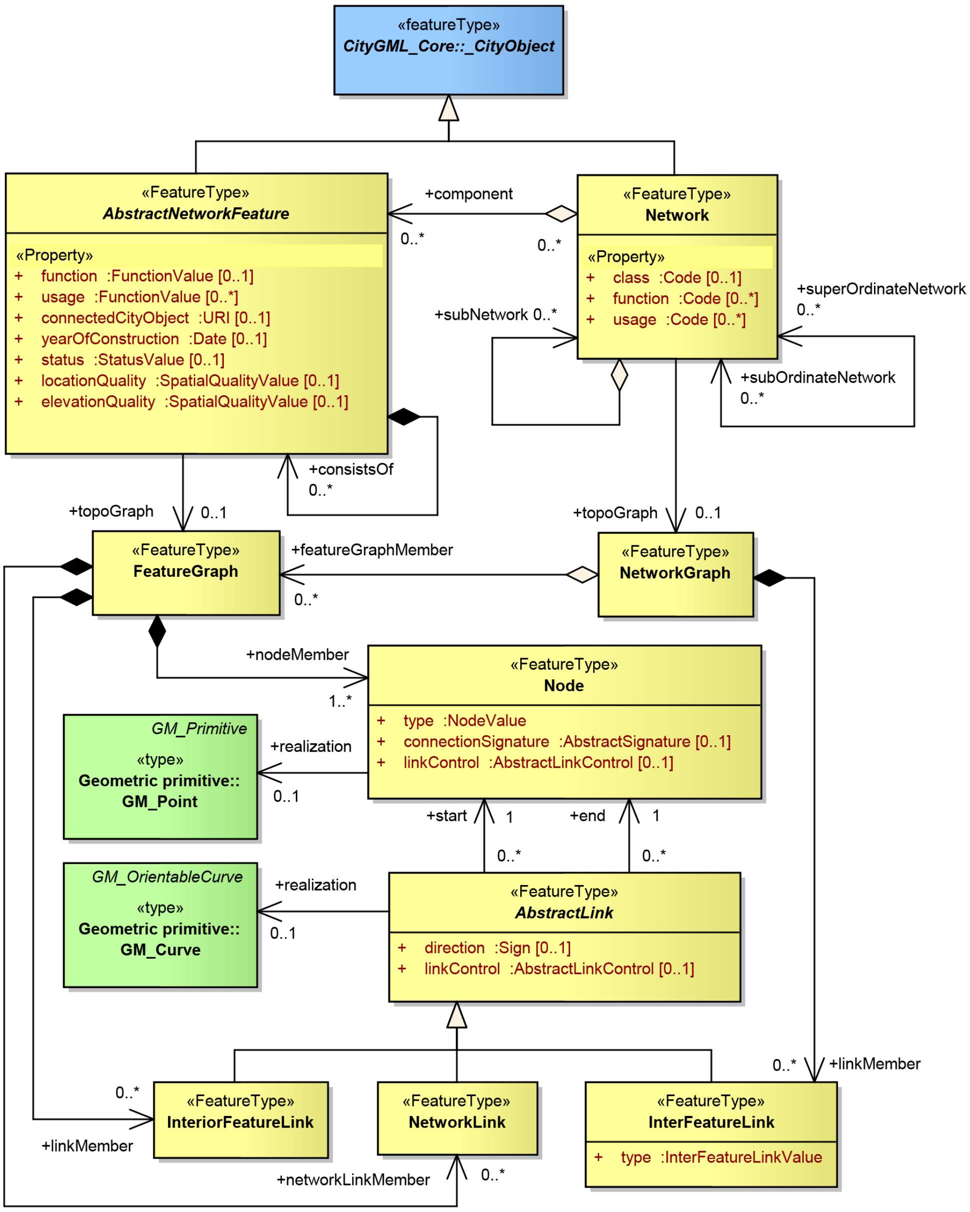


Figure 4.2: CityGML Utility Network - Core

Since electrical current is transported through cables, the Cable class is used to topographically represent all electricity cables. The Cable class inherits the properties of both the AbstractDistributionElement as well as of the AbstractNetworkFeature superclass. The physical characteristics of the electricity cables are similar, and electrical current can seamlessly be transported from a Low-Voltage cable to an OV-cable, or vice versa (see Section 3.2.1). Because of this reason, all electricity cables are part of the same Network. In order to still have a distinction between the different cables, the type is specified in the feature class description.

The class attribute of the AbstractDistributionElement is used to have a distinction between the main electricity line and the connection line to the streetlight. The codelists in the Network Components module provide some possible values for the class attribute (see Appendix A). The 'supplyLine' value is used for the connection line and the 'mainLine' value is used for the main electricity line. Furthermore, the status attribute the AbstractNetworkFeature is used to state whether a electricity cable is in use or not. Depending on how detailed the information is that comes with the dataset is, more attributes can be added.

The TerminalElement class is a-kind-of AbstractNetworkFeature. A TerminalElement represents the connection point of the below ground feature, where the cable is connected to the above ground feature. This could be the point where the electricity cable and the below ground part of the streetlight meet. A streetlight is a typical example of a city furniture object [Gröger et al., 2012], thus the CityFurniture class is used (Figure 4.4).

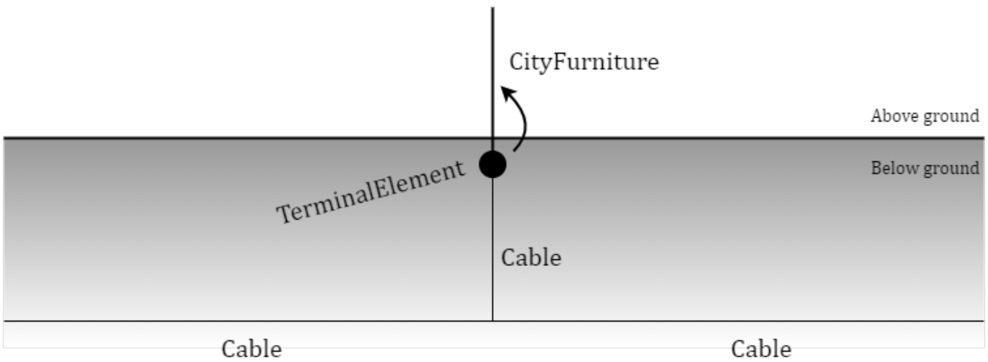


Figure 4.4: Electricity network topography

In Section 3.2.1 it is explained that streetlights can be connected to the main electricity line as a 'group' of multiple streetlights. In this case one of the streetlights is directly connected to the main electricity line. In case this connection line between the main electricity line and parent streetlights is broken, the other streetlights in the group are not functioning either.

This group of streetlights is grouped to a CityObjectGroup. CityObjectGroups aggregate CityObjects, e.g. TerminalElements. Each member of a group may be qualified by a role name, reflecting the role each CityObject plays in the context of the group. A 'parent' role is assigned to the streetlight that is connected to the main electricity line (the one on the right in Figure 4.5). In case the streetlight has no dependents, it will not be found as part of a CityObjectGroup. In this way it can easily be checked what streetlights are part of the same group and what streetlights are affected in case the parent streetlight is not functioning anymore. Moreover, the structure of the resulting CityGML file is more clear which helps when inserting the data into a database. Figure 4.5 and Figure 4.6 depict this situation respectively with a topographical representation and corresponding UML model. Note that only the main electricity lines and connection lines to the parent streetlights, and not the lines between the mutually connected streetlights, are represented.

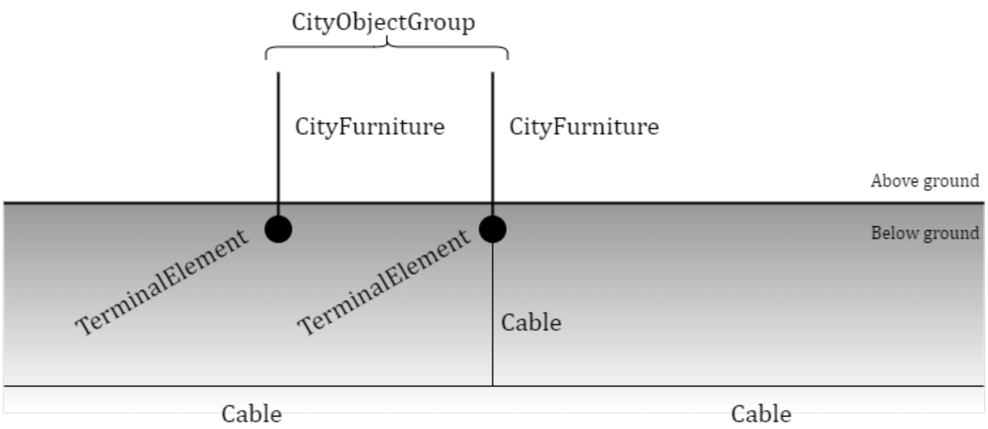


Figure 4.5: Grouping streetlights

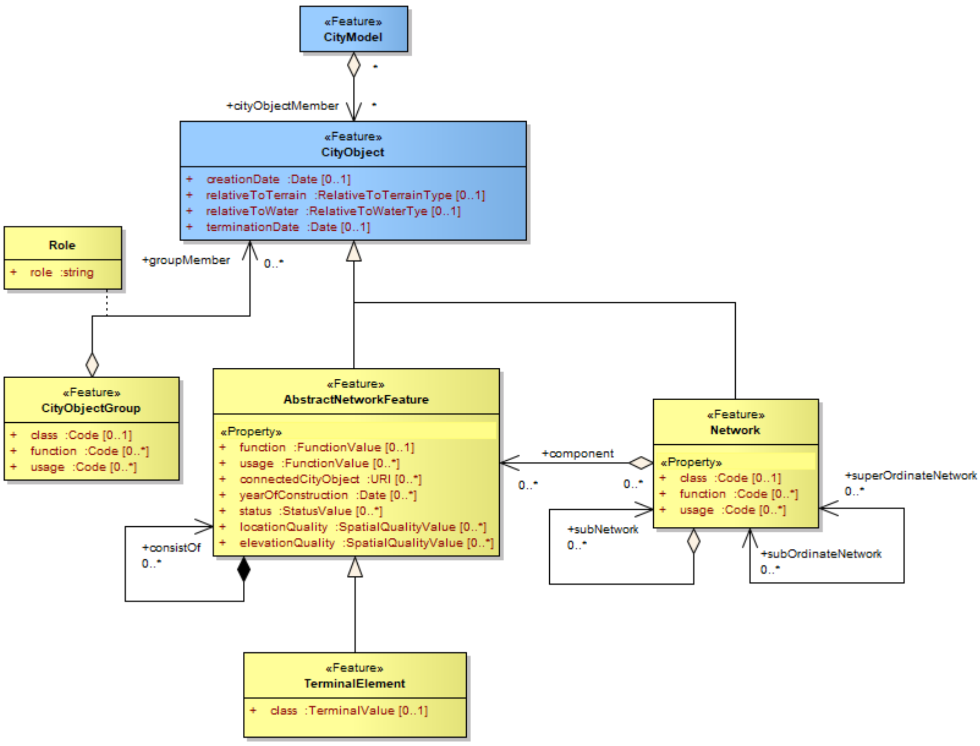
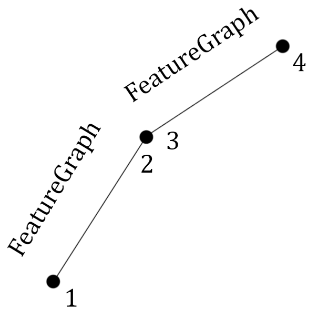


Figure 4.6: Grouping streetlights in UML

The FeatureGraph class is used in order to reflect the topological aspects of each element. In other words, a FeatureGraph is used for relating the different below ground utility network features. A FeatureGraph consists of at least one Node. A network component that is mapped to a single exterior Node is the minimum possible FeatureGraph representation.

The electricity cable is graphically represented by an InteriorFeatureLink marked with an exterior start and end Node. The topological relation between different network features is established by an InterFeatureLink. An InterFeatureLink is a connection between two exterior nodes belonging to a different FeatureGraph in the same Network. Figure 4.7 illustrates this concept with a simple example of two connecting lines. The nodes used as start and end of the InterFeatureLink have different id's but the exact same xyz coordinates.



InteriorFeatureLinks

id	start	end	geom
1	1	2	0 0 0, 2 3 1
2	3	4	2 3 1, 5 5 2

InterFeatureLinks

id	start	end
1	2	3

Nodes

id	geom
1	0 0 0
2	2 3 1
3	2 3 1
4	5 5 2

FeatureGraphs

id
1
2

Figure 4.7: Utility Network ADE topology principle

Figure 4.8 is a graphical representation of the below ground electricity network features, and their relationships. To have a complete image of the situation, the city furniture object is graphically represented as a simple vertical line.

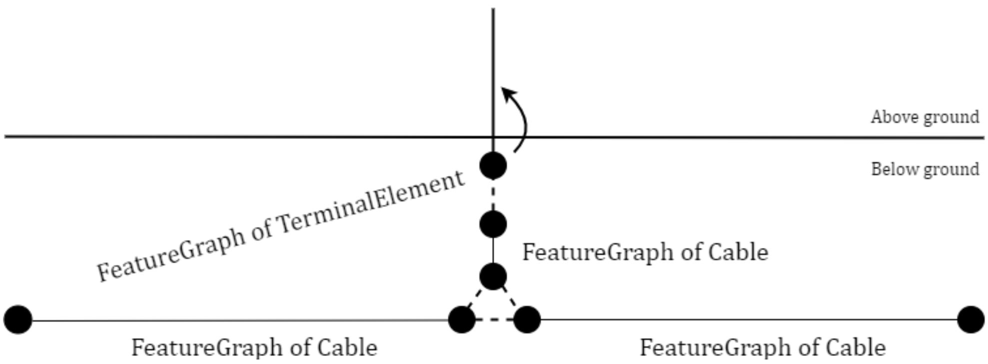


Figure 4.8: A graphical representation of the electricity network

Figure 4.8 shows three intersecting lines, with one point of intersection. This is the point where the connection line meets the main electricity line. The main electricity line is split up in order to do rather detailed (network) analyses later on. In addition, a streetlight is graphically represented as a simple vertical line. The dotted lines represent InterFeatureLinks that are used to topologically relate the different FeatureGraphs. More specific, an InterFeatureLink is used to describe the connection between two exterior nodes of different FeatureGraphs. In this particular case, InterFeatureLinks are used to connect the FeatureGraphs of the different Cables and to connect the FeatureGraph of the supply Cable and the FeatureGraph of the TerminalElement.

Since, the above ground city objects are not directly part of the utility network, it is not possible to have a relationship between the TerminalElement and the CityFurniture through an InterFeatureLink. This is where the useful 'connectedCityObject' attribute of the AbstractNetworkFeature comes in. This attribute is used to create a one-to-one topological relationship between the below ground network feature (TerminalElement) and the streetlight as above ground city object (CityFurniture). This is presented with the UML class diagram in Figure 4.9. The classes in blue belong to the CityGML core model and the classes in yellow belong to the Utility Network ADE.

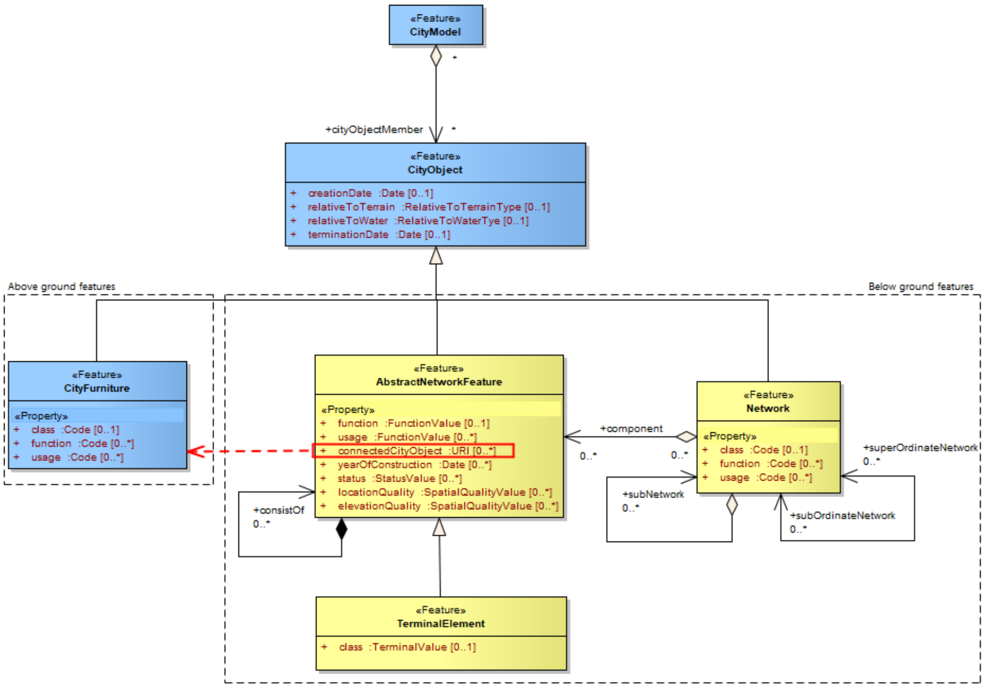


Figure 4.9: Linking below and above ground electricity network features in UML

4.1.2 Sewer network data mapping

This section describes how the sewer network is mapped to the CityGML Utility Network ADE. Figure 4.10 is a UML class diagram of the classes used for the topographical representation of the sewer network. Some detail is added through e.g. the AbstractFeatureMaterial and AbstractCommodity classes. However, only the classes belonging to the core module are depicted for readability purposes. After discussing the topographical part, the graphical representation will be discussed.

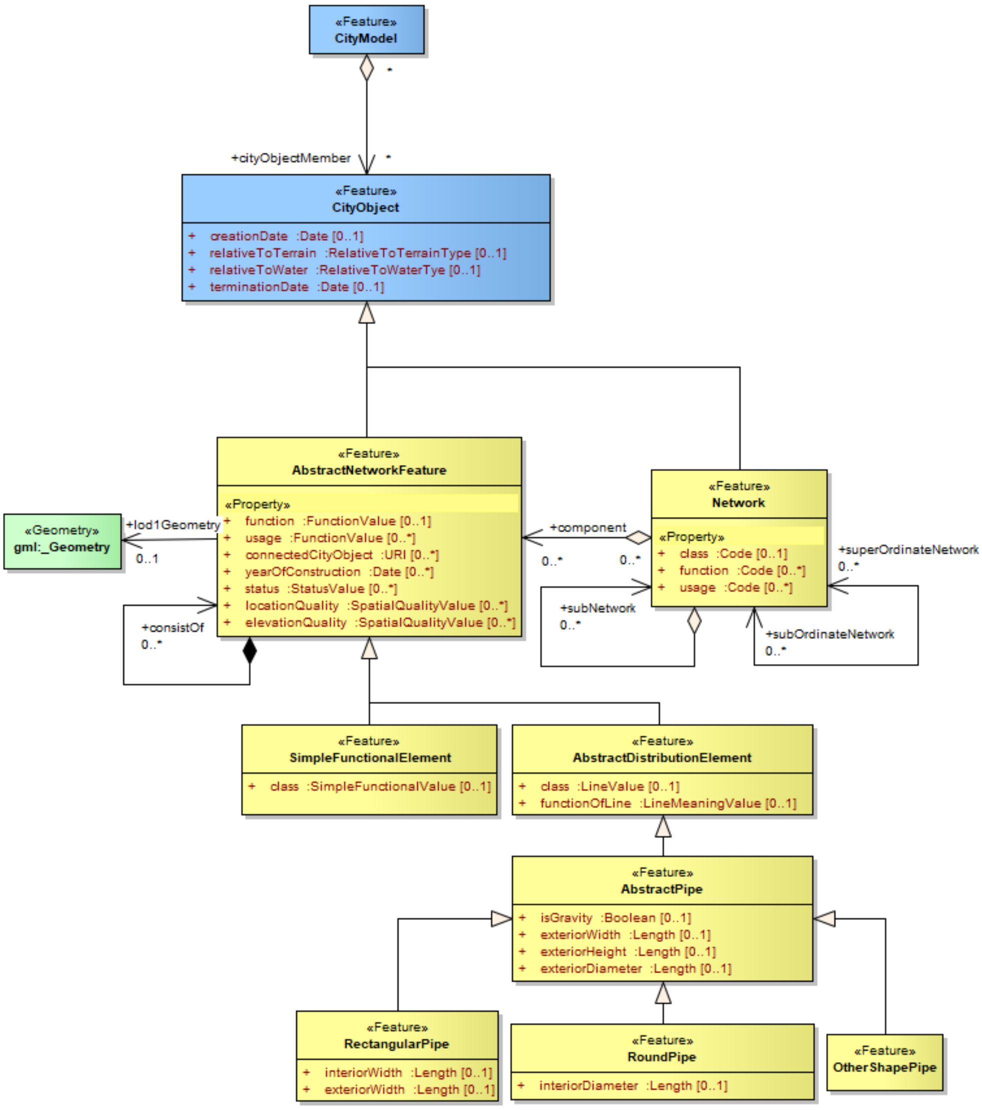


Figure 4.10: A UML diagram with all classes used for the topographical representation of the sewer network

In Section 3.2.2 it is explained that the sewer network data is gathered through measurements at the place of a manhole. As a result, each sewer network line is marked with a manhole at its start and end point. Hence, each sewer pipe is connected to a manhole at its end and start. Depending on the shape, the sewer pipe is topographically represented with one of the AbstractPipe subclasses (see Figure 4.10)

The AbstractPipe class inherits the properties of both the AbstractDistributionElement as well as of the AbstractNetworkFeature class. In case the cross section shape of the pipe is round, the interiorDiameter attribute is used to determine the inner diameter size. In other cases, the height can be different than the width. For this reason, the two possible attributed, interiorWidth and interiorHeight, are used to specify the size of a sewer pipe with a rectangular cross section. Sewer pipes with other possible cross section shapes, e.g. egg shaped, are mapped to OtherShapePipes.

