NETWORK MODELLING AND SEMANTIC 3D CITY MODELS: TESTING THE MATURITY OF THE UTILITY NETWORK ADE FOR CITYGML WITH A WATER NETWORK TEST CASE

I. Boates[1], G. Agugiaro[2], A. Nichersu[1]

[1] EIFER - European Institute For Energy Research, Emmy-Noether-Straße 11, 76131, Karlsruhe, Germany
{isaac.boates, alexandru.nichersu}@eifer.uni-karlsruhe.de
[2] AIT - Austrian Institute of Technology, Smart and Resilient Cities Unit, Giefinggasse 4, 1210 Vienna, Austria
giorgio.agugiaro@ait.ac.at

Commission IV, WG IV/10

KEY WORDS: Utility networks, CityGML, 3D City Database, pgRouting, water network

ABSTRACT:

Recent advances in semantic 3D city modelling and a demand from utility network operators for multi-utility data models integration have contributed to the emergence of an open Application Domain Extension (ADE) of the CityGML data model tailored to multiple types of utility networks. This extension, called the Utility Network ADE, is still in active development. However, work is already well underway to create data samples and to develop methods of modelling thereupon. In this paper, a mapping of the Utility Network ADE data model to a relational database schema is introduced. A sample of a freshwater network using the Utility Network ADE and based on data from the city of Nanaimo, Canada, is also presented. This sample has also been imported into a relational database schema built upon the 3DCityDB (a database implementation of CityGML) extended with a schema of the Utility Network ADE. Further to this, a series of basic network analysis functions have been defined and implemented in SQL to interact with the database so as to carry out sample atomic processes involved in network modelling, such as reading semantic properties of elements, calculating composite physical parameters of the network as a whole, and performing simple topological routing to serve as a guiding example for further and more complex development. A brief outlook is also presented, suggesting areas with high potential for future research and development of this nascent data model.

1. INTRODUCTION

Semantic 3D city models are an extremely powerful tool in smart planning processes, as they offer a coherent and - ideally - omni-comprehensive spatial and semantic representation of all relevant entities (included their properties and dependencies) which characterise a city, both above and belowground. In the context of integrated urban planning, for example, a detailed knowledge not only of the building stock, but also of the distribution networks that supply energy, water, heating, cooling, etc. plays a major role, as they represent the actual link between the "punctual" source of the resources (power plants, water basins, etc.) and the spatially distributed consuming/using points. In general, a number of specific data models and software tools exists to access specific supply networks, but interoperability is indeed still scarce (if not impossible at all) due to a general domain-specific "silo-approach" which hinders de facto the exchange of data and the development of multi-network applications. Some proprietary software solutions exist, indeed, but they rely often on proprietary data formats which are generally proprietary, closed-source, or not well documented. Nevertheless, smart integrated planning decidedly calls (also) for overcoming these interoperability issues. At the lowest level, data interoperability can be facilitated by adopting common/shared data models (and possibly open so as to allow for further development via large communities), upon which more advanced applications and services can be built.

In the context of semantic 3D city modelling, CityGML (Gröger et al, 2012) represents, as of today, the only existing open international standard, allowing for an integrated representation of all urban entities at city scale (buildings, bridges, vegetation, relief, transportation, water bodies, etc. and others via generic object) and the relationships between them. It also defines different standard levels of detail (LoDs) for the 3D objects. This standardised representation of urban objects can be exploited for different applications and purposes, such as simulations, urban data mining, facility management, and thematic inquiries.

A number of software tools already exist that allow work to be carried out with CityGML data. They range from data ETL (extraction, transform, load) tools, database implementations such as the free and open-source 3D City Database (or, in short, 3DCityDB) (Yao et al., 2018), up to web feature services and the possibility to visualise and explore 3D city models on 3D virtual globes platforms such as Google Earth or Cesium. One of the significant advantages of CityGML is the possibility to extend it by means of so-called Application Domain Extensions (ADE): depending on the specific needs, new features or properties can be added, hence greatly augmenting modelling capabilities offered by CityGML. As a matter of fact, several ADEs have been proposed at the time of writing. They aim at assessing specific domains such as urban noise (Kumar et al., 2017), energy (Agugiaro et al., 2018), indoor routing and positioning (Dutta et al., 2017), indoor facilities management (Kim et al., 2014) and cultural heritage (Li et al., 2017). In the case of utility networks the ADE mechanism also plays a significant role. Given that CityGML falls short when it comes to the definition of specific entities and attributes for utility networks, research and development work has been carried out to extend the existing data model. To this extent, the so-called Utility Network ADE is in an advanced stage of development. It offers a generic data model to cope with any utility network both in terms of geographical and topological representations (Kutzner and Kolbe, 2016).

With regards to the Utility Network ADE, this paper presents and discusses work carried out to test the overall state of maturity and suitability of this ADE. For this reason, the data model was implemented as database schema, extending and using the same principles of the already available 3DCityDB. Secondly, real network data from the city of Nanaimo (Canada) was used as test case. This was mapped to the corresponding classes of the Utility Network ADE and imported into the previously mentioned extended 3DCityDB. Finally, some routing functionality was implemented and tested using the PostgreSQL/PostGIS extension pgRouting.

2. BACKGROUND AND RELATED WORK

2.1 Current alternatives for Utility Network Modelling

There exists a multitude of existing methods of modelling various utility networks, but each one tends to focus on a specific network type (water, electric, gas, communication, etc.), and/or a specific geographic scale (building, city, country, etc.). In addition to this, in every alternative there are limited levels of support for all the characteristics that are necessary for formulating a complete model of a heterogeneous utility network system. These deficiencies are summarized by Kutzner and Kolbe (2016) and listed in Table 1.

