Digital geoTwin

A CityGML-Based Data Model for the Virtual Replica of the City of Vienna

Lehner, Hubert; Kordasch, Sara Lena; Glatz, Charlotte; Agugiaro, G.

DOI
10.1007/978-3-031-43699-4_32

Publication date
2024

Document Version
Final published version

Published in
Recent Advances in 3D Geoinformation Science

Citation (APA)
Lehner, H., Kordasch, S. L., Glatz, C., & Agugiaro, G. (2024). Digital geoTwin: A CityGML-Based Data Model for the Virtual Replica of the City of Vienna. In T. H. Kolbe, A. Donaubauer, & C. Beil (Eds.), *Recent Advances in 3D Geoinformation Science: Proceedings of the 18th 3D GeoInfo Conference* (pp. 517-541). (Lecture Notes in Geoinformation and Cartography). Springer. https://doi.org/10.1007/978-3-031-43699-4_32

Digital geoTwin: A CityGML-Based Data Model for the Virtual Replica of the City of Vienna

Hubert Lehner, Sara Lena Kordasch, Charlotte Glatz, and Giorgio Agugiaro

Abstract This paper presents a CityGML-based data model developed for the semantic 3D city model of Vienna, Austria. The data model consists in a profile of the CityGML 2.0 standard and has been extended by means of an Application Domain Extension (ADE) developed by the Department for Surveying and Mapping of the City of Vienna in order to comply with the current and future needs of the municipality. The definition and adoption of such data model are a fundamental part of Vienna's "Digital geoTwin" project. The core of the strategy is to process the 3D measurement data of the surveying and mapping department from existing as well as new measurement methods directly into a Digital geoTwin—a virtual, semantic 3D replica of all objects in the city—and to derive other geodata products (city map, elevation models, etc.) from this 3D model. Furthermore, the Digital geoTwin should serve as a geometric and semantic basis for a digital twin of the City of Vienna. In order to define the data model for the Digital geoTwin, 3D modelling of all city objects has been carried out in a test area of the city, followed by a mapping of the objects to the CityGML data model. In an iterative development process, conceptual gaps have been identified, analysed and eventually formalized into a UML-based Application Domain Extension. Additionally, the free and open-source CityGML 3D

This article was selected based on the results of a double-blind review of the full paper.

H. Lehner (✉) · S. L. Kordasch · C. Glatz
Department for Surveying and Mapping, City of Vienna, Muthgasse 62, 1190 Vienna, Austria
e-mail: hubert.lehner@wien.gv.at

S. L. Kordasch
e-mail: sara.kordasch@wien.gv.at

C. Glatz
e-mail: charlotte.glatz@wien.gv.at

G. Agugiaro
Faculty of Architecture and the Built Environment, Department of Urbanism, 3D Geoinformation Group, Delft University of Technology, Julianalaan 134, 2628BL Delft, The Netherlands
e-mail: g.agugiaro@tudelft.nl

T. H. Kolbe et al. (eds.), *Recent Advances in 3D Geoinformation Science*, Lecture Notes in Geoinformation and Cartography, https://doi.org/10.1007/978-3-031-43699-4_32

City Database (3DCityDB) has been used for storage after being extended accordingly, and FME workbenches have been created to transform and import the original source data into the 3DCityDB and therefore test the suitability of the developed data model.

Keywords Digital geoTwin · Urban digital twin · Data modelling · 3D city model · CityGML ADE

1 Introduction: Geodata from Maps Towards Urban Digital Twins

Urban digital twins are a current trend in the digital transformation of cities. Thus, digital twin projects and initiatives are found in many cities in Europe and all over the world (Digital European Urban Twins (DUET),[1] Connected Urban Twins (CUT),[2] digital twin of Singapore,[3] …). Originating from Industry 4.0, the Digital Twin is defined a multi-physical and multi-scale representation of a complex system. It uses realistic models of the system, its environment and sensor data in order to mirror physical life of the system in the digital world and vice versa (Durão et al. 2018; Rosen et al. 2015).

Cities are highly complex and growing ecosystems. According to the United Nations it is expected that by 2050 the world's population living in urban areas will increase to 68%.[4] Constant growth and global challenges like climate change reveal that new strategies such as urban digital twins are needed in order to manage and develop cities for further generations. Many data of many different disciplines and fields such as geodata, management data, sensor data, socioeconomic data, traffic data, etc. are necessary to build such urban digital twins. In our perception, an urban digital twin is not only a technical solution but also a new way of interdisciplinary collaboration. Thus, it is necessary to link many different expert knowledge systems in a virtual environment.

The resulting digital twin of the city can serve as a platform for various use cases and help simulate and understand actions before they are implemented in the real world. An urban digital twin is hardly built up from scratch but rather faces the challenges to develop and grow existing expertise and systems towards an interlinked virtual city.

Due to the complexity of the matter, the term urban digital twin is used in many different ways. While the wrong usage of the term—in our view—originates in some

[1] https://www.digitalurbantwins.com/.

[2] https://www.connectedurbantwins.de/en/.

[3] https://www.gim-international.com/content/article/singapore-s-journey-towards-a-nationwide-digital-twin.

[4] https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html.

cases from a too simple understanding or misinterpretation of the concept, in other cases it is due to marketing reasons. The term has been used as rebranding of existing products, technical solutions or software. Regarding geodata e.g. points clouds or textured 3D meshes have been named digital twin in some cases. These circumstances lead to the fact that the idea and concept of a digital urban twin is hard to define.

In the case of Vienna, there is a long-term vision to create a digital twin of the city. The vision encompasses several aspects related to digitalisation processes, the definition of new workflows to interact with it (both from inside the city administration, but also from outside in terms of services offered to the public or third parties), and the identification of applications and use cases exploiting it. Providing a detailed report on the overarching vision is beyond the scope of this paper.