In Section 3.2.2 it is explained that the sewer network as a whole consists of pipes with different functions. The collection of pipes with similar functional characteristics can be seen as a subNetwork. Since the focus is this research is put on the gravity sewer system, all pipes are part of the same Network. Again, the status attribute the AbstractNetworkFeature is used to state whether a sewer pipe is in use or not.

The SimpleFunctionalElement class is a-kind-of AbstractNetworkFeature. In the codelist some possible values for the SimpleFunctionValue attribute are given (see the Network Components module in Appendix A). The codelist includes a couple of different fittings as well as a 'manhole' value. The difference between a fitting and a manhole is that a manhole not only connects the below ground sewer pipes but also has a connection to the surface, with the manhole cover. This manhole can be topographically represented in LoD 1 to 4. To be consistent, the manhole is represented in the lowest possible LoD, as a point object.

Manhole covers are visible on the surface, so are part of the core CityGML model. However, in [Gröger et al., 2012] it is not stated in a clear manner to what class manhole covers should be mapped. City furniture objects can be found in traffic areas, residential areas, on squares or in built-up areas. This aligns with the definition of a manhole cover. Also, it is stated that city furniture objects are immovable. Though, the CityFurniture class is used for e.g. garbage cans. Just as garbage cans, manhole covers can potentially be moved, if needed. For this reason, the CityFurniture class is used for representation of the manhole covers. Figure 4.11 is a topographical representation of a manhole, mapped to a single node, as connection between two sewer pipes and a manhole cover.

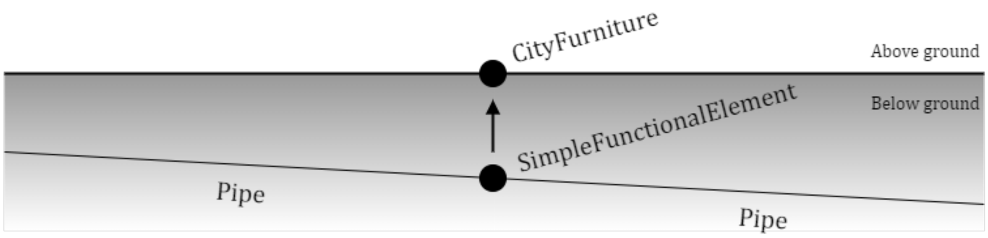


Figure 4.11: Sewer network topography

The sewer pipe is graphically represented by an InteriorFeatureLink marked with an exterior start and end Node. Figure 4.12 shows two connecting lines, with a manhole in between. The manhole connects both below ground sewer pipes but forms a connection to the above ground manhole cover too. InterFeatureLinks are used to have a topological link between (the exterior nodes of) the FeatureGraphs of the Pipe and the SimpleFunctionalElement. In a standard sewer system water flows from higher to lower elevations. Therefore, the node with the lowest z-value is marked as end node of InteriorFeatureLink. At the same time, this node is marked as start node of the InterFeatureLink to the SimpleFunctionalElement which is graphically represented by a single node. The node as part of the InteriorFeatureLink with the highest z-value is marked as start node of the InteriorFeatureLink and, at the same time, as end node of the InterFeatureLink.

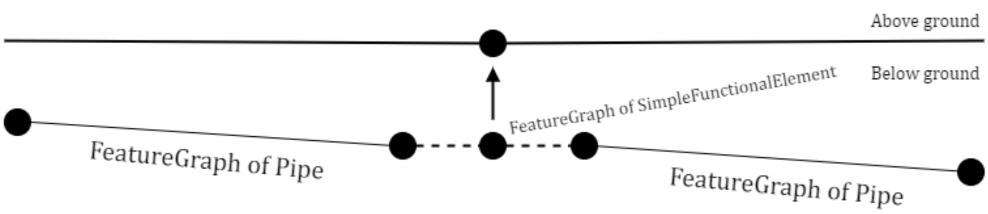


Figure 4.12: A graphical representation of the sewer network

The 'connectedCityObject' attribute of the AbstractNetworkFeature is used to create a directional one-to-one topological relationship between the below ground network feature (SimpleFunctionalElement) and the manhole cover as above ground city object (CityFurniture). This is illustrated with a UML class diagram in Figure 4.13. The classes in blue belong to the CityGML core model and the classes in yellow belong to the Utility Network ADE.

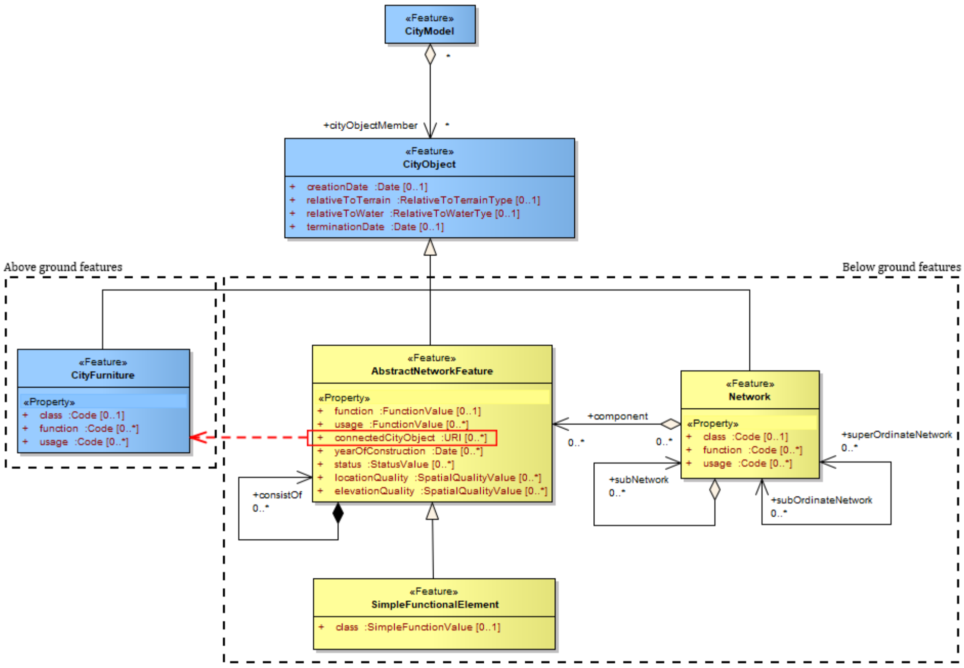


Figure 4.13: Linking below and above ground sewer network features in UML

4.2 DERIVATION OF THE RELATIONAL DATABASE

Two main parts can be defined; mapping the core CityGML model to a relational database, and extending the database by mapping the data model for utility networks to the relational database.

As CityGML datasets may become very large and objects may be arbitrarily nested leading to complex data structures, the efficient storage and input/output of CityGML data requires carefully optimized database schemas [Stadler et al., 2009]. One of the benefits is that a database ensures interoperable data access. Moreover, a database is essential in applications in which large amount of large-scale geodata need to be continuously maintained and managed [Zlatanova and Stoter, 2006].

4.2.1 Mapping the core CityGML model onto a relational-database model

This section describes how the object-oriented data model of CityGML is mapped to the relational structure of a spatial relational DBMS (SRDBMS) (henceforth referred as 'relational database' or just 'database'). Note that this database only allows for storage of city objects belonging to the core CityGML model, and e.g. not to the Utility Network ADE. The focus is put on the city objects that relate to the below ground electricity and sewer network; streetlights and manhole covers.

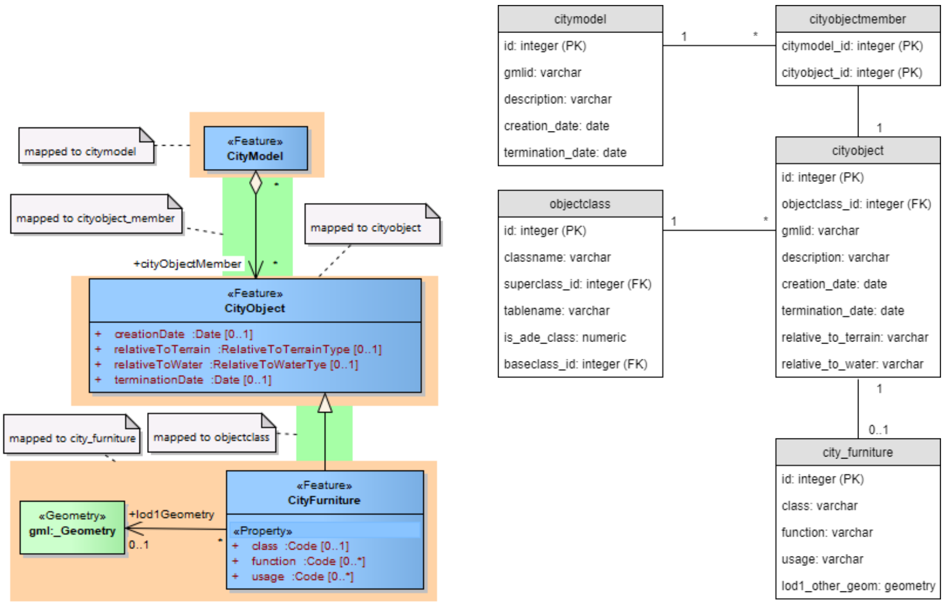
This mapping task is conducted by means of 3DCityDB. 3DCityDB is a free open source package consisting of a database schema and a set of software tools to import, manage, analyze, visualize, and export virtual 3D city models according to the CityGML standard. The database schema results from a mapping of the object-oriented data model of CityGML to the relational structure of a database.

Before the mapping process, the original CityGML model is slightly simplified allowing an optimized workflow and guaranteeing efficient processing time. Subsequently, all the feature classes are mapped to database tables. Generally, one or more classes are merged and mapped to a single table (see Section 2.3). In case representation is only possible by additional tables, a class is mapped to a single table. The attributes of the classes become columns of the corresponding table.

Although the 3DCityDB allows to map all thematic CityGML classes, this section focuses only on the mapping of those classes representing city objects that could possible relate to the below ground utility networks. As explained in Section 4.1.1 and Section 4.1.2, these are: streetlights and manhole covers which are modeled as city furniture objects.

The base class of all thematic classes within CityGML is the Cityobject class, which may be aggregated to a single CityModel. The CityObject class provides a creation and termination date for the management of histories of features as well as the position relative to the terrain or water (see Figure 4.14a). The subclasses of the CityObject class comprise the different thematic fields of a city model, e.g. city furniture. The complete CityGML model is very extensive [Gröger et al., 2012]. Though in this research the focus is only put on the CityFurniture class since the related above ground city objects are modeled as such.

Figure 4.14a depicts a UML diagram with 4 classes, representing the CityFurniture class with its parent classes and a geometry class. The CityFurniture class is a subclass of the CityObject class. Hence, city furniture objects inherit the properties of the city object.



(a) (b)

Figure 4.14: UML diagram and resulting database schema after the mapping

Figure 4.14b presents the database schema after mapping the presented CityGML classes. The four CityGML classes are mapped to five database tables. The classes that are mapped to a single table are shown as orange blocks in the UML diagram. Relationships, e.g. many-to-many relationships, that can only be presented by additional tables, are shown as green blocks.

All CityObject instances, and that of subclasses, are represented by tuples in the table 'cityobject'. The fields in the cityobject table are identical to the attributes of the corresponding UML class, plus some additional fields for metadata. In order to identify each city object, a unique identifier is essential which is achieved by the 'id' attribute. The cityobject table contains the attribute objectclass_id as foreign key which refers to the parent table 'objectclass'. This helps to identify the proper subclass tables, e.g. a building has a different 'objectclass_id' than a city furniture object. All object class names of the schema are managed in the objectclass table. In the 'objectclass' table, the relation of the subclass to its parent class is represented via the attribute superclass_id in the subclass tuple as a foreign key to the id of the parent class. For instance, the tuple representing the CityFurniture subclass has a superclass_id which refers to the CityObject class in the same cityobject table.

Besides having them stored as city objects, city furniture objects are represented by tuples in the table 'city_furniture'. The id, as primary key, of the city_furniture table refers to the cityobject table. The geometry of the city furniture object is stored in the field 'lod1_other_geom' of the city_furniture table. The geometry of city furniture objects can be represented in various ways, e.g. as surface-based geometry object or as point- or line-typed object. To be consistent and allow fast processing, the city furniture objects are represented as points or lines. In case of the sewer network, the manhole covers are mapped to point objects. In case of the electricity network, the streetlights are mapped to simple vertical line objects.

The table 'cityobjectmember' is created in order to allow aggregation of city objects to a single CityModel. This table has a composite primary key consisting of a citymodel_id and cityobject_id. This ensures that a city object cannot occur more than once in the same CityModel.

Figure 4.15 depicts an example of how a single city furniture object would be stored in a database according the presented database schema.

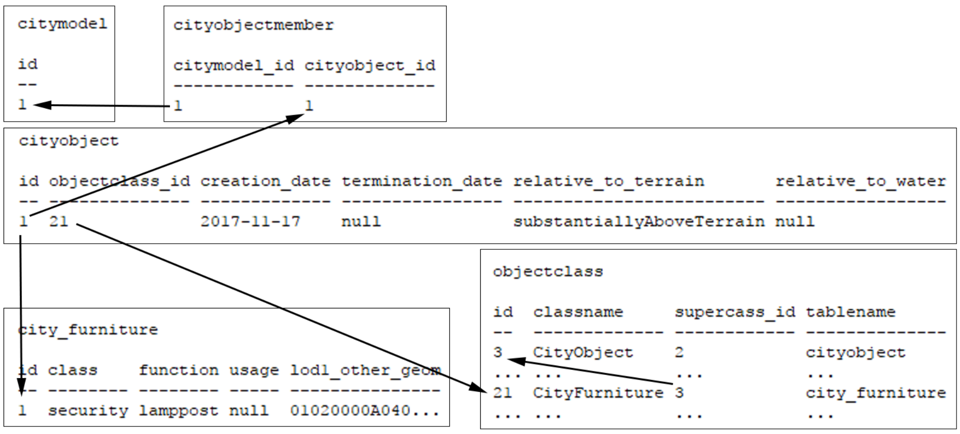


Figure 4.15: Storing a single city furniture object in the database according to the presented database schema

4.2.2 Mapping the CityGML Utility Networks ADE onto a relational-database model

This section describes how the 3DCityDB is extended in order to allow storage, representation and management of utility networks. Therefore the Utility Network ADE (objects-oriented data model) is mapped onto the relational database model (entity relationship model) and added to the 3DCityDB.

[Yao and Kolbe, 2017] presents how the 3DCityDB can be extended in order to allow for handling arbitrary CityGML ADE's. The proposed approach has been successfully implemented and tested based on a number of different CityGML ADE's including the Utility Network ADE. The 3DCityDB extension for the CityGML Utility Network ADE comes with a set of scripts that extends the relational database with numerous tables for representing the classes in all modules. However, only the Utility Network Core module is relevant for relating below ground utility network features and above ground city objects (see Section 4.1). The colored blocks in Figure 4.16 illustrate the mapped parts of the Utility Network ADE Core. Figure 4.17 presents the resulting entity relationship (ER) model for the Utility Network Core.

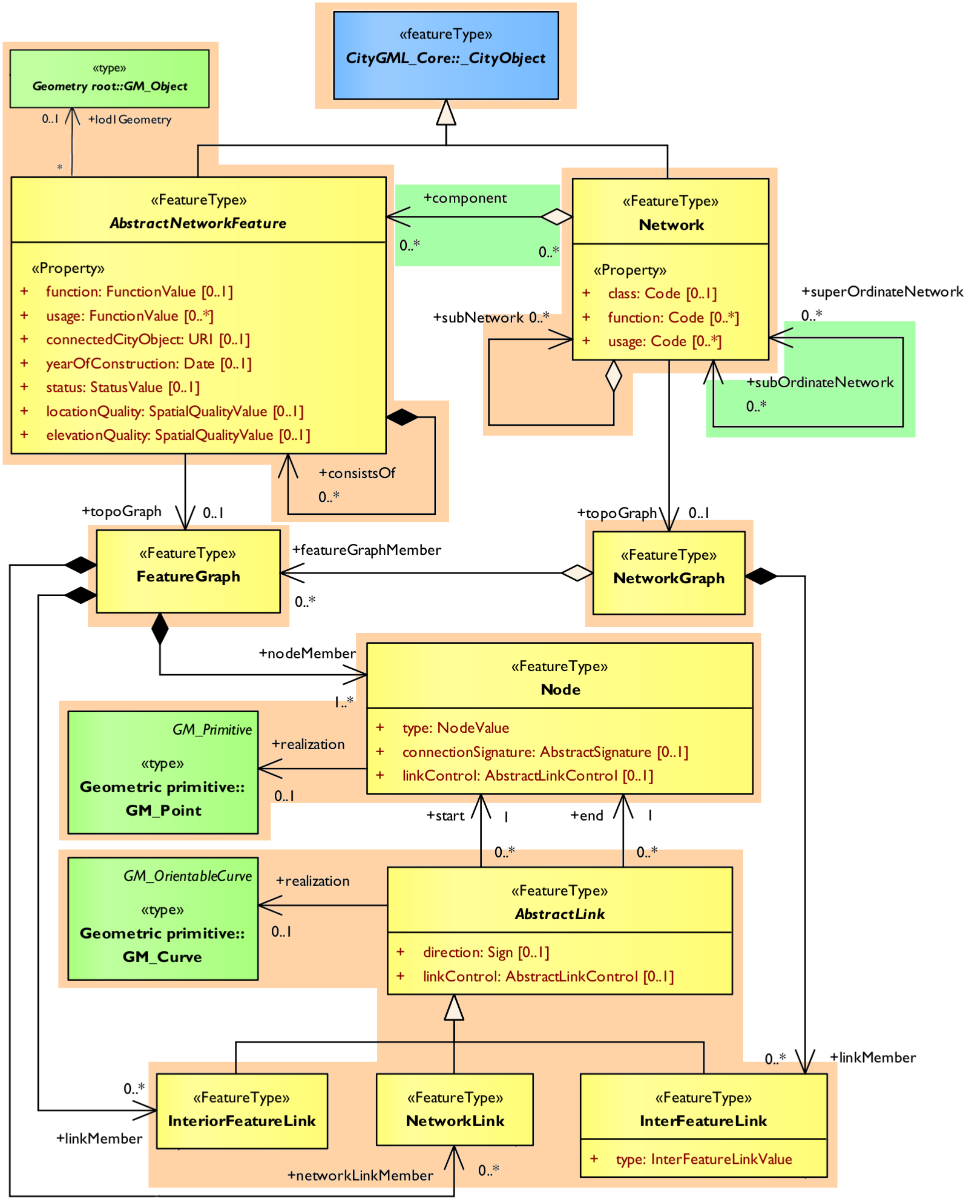


Figure 4.16: Utility Network ADE Core with colored blocks illustrating the mapped parts. Image from [Agugiaro, 2017]