	INSPIRE Utility Networks	IFC	ArcGIS Utility Networks	SEDRIS	Pipeline ML	CityGML UtilityNetwork ADE
Representation of heterogeneous networks	+	•	•	+	•	++
Dual representation	+	++	+	++	–	++
Topographic/graphic aspects	++	++	++	++	++	++
3D geometries	–	++	–	+	–	+
Functional aspects	–	–	–	–	–	•
Hierarchical modelling						
• networks/ subnetworks	++	–	–	++	–	++
• components/ subcomponents	++	++	•	–	•	++
Interdependencies between						
• network features and city objects	–	•	–	•	–	++
• network features of different network types	–	++	–	–	–	++

– = no support, • = basic support, + = sophisticated support, ++ = comprehensive support

Table 1. Levels of support for various utility network modelling systems. Table source: Kutzner and Kolbe (2016)

An additional review of existing alternatives for utility network modelling is provided by Becker et al. (2012), wherein it was found that a suitable unified model for utility network modelling does not yet exist. Ideally, a unified network model should be able to represent all networks carrying any sort of medium (water, steam, electricity, gas, light, etc.) within the same dataset. It should be able to explicitly model the topography, the topology and the semantic properties of all network elements. Furthermore, the model should be able to extend to arbitrary scales via subordinate and superordinate networks, and allow for the modelling of interdependencies between networks.

2.2 Utility Network ADE Data Model

The Utility Network ADE (Application Domain Extension) is an ambitious attempt to address all of the necessities previously outlined in a ideally omni-comprehensive open data model. By expanding upon the existing CityGML data model, it offers a Geography Markup Language (GML)-based representation of any type of network at any scale, both topographically and topologically. One of the first applications of the Utility Network ADE in a multi-network context was in project

SIMKAS-3D (*Simulation von intersektoriellen Kaskadeneffekten bei Ausfällen von Versorgungsinfrastrukturen*; in English: Simulation of intersecting cascading effects in case of outage of supply networks), wherein the interdependencies of various utility networks in Berlin were modelled (Federal Ministry of Education and Research, 2012). In a more recent example, Kutzner and Kolbe (2016) presented a means of performing supply and disposal modelling using the Utility Network ADE by introducing the concepts of supply-related properties into the data model. To date, however, there remain very few public examples of implementations of the Utility Network ADE in a city simulation/modelling context. An example based on a water network is presented herein to demonstrate advantages of using the Utility Network ADE in the context of city simulation/modelling. Furthermore, a mapping of the data model to a relational database is presented, which has allowed for a demonstration of network-based routing analysis tasks to be performed on the network.

3. METHODOLOGY

3.1 Software and tools used

3.1.1 PostgreSQL with PostGIS and pgRouting: PostgreSQL is an open-source relational database management system (RDBMS). The PostGIS extension enables a PostgreSQL database to store geospatial data, and is a popular choice for building geospatial systems due to its open-source nature, its speed and the fact that it is interfaceable with most GIS programs (e.g. QGIS) for quick visual display of the data. The pgRouting extension (pgRouting, 2018) further enables a PostgreSQL database to perform routing analysis tasks on topological graph structures. Utility networks can be expressed as topological graphs, and therefore pgRouting can be used to analyze their connectivity and find least-cost paths through them. A PostgreSQL database, extended with PostGIS and pgRouting provides all of the functionality required to model a Utility Network ADE network dataset in an RDBMS context, as well as perform functional analyses on it.

3.1.2 3DCityDB: The 3DCityDB is a database schema that represents a near-1:1 mapping of the CityGML data model (3DCityDB, 2018). It is available for both for Oracle Spatial and for PostgreSQL/PostGIS as open-source software, and it represents the reference database implementation of CityGML. Storing a semantic city model in a database allows, for example, for the querying and analysis of the data via structured query language (SQL) commands instead of processing directly CityGML XML files. The Java-based 3DCityDB Importer/Exporter allows to import and export CityGML instance documents to and from the database, but it is currently only able to handle "plain" CityGML (i.e. without ADE features).

3.2 Utility Network ADE Support in the 3DCityDB

In order to better test and exploit the potential of the Utility Network ADE, there has been a growing interest in having a suitable spatial Relational Database Management System (RDBMS) implementation of the Utility Network ADE, building upon and extending the 3DCityDB. As previously mentioned, the current version of the 3DCityDB (and the set of accompanying software tools like the Import/Export tool) do not provide a generic solution for handling CityGML ADE elements in CityGML instance documents. As a matter of fact,

the 3DCityDB development team is currently working to add support to 3DCityDB for any ADE. Planned is also the functionality to automatically perform the mapping between the Object-Oriented model and the Entity-Relationship model. Further details can be read in Yao et al. (2018). In the meanwhile, a manually derived database schema was implemented and is already freely available. The current implementation is based on the Utility Network ADE v. 0.9.2 and it follows the same approach that was defined for the conceptually similar implementation of the Energy ADE. Initial work started internally at the Austrian Institute of Technology (AIT) during version 0.6 of the Energy ADE (Agugiaro, 2016) and was later extended to the Utility Network ADE, too. The 3DCityDB extension for the Utility Network ADE requires the latest version of the 3DCityDB (v. 3.3.2) and is currently implemented for PostgreSQL only. As the goal of this extension is to facilitate direct connection and usage of the 3DCityDB relational database by users and applications programmers, several design and implementation decisions were taken according to the inspiring criteria briefly listed in the following:

a) Define a non-concurrent way of extending the 3DCityDB with other ADEs at the same time;
b) Build upon the existing objects of the 3DCityDB (relations, stored procedures, etc.), but do not break compatibility with the existing tools;
c) Adopt a similar design "style" to the existing 3DCityDB objects when it comes to tables, constraints, naming conventions, data types, etc.;
d) New table and attribute names in the database are kept as close as possible to the original names of the Utility Network ADE as indicated in the UML diagrams;
e) Group and merge multiple classes into one table – whenever meaningful –, in order to keep the database structure as reasonably compact as possible.