Within the broader vision, the Department for Surveying and Mapping of the Vienna City Administration started a sub-project to support the Digital Twin of the City of Vienna. In order to work in a virtual environment of a city 3D models are necessary. However, they need to be more than just visualisation models. The objects need semantic information and further attributes in order to support a broader usage. Thus, geometric as well as semantic modelling is necessary.

The Department for Surveying and Mapping has been responsible for delivering and maintaining basic geodatasets for decades. The classic and historic evolution of geodata is similar to many European cities. A more detailed overview of the development in Vienna is given in Lehner and Dorffner (2020). The development usually started with (2D) digital city maps and 2.5D height models to 3D city models. The latter encompassing solely 3D building models, in the beginning, and adding more 3D objects such as 3D bridge models, 3D tree models, etc. later on. Each of these datasets is usually created by specialised software with specialised functions and workflows. Sometimes different input data and sources are used for each dataset, while in other cases—such as in Vienna—it was attempted to use input data multiple times in several product chains. In order to create detailed semantic 3D models of a city, all these different geodatasets are usually combined into one semantic 3D city model (see Fig. 1). Problems usually occur due to temporal incoherence of the datasets. While for visualization purposes, these problems might be neglectable, they have to be tackled in case the resulting 3D city model should serve as basis for an urban digital twin.

This approach does not seem sufficient in order to create data models which can serve as geometric foundation for an urban digital twin. Thus, a new approach for a Digital geoTwin was created in order to overcome the drawbacks of the classical, historically evolved way of geodata production. The centre of the strategy is to use all 3D measurement data of the Department for Surveying and Mapping from existing as well as new measurement methods in order to directly model semantic vector-based 3D geo-objects of the whole city. More generalised geodata products shall be derived from the Digital geoTwin (see Fig. 2).

An advantage of this strategy is that temporal and content-related coherence is achieved for all the geodata products, which are derived from the Digital geoTwin. The prefix ***geo*** was chosen for the neologism **Digital *geo* Twin** not only to emphasize our focus on the geodetic, geometric aspect of creating semantic geo-objects, but also

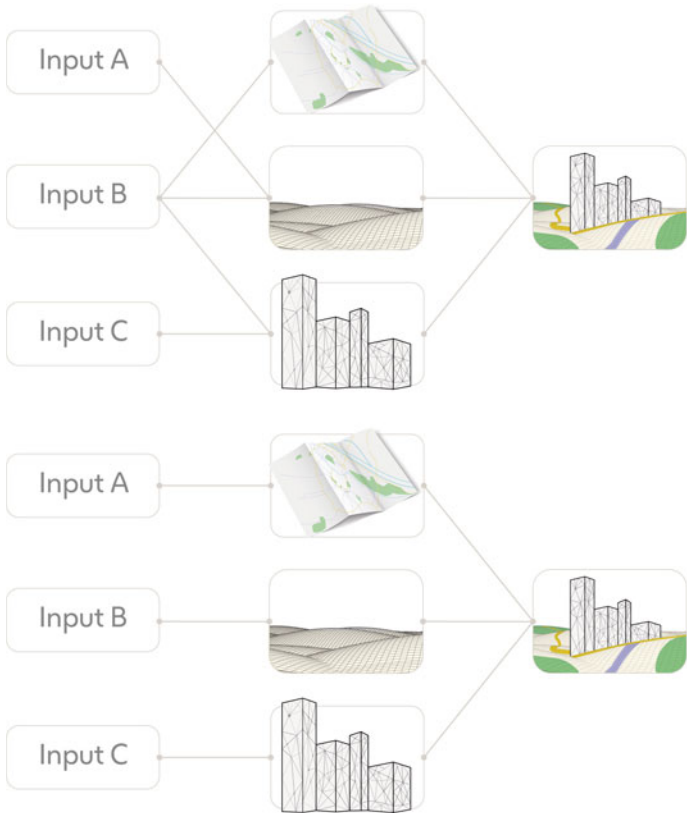

Fig. 1 "Classic" way of producing geodata sets and combining them to a 3D city model. Exemplary delineation of multiple usage of input data in several geodata sets (upper sketch) compared to separate input data for each dataset (lower sketch)

Fig. 2 Representation of the concept for modelling a Digital geoTwin and deriving other geo-datasets from it

to distinguish our concept and the resulting model from the wide field of digital urban twins.

2 Digital geoTwin—A Virtual Replica of the City of Vienna

The concept of the Digital geoTwin was already published in 2020 by Lehner and Dorffner. The goal was to rethink the creation of geospatial data of the Department for Surveying and Mapping, detached from existing established systems and structures. Contrary to serial production of several datasets via different workflows and software, all input data shall be used once to create semantic vector-based 3D geo-objects of the whole city. Other datasets of the department shall be derived from this dataset.

With the concept of the Digital geoTwin, the Department for Surveying and Mapping of the City of Vienna is pursuing a unique and innovative approach for which, to the best of current knowledge, there are no comparable examples nationally or internationally. Accordingly, it was not possible to draw upon references. In order to verify the feasibility of the approach a proof of concept (PoC) was established. In the PoC, the focus was on determining whether technical solutions are available for this inverted approach and whether the concept can be implemented productively in the future.

When setting up the project, a deliberate distinction was made between inward-facing and outward-facing project objectives.