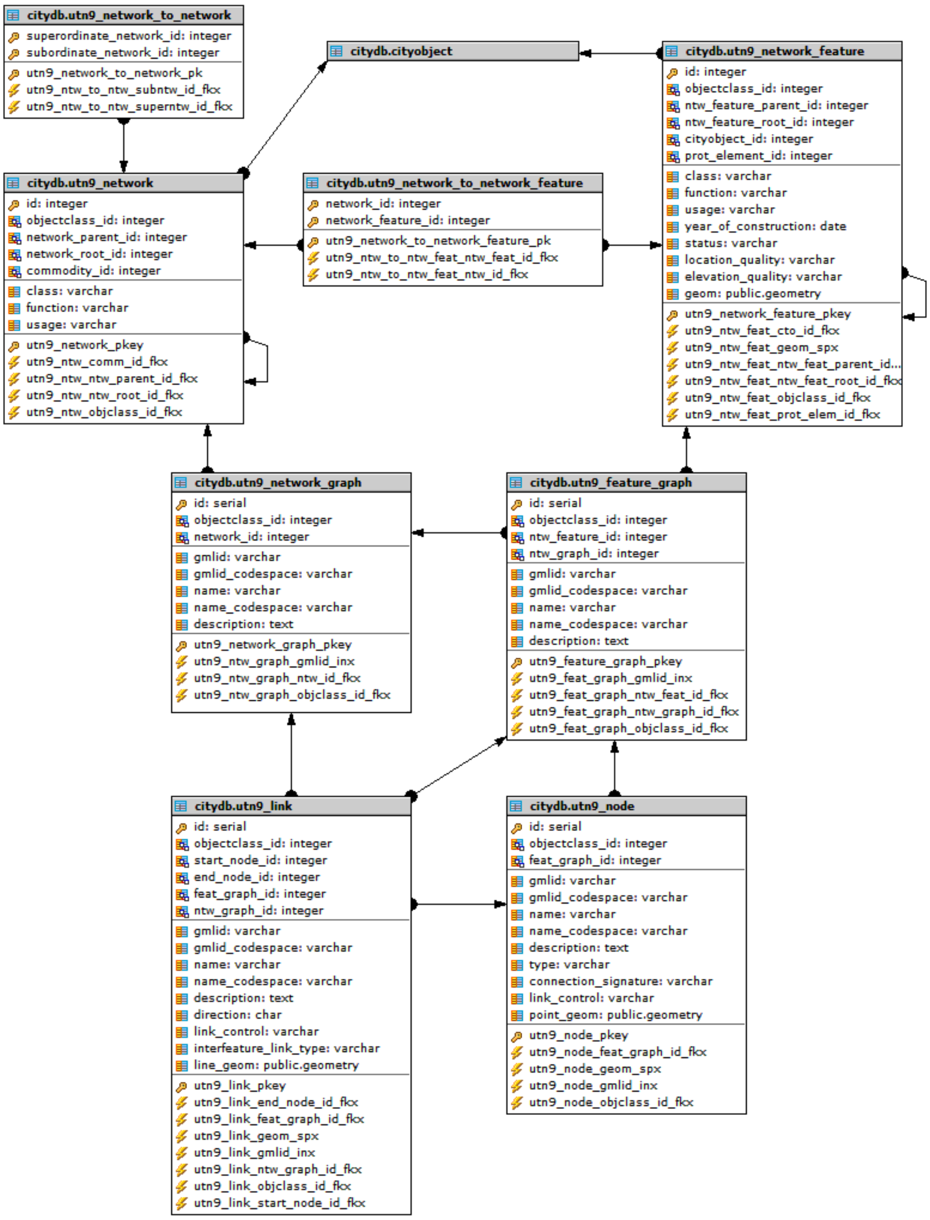


Figure 4.17: ER model for the Utility Network ADE Core. Image from [Agugiaro, 2017]

In addition, the objectclass table is extended with the Utility Network ADE classes.

The utn9_network_feature table contains a cityobject_id field which is a foreign key that refers to the id of the cityobject table. This allows for relating the below ground utility network features to the above ground city furniture objects, as shown in the resulting ER model that consists of the tables of interest used for the topographical representation (see Figure 4.18). In addition to the ones presented in Figure 4.14 and Figure 4.16 a utn9_distrib_element is introduced. The utn9_distrib_element allows for adding more detail to Pipes and Cables e.g. section shape dimension parameters.

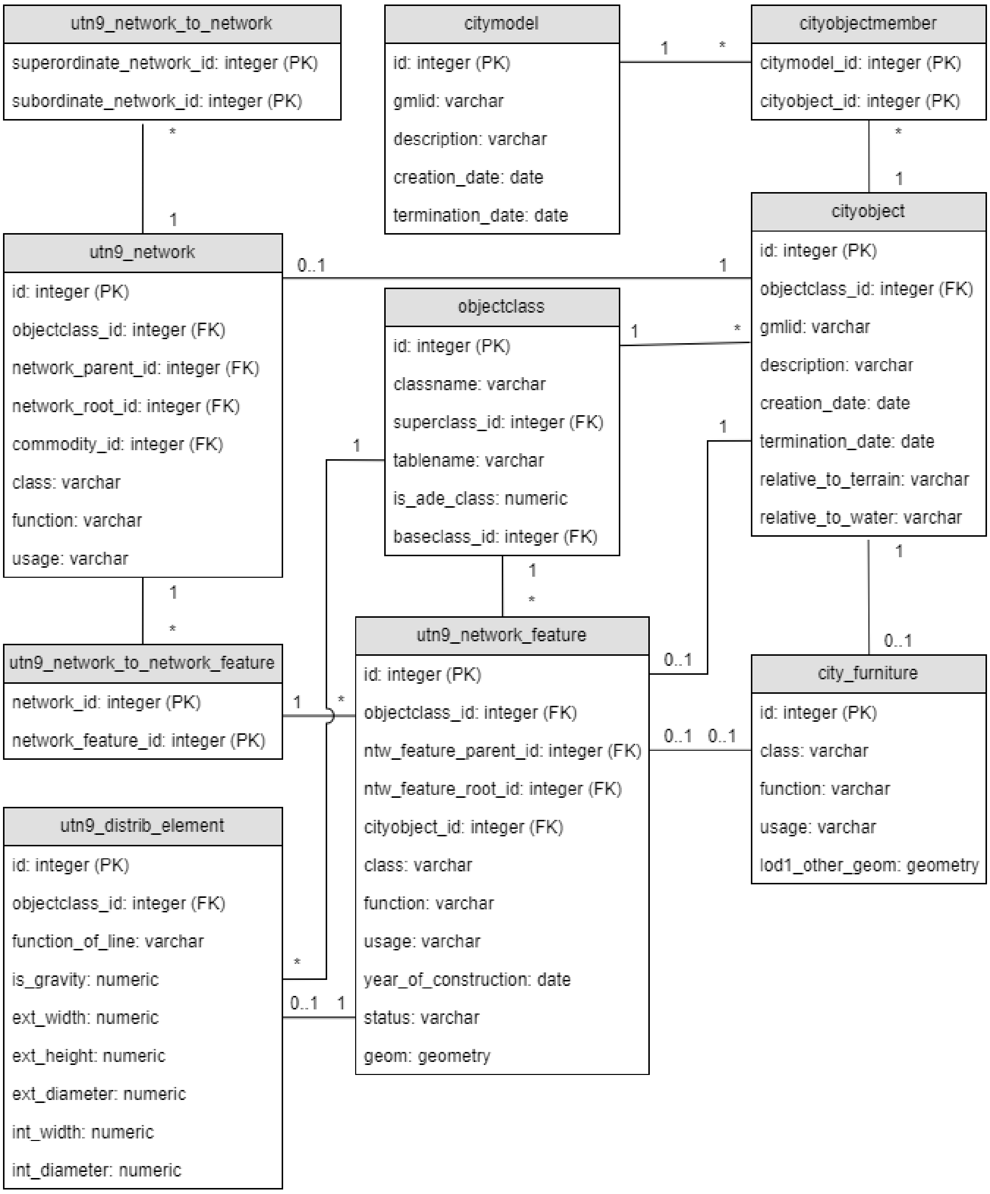


Figure 4.18: ER model for the topographical representation of the relevant utility network features and city furniture objects

The classes and derived tables used for the topographical representation of the utility networks can be found in Appendix B.

4.3 IMPORTING THE CITYGML DATA INTO THE RELATIONAL DATABASE

This section describes the process of importing the generated CityGML data into the relational database. This includes both the importing of the above ground city furniture objects as well as the below ground utility network features. First the importing of the electricity network data will be discussed, followed by the sewer network data.

After modeling the utility network data according to the Utility Network ADE, the data is put into the database. The modeled data contains a varied set of different types of features. To populate the tables in the database either only one or multiple types of features are used. The table names correspond with the names in the ER model.

4.3.1 Importing the electricity network data

Table 4.1 lists what types of features are used to populate the tables in the database for the electricity network data.

	Feature	Table
1	Network CityFurniture TerminalElement Cable	cityobjects
2	CityFurniture	city_furniture
3	TerminalElement CityObjectGroup	cityobjectgroup
4	TerminalElement CityObjectGroup	group_to_cityobject
5	Network	utn9_network
6	NetworkGraph	utn9_network_graph
7	TerminalElement Cable	utn9_network_feature
8	Cable	utn9_distrib_element
9	FeatureGraph	utn9_feature_graph
10	Node	utn9_node
11	InterFeatureLink InteriorFeatureLink	utn9_link

Table 4.1: CityGML features and table names used for the electricity network

4.3.2 Importing the sewer network data

Table 4.2 lists what types of features are used to populate the tables in the database for the sewer network data.

	Feature	Table
1	Network CityFurniture SimpleFunctionalElement RoundPipe RectangularPipe OtherShapePipe	cityobject
2	CityFurniture	city_furniture
3	Network	utn9_network
4	NetworkGraph	utn9_network_graph
5	SimpleFunctionalElement RoundPipe RectangularPipe OtherShapePipe	utn9_network_feature
6	RoundPipe RectangularPipe OtherShapePipe	utn9_distrib_element
7	FeatureGraph	utn9_feature_graph
8	Node	utn9_node
9	InterFeatureLink InteriorFeatureLink	utn9_link

Table 4.2: CityGML features and table names used for the sewer network

Note that the 'cityobjects' table is populated with below as well as above ground features. The cityobjects table is a collection of all city objects, above and below ground, in the city model. Several tables exist that relate to the cityobject table. For this reason, some features are used to populate more than one table. The same goes for the TerminalElement and CityObjectGroup features in the electricity network data. Both feature types are used to populate the cityobjectgroup table as well as the group_to_cityobject table.

Figure 4.17 has shown that many tables are linked through foreign keys which makes the order of inserting the data an important task that must be handled with care. For example, the cityobjects table has to be populated before populating the utn9_network table since the id of the utn9_network table is a foreign key that refers to the id, as primary key, of the cityobjects table. Hence, the id must exist in the cityobjects table before populating the utn9_network table. The order in which Table 4.2 lists the features and tables is also the order of inserting used.

Moreover, assigning the id's has to be done with care since these are all primary keys and thus have to be unique. Inserting the different features into the cityobjects table is done in one batch. All the features are inserted one after each other having a unique numeric id. Also, care must be taken on referencing the id's in different tables. This means that e.g. the id of a feature in the city_furniture table must be the same as it is in the cityobjects table. This is achieved by assigning the id's as the very first step in the process of inserting.

5 TECHNICAL IMPLEMENTATION AND RESULTS

This chapter describes the details of the experiments (including datasets used) and the actual implementation of the data. The implementation details (e.g. language details, classes, bits and bytes, running time) will be presented and discussed.

5.1 TOOLS AND LIBRARIES

Before the actual processing of the data, some pre-processing is done in Python and with the FME software. The Python Shapefile Library (pyshp) is used for reading and writing ESRI Shapefiles. Shapely is used for manipulation and analyses (e.g. nearest neighbour) of the Shapefiles. Shapely is based on the widely deployed GEOS (the engine of PostGIS) and JTS (from which GEOS is ported) libraries. Translating ESRI Shapefiles to CityGML data is completely done in FME. Moreover, PgAdmin is used as development platform for PostgreSQL object-relational database. PostGIS is used as a spatial database extender for PostgreSQ. It adds support for geographic objects allowing location queries to be run in Structured Query Language (SQL). In order to visualize the data, Quantum GIS (QGIS), ArcGIS, FME's Data Inspector or the FZKViewer is used. QGIS is often used for quick inspections and minor analyses of mostly 2D ESRI Shapefiles. FME's Data Inspector and the FZKViewer are used to visualize the datasets in CityGML format. ArcGIS is mainly used to visualize the data stored in the database. In addition, Enterprise Architecture is used for UML modeling.

5.2 SPATIAL SCOPE

The proposed data modeling approach is tested on an area in 'Hoogvliet', located in the southern part of Rotterdam (Figure 5.1). Due the current developments, above and below ground, this is an interesting area. Furthermore, this area has a diverse network of pipes and cables and assets. Therefore, this area could be representative for the rest of the city of Rotterdam.

Figure 5.1: Hoogvliet, Rotterdam

5.3 ASSET MANAGEMENT IN ROTTERDAM

The term 'asset management' is included in the main research question and therefore needs to be defined. Asset management can be defined as an integrated framework for the maintenance of the assets in public space, above and below ground. In order to spend the maintenance budget as wisely as possible and better decision making an optimum balance is needed between costs, performance and risk. Asset management links technical maintenance directly to the most important goals of the city. As such through maintenance of the city value is created for the development of Rotterdam [Municipality of Rotterdam, 2017].

The following assets, managed by the municipality of Rotterdam, can be distinguished:

- Subsurface
- Green infrastructure
- Playgrounds
- Streetlights
- Roads
- Bridges and tunnels
- Surface water
- Sewerage
- Visual art and monuments

5.4 DATA

Thanks to municipality of Rotterdam a varied selection of datasets were available to use for experimentation. Both a dataset containing the pipes and cables as well as several related above ground city objects, e.g. streetlights and manholes, are provided. The data is originally extracted from the LVZK which is used to manage all pipes and cables in Rotterdam (Figure 5.2).

Appendix C describes the datasets in more detail. Following are a short description of each dataset:

5.4.1 Electricity network data

Electricity lines

An ESRI Shapefile containing 3D line objects representing the electricity lines (LV and OV).

Streetlights

An ESRI Shapefile containing point objects representing the location of a streetlight. The type of connection and physical properties, e.g. the height, are put in the attribute fields.

5.4.2 Sewer network data

Sewer network lines

An ESRI Shapefile containing 3D line objects representing the standard sewer pipes.

Figure 5.2: Pipes and cables in Hoogvliet, Rotterdam

Manholes

An ESRI Shapefile containing point objects representing the location of the manholes.

5.4.3 Rotterdam 3D

A 3D city model of Rotterdam in CityGML format including buildings, trees, ground level, design and building information. Pipes and cables are also part of the 3D city model but are modeled as generic city objects.

In addition to the Rotterdam data, a 2D dataset from Stedin is examined and also detailed in Appendix C.

5.5 PRE-PROCESSING THE ELECTRICITY NETWORK DATA

As described in Section 3.1, the connection line between the above ground feature and main electricity line is not registered. Hence, it is currently unknown how the streetlights are connected to the below ground electricity lines. In order to check whether the CityGML Utility Network data model is suitable for modeling such detailed network structures, an algorithm is written. Having the streetlights, as points, and the electricity cables, as lines, as input a best estimate of what streetlights are connected to what streetlights and what streetlights are connected to what electricity line is computed in Python (see Figure 5.3 for a flowchart of the algorithm).

In Section 3.2.1 it is described that streetlights can be connected to the electricity line in different ways. The constraints for the developed algorithm are listed below.

- A streetlight cannot be directly connected to a HV or MV voltage cable
- Any kind of streetlight can directly connected to an OV cable
- Streetlights with an assisting wire are directly connected to an OV or to a LS cable
- Streetlights with a fuse connection are connected to another streetlight or to an OV cable
- The connection line prefers the shortest distance

After running the pre-processing algorithm the original ESRI Shapefile is enriched with an attribute field specifying the group. Moreover, the algorithm outputs the connection lines. As a result, the coordinates of the end point of the connection line is equal to the base of the streetlight (Figure 5.4). The source and target object, streetlight and electricity line, are specified as an attribute. In each group of streetlights there is always one streetlight that has a direct connection to the main electricity line.

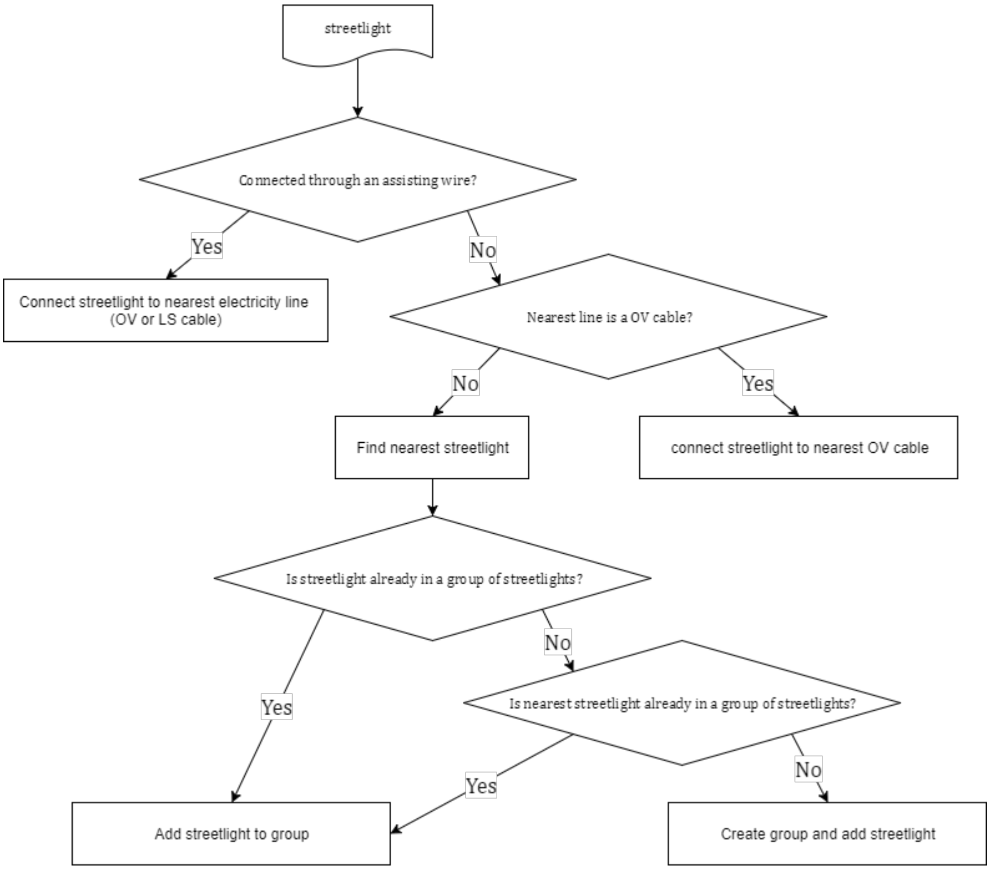


Figure 5.3: A flowchart representing the pre-processing electricity network data algorithm

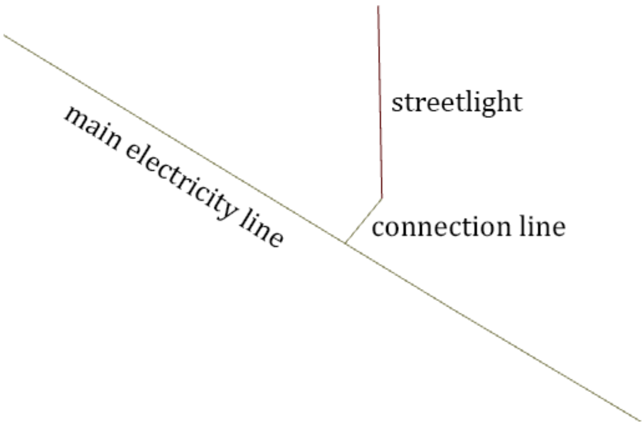


Figure 5.4: Main electricity line and connection line to streetlight

5.6 MAPPING THE DATA TO CITYGML

This section describes in more detail how the utility network data, as ESRI Shapefile, is mapped to CityGML including Utility Network ADE. This translation is completely done in FME, for both the electricity network and the sewer network. The principle of FME is reading the source data, manipulating the data structure and content with so-called transformers and subsequently writing the data. To be able to write CityGML files in FME, the different feature types need to be specified to the CityGML Writer first. In addition to the standard core CityGML feature types, the feature types of the CityGML Utility Network ADE are specified. The complete translation is complex. Over a hundred FME transformers are used. Therefore, this section only elaborates on the most important parts. The creation of the CityGML file is different for both utility networks. However, the different types of used FME transformers is similar.