All resources of the database implementation of the Utility Network ADE (DDL procedures, documentation, test data, etc.) are already freely available and can be downloaded from GitHub (3DCityDB-ADE, 2018). Before proceeding with the Utility Network ADE itself, a number of changes and additions to the original 3DCityDB were required in order to add generic support for ADEs. Therefore, rules were first defined and then implemented covering the following three main areas:

a) Metadata module to "register" different ADEs;
b) Rules to map ADE classes to new or existing tables;
c) Rules to deal with database stored procedures.

A detailed description of each point is beyond the scope of this paper. A detailed explanation of the rationale behind them, as well as of the concrete implementation solutions can be found in Agugiaro et al. (2018) and in Agugiaro and Holcik (2017). An example of mapping is presented in Figure 1. For intuitive understanding, classes that are (sometimes merged and) mapped to a single table in the relational schema are shown in light orange blocks in the UML diagram. Light green blocks represent association tables in the database used to map many-to-many associations.

3.3 Interacting with the extended 3DCityDB

3.3.1 FME: Den Duijn (2018) successfully converted the sewer and low-voltage electrical networks of Rotterdam, The Netherlands, from its original ESRI shapefile (SHP) format to a representation in a PostgreSQL database with 3DCityDB and Utility Network ADE using SAFE Software's Feature Manipulation Engine (FME) software. The process involved inserting utility network data in a specific order in a series of FME Workbench files. The end result is a dataset containing both networks with functional linkages between the topography, topology and semantical properties of each respective element. It was also demonstrated that streetlights affected by a breakage in the network could be identified using the pgRouting extension by finding which nodes could not be reached on the network graph, and tying this information back to their corresponding topographical features.

3.3.2 Database-side data layer: Despite the adopted design criteria to keep the database implementation of the Utility Network ADE as simple as possible, the database structure can be sometimes complex in terms of CRUD operations (i.e. select, insert, update and delete), especially when dealing with the tables added by the ADE. Data belonging to a specific object can be split and stored in different nested tables. For example, data for a "RoundPipe" object will need to be written to tables CITYOBJECT, UTN9_NETWORK_FEATURE, UTN9_-DISTRIB_ELEMENT, UTN9_NETWORK_TO_NETWORK-_FEATURE respectively, at least as far as the geographical representation is concerned. If topological information is also included, then the additional data will have to be inserted into tables UTN9_LINK and UTN9_NODE. Given the hierarchy existing in the table structure, data have to be inserted (selected, updated, deleted) according to a certain order, in order for referential integrity to hold. The process requires therefore a deep knowledge of the database structure and CRUD operations can lead to complex SQL statements. For this reason, and to facilitate CRUD operations at database level, different strategies were implemented. First of all, stored procedures (or "functions", in the PostgreSQL jargon) to delete ADE objects were implemented following the same logic as those shipped with the 3DCityDB. Additionally, several additional functions for inserting data were implemented, too. They can be divided into two main classes:

- Low-level insert functions, allowing for direct insert operations at single-table level;
- "Smart" insert functions, building upon the previous ones and allowing for insert operation over multiple tables simultaneously, and providing some additional functionality (see SQL examples in Appendix A).

Finally, and building upon the previously mentioned "smart" insert functions, a series of updatable views were also implemented. Views represent a facilitated way to access data, ideally providing one single "virtual" table that hides the complex structure of the actual database. Moreover, if a view is also updatable, entries can be updated, inserted and deleted directly interacting with the view as if it were a normal table. For example, in the case of a "RoundPipe" object, one single view UTN9_NTW_FEAT_DISTRIB_ELEM_PIPE_ROUND merges the previously mentioned tables CITYOBJECT, UTN9_NETWORK_FEATURE, UTN9_DISTRIB_ELEMENT and provides the additional functionality offered by the corresponding "smart" functions. The availability of such stored procedures and updatable views represents therefore a first-level, database-side data layer allowing for easier interaction with the database structure. Additionally, the stored procedures can be embedded and called in routines/functions/code in any programming language (e.g. Python), or by means of existing Extract Transform and Load (ETL) tools.