The inward-facing project objectives deal with the necessary components from the creation of 3D objects up to the automated derivation of geospatial data products:

- Modelling of semantic, vector-based 3D geo-objects of all objects of the city based on 3D measurement data;
- Usage of point cloud data from different technologies in the modeling process to support filling missing content in the 3D measurement data;
- Development of a 3D semantic, object based data model and a 3D object database for the storage of 3D semantic geo-objects;
- Conceptualization of the automatic derivation of geospatial data from the 3D object inventory.

The outward-facing project objectives encompass interfaces to application scenarios of a Digital Twin of the city of Vienna. They were considered from the beginning, as they may have implications on the creation and maintenance process of the Digital geoTwin. These goals include interlinking the semantic 3D city model with maintenance or management data or sensor data in order to increase the level of information of the individual objects and the whole virtual model to a great extent. Furthermore, it plays a significant role in breaking down data silos within the city administration and enhancing the quality of datasets, contributing significantly to the Data Excellence (DX) strategy of the City of Vienna (Lutz 2019). Another goal deals with the usage of the Digital geoTwin as basis in planning processes. Finally, all parts together—the Digital geoTwin, interlinked domain-specific data and sensor data as well as planning data—can be used in various urban simulation processes.

Within the Digital geoTwin project a number of different sub-tasks and preliminary studies are being carried out. One of the sub-tasks has the main goal of identifying the requirements and then defining (and adopting) an underlying data model for the semantic 3D city model that will comply with the overall "Digital geoTwin vision". The developed data model is based on the international standard CityGML 2.0, from which a profile has been extracted, on the one hand, while additional features have been added by means of the Application Domain Extension (ADE) mechanism. The specific parts regarding the generation of the ADE have taken inspiration from existing previous experiences, as described in Agugiaro et al. (2018), Kumar et al. (2019) and Ying et al. (2022). A rather exhaustive overview of CityGML ADEs can be found in Biljecki et al. (2018). However, at least in the authors' opinion, there are not so many previous experiences regarding a continuous 3D modelling of all objects of a city and an extensive mapping of these objects to nearly all CityGML modules as in the case of Vienna.

3 Digital geoTwin Data Model Requirements

As a result of the first data and user analysis, the following requirements have been identified when it comes to the semantic 3D city model of Vienna, also considering the type of products to be obtained from it. The 3D city model should offer the possibility to:

- Store not only vector-based 3D geometries, but also semantics for each object. This applies also to features generally needed to generate a digital terrain model. For example: distinction between a river bed and the river surface, identification of the terrain surfaces that existed before being covered or built up (e.g. by a building);
- Store objects represented in different Level of Details (LoD);
- Decompose complex objects into smaller components (e.g. buildings and building parts);
- Model and store objects that might be repeated/cloned multiple times by means of templates (e.g. a park bank, a traffic light, or a bus station);
- Store all geometries natively in 3D. In particular, in case of non-planar surfaces, it must be possible to derive them in form of semantic 3D meshes. For example, this applies *also* to terrain models which may contain features that cannot be modelled in 2.5D (such has vertical elements, or overhangs);
- Automatically derive 2D maps or 2.5D height models, which contain a seamless tessellations of space. This means, for example, no holes in a terrain model due to buildings being "taken out";

- Store some metadata at the level of each object, regarding for example the geometrical accuracy resulting from the surveying campaign, or information about the creation timestamp and the validity/existence interval of time of a given object.

Such shift in the 3D city modelling process does not only call for a revision of the data ETL (Extract, Transform and Load) and integration strategies, but also for the identification of a suitable data model able to store such a 3D city model. Given the above listed set of requirements, the open and international standard CityGML was taken into consideration in order to evaluate its suitability.

3.1 Evaluation of CityGML

Thanks to its availability as open data model, its growing world-wide adoption by several international cities, and its set of characteristics that already align very well with the needs of the Department for Surveying and Mapping, CityGML was considered as the candidate data model for the Digital geoTwin of Vienna.

Despite the availability of CityGML 3.0 in terms of the conceptual model (Kolbe et al. 2022), it was decided to test and evaluate CityGML version 2.0 instead (Gröger et al. 2012) because of the current availability of software tools to work with it, such as the 3D City Database suite (Yao et al. 2018), which includes the free and open-source 3DCityDB and the accompanying Importer/Exporter tool. At the same time, enough documentation and experiences regarding ADE modelling and the associated tools already exist, while the same cannot be said about CityGML version 3.0 to the same extent, yet.

In general, CityGML is a modular and well-structured standard which comes with a set of modules (and their relative classes) that already cover most objects of a city, as represented in Fig. 3. At the same time, it is extendible, i.e. it can be customized by means of the Generics module, or via ADE. At the same time, if certain features (or feature attributes) are not needed, they can be omitted in a subset of CityGML, which is a so-called *profile.*

In the case of Vienna, both strategies were adopted: some features (or some of their attributes) were removed, therefore defining a profile, and at the same time some features were added by means of the ADE mechanism. Although the details of the Digital geoTwin data model will be given in the next chapter, the overarching design decisions will be described here.

3D modelling of all city objects has been carried out in a test area of the city (see Chap. 4.1), followed by a mapping of them, as far as possible, to the closest CityGML equivalent. This means that an analysis was carried out in terms of semantics, but also in terms of available (or missing) attributes, available LoDs and geometry types (solid vs multi-surfaces, implicit vs explicit geometries, etc.). For certain attributes, codelists were defined.