The following FME Transformers are used:

- SpatialFilter
- Intersector
- TopologyBuilder
- DuplicateFilter
- Creator
- UUIDGenerator
- Tester
- TestFilter
- FeatureMerger
- ListBuilder
- ListExploder
- NeighborFinder
- StringConcatenator
- SurfaceDraper
- GeometryRemover
- GeometryPropertySetter
- VertexCreator
- AttributeCreator
- AttributeRenamer

Relationships (associations, aggregations, compositions) between the classes are encoded in a CityGML file such that the referenced features are represented as child elements of the referencing features (parent elements). When a UML class is referenced by several other UML classes, relationships are realized in CityGML using XLinks. Xlinks are used for pointing to remote objects within the same or an external document. Figure 5.5 presents how the relationships (child-parent or xlink) are encoded in the Utility Network ADE Core model.

Figure 5.6 depicts a small segment of the resulting CityGML file. The InteriorFeatureLink and Nodes, as child elements, are indented into the parent FeatureGraph. A reference to the start and end Node is built by assigning their identifiers to xlink attributes of the InteriorFeatureLink.

The relationships need to be provided explicitly for each referenced feature in the form of the role names in the UML diagram (see Appendix A). The AttributeCreator transformer is used in FME to add the 'citygml_feature_role' attribute to the referenced features. The attribute value is the role name of the referenced UML class. For example, for the features Cable and RoundPipe the attribute value is 'component' (Figure 5.7). The same procedure, but different role names, is used to create relationships between other CityGML features.

The complete CityGML files can be found in [den Duijn, 2017].

5.6.1 Mapping the electricity network data to CityGML

This section elaborates on the mapping of the electricity network data, as ESRI Shapefiles, to CityGML including the Utility Network ADE. The following CityGML classes are written:

- CityGML Core
  - CityFurniture
  - CityObjectGroup

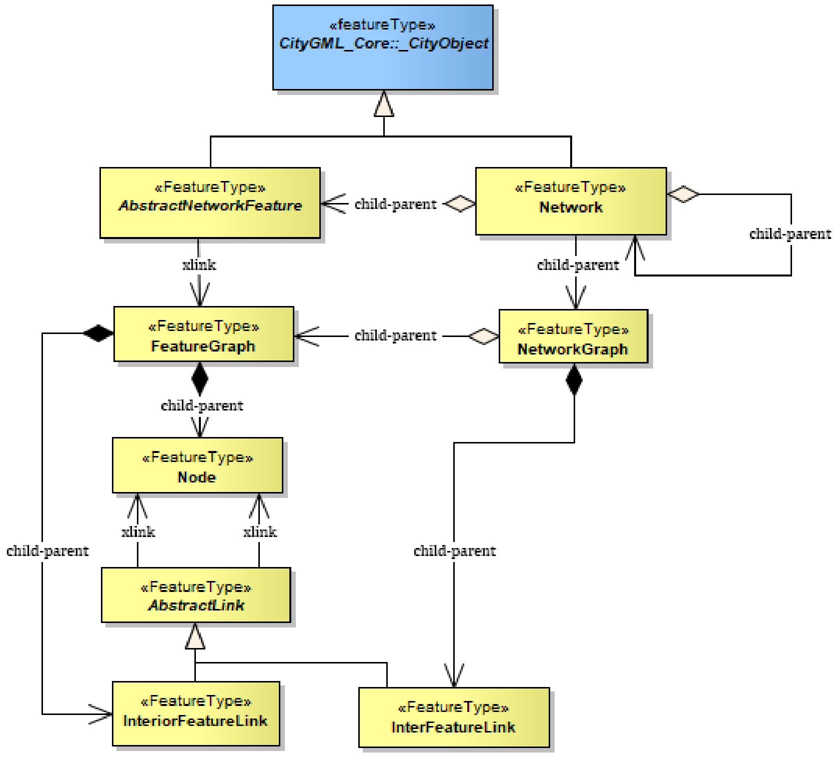


Figure 5.5: Mutual relationships in simplified Utility Networks ADE

- CityGML Utility Network ADE
  - Topographical
    - Network
    - TerminalElement (AbstractNetworkFeature)
    - Cable (AbstractDistributionElement, AbstractNetworkFeature)
  - Graphical
    - NetworkGraph
    - FeatureGraph
    - InterFeatureLink
    - InteriorFeatureLink
    - Node
  - Additional
    - ExteriorMaterial
    - ElectricalMedium

The connection lines, between the main electricity line and streetlight, are output after running the algorithm, as explained in Section 5.5. Before determination of the topological relationships, it is important to ensure that the connection lines touch the main electricity line. This is achieved by extending the connection lines a little (0.01 ground units) first. Then, the intersections between the lines are computed with the Intersector Transformer, where after the short lines are neglected. This process is depicted in Figure 5.8.

Subsequently, the topological relationships are determined with a TopologyBuilder transformer. The TopologyBuilder is set to 'assume clean data' in order to prevent that lines are splitted in case they intersect in 2D but not in 3D. The topologically

```
<utility:featureGraphMember>
  <utility:FeatureGraph gml:id="UUID_fc3aa9fa-65ef-43d5-8a08-64b5f9ee9614">
    <utility:linkMember>
      <utility:InteriorFeatureLink gml:id="UUID_d39ddc52-9961-42e0-8af3-1044bc0f67bc">
        <utility:realization>
          <gml:LineString srsName="epsg:28992" srsDimension="3">
            <gml:posList>83433.28 431561.93 0.27 83429.88 431564.76 0.24</gml:posList>
          </gml:LineString>
        </utility:realization>
        <utility:start xlink:href="UUID_ef69d839-5252-4b49-aaae-72150f2a5346"/>
        <utility:end xlink:href="UUID_b1602879-1d9a-4ed6-a3a1-09ca2f6f7cbe"/>
      </utility:InteriorFeatureLink>
    </utility:linkMember>
    <utility:nodeMember>
      <utility:Node gml:id="UUID_ef69d839-5252-4b49-aaae-72150f2a5346">
        <utility:type>exterior</utility:type>
        <utility:realization>
          <gml:Point srsName="epsg:28992" srsDimension="3">
            <gml:pos>83433.28 431561.93 0</gml:pos>
          </gml:Point>
        </utility:realization>
      </utility:Node>
    </utility:nodeMember>
    <utility:nodeMember>
      <utility:Node gml:id="UUID_b1602879-1d9a-4ed6-a3a1-09ca2f6f7cbe">
        <utility:type>exterior</utility:type>
        <utility:realization>
          <gml:Point srsName="epsg:28992" srsDimension="3">
            <gml:pos>83429.88 431564.76 0</gml:pos>
          </gml:Point>
        </utility:realization>
      </utility:Node>
    </utility:nodeMember>
  </utility:FeatureGraph>
</utility:featureGraphMember>
```

Figure 5.6: Relationships between classes encoded in CityGML

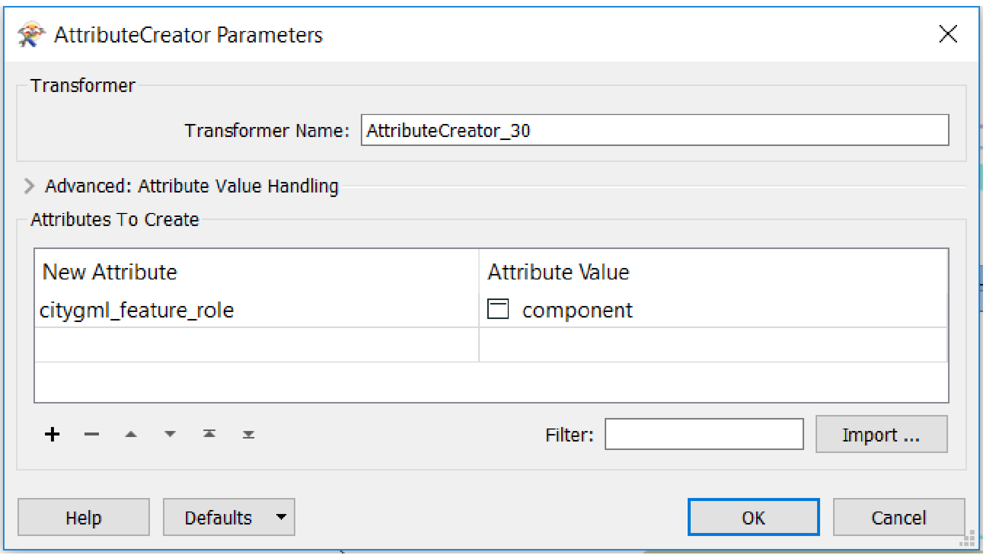


Figure 5.7: Reference features using the FME AttributeCreator

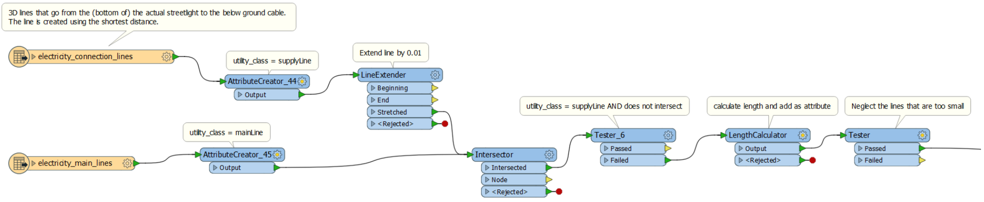


Figure 5.8: FME transformers used to ensure correct determination of topological relationships

significant nodes (point geometries) and edges (curve geometries) are output. These features, together with the point geometries in the Streetlights ESRI shapefile, form the basis for the rest of the translation.

The output edges are mapped to the following features:

- Cables
- InteriorFeatureLinks as part of the FeatureGraph of the Cable

The output nodes are used to write the following features:

- Nodes as part of the FeatureGraph of the Cable
- Nodes as part of the FeatureGraph of the TerminalElement (the end of the connection line corresponds with the location of the streetlight)
- InterFeatureLinks
  - between the exterior nodes as part of the different FeatureGraphs of Cables
  - between the exterior node as part of the FeatureGraph of a TerminalElement and the exterior node as part of the FeatureGraph of a Cable
- CityFurniture (the Neighborfinder transformer is used to find what nodes should be enriched with the streetlight attributes and represented as streetlights. The height attribute is used to create a vertical line which topographically represents the streetlight as a city furniture object)

The Creator transformer is used to write the Network and NetworkGraph features. In addition, the 'SUBGROUP' attribute, that is added to each streetlight after the pre-processing, is used to write the CityObjectGroups.

5.6.2 Mapping the sewer network data to CityGML

This section elaborates on the mapping of the sewer network data, as ESRI Shapefiles, to CityGML including the Utility Network ADE. The following CityGML classes are written:

- CityGML Core
  - CityFurniture
- CityGML Utility Network ADE
  - Topographical
    * Network
    * SimpleFunctionalElement (AbstractNetworkFeature)
    * RoundPipe (AbstractDistributionElement, AbstractNetworkFeature)
    * RectangularPipe (AbstractDistributionElement, AbstractNetworkFeature)
    * OtherShapePipe (AbstractDistributionElement, AbstractNetworkFeature)
  - Graphical
    * NetworkGraph
    * FeatureGraph

    * InterFeatureLink
    * InteriorFeatureLink
    * Node
  - Additional
    * ExteriorMaterial
    * LiquidMedium

Again, the topological relationships are determined with a TopologyBuilder transformer. The topologically significant nodes (point geometries) and edges (curve geometries) are output. These features, together with the point geometries in the manholes ESRI shapefile, form the basis for the rest of the translation.

The output edges are subsequently mapped to the following features:

- Pipes (RoundPipes, RectangularPipes or OtherShapePipes)
- InteriorFeatureLinks as part of the FeatureGraph of the Pipe

The output nodes are used to write the following features:

- Nodes as part of the FeatureGraph of the Pipe
- Nodes as part of the FeatureGraph of the SimpleFunctionalElement (A sewer pipe always starts and end at a manhole)
- InterFeatureLinks
  - between the exterior nodes as part of the different FeatureGraphs of Pipes
  - between the exterior node as part of the FeatureGraph of a SimpleFunctionalElement and the exterior node as part of the FeatureGraph of a Pipe
- CityFurniture (the manhole cover is represented as a single node, draped on a Digital Terrain Model (DTM))

The Creator transformer is used to write the Network and NetworkGraph features.

5.6.3 Linking below ground utility network features and above ground city furniture objects

For both the electricity and sewer network, the relationship between the below ground utility network features and above ground city objects are created similarly. In case of the electricity network, the TerminalElement represents the below ground utility network feature. In case of the sewer network, the SimpleFunctionalElement (below ground part of the manhole) represents the below ground utility network feature. In both cases, the utility network feature is duplicated and assigned a unique identifier. Subsequently, the 'connectedCityObject' attribute is assigned to the utility network feature. A Uniform Resource Identifier (URI), as attribute value of the connectedCityObject, is used to make a one-to-one reference to the above ground city object. A URI reference is defined as an optional choice between an absolute or relative URI, followed by object identifier that consists of a crosshatch ("#") and additional reference information (the identifier of the above ground city object). The crosshatch followed by the CityFurniture object identifier is used to refer the TerminalElement or SimpleFunctionalElement to the CityFurniture object (Figure 5.9).

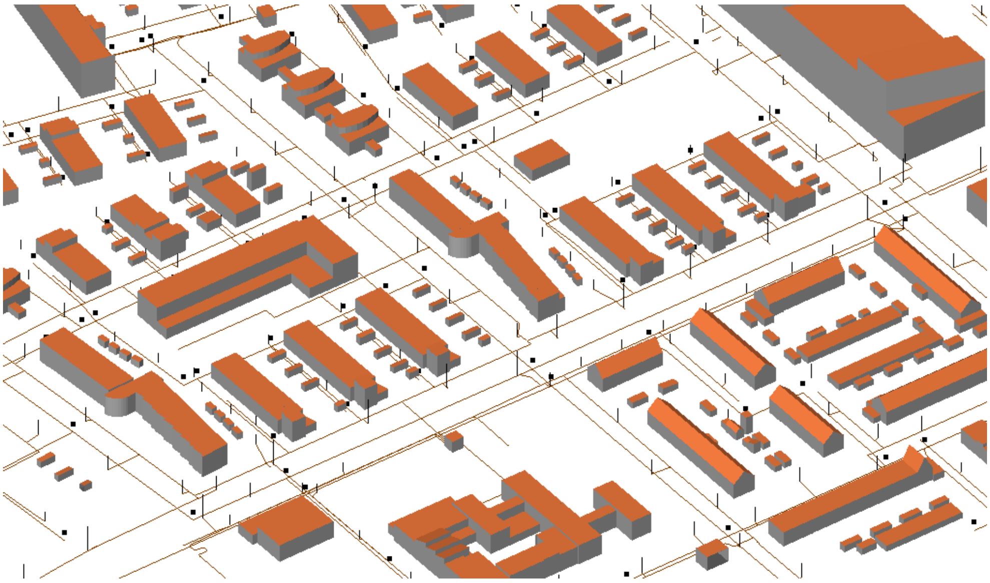


Figure 5.10: 3D view of both utility networks and city objects

```
<utility:component>
  <utility:TerminalElement gml:id="UUID_44c9797e-4259-42be-99bb-4fddf85e11b3">
    <utility:connectedCityObject>#UUID_011b78d3-7fef-408a-9a7b-8ffd378ab42a
    </utility:connectedCityObject>
    <utility:topoGraph xlink:href="#UUID_40c9b3fd-f66d-4023-8a37-6c014de015b9"/>
    <utility:lod1Geometry>
      <gml:Point srsName="epsg:28992" srsDimension="3">
        <gml:pos>83191.2 431661.31 3.34601202869416</gml:pos>
      </gml:Point>
    </utility:lod1Geometry>
    <utility:class>streetLight</utility:class>
  </utility:TerminalElement>
</utility:component>
```

Figure 5.9: URI as connectedCityObject attribute value to refer a TerminalElement to a CityFurniture object

5.6.4 Visualization of the data in CityGML format

Yet, the support of data in the core CityGML format is limited. Especially when it comes to CityGML data including ADE's such as the Utility Network ADE. Only a few gis applications are able to read and visualize the data, e.g. the FME Data Inspector or FZKViewer can be used [KIT: Karlsruhe Institute of Technology, 2017]. These applications allow for quick inspections of the data in 3D (Figure 5.10). Moreover, it is possible to export the data to another data format if desired. Further possibilities, such as GIS analyses are, however, limited. Having the CityGML data put into a relational database allows the user to do more operations. Section 5.7 elaborates on the mapping of the CityGML Utility Network ADE onto a relational database.

5.7 MAPPING THE CITYGML CORE AND UTILITY NETWORKS ADE ONTO A RELATIONAL DATABASE

First, the core CityGML model is mapped onto a relational PostgreSQL/PostGIS database by means of 3DCityDB. The SQL scripts that come with the installation, are used to set up the database [Chair of Geoinformatics, Technical University of

Munich et al., 2017]. Subsequently, the database is extended by running the SQL scripts shipped with the 3DCityDB extension for the Utility Network ADE [Agugiaro, 2017]. Figure 5.11 is a screen dump of one of the many 3DCityDB tables in pgAdmin.