3.3.3 Python-based data layer: The previously described "smart" insert functions were used to import a subset of a water
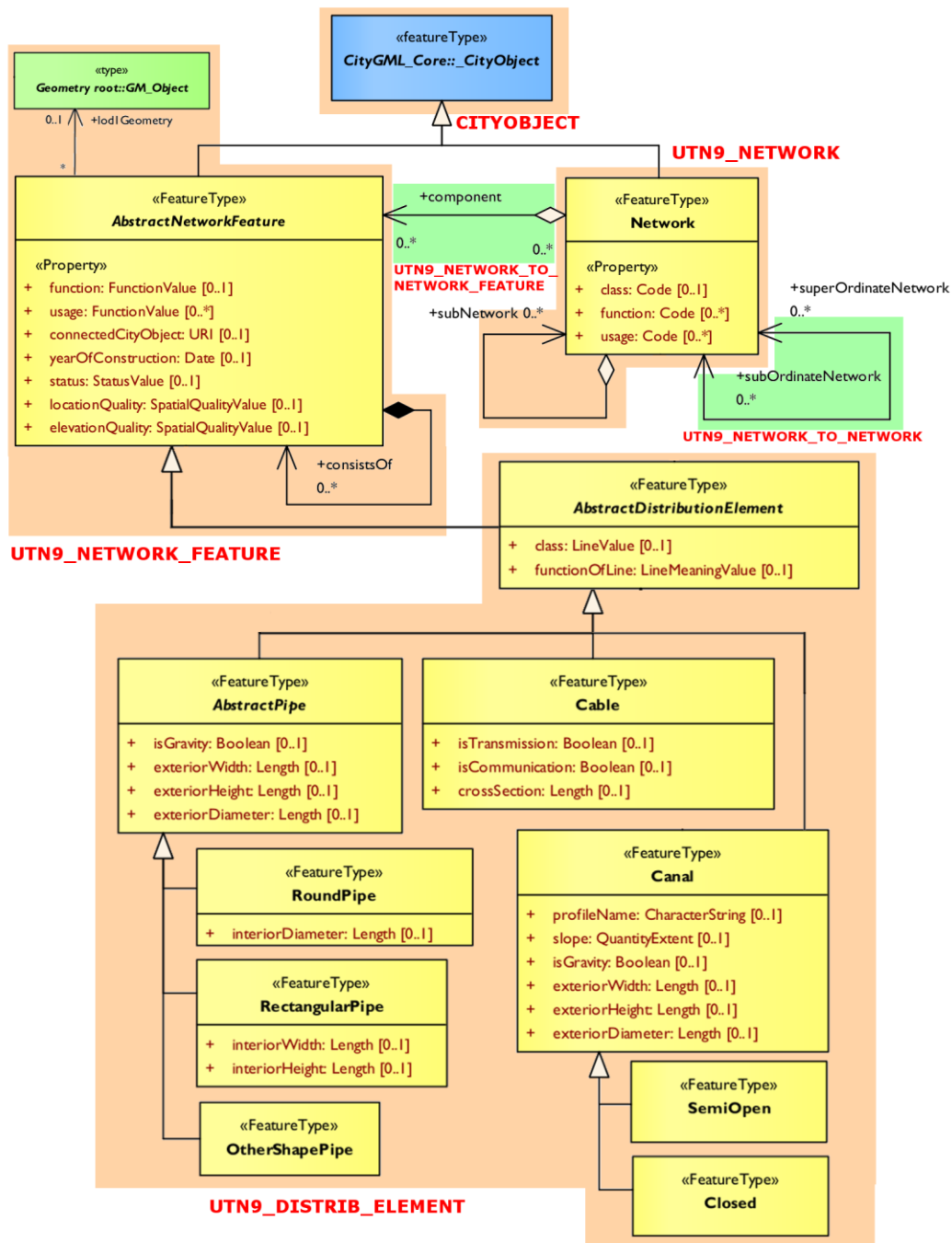
Figure 1. Excerpt of the UML model of the Utility Network ADE. Light orange (for classes) and green (for many-to-many associations) boxes represent the mapping of from the Object-Oriented model to the Entity-Relationship model. The corresponding table names in the 3DCityDB are written in red

network dataset from Nanaimo, Canada, already in CityGML and Utility Network ADE format, into the extended 3DCityDB database. A Python-3 script used the "ElementTree" and "psycopg2" libraries to read the existing CityGML file (ElementTree) and executed previously prepared SQL statements calling the "smart" insert functions (psycopg2). Upon successful import of the data into the 3DCityDB, the database was used, on one hand, to directly visualize the network (e.g. in QGIS or any other visualisation tool), as well as to perform routing analyses, such as counting the number of traversals across pipes when routing from houses to the local reservoir, and selectively excluding some elements based on their status (i.e. if a valve is closed, it cannot be routed through).

3.4 Nanaimo Water Network Case Study

The City of Nanaimo in British Columbia, Canada has released several of its utility networks for download in ESRI shapefile format (Nanaimo, 2018). An FME workbench was created combining the "Water System" (freshwater) dataset, including water pipes, house service lines and various network appurtenances, with a DTM (Digital Terrain Model) of the area to create a 3D dataset in CityGML with Utility Network ADE data. This work has been done to provide a public data sample to assist in further development of the Utility Network ADE data model (Boates, 2017). One advantage of the Utility Network ADE highlighted in this data sample is the explicit definition of topography and topology. In the source data, the

two are inseparable, with topology being defined implicitly by geometry and attributes of the source geographical features. In the Utility Network ADE, the two are linked, but are alternative distinct representations of the same reality. This "dual representation principle" (Becker et al., 2011) allows for simultaneous modelling of two aspects of networks: one regarding the physical reality and the other its topological counter-part. This concept is shown in Figure 2.
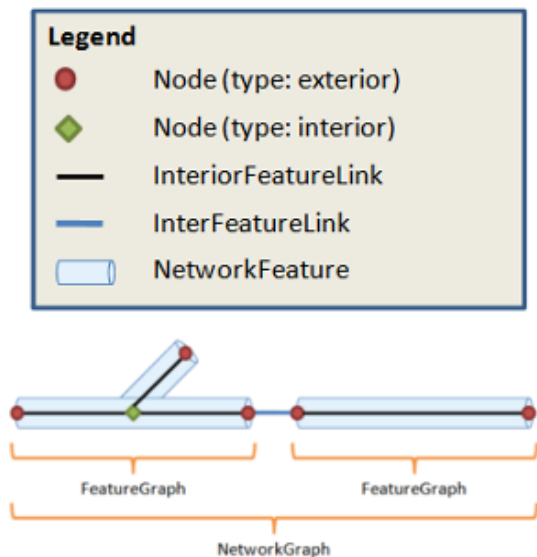