One of the aspects that required major attention was the "mapping" of the digital terrain objects. Given the necessity to store fully 3D objects (e.g. vertical walls), this

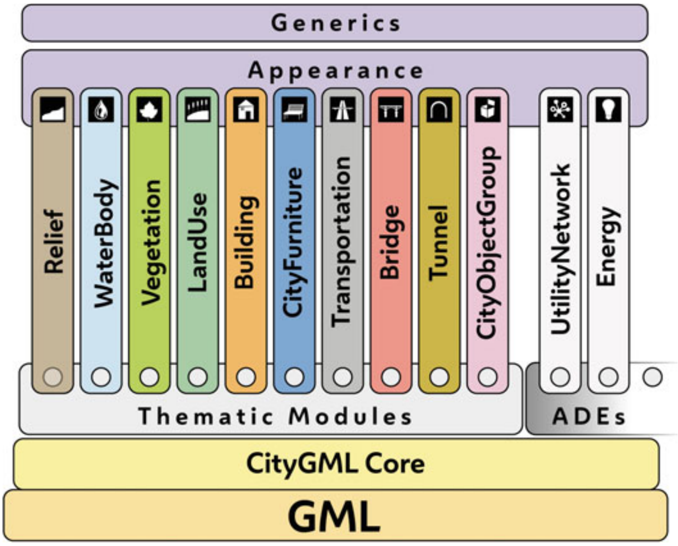

Fig. 3 Overview of the CityGML 2.0 thematic modules. Image adapted from Löwner et al. (2012)

automatically ruled out the use of CityGML's Relief module, which allows only for 2.5D triangulated meshes in terms of polygon-based geometries.[5]

As a result, a major change in the mapping and modelling rules was introduced for "terrain objects": the Relief module was discarded and the LandUse module was chosen instead as it is able to store 3D meshes. Additionally, semantically different "terrain" objects such as transportation or waterbody objects have been mapped directly to the corresponding CityGML modules.

When it comes to the intersection between a building (or bridge) and the terrain, the CityGML concept of TerrainInstersectionCurve was adopted to store the resulting 3D polyline representing such intersection. However, in order to also store the actual portion of digital terrain model "cut out" by a building (or a bridge) the additional class *LandUseClosureSurface* was added via the ADE mechanism. More details will be given later. The *LandUseClosureSurface* concept extends the CityGML *ClosureSurface*. ClosureSurfaces do not exist in reality and are mainly there to virtually close volumetric objects. They are for example already intended for the Tunnel module. Here the ClosureSurface is used to virtually seal the tunnel portal. The outline of the portal defines the boundaries of a 3D surface which is then in instance of the TunnelClosureSurface. A different usage can be found in the WaterBody module. The WaterClosureSurface intends to virtually define the end of the modelled area of bigger water bodies in case they exceed the area of interest (e.g. outside the boundaries of the city). The WaterClosureSurface, thus, can be defined arbitrarily without any representation in nature.

[5] Please note that 3D breaklines and 3D masspoints are also supported, but they were considered not suitable for the needs of Vienna.

In order not to store geometries twice, which are deployed in more than one object, CityGML incorporates the xLink concept of the GML-Standard (Gröger et al. 2012). The geometry is stored in one object and can be linked to another object via xLink. Building models can be modelled via thematic surfaces (wall, roof, ground, etc.) and xLink the surfaces to the volume of the building. Another usage is to xLink facade surfaces, which are shared by adjacent buildings. The standard furthermore recommends in the LandUse module to store line segments of adjacent surfaces only once and xLink them from one polygon to the other. While the first example of linking thematic surfaces to volumes is commonly used the other two examples are hardly used in practise. The concept of such complex hierarchical data structures are beneficial for updating city-wide semantic models, as intended with the Digital geoTwin. However, they are hardly suitable for exchanging sematic models. Most software products simply do not support models which are modelled in such complex hierarchical way. Thus, the hierarchical data structure was just used while modelling the data but not when mapping the objects in CityGML.

Regarding thematic modules, CityGML 1.0 was already quite comprehensive. Only the modules bridge and tunnel were missing compared to CityGML 2.0. Still, the focus of 3D city models is the building module. Many software solutions focus solely on 3D building models and not much on other thematic modules. Thus, 3D building models were simply understood as 3D city models. This can also be seen in freely available CityGML datasets. CityGML Wiki,[6] Github[7] and the 3D Geoinformation research group of TU Delft[8] links data sets of cities, regions and in some cases even nations. The datasets which can be found there contain only building models in many cases. Next to buildings some datasets also provide a terrain, tree or transportation models. These reference lists do not claim to be exhaustive and do not show whether there are other datasets, which do use more thematic modules. Still they give a good overview on how and where CityGML is used in practise.

The following chapter will provide further details about the Vienna semantic 3D city data model.

4 The Digital geoTwin Data Model for the Semantic 3D City Model of Vienna

As already mentioned before, one of the main purposes of Vienna's 3D city model is to allow for coherent and integrated representation and storage of all 3D city objects being modelled based on data from different surveying techniques and campaigns. Additionally, it is meant to represent the unique source of all different geodata products that are otherwise provided and used by the City of Vienna. This includes for example the automatic derivation of a city-wide DTM which must geometrically be

[6] https://www.citygmlwiki.org/index.php?title=open_data_initiatives.

[7] https://github.com/OloOcki/awesome-citygml.

[8] https://3d.bk.tudelft.nl/opendata/opencities/.

closed and topologically correct. In other words, the semantic 3D city model will contain all urban 3D geo-objects, but still, a closed (i.e. “hole-free”) surface can be derived from it to represent just the terrain.