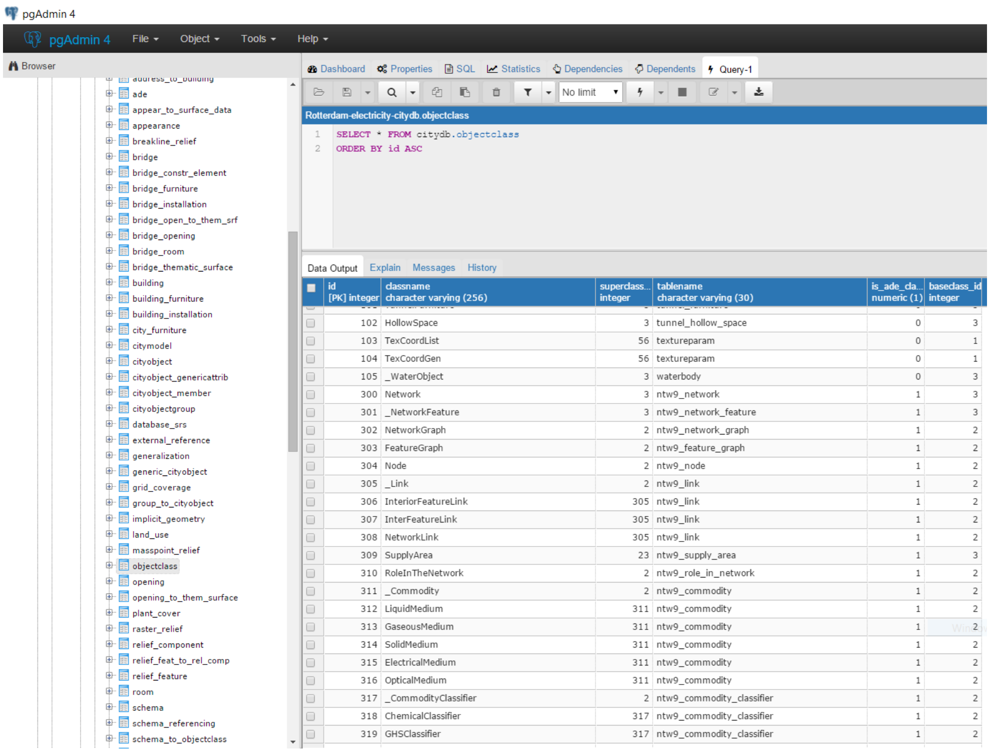


Figure 5.11: 3DCityDB tables

5.8 IMPORTING THE CITYGML DATA INTO THE RELATIONAL DATABASE

The modeled data is inserted into the tables of the created relational database using FME. The different feature types of the resulting CityGML file are used as readers. As pointed out in Section 4.3, the order of importing is a task that must be handled with care. Parallel importing of all feature types into the tables in a single workspace might cause an error due to referencing to not existing id's. Therefore, a command line batch file is created that runs the different workspaces, used to populate the different tables with the CityGML data, after each other (Figure 5.12).

```
:: Created by Xander den Duijn 2017-10-19
:: Last edited by Xander 2018-01-08

:: step 1
fme cityobject.fmw --SourceDataset_CITYGML sewer.gml

:: step 2
fme city_furniture.fmw --SourceDataset_CITYGML sewer.gml

:: step 3
fme utn9_network.fmw --SourceDataset_CITYGML sewer.gml

:: step 4
fme utn9_network_graph.fmw --SourceDataset_CITYGML sewer.gml

:: step 5
fme utn9_network_feature.fmw --SourceDataset_CITYGML sewer.gml

:: step 6
fme utn9_distrib_element.fmw --SourceDataset_CITYGML sewer.gml

:: step 7
fme utn9_feature_graph.fmw --SourceDataset_CITYGML sewer.gml

:: step 8
fme utn9_node.fmw --SourceDataset_CITYGML sewer.gml

:: step 9
fme utn9_link.fmw --SourceDataset_CITYGML sewer.gml
```

Figure 5.12: Populating the 3DCityDB tables by running the FME workspaces after each other in a single batch file

By connecting to the designed PostgreSQL/PostGIS database in a GIS application such as QGIS or ArcGIS it is possible to view the stored data. Visualization in ArcGIS is preferred since the data can be viewed in 3D (Figure 5.13). Besides just pure visualization of the geometries in the tables, it allows for visualization of several spatial analyses which will be discussed in Chapter 6.

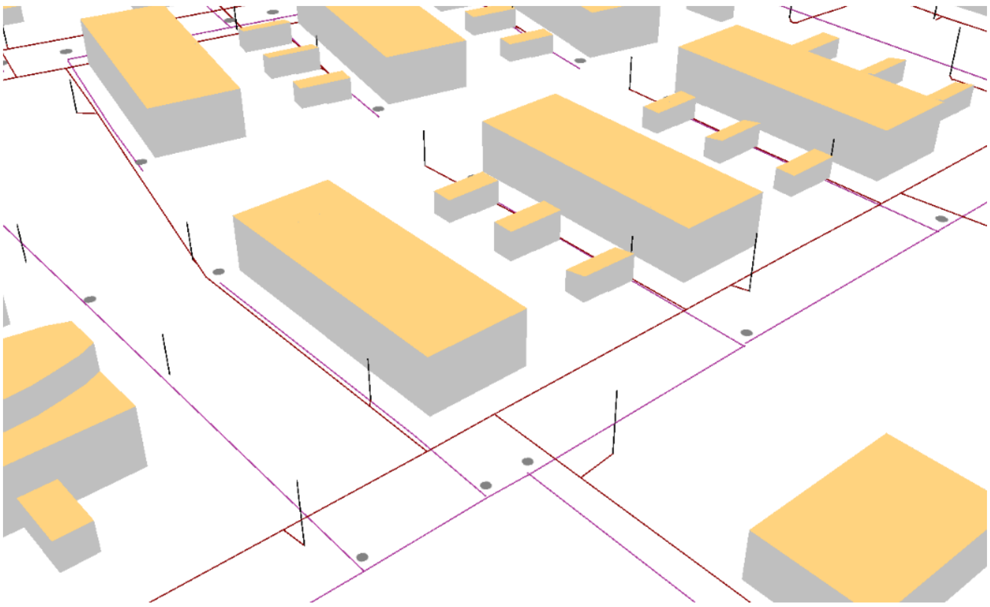


Figure 5.13: Utility networks and related above ground city objects in ArcGIS

6 TESTING AND VALIDATION; QUERYING THE RELATIONAL DATABASE

Several SQL scripts are written in order to conduct (network) analyses on the spatial data in the relational database. This section details two scenarios, one for each type of utility network. Moreover, the queries are used to validate whether the relationships are built properly.

6.1 SCENARIO #1: A UTILITY STRIKE DURING A MAINTENANCE OPERATION

The registered location and depth of the utilities are often not accurate. A trench is commonly dug in order to perform a maintenance operation and acquire more accurate location information. Because the registered location of the below ground utilities are initially not accurate, utility strikes often occur. In order to know the consequences of such an event, several steps have to be taken which are detailed in this section.

What utility network features are affected by a trench?

A trench is a linear excavation perpendicular to the utility lines on the map. Due this, and its length, there is a high chance of finding the utility line. Figure 6.1 illustrates this with an example in the electricity network dataset.

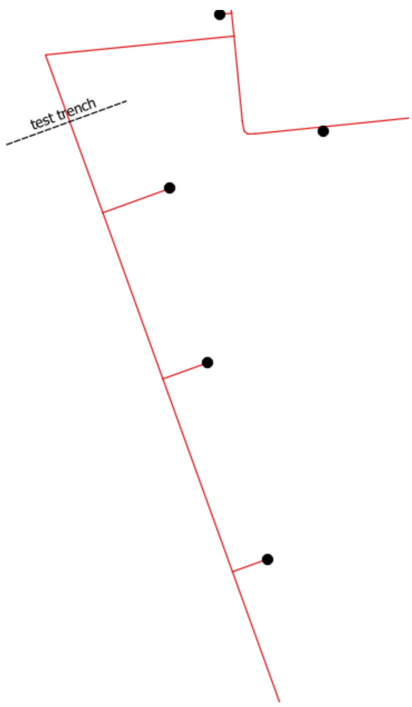


Figure 6.1: A trench, as 2D line, in the electricity network dataset

The query in Figure 6.2 checks with what geometries in the utn9_network_feature table the trench, represented as a 2D line, intersects.

```
SELECT intersect_check.id, objectclass.classname, intersect_check.st_intersects
FROM (
      SELECT id, objectclass_id, ST_Intersects(ST_GeomFromText('LINESTRING(
      83241.524 431159.165, 83254.870 431164.029)', 28992), geom)
      FROM utn9_network_feature
     ) AS intersect_check
JOIN objectclass ON intersect_check.objectclass_id = objectclass.id
WHERE st_intersects is true
```

Figure 6.2: A SQL query used to find what utility network features intersect with the trench as 2D `LINESTRING`.

After executing the query a single network feature is output (Figure 6.17). By joining the subquery table and the objectclass table, the objectclass of the network feature is found which is in this case a Cable.

id	classname	st_intersects
1036	Cable	True

Figure 6.3: SQL query result; affected network features

Flow direction and affected corresponding nodes

Each utility line is graphically represented by a FeatureGraph. The FeatureGraph of a utility line, for both the electricity and the sewer network, consists of an Interior-FeatureLink and two exterior Nodes (Figure 6.4). TerminalElements are graphically represented as single exterior Nodes.

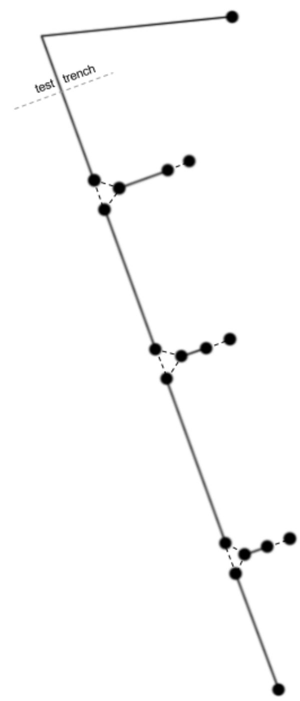


Figure 6.4: Graphical representation of a segment of the electricity network dataset

The ntw_feature_id field of the utn9_feature_graph table refers to the identifier of the utn9_network_feature table. The query below is used to find the corresponding FeatureGraph id's to the network features.

```
SELECT utn9_feature_graph.id
FROM
    (SELECT intersect_check.id
    FROM (
          SELECT id, objectclass_id, ST_Intersects(ST_GeomFromText('LINESTRING(
          83241.524 431159.165, 83254.870 431164.029)', 28992), geom)
          FROM utn9_network_feature
         ) AS intersect_check
    JOIN objectclass ON intersect_check.objectclass_id = objectclass.id
    WHERE st_intersects IS true) AS ntw_feature_ids
JOIN utn9_feature_graph ON ntw_feature_ids.id = utn9_feature_graph.id;
```

Figure 6.5: A SQL query used to find the affected FeatureGraph

The utn9_node table contains the feat_graph_id field that refers to the identifier of the FeatureGraph. The affected corresponding nodes can be found by extending the query in Figure 6.5 as follows:

```
SELECT id, objectclass_id, gmlid, type, feat_graph_id, point_geom
FROM utn9_node
WHERE feat_graph_id IN (
    SELECT utn9_feature_graph.id
    FROM
        (SELECT intersect_check.id
        FROM (
              SELECT id, objectclass_id, ST_Intersects(ST_GeomFromText('LINESTRING(
              83241.524 431159.165, 83254.870 431164.029)', 28992), geom)
              FROM utn9_network_feature
             ) AS intersect_check
    JOIN objectclass ON intersect_check.objectclass_id = objectclass.id
    WHERE st_intersects IS true) AS ntw_feature_ids
JOIN utn9_feature_graph ON ntw_feature_ids.id = utn9_feature_graph.id);
```

Figure 6.6: A SQL query used to find the affected corresponding nodes

The affected corresponding nodes are found after executing the query:

id	objectclass	gmlid	type	feat_graph_id	point_geom
1648	304	UUID_2df ...	exterior	1036	01010000A0 ...
1647	304	UUID_44c ...	exterior	1036	01010000A0 ...

Figure 6.7: SQL query result; found affected corresponding nodes

In a standard sewer network water flows from higher to lower elevations. In case of the electricity network, an expert needs to determine the flow direction of the current. This is not modeled in the current approach.

Graph connectivity

A graph is connected when there is a path between every pair of vertices. In a connected graph, there are no unreachable vertices. A graph that is not connected is disconnected. A graph is said to be disconnected if there exist two nodes in the graph such that no path has those nodes as endpoints (Figure 6.8).

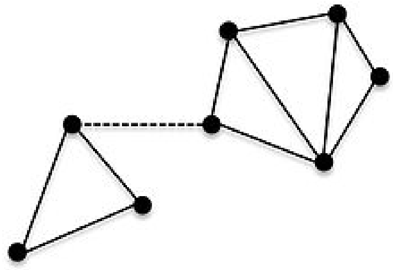


Figure 6.8: Illustration of a graph that becomes disconnected when the dashed edge is removed. Image from [WikiMedia Commons, 2015]

When the flow direction is known it is checked what network features can be reached from the affected, marked dead, node. Assuming the commodity flows through each network feature one after the other (a series circuit), the reached network features, and city objects, are marked dead as well.

Often the utility network is modeled in such a way that the initially marked dead network feature can be reached from still functioning network features. This can be the other node of the affected FeatureGraph, or features coming from an transformer substation that feeds the network. In this case the network graph is not disconnected, since each network feature, and city object, can still be reached.

To check whether there exists a path between the affected nodes of the FeatureGraph, pgRouting is used. The pgRouting library extends the PostGIS/PostgreSQL geospatial database to provide geospatial routing functionality [pgRouting Community, 2017].

The `WHERE` clause in the subquery ensures restricted access to the 'broken' link' in the network, the one affected by the dug trench (see Figure 6.10).

```
SELECT sum(cost) AS sum_cost, node as node_id
FROM pgr_dijkstra(
    'SELECT id, start_node_id::int4 AS source, end_node_id::int4 AS target,
    cost::double precision
    FROM utn9_link
    WHERE id != 2440',
    1647, 1648, false)
GROUP BY node_id;
```

Figure 6.9: A SQL query used to check the connectivity between two nodes, with restricted access to the broken link

The 'sum_cost' table defines the number of links it passes before reaching the specified network feature. This includes both InteriorFeatureLinks and InterFeatureLinks. In case nothing it output, no path exists between the nodes and thus the graph is disconnected. Only when e.g. a transformer station is connected to the disconnected graph, the streetlight would work. If this is not the case we can assume that all network features that are part of the disconnected graph are not functioning either.

The features that are part of the disconnected graph, the ones 'in reach', are found by means of pgRouting as well. Figure 6.10 depicts the use of pgRouting to find the features in reach.

```
SELECT sum(cost) AS sum_cost, end_vid as node_id
FROM pgr_dijkstra(
    'SELECT id, start_node_id::int4 AS source, end_node_id::int4 AS target, cost::double precision
    FROM utn9_link
    WHERE id != 2440',
    1647, array[1,2,3...], false)
GROUP BY node_id
ORDER BY sum_cost ASC;
```

Figure 6.10: A SQL query used to find the nodes in reach

The query in Figure 6.10 assumes that the node with id number 1647 is the very first broken affected node, and thus start node. From this node on, the query checks what nodes can be reached.

sum_cost	node_id
1	2046
1	1650
2	2045
2	1649
3	2158
3	1652
3	142
4	2157
4	1651
5	1654
5	151
5	2228
6	2227
6	1653
7	159

Figure 6.11: SQL query result; reached nodes

In this case, 15 nodes can be reached (Figure 6.11). These nodes can either represent exterior nodes as part of the FeatureGraph representing the Cable or the TerminalElement. Each TerminalElement has a one-to-one relationship with a streetlight. Note that InteriorFeatureLinks as well as the InterFeatureLinks are included in the computation of the sum_cost. The query in Figure 6.11 is extended in order to find what network features, Cables or TerminalElements, and related city objects are affected (Figure 6.12).

```
SELECT ntw_feature_id, classname, cityobject_id, lod1_other_geom
FROM(
    SELECT ntw_features.id as ntw_feature_id, classname, cityobject_id
    FROM(
        SELECT id, objectclass_id, cityobject_id
        FROM(
            SELECT utn9_feature_graph.ntw_feature_id
            FROM(
                SELECT feat_graph_id
                FROM(
                    SELECT sum(cost) AS sum_cost, end_vid as node_id
                    FROM pgr_dijkstra(
                        'SELECT id, start_node_id::int4 AS source,
                        end_node_id::int4 AS target, cost::double precision
                        FROM utn9_link
                        WHERE id != 2440',
                        1647, array[1,2,3...], false)
                    GROUP BY node_id
                    ORDER BY sum_cost ASC) as nodes_in_reach
                JOIN utn9_node ON nodes_in_reach.node_id = utn9_node.id
                GROUP BY feat_graph_id) as feat_graph_ids
            JOIN utn9_feature_graph ON feat_graph_ids.feat_graph_id =
            utn9_feature_graph.id) as ntw_feature_ids
        JOIN utn9_network_feature ON ntw_feature_ids.ntw_feature_id =
        utn9_network_feature.id) as ntw_features
    JOIN objectclass ON ntw_features.objectclass_id = objectclass.id)
    as terminalelement_ids
LEFT OUTER JOIN city_furniture ON terminalelement_ids.cityobject_id =
city_furniture.id
```

Figure 6.12: A SQL query used to find the affected network features and related city furniture objects

The graphical explain plan provides a view of the query and illustrates the large number of table joins (Figure 6.13).

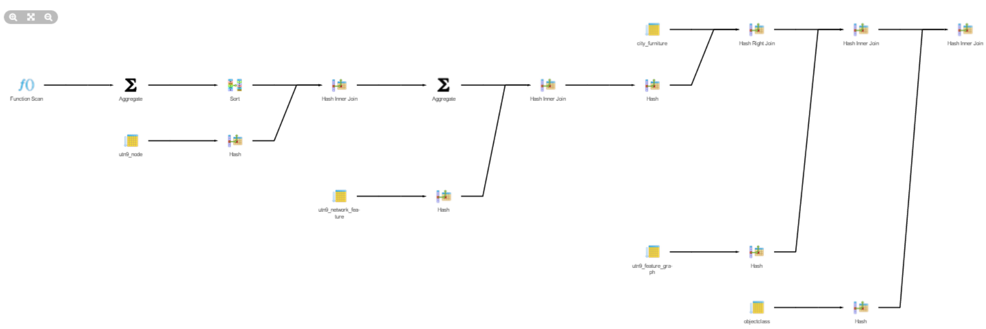


Figure 6.13: Graphical explanation of the query in Figure 6.12

Figure 6.14 depicts the tabular output.

id	classname	cityobject_id	lod1_other_geom
142	TerminalElement	1560	01020000A040...
151	TerminalElement	1569	01020000A040...
159	TerminalElement	1577	01020000A040...
1037	Cable	null	null
1038	Cable	null	null
1039	Cable	null	null
1235	Cable	null	null
1291	Cable	null	null
1326	Cable	null	null

Figure 6.14: SQL query result; affected network features and related city furniture objects

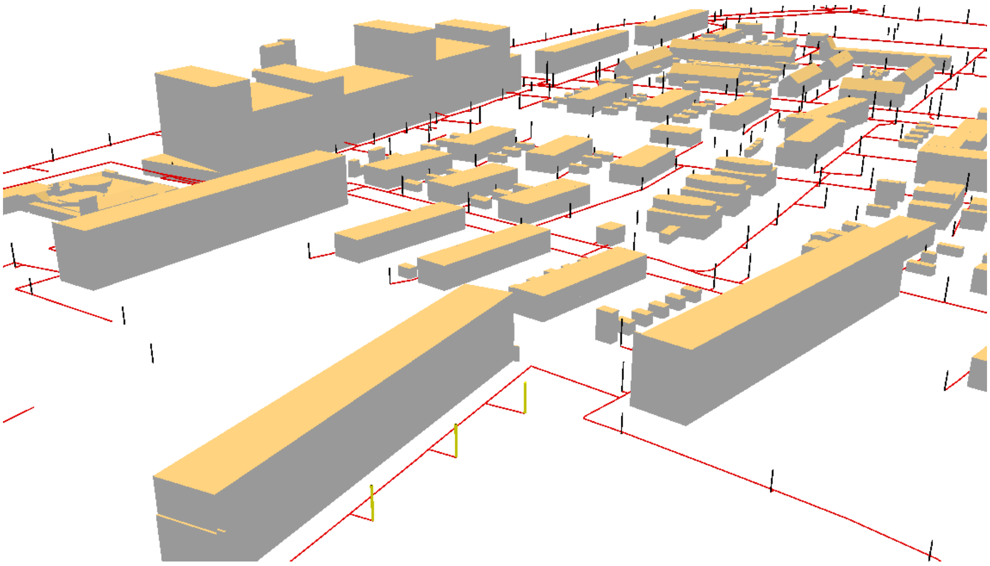


Figure 6.15: Affected streetlights visualized in ArcGIS

A broken parent streetlight

In Section 3.2.1 it is explained that one of the streetlights in a group with mutually connected streetlights has a parent role. This streetlight is responsible for the functioning of the other streetlights (in series connection). Hence, in case this streetlight is broken, the other streetlights are not functioning either. Figure 6.16 is used to find what streetlights are not functioning when it is known that a parent streetlight is broken (in this case streetlight with id=248)

```
SELECT terminal_element_id, connected_cityobject, cityobjectgroup_id, role, lod1_other_geom
FROM(
    SELECT terminal_element_id, utn9_network_feature.cityobject_id as connected_cityobject,
    cityobjectgroup_id, role
    FROM(
        SELECT cityobject_id as terminal_element_id, cityobjectgroup.cityobjectgroup_id, role
        FROM (
              SELECT cityobjectgroup_id
              FROM group_to_cityobject
              WHERE cityobject_id = 248
             ) as cityobjectgroup
        JOIN group_to_cityobject ON cityobjectgroup.cityobjectgroup_id =
        group_to_cityobject.cityobjectgroup_id) as terminalelements_in_group
    JOIN utn9_network_feature ON terminalelements_in_group.terminal_element_id =
    utn9_network_feature.id) as cityobjects_in_group
JOIN city_furniture ON cityobjects_in_group.connected_cityobject = city_furniture.id;
```

Figure 6.16: A SQL query used to find what streetlights are not functioning as a result of a broken parent streetlight

After executing the query, several features are output, representing the streetlights that are not functioning anymore (Figure 6.17).

cityobject_id	cityobjectgroup_id	role	lod1_other_geom
244	1864	null	01020000A040 . . .
245	1864	null	01020000A040 . . .
246	1864	null	01020000A040 . . .
247	1864	null	01020000A040 . . .
248	1864	parent	01020000A040 . . .