Figure 2. The "NetworkFeature" elements represent the physical reality of the network elements – their actual position, size, length and semantical properties. They are explicitly defined from, but ultimately linked to, the "InteriorFeatureLink" and "Node" elements (themselves contained within "FeatureGraph" elements, which represent a simplified topological view suitable for understanding and modelling functional connectivity of network elements. Figure modified from (Becker et al., 2011)

This advantage is leveraged in the created Nanaimo water pipe network sample. The source data from Nanaimo does not contain any information about physical water pipe features connecting the water network directly to the buildings. The dual representation of topography (Figure 3) and topology (Figure 4) allows for the linkage to be established via a topological link. This in turn gives us the possibility of functional routing analysis and a clear visual representation of topology, while not changing or adding any physical pipe features. In the end, this allows for the representation of topography to remain true to what is known from the source data. (Figure 4) exemplifies this.
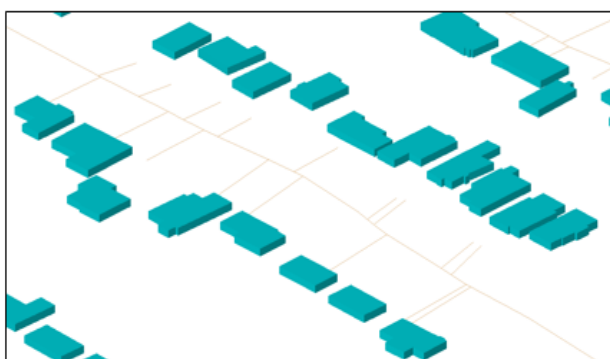

Figure 3. Representation of topography in the Nanaimo water network modelled according to the Utility Network ADE. Figure from "FZK Viewer" software
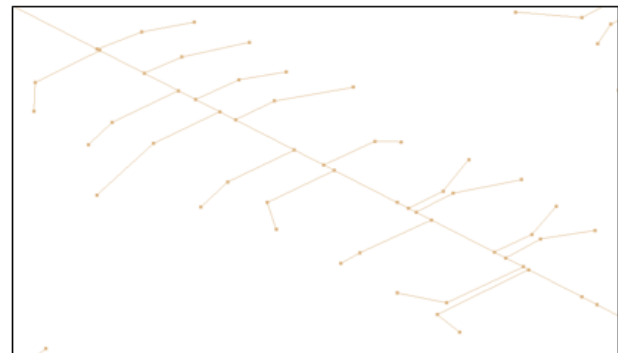

Figure 4. Representation of topology in the Nanaimo water network modelled according to the Utility Network ADE. Figure from "FZK Viewer" software

4. RESULTS

4.1 Nanaimo Water Network in the 3DCityDB

The Nanaimo Water Network Data Sample described in section 3.4 and imported into the 3DCityDB by the Python script mentioned in section 3.3.3 could be visualized in QGIS, as well as used for various routing algorithms with pgRouting. The semantic properties of the network elements are also leveraged by making use of the "status" attribute: only elements with a "status" value of "inUse" can be traversed. This allows the functional properties of the network to be modified simply by changing a status = "inUse" element to be status = "outOfService", which would correspond to a topological disconnection at that element. Therefore, a routing algorithm with a target element on the other side of an element with status = "outOfService" would be forced to find an alternative route, or return no results at all. This effect is exemplified in (Figure 5), where after being imported into the 3DCityDB, a couple of network elements were modified to be of status = "outOfService" in order to test the ability of selective exclusion when performing routing analysis.
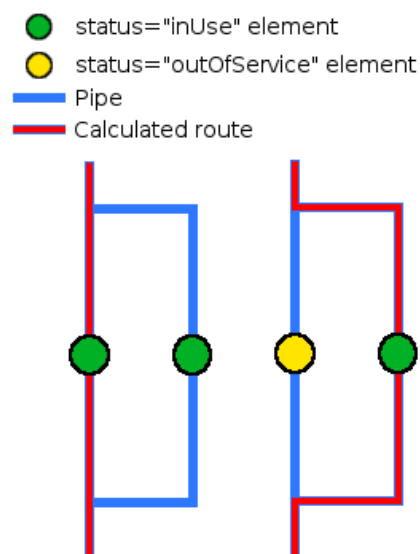