From the development and implementation point of view, the procedure followed to define the data model was based on the following main steps:

1. 3D modelling of all city objects based on existing surveying data and point clouds in the test area;
2. Mapping of the city objects to the CityGML data model, identification and analysis of correspondences and gaps;
3. Extension of the CityGML data model by means of an ADE in order to cope with the specific additional requirements stemming from the previous analysis. Formalisation of the ADE by means of a UML class diagram and generation of the resulting XSD file. The UML class diagram was created in Enterprise Architect, the XSD file was generated by means of ShapeChange, following a similar approach as described by Van den Brink et al. (2013);
4. Generation of DDL database scripts to extend the 3DCityDB and add support for the ADE. Please note that this functionality is already available with version 4.x of the 3DCityDB;
5. Definition of pipelines to convert the source data into CityGML and to import them into the extended 3DCityDB. These ETL procedures were developed using FME 2022;
6. Test of the developed data model and the ETL procedures using the datasets collected for a chosen test area in Vienna;
7. Evaluation of the results.

Please note that, due to the exploratory nature of this work, there have been some iterations in the steps listed above as the process is not completely straightforward.

The following subsections will provide further details on how the CityGML 2.0 data model was adapted in order to comply with the overall requirements described in the previous chapter. To ease readability, first the test area in Vienna will be presented and then the specific mapping decisions regarding the 3D modelling results of the test area will be discussed. For the sake of logical simplicity, the overall structure of CityGML in terms of modules will be followed.

4.1 Test Area

A test area of circa 0.314 km^2 was defined in the third district of Vienna along the Danube Canal near the Stadionbrücke (see Fig. 4). In this area, a good heterogeneity of urban fabric can be found, ranging from small garden settlement to densely built residential area, forest, waterbodies, irregular terrain, etc. (see Fig. 5).

The available geospatial datasets consist of surveyed 3D points or 3D polylines, associated attributes and codes. An example is given in Fig. 6. All datasets, their characteristics and metadata were first collected, then used to model all city objects,

Fig. 4 Map of Vienna (left) with highlighted position of the test area (right), which is represented in detail (*Source* Stadtplan3D map—Department for Surveying and Mapping Vienna)

Fig. 5 Oblique aerial view of part of the test area (*Source* oblique image 2020—Department for Surveying and Mapping Vienna)

which were finally used as foundation for the development of the data model. The mapping between the 3D objects in the test area and the CityGML classes was carried out manually, while the transformation from the source datasets into CityGML files was carried out using FME.

4.2 LandUse, Transportation and WaterBody Modules

Objects, semantically belonging to these CityGML modules, were generated from 3D points and polylines as 3D meshes. 3D points come from laserscanning or from classic topographic surveying. The 3D points can have a regular as well as irregular

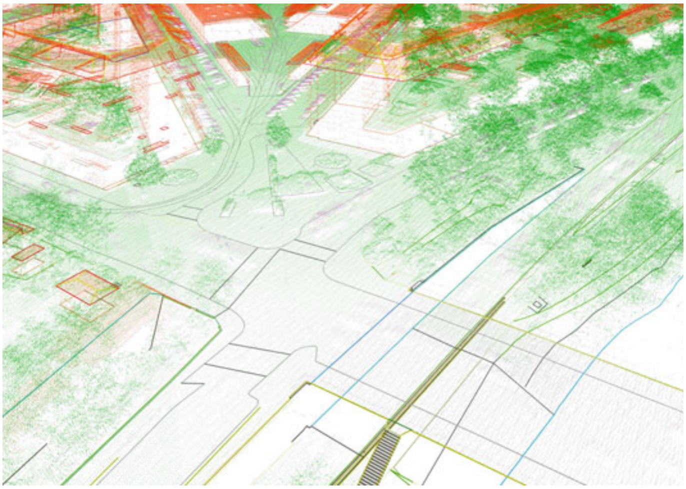

Fig. 6 Example of data available in the test area: 3D surveyed data overlaid with a classified ALS-based point cloud (*Source* Department for Surveying and Mapping Vienna)

pattern. For example, vertical walls can be represented by means of vertical surfaces without the need to change the underlying data model, as LandUse, Transportation and WaterBody modules already allow to model geometries as MultiSurfaces (instead of TINs), thus overcoming the 2.5D limitation (see Fig. 7).

As in general, there are no clear indications in the CityGML specifications regarding which level of detail to use with the LandUse model. Thus, it was decided to use only LoD2. Unneeded attributes and properties/associations to unnecessary LoDs were therefore removed (see Fig. 8).

When it comes to the Transportation module, all existing classes and subclasses (TrafficArea, AuxiliaryTrafficArea) were used to map and preserve the semantic of the objects available in the test area.

Regarding the WaterBody module, the existing classes and attributes were used to model both waterbodies for which 3D data are available (e.g. the Danube canal) and minor affluent streams (see Fig. 9). An example is given in Fig. 10. In the case of the Danube canal, the ClosureSurface classes were used to seal the river volume contained between the WaterSurface and the GroundSurface.

As mentioned before, in order to create a sealed (e.g. hole-free) "terrain" model once the 3D objects (e.g. buildings) are removed, a new class, called LandUseClosureSurface was defined by means of the ADE mechanism. The class LandUseClosureSurface is associated with the corresponding CityObject it substitutes, and consists geometrically of a 3D mesh. In the case of a building, for example, the LandUseClosureSurface will be a surface whose boundaries coincide with the TerrainIntersectionCurve of that building (see Fig. 7).