Figure 6.17: SQL query result; malfunctioning streetlights

The above analyses are particularly useful before performing operations on the utility network. In case the operation comes with a high risk, e.g. streetlights in a large part of the city might go off, the planned activities must be handled with extra care.

The shown analyses assume the location of the utility strike is known. Often, it is the other way around. In this case, the consequences of a problem in the network are visible. This makes finding the problem in the network challenging. First, it needs to be checked if the consequences are caused by the same problem. For instance, a group of streetlights is off. In case this group is fed by the same electricity cable, it can be assumed the streetlights are off caused by the same problem.

Subsequently, we check what part of the network is not causing the problem. The part of the network that is used to carry electrical current to still functioning streetlights is not causing the problem and can be disregarded. The part of the network that is not used but does connect to the broken streetlights, is where the problem probably lies (Figure 6.18).

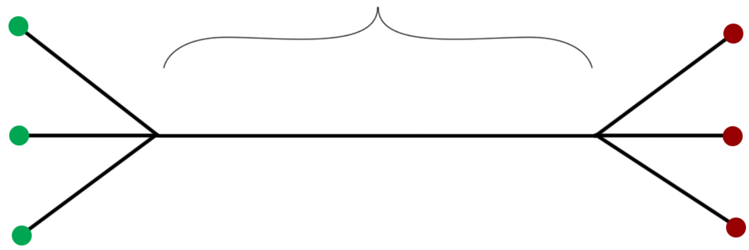


Figure 6.18: Finding the problem in a network only not knowing the location of a utility strike with on the left side the still functioning related objects in green, and on the rights side the not functioning related objects in red

6.2 SCENARIO #2: WATER FLOW IN A STANDARD SEWER SYSTEM

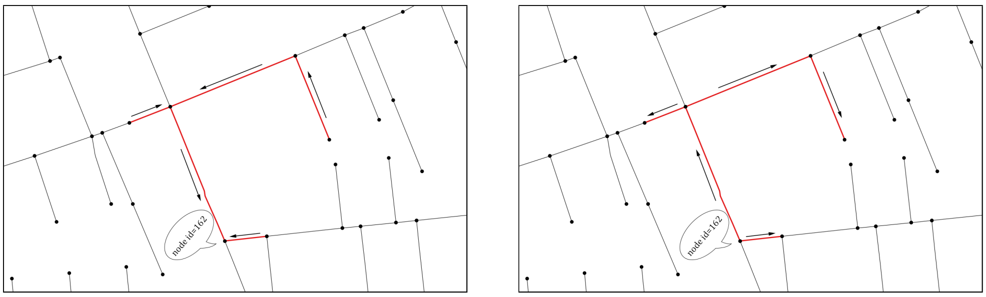
It is important to know what part of the sewer network relies on the functioning of a pumping station. Therefore we need to find out what sewer pipes are used to transport waste water by gravity flow in a sewer sub network. This section details a small experiment that illustrates the relevance of implementing the flow direction in a standard sewer pipe.

The query below checks through what sewer pipes and manholes the water flows before reaching a random node within the network of which the id is known.

```
SELECT ntw_features.id as ntw_feature_id, classname, geom
FROM(
    SELECT id, objectclass_id, geom
    FROM(
        SELECT utn9_feature_graph.ntw_feature_id
        FROM(
            SELECT feat_graph_id
            FROM(
                SELECT sum(cost) AS sum_cost, end_vid as node_id
                FROM pgr_dijkstra(
                    'SELECT id, end_node_id::int4 AS source,
                    start_node_id::int4 AS target,
                    cost::double precision
                    FROM utn9_link',
                    612, array[1,2,3...], true)
                GROUP BY node_id
                ORDER BY sum_cost ASC) as nodes_in_reach
            JOIN utn9_node ON nodes_in_reach.node_id = utn9_node.id
            GROUP BY feat_graph_id) as feat_graph_ids
        JOIN utn9_feature_graph ON feat_graph_ids.feat_graph_id =
        utn9_feature_graph.id) as ntw_feature_ids
    JOIN utn9_network_feature ON ntw_feature_ids.ntw_feature_id =
    utn9_network_feature.id) as ntw_features
JOIN objectclass ON ntw_features.objectclass_id = objectclass.id
```

Figure 6.19: A SQL query used to find what manholes and sewer pipes can reach a node in a directed graph network

The directed parameter in the route query is set to true, since we are working with a directed graph network. However, the source and target columns are swapped. Doing this, reverses the direction and enables you to find what nodes can be reached from the node with id number 612 (Figure 6.20). Actually, the query checks what part of the network can reach this node.



(a) original direction of flow **(b)** reversed direction of flow

Figure 6.20: Swapping the direction of flow in order to find what part of the network can reach a certain node

It is, however, not known whether the node with id number 612 is the final destination or not. Therefore, we need to check what nodes can be reached from this node. In order to do so, the start_node and end_node columns are swapped back in the route query (Figure 6.21).

```
SELECT ntw_features.id as ntw_feature_id, classname, geom
FROM(
    SELECT id, objectclass_id, geom
    FROM(
        SELECT utn9_feature_graph.ntw_feature_id
        FROM(
            SELECT feat_graph_id
            FROM(
                SELECT sum(cost) AS sum_cost, end_vid as node_id
                FROM pgr_dijkstra(
                    'SELECT id, start_node_id::int4 AS source,
                    end_node_id::int4 AS target,
                    cost::double precision
                    FROM utn9_link',
                    612, array[1,2,3...], true)
                GROUP BY node_id
                ORDER BY sum_cost ASC) as nodes_in_reach
            JOIN utn9_node ON nodes_in_reach.node_id = utn9_node.id
            GROUP BY feat_graph_id) as feat_graph_ids
        JOIN utn9_feature_graph ON feat_graph_ids.feat_graph_id =
        utn9_feature_graph.id) as ntw_feature_ids
    JOIN utn9_network_feature ON ntw_feature_ids.ntw_feature_id =
    utn9_network_feature.id) as ntw_features
JOIN objectclass ON ntw_features.objectclass_id = objectclass.id
```

Figure 6.21: A SQL query used to find what manholes and sewer pipes can be reached from a certain node

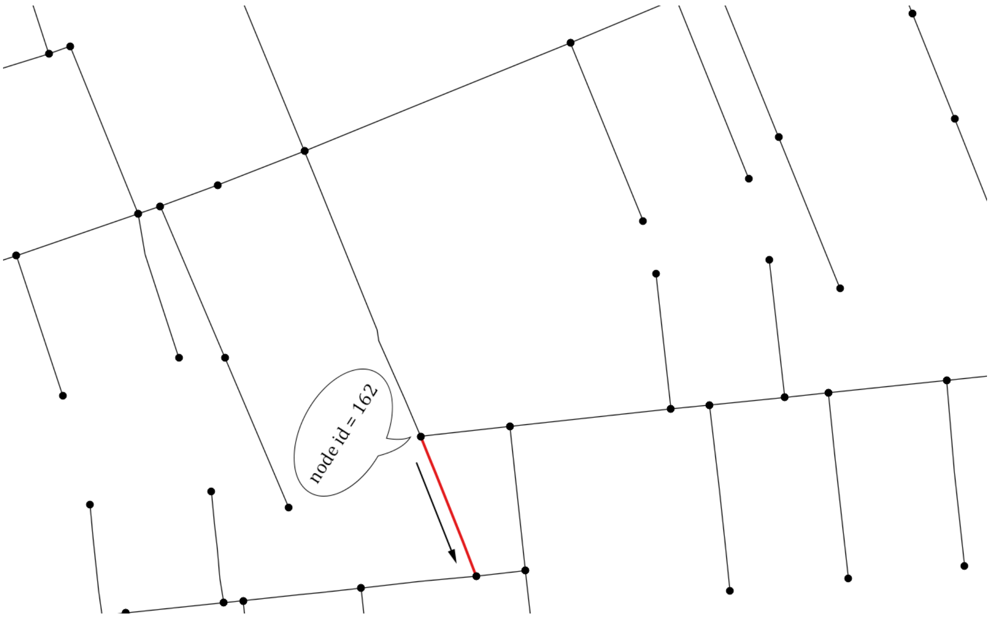


Figure 6.22: Water flow from a node in a directed graph network

Combining the results of the queries in Figure 6.19 and Figure 6.21 shows what network part is used to transport waste water by gravity flow (Figure 6.23).

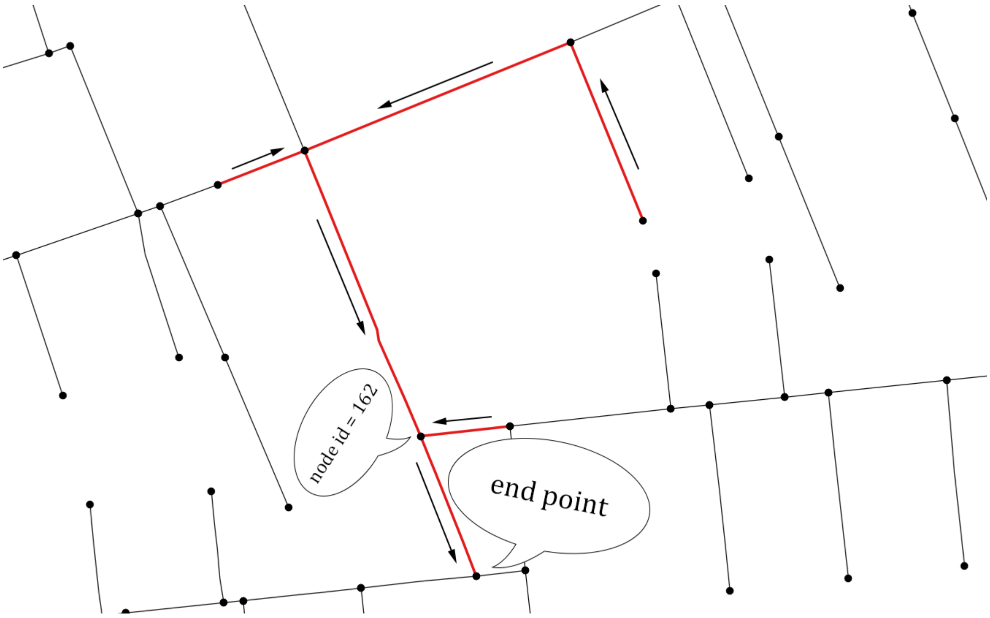


Figure 6.23: Network part used to transport waste water by gravity flow

The results of such analyses on directed network graph networks relies on the 'correctness' of the edge directions. In this case, the direction of flow is based on the z-values of the nodes which mark the start and end of the sewer pipe. In case a wrong z-value is assigned to a certain node and, as a result, the direction of flow is wrong, the analysis becomes useless. The results of the queries show a sewer network consisting of 6 sewer pipes while, normally, such networks are bigger. Possibly, some z-values of the source data are incorrect or the start/end attribute of a node is wrong due e.g. rounding errors during processing of the data.

7 CONCLUSIONS AND FUTURE WORK

The objective of this thesis was to propose a suitable data modeling approach for below ground utility network features and related above ground city objects. First, this section will answer the defined sub-research questions, where after a conclusion will be drawn by answering the main research question. Secondly, the future work and recommendations will be presented. Lastly, this section gives a short substantiated explanation to account for the results of the research in the graduation phase.

7.1 RESEARCH QUESTIONS

In this section the sub-research questions, proposed in Section 1.4, will be answered. They form the basis for the answer to the main research question.

- *What dependencies and relationships between below and above ground utility network features are of importance for the municipality of Rotterdam in order to facilitate asset management?*

  As stated in Section 5.3, asset management can be defined as an integrated framework for the maintenance of the assets, above and below ground, in public space. In order to spend the maintenance budget as wisely as possible and better decision making, an optimum balance is needed between costs, performance and risk. Important relations are the ones from hierarchical high level assets e.g. relations with pumping stations and transformer substations. The sewer network relies on the functioning of a pumping station. Transformer substations are responsible for the supply of electrical current to the lower level electricity network. These high level relationships are the ones that could cause harm or damage to humans, property, or the environment in case of malfunctioning or missing. A missing relation to/from such high level assets could lead to negative consequence on large scale. However, the number of high level relations in a network is low and are commonly carefully monitored.

  Still the relationships between utility networks and hierarchical low level assets, e.g. relations with streetlights, are of importance. These relationships occur more often and are more likely to be affected. Moreover, there is a need for the modeling of such relationships since it allows for detailed network operations.

  In Chapter 6 the relationships between the utility network features and city objects are validated by querying the relational database. It is shown that without much effort it is possible to check what features are affected in case of a 'broken' link/relationship. Streetlights are used as an example, but other city objects of interest could be modeled and found in a similar way. In addition, the importance of relating below ground utility network features is shown by finding what part of the sewer network is used for transporting waste water in a standard sewer system.

- *To what extent can the current state of an existing utility network data model such as the CityGML Utility Network ADE fulfill the needs of the municipality of Rotterdam?*

  Section 3.4 concludes that the CityGML Utility Network ADE is the most promising data model for the purpose of this research. CityGML is the international OGC standard open data model and XML-based format for representing and exchanging semantic 3D city models. This research is limited to the mapping of electricity and sewer network data. Both utility networks can be very complex in reality, but are modeled in a rather low LoD. For instance, a manhole can integrate more components such as valves but is mapped to a single node in the current approach.

  The municipality of Rotterdam particularly has a need for a data model that allows for relationships and hierarchies of 3D city objects, above and below ground. Chapter 4 and Chapter 5 have shown that the Utility Network ADE is suitable for relating utility network features as well as utility network features and above ground 3D city objects. In this research only one-to-one relationships between below ground utility network features and above ground city objects are created. One-to-many, many-to-one or many-to-many relationships are not considered. These type of relationships could occur in case e.g. the below ground utility network is represented according a different level of abstraction as the above ground city objects. For instance, a single utility line that represents multiple utility lines could relate to multiple city objects. The one-to-one relationship between the below ground utility network feature and the above ground city object is modeled by means of a XLink. A disadvantage of the XLink topology is that navigation between related objects can only be performed in one direction, not bidirectional. As a result, a below ground utility network feature can be related to an above ground city object, but not the other way around.

- *Is the proposed data modeling approach suitable for implementation of existing utility network data and city objects in 3D city models?*

  Chapter 5 details the creation of CityGML Utility Network ADE data that proves that the proposed data modeling approach is suitable for implementation of existing utility network data and 3D city objects. Having existing utility network data and city objects as input, a file in CityGML Utility Network ADE format is created. The manipulation of the data structure and content, according the proposed data modeling approach, presented in Chapter 4, is completed in FME. FME is a useful tool since it allows for writing CityGML Utility Network ADE data.

  Building the network topology is one of the very first steps in the developed FME workspace. The success of the implementation strongly relies on the quality of the input data. In this research most experiments are conducted with the data provided by the municipality of Rotterdam. Generally, the Rotterdam data is appropriate since the input vector data is topologically 'clean'. Building the topology, as one of the first steps in the implementation, would be greatly hampered when the input data is not clean, e.g. under- or overshoots appear. In that case the data must be cleaned up first which can be either done by a GIS analysis or manually by an expert.

  The developed FME workspace is rather complex and is particularly designed for the Rotterdam data in vector file format. Small adjustment to the FME workspace are expected when working with data in a similar vector file format. The FME workspace requires bigger adjustments in case of working with unclean data, different structured data and/or modeling of feature classes from other modules is desired.

- *Which mapping methods are required to derive a relational database from the designed data model?*

  Mapping the object-oriented CityGML data model to a relational database may follow one of three different mapping methods:

  - mapping all classes to a single table
  - mapping non-abstract classes to their own tables
  - mapping each class to a single table

  The criteria for the most suitable mapping method are 1) the expected number of tuples within the tables, 2) the number of joins to reconstruct the objects and 3) the overall complexity of the most important queries (with respect to the CityGML object structure). A mixture of mapping methods is used in this research by means of the 3DCityDB.

  Generally, most abstract classes are mapped to a single table with an additional field used to specify the object class. Additional tables had to be created for establishing the one-to-many, many-to-one or many-to-many relationships between classes. Several specific object class attributes are neglected in order to limit the amount of table fields. Moreover, a large number of object classes, including the used TerminalElements and SimpleFunctionalElements, are grouped and mapped to a single table. As a result, the possible level of detail is limited.

  In the current version of the Utility Networks ADE, these grouped object classes have a limited number of attributes which makes them suitable for mapping to a single table. Electrical terminals and sewer manholes are, however, completely different types of objects. Due generalizing these objects, a more detailed modeling is currently not possible.

  In order to model the different components in more detail, the Utility Network NetworkComponents module should be optimized. Adding classes and attributes for the different components allow for a more detailed mapping to additional tables. A manhole is, for instance, typically used to connect network components. Therefore, the introduction of a ConnectionComponent is suggested, just as for StorageComponents, ControllerComponents and MeasurementsComponents.

- *Can the designed relational database be used to perform essential (spatial) operations?*

  In Chapter 6 examples of relevant (network) analyses are performed by querying the designed relational database. It shows the possibility to simulate 1) what network features are affected by e.g. a utility strike and 2) the water flow in a standard sewer system by means of pgRouting. What specific operations can be performed depends on the modeled data. In this case it is e.g. not possible to check how many households are affected by a utility strike since no relationships between the utility network and houses are modeled. Though, no problems are expected since modeling the relationship between the utility network and a house would be similar as to a streetlight. Newly designed queries would require more relationships, and vice versa.

- *How to visualize the modeled data and (spatial) operations on the designed relational database?*

  Yet, the support of CityGML Utility Network ADE data by viewers and gis applications is limited. Only a few gis applications are able to read and visualize the data, e.g. the FME Data Inspector or FZKViewer can be used. These applications allow for quick inspections of the data in 3D. Having the CityGML data put into a relational database, however, allows the user to visualize the data and to do more operations. Chapter 6 has shown that the relational database allows for performing spatial network analyses. A query on the relational database in a DBMS will, however, always have a tabular output. Plus, it is not possible to select/highlight the affected features by executing the query in a gis application such as ArcGIS. The tabular output can be visualized by executing the query in a DBMS and subsequently adding the result, as a new layer, to a GIS.

Moreover, possibilities exist for visualization and interactive exploration of 3D city models in CityGML through e.g. the 3DCityDB WebClient [Cesium Consortium, 2017]. The team at Chair of Geoinformatics at TUM, is currently working on the development of an enhanced version of the open source 3DCityDB WebClient including ADE's.

Visualizing what objects are located below ground and what objects are located above ground is challenging. Besides the utility networks, only buildings, streetlights and manholes are visualized in the current approach. A DTM and more above ground city objects would help in understanding the subdivision between objects below and above ground.

7.2 CONCLUSION

In conclusion to the responses of the sub-research questions a answer can be given to the main research question:

How to efficiently model below ground utility networks and related above ground city objects in order to facilitate integrated asset management?

Based on a comparison between several existing utility network data models, the CityGML Utility Network ADE turned out to be the most promising data model. In contrast with other data models, the Utility Network ADE is capable of relating utility network features as well as utility network features and above ground 3D city objects which is a key requirement in this research. The data model allows for topographical representation as well as a graphical representation used to ensure topology.

The suitability of the CityGML Utility Network ADE is proved by the implementation of existing electricity and sewer network data. However, several other different types of utility networks exist and therefore further research on the suitability of the data model is needed.