Figure 5. Different routing results obtained when switching an element's "status" attribute
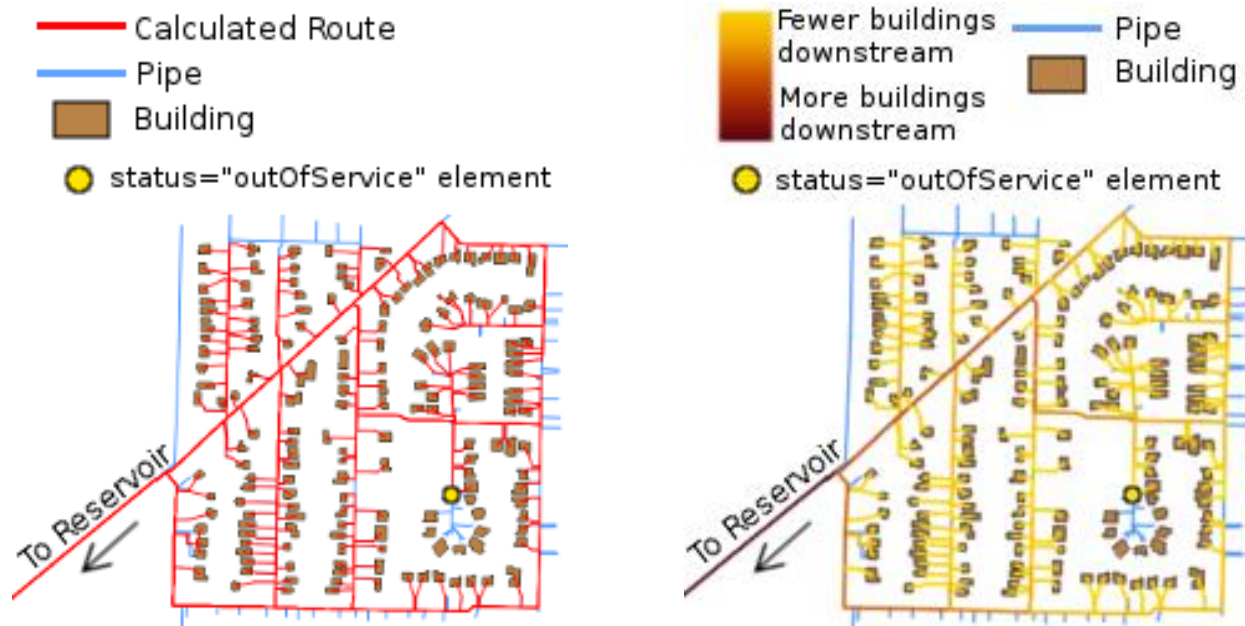
Figure 6. Visual representation of the least-cost path from the reservoir to the buildings (left). Visual representation of the stress on each link in the network along the routing of the reservoir to the buildings (right). Both examples are obtained using a subset of the Nanaimo data sample from Boates (2017)

4.2 Routing Analysis

A number of SQL queries for routing analysis were written to showcase the power and flexibility of using pgRouting on an instance of the extended 3DCityDB. The following list is not intended to be an exhaustive list of functionality, but rather a starting point for getting familiar with the extended 3DCityDB schema and how network routing algorithms can be performed on it.

4.2.1 Get total network volume: The first example returns the total volume in the network that is accessible from the local water reservoir (modelled as a "StorageDevice" in the Utility Network ADE). By using the pgRouting function "pgr_drivingDistance()" with a "distance" value set to infinity, all topological links from the network are scoured, starting from the reservoir, and, through a series of joins, the inner diameter property of every connected pipe element is used with the length of the pipe to calculate the total volume of water in the network. Changes in the network (opening/closing valves, elements being broken/fixed, etc.) would result in a change in the total network volume, so a dynamic method of calculating the total network volume in its current state would be quite useful.

4.2.2 Get routes from reservoir to houses: The second example returns the sets of links from the local water reservoir to each of the houses along the topological graph. The pgRouting function "pgr_Dijkstra()" was used in "one-to-many" mode, routing the reservoir element to all houses (modelled as "TerminalElements" in the Utility Network ADE). This function implements the well-known Dijkstra's algorithm to find, for example, the least-cost path between two nodes along the edges of a topological graph (Dijkstra, 1959). This could serve as an indication that the house is indeed reachable from the reservoir, and if so, what is the least-cost path to reach it. This is useful to know which buildings will lose or gain water supply in the event of a planned/unplanned change in the network (opening/closing valves, elements being fixed, broken,

etc.), and along which path. The result of this operation can be seen in Figure 6 (left).

4.2.3 Get number of traversals through each link: The third example returns the set of links comprising the least-cost paths from the water reservoir to all of the houses, indicating how many times each link was traversed in reaching all the houses, once again using the pgRouting function "pgr_Dijkstra". The process is similar to that of the example found in section 4.2.2, however it takes the extra step of consolidating all traversals across each link and adding an attribute counting the number of traversals. This could be considered as a measurement of "stress", where a higher number indicates a greater amount of buildings downstream from the current element. This is useful to see changes in the event of a planned/unplanned change in the network (opening/closing valves, elements being fixed, broken, etc.), and along which path. The result of this operation can be seen in Figure 6 (right)

4.3 Code Resources

All data and code involved in the writing of this paper is freely available in the form of a git repository, hosted on GitHub (Nanaimo-3DCityDB, 2018). Included is a subset of the Nanaimo water network sample dataset created by Boates (2017), the Python-3 script specified in section 3.3.3, as well as the SQL queries used to perform the routing analysis functions in section 4.2.

5. CONCLUSIONS

This paper has presented an example of implementation of the Utility Network ADE as common data model for network modelling in the context of semantic 3D city modelling based on the international open standard CityGML. The Utility Network ADE is a versatile data model for use in multi-network modelling. The possibility to import the data into the extended 3DCityDB (to support the Utility Network ADE and

pgRouting) opens the door to a number of challenges and interesting possibilities. In the test case presented in this paper, the sample data used for the examples of routing analyses in section 4.2 is for a freshwater network, but the unified model provided by the Utility Network ADE is independent of network type and other semantical properties, therefore it allows for the same functions to be quickly adapted and re-applied to other types of networks, at any scale. Furthermore, pgRouting's inherent customizability for building topological graph structures on the fly for routing analysis purposes means that using the semantical properties of network features is a feasible method of controlling network state and simulation behaviour. Modifying these properties, either by direct user intervention or as a consequence of another event elsewhere in the network provides a number of possibilities for all kinds functional network modelling.