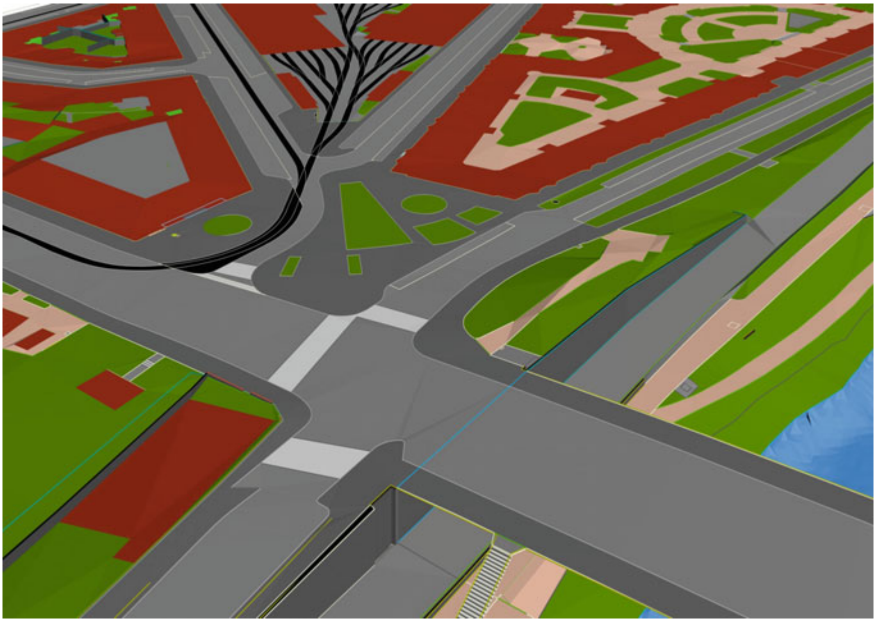

Fig. 7 Example of objects found in the test area modelled using classes from the CityGML LandUse, Transportation and WaterBody modules. Additionally, LandUseClosureSurface objects are represented in red

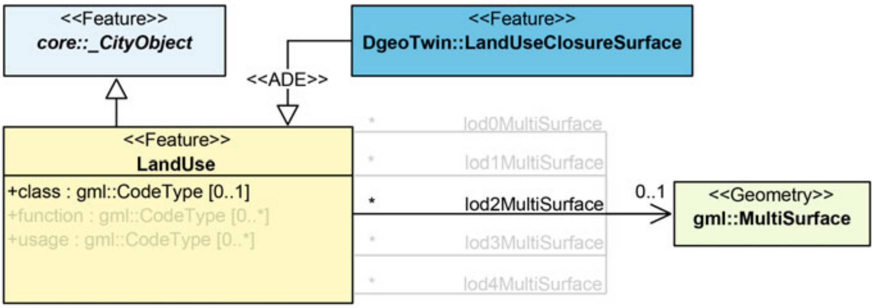

Fig. 8 Class diagram of the Digital geoTwin data model for the LandUse module, consisting in the profile derived from the CityGML corresponding module and the ADE class represented in dark azure. Original CityGML properties not used are represented in light gray. Image adapted from Gröger et al. (2012)

Finally, in order to store information about the status of the surface sealing, an additional attribute (SurfaceSealing) was added via ADE to the core module (see later).

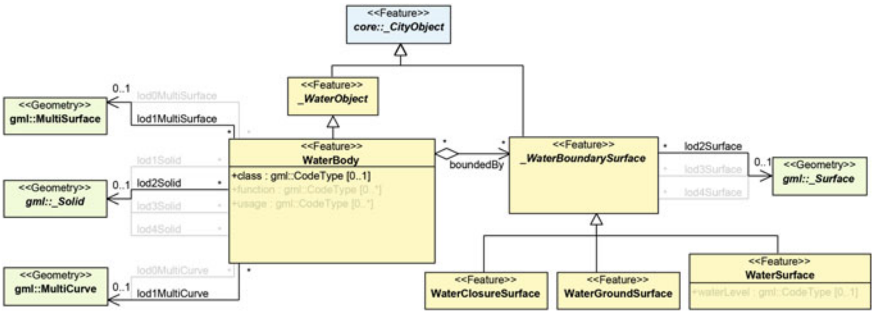

Fig. 9 Class diagram of the Digital geoTwin data model for WaterBody module, represented using the same approach as Fig. 8. Image adapted from Gröger et al. (2012)

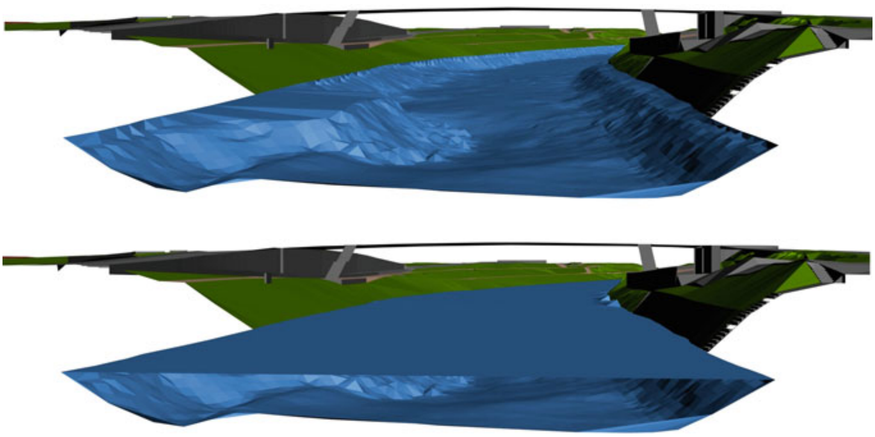

Fig. 10 Example of semantic 3D modelling of the Danube canal in the test area: WaterGroundSurface (top), WaterGroundSurface and WaterSurface (bottom)

4.3 Vegetation and CityFurniture Modules

For both the vegetation and city furniture modules, similar mapping and modelling rules apply. In general, CityObjects are generated automatically from the surveyed source data and both explicit (BRep or solids) or implicit geometries are used, depending on the type of CityObject.