The Utility Network ADE is part of the well-known matured CityGML standard and therefore allows for implementation into 3D city models. In addition to the topological relationships between below ground network features, the data model allows for one-to-one attribute relationships between below ground network features and above ground city objects. The proposed data modeling approach is successfully examined by implementing relationships between 1) the below ground electricity network and above ground streetlights and 2) between the sewer network and the above ground manhole covers. Further investigation is however needed since several other types of relationships exist but are not considered.

The manipulation of the existing utility network data structure and content is a rather complex translation. The success of the translation strongly relies on the type and quality of the input data. Subsequently, a simplified version of the object-oriented data model is mapped to a relational database. The derived relational database has proven to be suitable for storage and management of the modeled data but in a limited level of detail.

To facilitate asset management, utility services operation and maintenance as well as emergency management and disaster response are of great importance for the municipality of Rotterdam. The designed data model allows for performing several types of (network) operations, which are performed on the database, that support these activities. For instance, the designed data model allows for finding the affected city objects in case of a utility strike and simulating the flow of water in a standard sewer system. Also, knowing the risk of maintenance or construction operations can assist asset management with respect to better decision making. Still more other types of relationships would require new queries, and vice versa.

This research proposed a suitable 3D data modeling approach for below ground utility networks features (viz. electricity and sewer) and related above ground city objects (viz. streetlights and manhole covers) in order to facilitate asset management. The proposed data modeling approach paves the way for future research on the use and implementation of the CityGML Utility Network ADE, as standard data model and as file format. Moreover, the object-oriented CityGML model is successfully mapped to a relational database which has proven to be efficient for storing, management and analyses by the performed (network) operations.

7.3 FUTURE WORK AND RECOMMENDATIONS

This research made one of the first attempts to thoroughly map existing utility network data and city objects to the CityGML Utility Network ADE. Further research could optimize the proposed data modeling approach for better decision making in the field of asset management. Following are the future work:

- Modeling multiple different utility networks and city objects

  This research is limited to the modeling of the Low-Voltage electricity network and the conventional gravity sewer network. Other types of networks, e.g. oil, gas and telecommunication, are not considered. The challenges that come along with the modeling of these other types of networks will enhance the data model. Moreover, these networks have relationships to other types of above ground city objects which are not taken into account in this research.

  An interesting case would be a pumping station. Pumping stations are used to move waste water from one place to another. In addition to the relationship with the (waste) water network, a pumping station relies on the electricity and telecom network.

- Modeling in a higher LoD

  In this research most objects are modeled in a low LoD. A higher LoD allows for a more thorough use of the designed data model. For instance, subsurface mass computation can be conducted more detailed when below ground objects such as pipes and cables are modeled as geometric shapes instead of single lines and nodes.

  Furthermore, there is a need for additional class attributes. Besides round and rectangular shaped pipes, pipes can e.g. be egg-shaped too. At the moment it is not possible to assign a value to the interior width and height of pipes shaped like these. A more detailed modeling of the shape of pipes and cables would e.g. facilitate the determination of available design space.

- Exporting a CityGML file from the relational database

  The 3DCityDB comes with a Importer/Exporter tool but currently only allows exporting from the core 3DCityDB instance. Hence, exporting the utility network features that are stored by means of the Utility Network extension of the 3DCityDB is not supported. This hampers the interoperable exchange of, and access to the data.

- Detailing the CityGML Utility Network classes and use

The state-of-the-art CityGML Utility Network ADE consist of several classes and relationships that are not clearly defined. A thorough documentation of the data model is missing. For this reason, some classes are used based on intuitive decision making in this research. The same goes for the modeling of objects in specific situations, such as the grouped streetlights discussed in Section 4.1.1. The result of using the CityObjectGroup class is satisfying but modeling it that way was cumbersome.

- Better investigating on more types of analyses

  Modeling more types of utility networks and city objects come with new types of relationships. These relationships require newly designed queries.

- Implementing larger datasets

  The proposed data modeling approach is tested on an small area in the city of Rotterdam. Implementation of larger datasets allows for validation and optimization of the designed workspaces and algorithms.

- Implementing datasets with a different accuracy

  The workspaces in FME are initially designed particularly for the Rotterdam data. The success of this translation relies on the quality and type of input data. Implementation of data with a different accuracy, e.g. from Stedin, will facilitate the optimization of the proposed data modeling approach.

- Better investigating on visualization of the data

  The 3DCityDB content can be directly exported and subsequently visualized in a broad range of applications like Google Earth, ArcGIS, and the WebGL-based Cesium Virtual Globe. CityGML data including ADE's such as the Utility Network ADE is however not supported. Further investigation on the support of CityGML Utility Network ADE for visualization in Cesium is needed.

  Moreover, investigation is required on how to clearly distinguish below ground network features from above ground city objects in the visualization.

- Investigating on the modeling of different types of relationships

  Different types of relationships and dependencies between below and above ground utility networks features exist. In this research only directional (from below to above) one-to-one physical relationships, as they exist in the real world, are created. Other types of relationships, many-to-one, one-to-many and/or many-to-many , could occur in case e.g. the below ground utility network is represented according a different level of abstraction as the above ground city objects. Investigation on how to cope with these other types of relationships is part of the future research.

  Many-to-one relationships could be established by using the connectedCityObject attribute, just as with the one-to-one relationships. In this case, many below ground network features are related to the same identifier of a single above ground city object. The other way around, relating a single below ground network feature to multiple above ground objects, is not possible since the connectedCityObject attribute only allows for having a single attribute value. A possible way to establish one-to-many relationships, would be by grouping the above ground city objects into a CityObjectGroup. In this way, the below ground network features can be related to the single above ground CityObjectGroup. The same procedure could be used for many-to-many relationships. The below ground network features can be related to the identifier of the above ground CityObjectGroup.

7.4 REFLECTION

The Geomatics for the Built Environment programme at the TU Delft provides vital spatial knowledge about the built environment. The science Geomatics is concerned with the 1) acquisition, 2) analysis, 3) modeling/management and 4) visualization of geographic data with the aim of gaining knowledge and a better understanding of the built and natural environments.

The proposed data modeling approach presented in this thesis is examined by implementation of 3D data representing two types of utility networks (viz. low-voltage electricity network and standard sewer network) and related above ground city objects (viz. streetlights and manhole covers). The different numbered aspects Geomatics concerns perfectly align with the steps conducted taken in the thesis research. When working with geographic data it is import to know something about the acquisition method used. Several methods of gathering utility network data exist (Section 3.1). The geographic data used in the thesis research is mostly acquired through GPS measurements which come with a varying accuracy. The analysis, modeling, storing and management of the geographic data is described in Chapter 4 and Chapter 5. After a little pre-processing, the data is mapped to the CityGML Utility Network ADE. The translation in which the data structure and content is manipulated, outputs a single CityGML file. The data is subsequently imported into a relational database which is created by means of 3DCityDB. A database allows for efficient storage and management of the CityGML data. Moreover, it ensures interoperable data access and detailed (network) analyses. After having the data imported into the relational database, the data can be viewed in a GIS. Besides just visualization, it is possible to visualize the results of several network analyses.

To be more specific, the Geomatics courses on Python Programming, GIS and Cartography, Geo-DBMS, 3D Modeling and Geodatasets (GEO3001, GEO1002, GEO1006, GEO1004, GEO1008) in particular proved to be very useful.

BIBLIOGRAPHY

Agugiaro, G. (2017). 3D City Database extension for the CityGML Utility Network ADE. https://github.com/gioagu/3dcitydb_ade/tree/master/03_utility_network_ade. Accessed: 2017-10-30.

Becker, T., Bartels, M., Hahne, M., Hempel, L., and Lieb, R. (2012a). Cascading effects and interorganisational crisis management of critical infrastructure operators. Findings of a research project. *on Geo-information for Disaster Management–Best Practices*, page 105.

Becker, T., Nagel, C., and Kolbe, T. H. (2011). *Integrated 3D modeling of multi-utility networks and their interdependencies for critical infrastructure analysis*, pages 1–20. Springer.

Becker, T., Nagel, C., and Kolbe, T. H. (2012b). CityGML UtilityNetworkADE. http://www.citygmlwiki.org/index.php/CityGML_UtilityNetworkADE. Accessed: 2017-06-10.

Becker, T., Nagel, C., and Kolbe, T. H. (2013). *Semantic 3D modeling of multi-utility networks in cities for analysis and 3D visualization*, pages 41–62. Springer.

Benner, J., Geiger, A., and Leinemann, K. (2005). Flexible generation of semantic 3D building models. In *Proceedings of the 1st international workshop on next generation 3D city models, Bonn*, pages 17–22.

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., and Çöltekin, A. (2015a). Applications of 3D city models: State of the art review. *ISPRS International Journal of Geo-Information*, 4(4):2842–2889.

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., and Çöltekin, A. (2015b). Applications of 3D city models: State of the art review. *ISPRS International Journal of Geo-Information*, 4(4):2842–2889.

Bitenc, M., Dahlberg, K., Doner, F., van Goor, B., Lin, K., Yin, Y., Yuan, X., and Zlatanova, S. (2008). Utility registration: Slovenia, China, Sweden and Turkey. *GISt report*, 49:48.

Cesium Consortium (2017). 3DCityDB WebClient. https://www.iai.kit.edu/1302.phphttps://cesiumjs.org/demos/3DCityDB/. Accessed: 2017-11-29.

Chair of Geoinformatics, Technical University of Munich, virtualcitySYSTEMS GmbH, and M.O.S.S. Computer Grafik Systeme GmbH (2017). 3D City Database. https://github.com/3dcitydb/3dcitydb/tree/master/Documentation. Accessed: 2017-11-16.

CityGML (2017). CityGML Homepage. https://www.citygml.org/. Accessed: 2017-06-10.

den Duijn, X. (2017). CityGML-Utility-Network-ADE. https://github.com/XanderdenDuijn/CityGML-Utility-Network-ADE.

Du, Y., Zlatanova, S., and Liu, X. (2006). Management and 3D visualisation of pipeline networks using DBMS and AEC software. In *Proceedings of the ISPRS commission IV symposium on geospatial databases for sustainable development*, pages 27–30.

Döner, F., Thompson, R., Stoter, J., Lemmen, C., Ploeger, H., van Oosterom, P., and Zlatanova, S. (2011). Solutions for 4D cadastre–with a case study on utility networks. *International journal of geographical information science*, 25(7):1173–1189.

Emgard, K. and Zlatanova, S. (2007). Design of an integrated 3D information model. *Urban and regional data management: UDMS annual*, pages 143–156.

Emgård, L. and Zlatanova, S. (2008). *Implementation alternatives for an integrated 3D Information Model*, pages 313–329. Springer.

ESRI (2016). The Road Ahead for Network Management.

Gröger, G., Kolbe, T., Nagel, C., and Häfele, K. (2012). OGC city geography markup language (CityGML) encoding standard, Version 2.0, OGC doc no. 12-019. *Open Geospatial Consortium*.

Halfawy, M. R. (2010). Municipal information models and federated software architecture for implementing integrated infrastructure management environments. *Automation in Construction*, 19(4):433–446.

Hijazi, I., Ehlers, M., Zlatanova, S., Becker, T., and van Berlo, L. (2011). *Initial investigations for modeling interior Utilities within 3D Geo Context: Transforming IFC-interior utility to CityGML/UtilityNetworkADE*, pages 95–113. Springer.

Hijazi, I., Ehlers, M., Zlatanova, S., and Isikdag, U. (2009). IFC to CityGML transformation framework for geo-analysis: a water utility network case. In *4th International Workshop on 3D Geo-Information, 4-5 November 2009, Ghent, Belgium*.

Hijazi, I., Kutzner, T., and Kolbe, T. H. (2017). Use Cases and their Requirements on the Semantic Modeling of 3D Supply and Disposal Networks.

Hijazi, I. H., Ehlers, M., and Zlatanova, S. (2012). NIBU: a new approach to representing and analysing interior utility networks within 3D geo-information systems. *International Journal of Digital Earth*, 5(1):22–42.

INSPIRE Thematic Working Group Utility and Governmental Services (2013). Data Specification on Utility and Government Services – Technical Guidelines. http://inspire.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_US_v3.0.pdf. Accessed: 2017-06-10.

ISO 16739 (2016). Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries.

Jaw, S. W. and Hashim, M. (2013). Locational accuracy of underground utility mapping using ground penetrating radar. *Tunnelling and Underground Space Technology*, 35:20–29.

Kadaster (2017). KLIC (WION). https://www.kadaster.nl/klic-wion. Accessed: 2017-09-27.

KIT: Karlsruhe Institute of Technology (2017). FZKViewer. https://www.iai.kit.edu/1302.php. Accessed: 2017-10-10.

Kolbe, T. H. (2009). *Representing and exchanging 3D city models with CityGML*, pages 15–31. Springer.

Kremers, E. A. (2013). *Modelling and simulation of electrical energy systems through a complex systems approach using agent-based models*. KIT Scientific Publishing.

Kutzner, T. and Kolbe, T. H. (2016). Extending Semantic 3D City Models by Supply and Disposal Networks for Analysing the Urban Supply Situation.

McMorran, A. W. (2007). An introduction to iec 61970-301 61968-11: The common information model. *University of Strathclyde*, 93:124.

Municipality of Rotterdam (2017). Slim beheer - assetmanagement in Rotterdam. Accessed: 2017-09-27.

OGC (2016). PipelineML. http://www.pipelineml.org/. Accessed: 2017-06-10.

olde Scholtenhuis, L., Zlatanova, S., and den Duijn, X. (2017). 3D approach for representing uncertainties of underground utility data.

pgRouting Community (2017). pgRouting Project - Open Source Routing Library. http://pgrouting.org/. Accessed: 2017-06-17.

Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., and Lorensen, W. E. (1991). *Object-oriented modeling and design*, volume 199. Prentice-hall Englewood Cliffs, NJ.

Semm, S., Becker, T., and Kolbe, T. H. (2012). Simultaneous Visualization of Different Utility Networks for Disaster Management. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1:159–164.

Stadler, A., Nagel, C., König, G., and Kolbe, T. H. (2009). *Making Interoperability Persistent: A 3D Geo Database Based on CityGML*, pages 175–192. Springer Berlin Heidelberg, Berlin, Heidelberg.

Stedin (2017). Stedin. https://www.stedin.net/. Accessed: 2017-09-27.

van den Brink, L., Janssen, P., and Quak, W. (2016). IMKL2015 - Dataspecificatie Utiliteitsnetten. Report.

Van Oosterom, P., Stoter, J., Quak, W., and Zlatanova, S. (2002). *The balance between geometry and topology*, pages 209–224. Springer.

WikiMedia Commons (2015). Illustration of a graph that becomes disconnected when the dashed edge is removed. https://commons.wikimedia.org/wiki/File:Sample-graph.jpg. Accessed: 2017-11-01.

Yao, Z. and Kolbe, T. H. (2017). Dynamically Extending Spatial Databases to support CityGML Application Domain Extensions using Graph Transformations.

Zlatanova, S. (2006). 3D geometries in spatial DBMS. *Innovations in 3D geo information systems*, pages 1–14.

Zlatanova, S. and Stoter, J. (2006). *The role of DBMS in the new generation GIS architecture*, pages 155–180. Springer.

Zlatanova, S., Van Berlo, L., Van den Brink, L., Reuvers, M., and Stoter, J. (2011). 3D Pilot: Eindrapport werkgroep 3D Standaard NL. *NCG Nederlandse Commissie voor Geodesie Netherlands Geodetic Commission 53*.