The direct relationship of the Utility Network ADE with CityGML opens a great possibility of integrating network functionality within pre-existing smart city models that already operate based on a CityGML-based semantic city data model. For instance, a simulation of heat gains and losses via solar irradiance, such as that conducted in (Murshed et al., 2018), could be expanded to model and optimize local energy grid as the individual energy demand on each building grows and wanes over time. The current version of the Utility Network ADE, as well as the ancillary set of tools (database schema, data layers, routing functions, etc.) have allowed to perform a number of analyses on a water network. This proves that the Utility Network ADE, although still in active development, is mature enough to be used successfully.

5.1 Outlook

The work conducted in this paper is a first step when it comes to network modelling in CityGML. As a matter of fact, the water network used in this test case is only a subset of the entire city, which itself has much more semantical detail and information. For instance, there is more than just one reservoir (requiring multi-supply modeling, and there is a hierarchy of networks with mainlines feeding local distribution networks. The complexity of such systems will be exploited further to investigate the suitability and level of maturity of the Utility Network ADE in terms of modelling capabilities for multi-scale networks. Additionally, it is planned to test the Utility Network ADE with a number of integrated network types, so called multi-utility networks. For example, the proliferation of distributed energy generation in many countries in recent years (Georgilakis and Hatziargyriou, 2015) brings forth the need for a unified data model for application of electrical network simulation models and methods on multiple scales and in complex hierarchical networks. This need is not unique to electrical networks - indeed all networks require planning and modelling of one sort or another, and the Utility Network ADE can serve as a unified ecosystem in which multi-network multi-source applications can be developed.

Attention should be given to the capabilities of modelling interdependencies between different networks. In many cases, the state and functionality of elements on one network are directly influenced by changes in another. The Utility Network ADE provides capability for modelling these dependencies, however very little has been published in this area. More test cases and experiences from real life applications are needed in order to provide constructive feedback and input for further improvements.

REFERENCES

3DCityDB, 2018, 3D City Database homepage, http://www.3dcitydb.org (30 April 2018)

3DCityDB-ADE, 2018, Extended-3DCityDB GitHub repository, https://github.com/gioagu/3dcitydb_ade (30 April 2018)

Agugiaro, G., 2016, Enabling "energy-awareness" in the semantic 3D city model of Vienna. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-4/W1, pp. 81-88, doi:10.5194/isprs-annals-IV-4-W1-81-201

Agugiaro, G., Benner, J., Cipriano, P., Nouvel, R., 2018, The Energy Application Domain Extension for CityGML: Enhancing interoperability for urban energy simulations. *Open Geospatial Data, Software and Standards* 2018 3:2, doi: 10.1186/s40965-018-0042-y

Agugiaro, G., Holcik, P., 2017, 3D City Database extension for the CityGML Energy ADE 0.8 PostgreSQL Version - Documentation. Online, https://github.com/gioagu/3dcitydb_ade/blob/master/02_energy_ade/manual (30 April 2018)

Federal Ministry of Education and Research, 2012, Simulation of intersectorial cascade effects through breakdowns in supply infrastructures using a virtual 3D city model of Berlin (SIMKAS-3D), Online https://www.bmbf.de/pub/Civil_Security_Protection_of_Supply_Infrastructures.pdf (30 April 2018)

Becker, T., Nagel, C., Kolbe, T.H., 2011, Integrated 3D modeling of multi-utility networks and their interdependencies for critical infrastructure analysis. In: Kolbe, T.H., König, G., Nagel, C. (Eds) *Advances in 3D Geo-Information Sciences, Lecture Notes in Geoinformation and Cartography*. Springer, Berlin Heidelberg, pp. 1-20.

Becker, T., Nagel, C., Kolbe, T.H., 2012, Semantic 3D modeling of multi-utility networks in cities for analysis and 3D visualization. In: Pouliot, J., Daniel, S., Hubert, F., Zamyadi, A., (Eds) *Progress and New Trends in 3D Geoinformation Sciences, Lecture Notes in Geoinformation and Cartography*. Springer, Berlin Heidelberg, pp. 41-62.

Boates, I., Nichersu, A., Kutzner, T., 2017, Nanaimo Water Pipes Utility Network ADE Sample Current stage of development, European Institute for Energy Research (EIFER)

https://en.wiki.utilitynetworks.sig3d.org/images/upload/2017-12-08_UtiltyNetworkADE_Karlsruhe.pdf (30 April 2018)

den Duijn, X., 2018, A 3D data modeling approach for integrated management of below and above ground utility network features, Master Thesis, TU Delft, The Netherlands. https://repository.tudelft.nl/islandora/object/uuid%3Afed24b16-cf95-4fa0-a109-ece6e91b61e9

Dijkstra, E.W., 1959, A Note on Two Problems in Connexion with Graphs, *Numerische Mathematik 1*, pp. 269-271.

Dutta, A., Saran, S., Kumar, A.S., 2017, Development of CityGML Application Domain Extension for indoor routing and positioning. *Journal of the Indian Society of Remote Sensing*, 2(1), pp. 1-12.

Georgilakis, P., Hatziargyriou, N., 2015, A review of power distribution planning in the modern power systems era: Models, methods and future research. *Electric Power Systems Research*, 121, pp. 89-100.

Gröger, G., Plümer, L., 2012, CityGML – Interoperable semantic 3D city models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 71, pp. 12-33.