In the case of the Vegetation module, the class SolitaryVegetationObject is used to model single trees, while bushes and cluster of trees are mapped to the PlantCover class (see Fig. 11). For single trees, as the source data only consist of surveyed points and some attributes, the modelling approach consists of using different tree templates and model them geometrically using implicit geometries. Different templates are defined also depending on the LoD (see examples in Fig. 12 covering LoDs from 0 to 3). For the LoD0 implicit representation, this had to be added via ADE as it is not

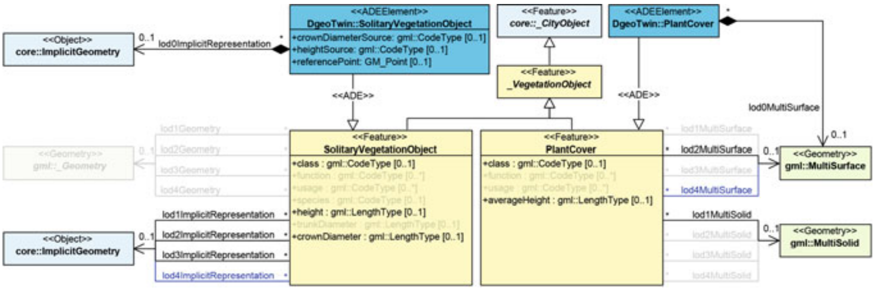

Fig. 11 Class diagram of the Digital geoTwin data model for the Vegetation module, represented using the same approach as Fig. 8. Image adapted from Gröger et al. (2012)

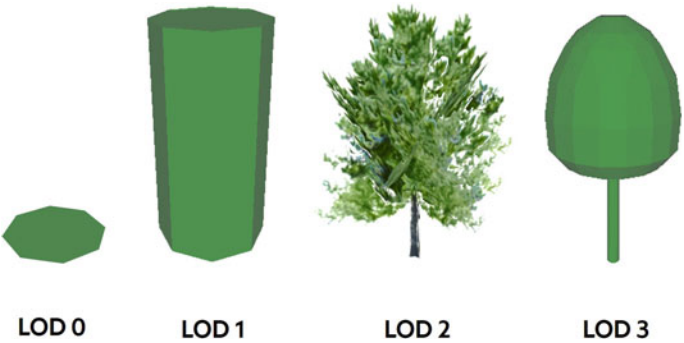

Fig. 12 Example of LoDs for SolitaryVegetationObject trees belonging to class „Laubbaum" (deciduous tree)

included in the default CityGML class. Attributes are mapped either to the existing or to new classes added via ADE. In particular, a customized codelist was created for attribute "class".

The PlantCover class is used to model bushes and trees groups, for which it is not possible to distinguish the single plant (see Fig. 13). Source data attributes are mapped to existing ones of class PlantCover (again, a customized codelist was created for attribute "class"), while for geometries, the following rules were defined:

- LoD0 represents the "footprint" of the bush or tree group and is modelled as MultiSurface
- LoD1 represents the closed (solid) geometry of the vegetation group. The height is the average height computed from the corresponding normalised DSM
- LoD2 is used instead only to model (as MultiSurface) the 3D mesh resulting from the vectorization of the corresponding DSM area covered by the PlantCover object.

Similarly to the SolitaryVegetationObject, the LoD0 geometries had to be added via ADE.

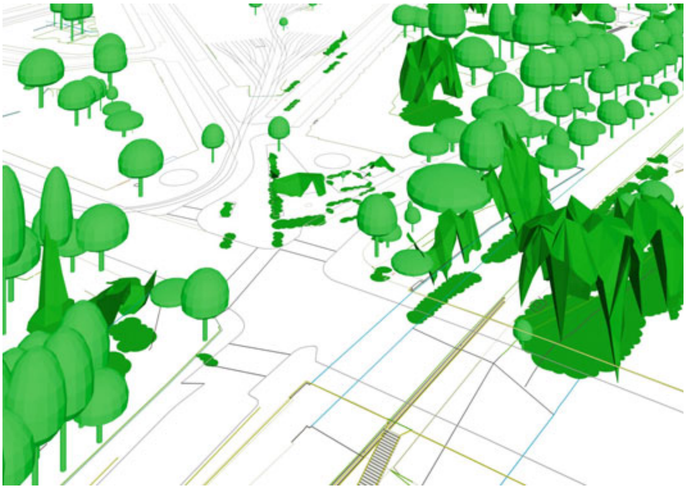

Fig. 13 Example of Vegetation objects in the test area, modelled using both SolitaryVegetationObject and PlantCover classes

When it comes to the CityFurniture module, this was used to represent city furniture objects, although some data transformation rules had to be defined to convert data from the source data accordingly. As the source data consists mainly of surveyed point, line and area objects, enriched with some attributes regarding dimensions and thematic data, these criteria were defined (see Fig. 14):

- CityGML "class" and "function" attributes are used to map the thematic attributes. Customized codelists are created therefore
- Source attributes regarding height, diameter, azimuth were used to generate implicit geometries and stored as metadata via additional ADE attributes

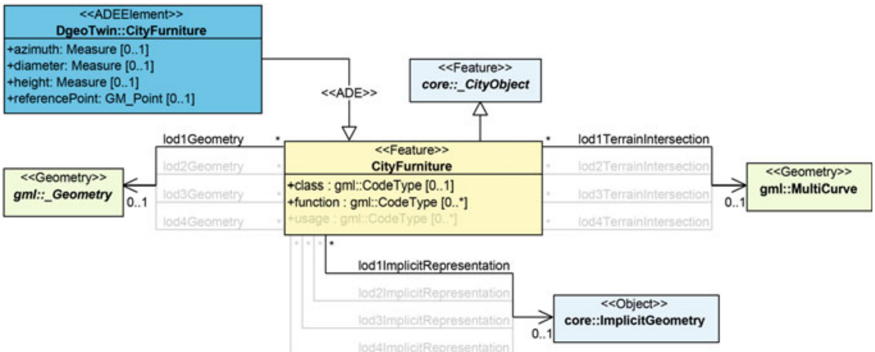

Fig. 14 Class diagram of the Digital geoTwin data model for the CityFurniture module, represented using the same approach as Fig. 8. Image adapted from Gröger et al. (2012)