A | CITYGML UTILITY NETWORK ADE

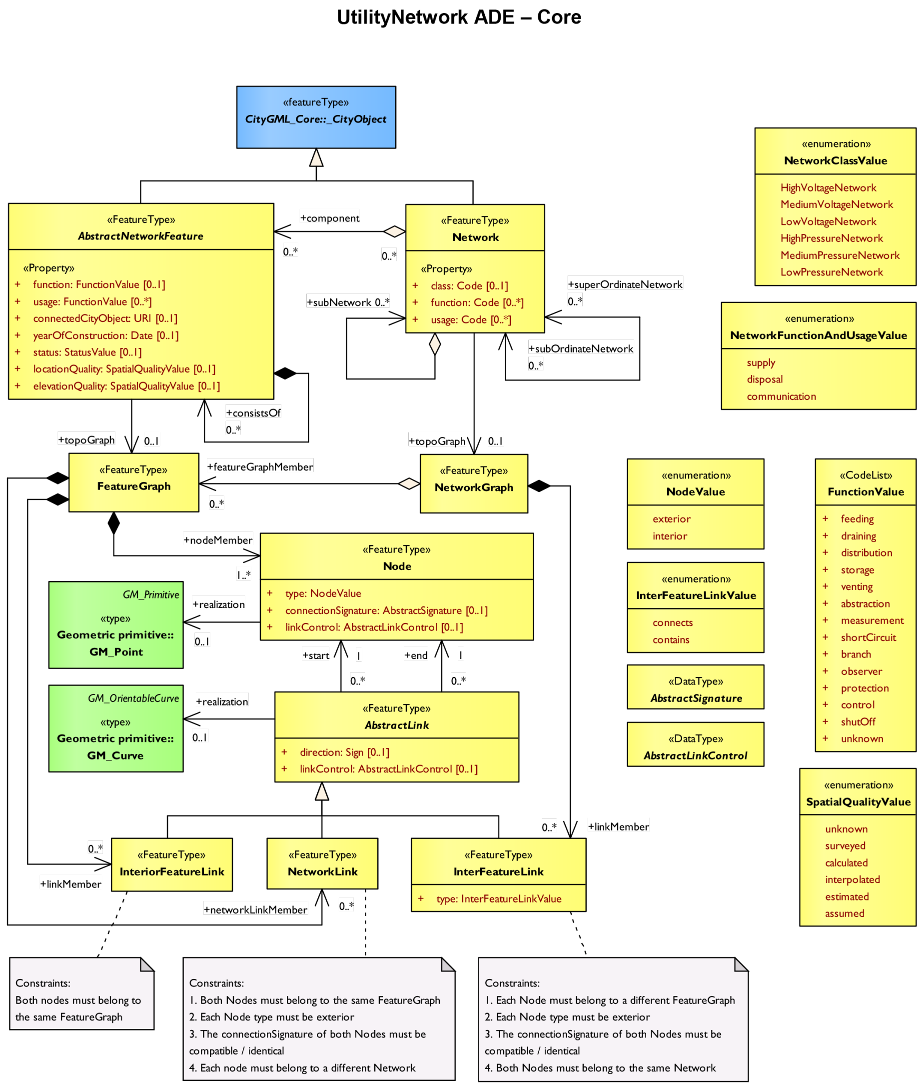


Figure A.1: UML diagram for UtilityNetwork ADE - Core

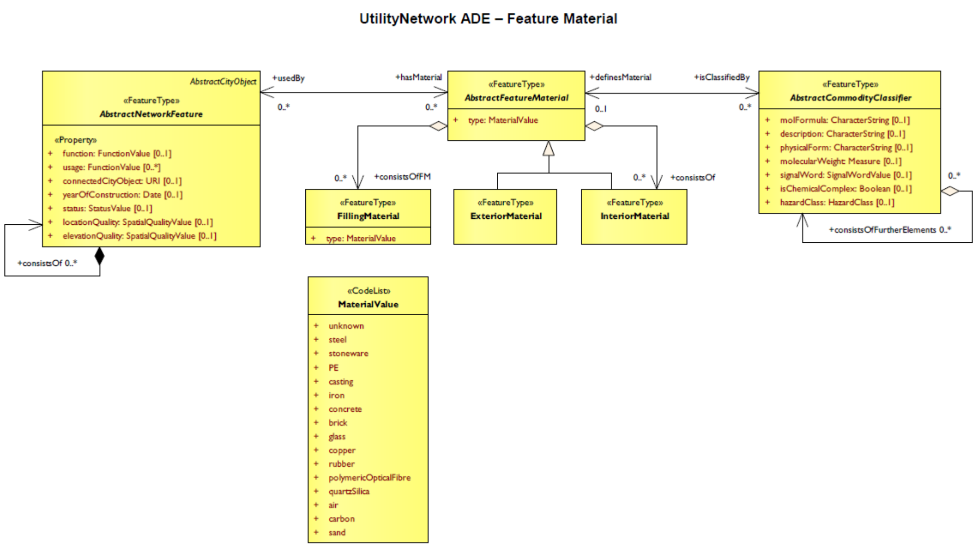


Figure A.2: UML diagram for UtilityNetwork ADE - Feature Material

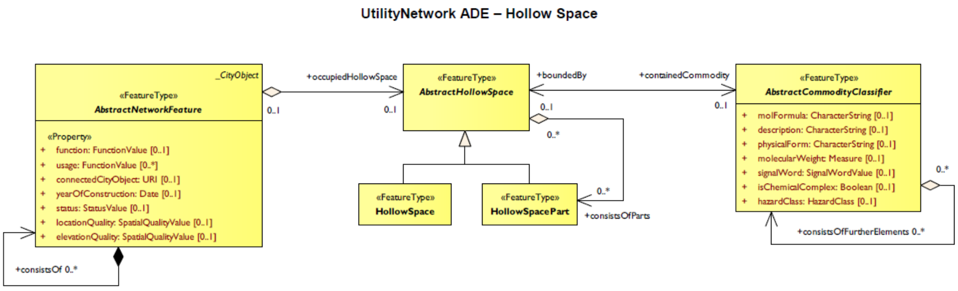


Figure A.3: UML diagram for UtilityNetwork ADE - Hollow Space

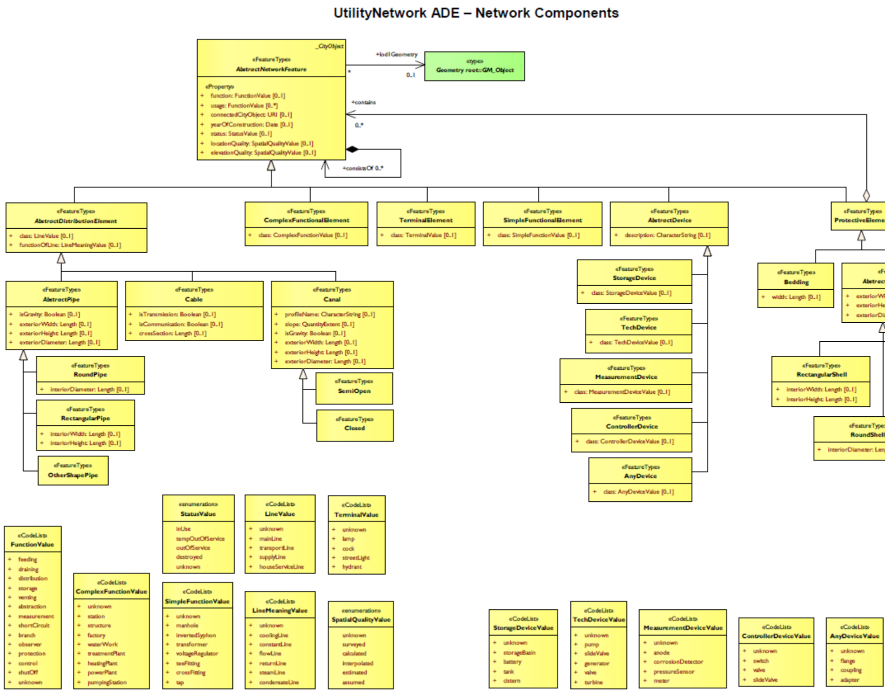


Figure A.4: UML diagram for UtilityNetwork ADE - Network Components

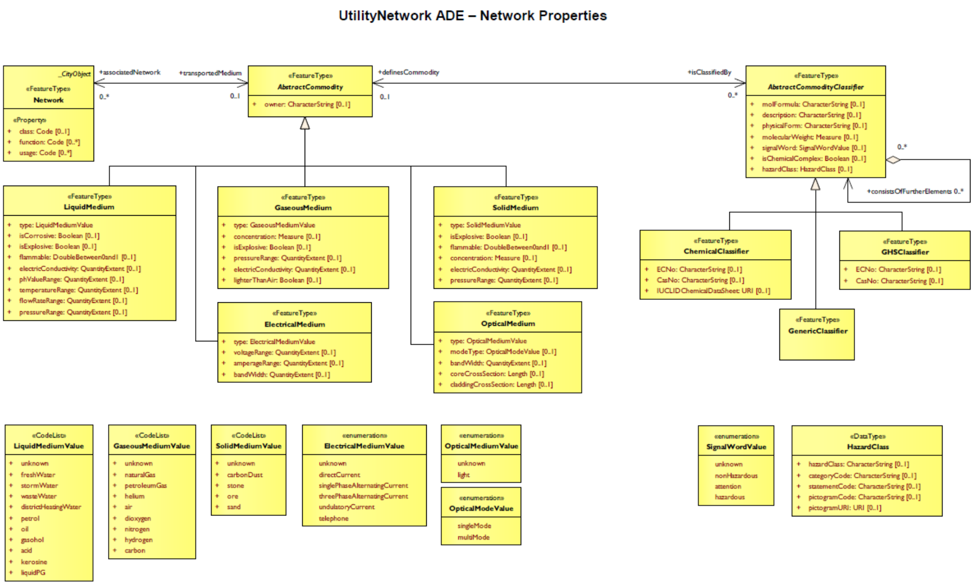


Figure A.5: UML diagram for UtilityNetwork ADE - Network Properties

B UML MAPPING DIAGRAMS

B.1 ELECTRICITY NETWORK MAPPING

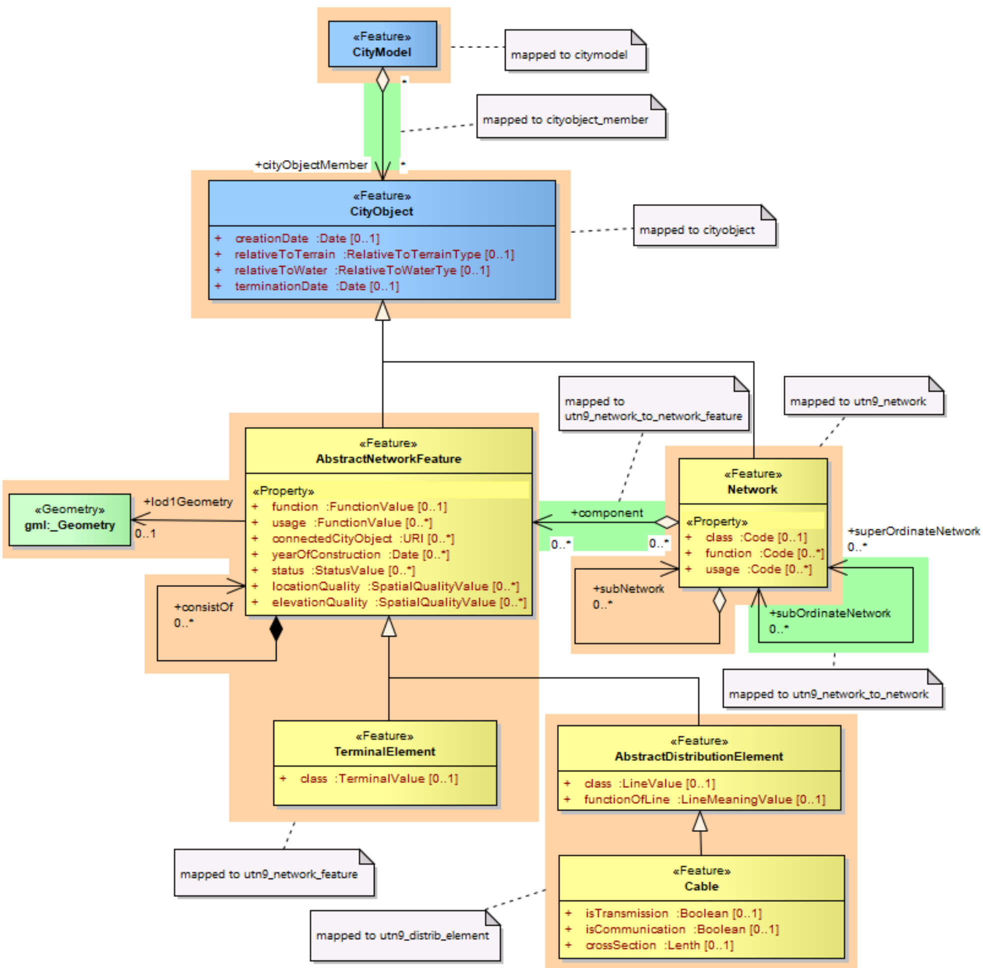


Figure B.1: Mapped classes for the topographical representation of the electricity network

B.2 SEWER NETWORK MAPPING

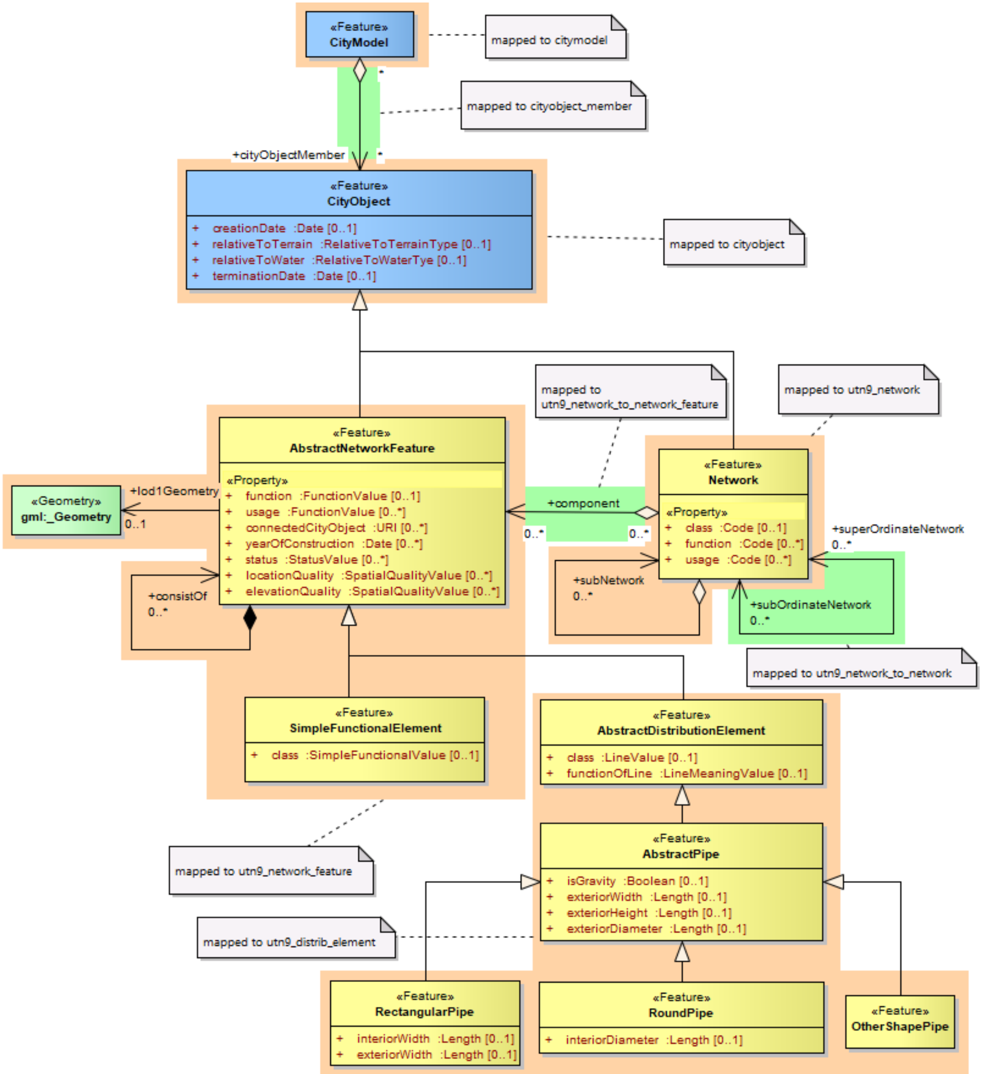


Figure B.2: Mapped classes for the topographical representation of the sewer network

C DATASETS

This chapter details the different datasets that were used for testing and analysis in this thesis. The majority of datasets were provided to me by the municipality of Rotterdam. They originate from the similar geographical location; Hoogvliet in Rotterdam. However, the characteristics of the datasets differ (Table C.1). The Rotterdam Streetlights, Electricity lines, Manholes and Sewer pipes datasets were all delivered as ESRI Shapefiles. Both datasets with line objects, Electricity and Sewer, are in 3D. The Manholes and Streetlights are 2D datasets. These datasets are extracted from the LVZK. The datasets have numerous attributes with detailed information. Moreover, a the city model, created by the municipality of Rotterdam, is used. This CityGML dataset is publicly available.

All datasets are presented in the following sections.

C.1 TECHNICAL DETAILS

Table C.1 presents the technical details of the used datasets provided by the municipality of Rotterdam. It should be mentioned that, for faster processing and visualization purposes, the size of the original datasets is limited to an area of about 4 square kilometers.

	Rotterdam 3D	*Electricity*	*Streetlights*	*Sewer*	*Manholes*
File format	CityGML	ESRI Shapefile	ESRI Shapefile	ESRI Shapefile	ESRI Shapefile
File size	225 MB	65 KB	12 KB	45 KB	7 KB
Object type	Any	3D Lines	2D Points	3D Lines	2D Points
CRS	EPSG:25833	EPSG:28992			

Table C.1: Details of the Rotterdam data

Note that the file size of the Rotterdam 3D dataset decreases to ca. 5 MB when pipes and cables (GenericCityObjects) and trees (SolitaryVegetationObjects) are neglected. This is due the large amount of surface geometries.

In addition to the Rotterdam data, a 2D dataset from Stedin is examined. Table C.2 presents the technical details of the data provided by Stedin. The 'Points of supply' represent the objects that are supplied by electrical current through OV cables.

	LV cables	*OV cables*	*Points of supply*
File format	Esri Shapefile		
File size	43 KB	14 KB	475 KB
Object type	2D Lines	2D Lines	2D Points
CRS	EPSG:28992		

Table C.2: Details of the Stedin data

In contrast with the Rotterdam data, all cables are represented by single electricity lines in the Stedin data. In case several electricity cables are of the same kind and located next to each other, they are mapped to a single line in the Rotterdam data. An attribute value specifies the type and number of cables it represents.

The following sections depict all datasets.

C.1.1 Rotterdam data - Electricity

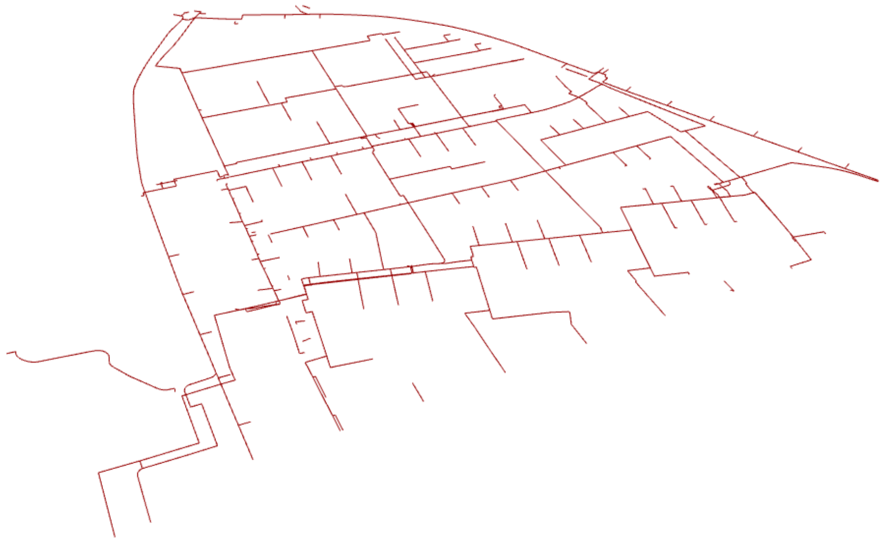


Figure C.1: Rotterdam data; 3D electricity lines

C.1.2 Rotterdam data - Streetlights

Figure C.2: Rotterdam data; Streetlights

C.1.3 Rotterdam data - Sewer

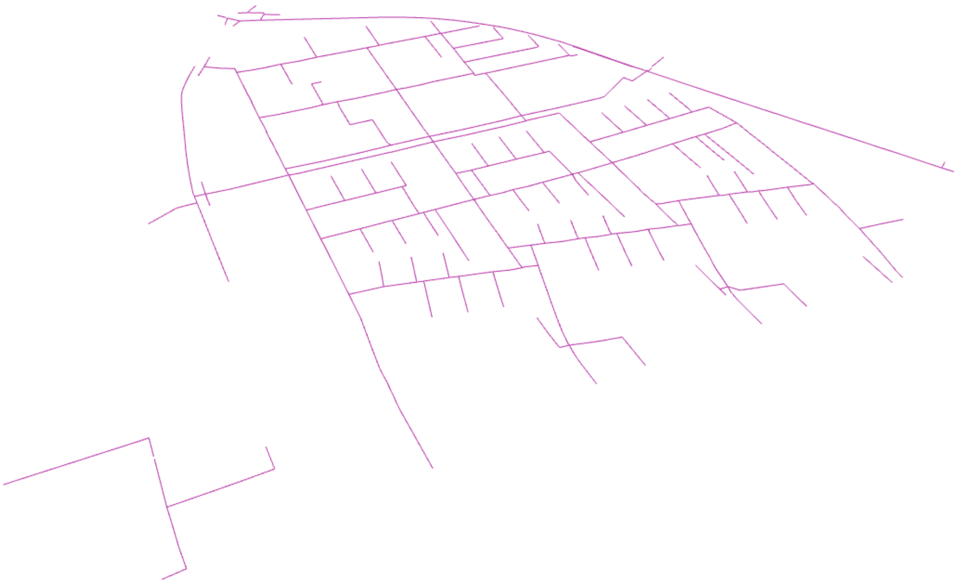


Figure C.3: Rotterdam data; 3D sewer pipes

C.1.4 Rotterdam data - Manholes

Figure C.4: Rotterdam data; Manholes

C.1.5 Rotterdam 3D

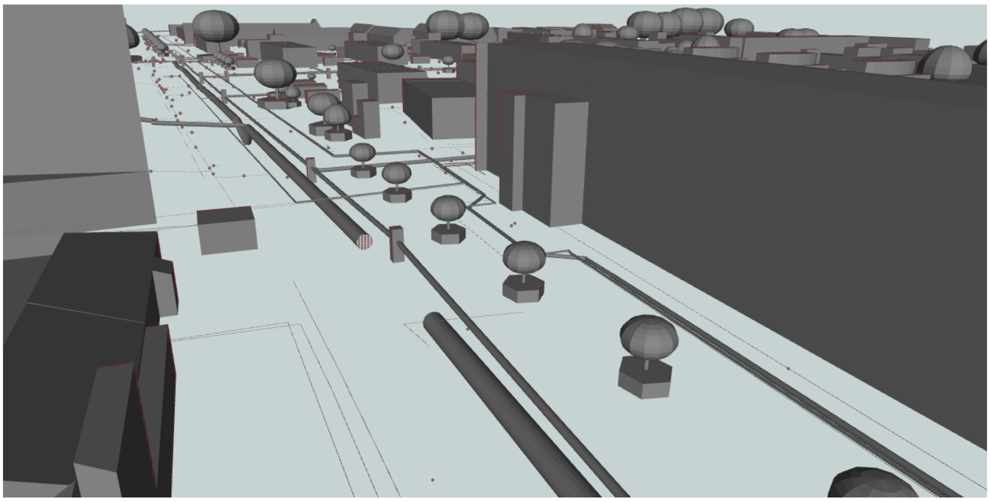


Figure C.5: Rotterdam 3D

C.1.6 Stedin - OV and LV cables

Figure C.6: Stedin; OV and LV cables

C.1.7 Stedin - Points of supply

Figure C.7: Stedin; Points of supply

COLOPHON

This document was typeset using LaTeX. The document layout was generated using the arsclassica package by Lorenzo Pantieri, which is an adaption of the original classicthesis package from André Miede.

Illustrative figures and diagrams were mostly drawn with Google Drawings. Visualizations were mostly made using the FME Data Inspector, FZKViewer or with GIS applications such as QGIS or ArcGIS. BibTeX was used to generate the bibliography.