IntegrCiTy, 2018, Project IntegrCiTy homepage, http://iese.heig-vd.ch/projets/integrcity (30 April 2018)

Kim, Y., Kang, H., Lee, J., 2014, Developing CityGML Indoor ADE to Manage Indoor Facilities. In Isikdag, U. (Ed.), *Innovations in 3D Geo-Information Sciences*, Springer International Publishing, pp. 243-265.

Kolbe, T.H., Yao, Z., Nagel, C., Redweik, R., Willkomm, P., György, H., Arda, M., Kunde, F., 2016, 3D City Database for CityGML Version 3.3.0 - Documentation. Online, https://www.3dcitydb.org/3dcitydb/fileadmin/downloaddata/3DCityDB_Documentation_v3.3.pdf (30 April 2018)

Kumar, K., Ledoux, H., Commandeur, T.J.F., Stoter, J., 2017, Modelling urban noise in CityGML ADE: case of the Netherlands. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-4-W5, pp. 73-81.

Kutzner, T., Kolbe, T.H., 2016, Extending semantic 3D city models by supply and disposal networks for analysing the urban supply situation. In: *Lösungen für eine Welt im Wandel, Dreiländertagung der SGPF*, DGPF und OVG, 36. Wissenschaftlich-Technische Jahrestagung der DGPF, Deutsche Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation e.V., pp. 382-394.

Li, L., Tang, L., Zhu, H., Zhang, H., Yang, F., Qin, W., 2017, Semantic 3D modeling based on CityGML for ancient Chinese-style architectural roofs of digital heritage. *ISPRS International Journal of Geo-Information*, 6(5), pp. 1-19.

Murshed, S., Simons, A., Lindsay, A., Picard, S., De Pin, C., 2018, Evaluation of Two Solar Radiation Algorithms on 3D City Models for Calculating Photovoltaic Potential. In: *Proc. of the 4th International Conference on Geographical Information Systems Theory, Applications and Management*, 1, pp. 296-303.

Nanaimo, 2018, City of Nanaimo Data Catalogue, https://www.nanaimo.ca/open-data-catalogue (30 April 2018)

Nanaimo-3DCityDB, 2018, Nanaimo dataset GitHub repository, https://github.com/iboates/nanaimo_3dcitydb (30 April 2018)

pgRouting, 2018, pgRouting homepage, http://pgrouting.org (30 April 2018)

Yao, Z., Nagel, C., Kunde, F., Hudra, G., Willkomm, P., Donaubauer, A., Adolphi, T., Kolbe, T.H., 2018, 3DCityDB - A 3D Geodatabase Solution for the Management, Analysis, and Visualization of Semantic 3D City Models based on CityGML. *Open Geospatial Data, Software and Standards*, 2018 3:5, doi: 10.1186/s40965-018-0046-7

APPENDIX A

The following SQL statements exemplify different possibilities to insert into the extended 3DCityDB a RoundiPipe object for distribution, with an external diameter of 400 mm and an internal diameter of 350 mm, constructed in 1997-01-01, currently in use, with an elevation value that is estimated. In the SQL statements, red attributes are compulsory, blue ones can be omitted.

Example 1, using the low-level insert functions of the (extended) 3DCityDB:

```
WITH
 obj AS (SELECT id AS obj_id FROM citydb.objectclass WHERE classname='RoundPipe'),
 co  AS (SELECT citydb_pkg.insert_cityobject(id := NULL, objectclass_id := obj.obj_id, gmlid := 'RoundPipe_ID_2')
         AS co_id FROM obj),
 nf  AS (SELECT citydb_pkg.utn9_insert_network_feature(id := co.co_id, objectclass_id := obj.obj_id,
         function := 'distribution', year_of_construction := '1997-01-01', status := 'inUse',
         elevation_quality := 'estimated', geom := ST_GeomFromText('LINESTRINGZ(0 0 0, 1 1 1)', 26910))
         AS nf_id FROM co, obj)
SELECT citydb_pkg.utn9_insert_distrib_element(id := nf.nf_id, objectclass_id := obj.obj_id,
 function_of_line := 'flowLine', ext_diameter := 400, ext_diameter_unit := 'mm', int_diameter := 350,
 int_diameter_unit := 'mm') AS co_id FROM nf, obj;
```

Example 2, using the "smart" insert function of the (extended) 3DCityDB:

```
SELECT citydb_view.utn9_insert_ntw_feat_distrib_elem_pipe_round(id := NULL, gmlid := 'RoundPipe_ID_3',
 function := 'distribution', year_of_construction := '1997-01-01', status := 'inUse',
 elevation_quality := 'estimated', geom := ST_GeomFromText('LINESTRINGZ(0 0 0, 1 1 1)', 26910),
 function_of_line := 'flowLine', ext_diameter := 400, ext_diameter_unit := 'mm', int_diameter := 350,
 int_diameter_unit := 'mm') AS co_id;
```

Example 3, using the updatable view of the (extended) 3DCityDB:

```
INSERT INTO citydb_view.utn9_ntw_feat_distrib_elem_pipe_round (id, gmlid, function, year_of_construction, status,
 elevation_quality, geom, function_of_line, ext_diameter, ext_diameter_unit, int_diameter, int_diameter_unit)
 VALUES (NULL, 'RoundPipe_ID_4', 'distribution', '1997-01-01', 'inUse', 'estimated',
 ST_GeomFromText('LINESTRINGZ(0 0 0, 1 1 1)', 26910), 'flowLine', 400, 'mm', 350, 'mm');
```