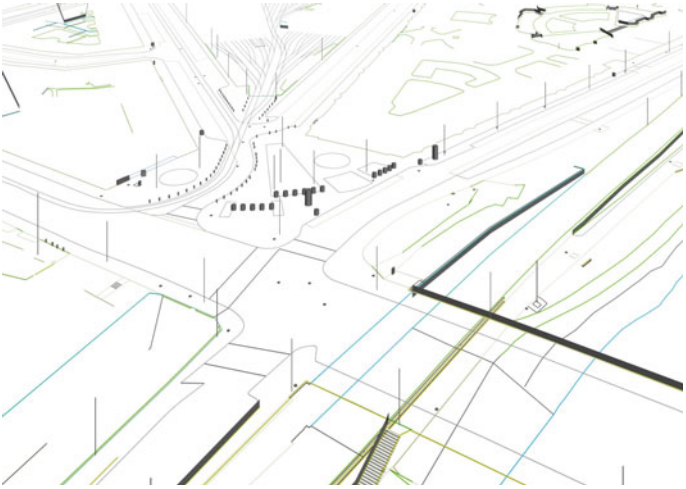

Fig. 15 Example of objects in the test area, modelled using classes from the CityFurniture module

In particular, regarding geometry:

- Only LoD1 was used
- For point-based objects, either implicit geometries for simple shapes (cylinder or box) or simple surfaces were generated, using the dimensional attributes mentioned before for scaling. Volumetric templates are used for objects like masts, bollards, etc. while simple surface templates are used, for example, for manhole lids.
- For line-based objects (e.g. fences or railings), multi-surface geometries are used. Depending on the origin of the information about their height (photogrammetry or terrestrial survey) they are vertically extruded upwards or downwards
- For area-based objects (e.g. the footprints of phone booths), the surveyed 3D multiline is stored as TerrainIntersectionCurve. Additionally, the 3D polyline is then flattened to the lowest surveyed height and stored as polygon in order to be extruded using default values depending on the type of object.

An example of the CityFurniture objects is given in Fig. 15.

4.4 Building, Bridge and Tunnel Modules

The last group of thematic modules deals with objects such as buildings, bridges and tunnels. An extensive study regarding building LoDs has already been published (Lehner and Dorffner 2020). The study is based on the works of Biljecki et al. (2016) and extends the therein-included refined specifications for the LoD concepts of buildings. Of particular relevance for Vienna is the systematic classification of buildings depending on their geometric characteristics, i.e. in case of architectonical

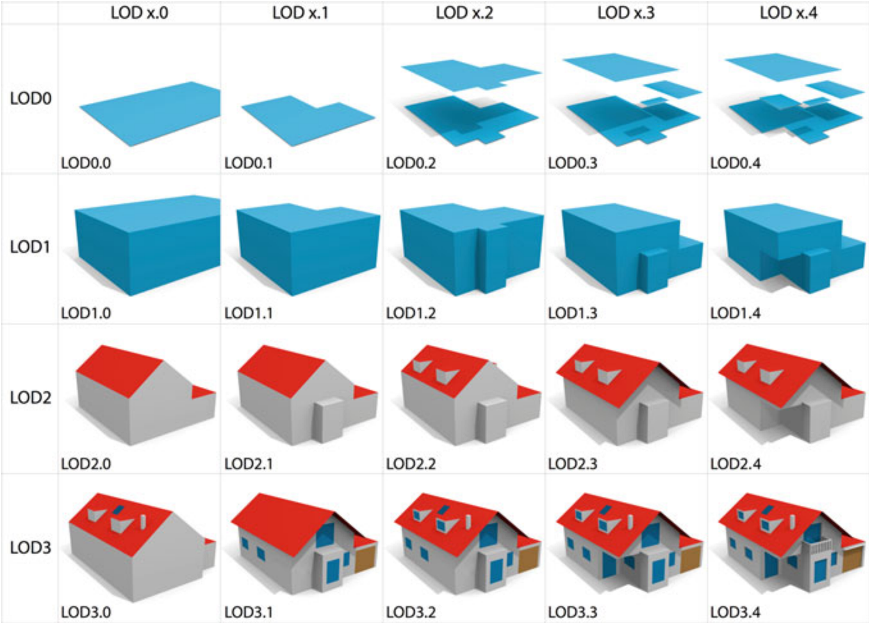

Fig. 16 Refined LoD concept for Building according to Lehner and Dorffner (2020), further developed from Biljecki et al. (2016)

elements such as arcades, passages, bay windows, etc. which eventually best fit with the level 2.4 of the new proposed LoD schema (see Fig. 16).

The underlying idea is to have the building geometry stored at the highest available level of detail, and to procedurally derive lower-LoD models by means of 3D generalization algorithms. At the same time, the LoD2.4 is considered the most future-proof in order to add further details like openings (e.g. windows and doors) whenever they will be available (and thus augmenting the LoD to 3.x). It is noteworthy to remind here that the TerrainIntersectionCurve of a building must correspond with the outer boundary of its LandUseClosureSurface.

In the case of the Bridge module, the bridge contained in the test area was mapped without major issues to the corresponding CityGML class. For the geometry, the chosen level of detail was LoD2. Again, analogously to the Building module, the TerrainIntersectionCurve of the bridge must correspond to the outer boundary of the associated LandUseClosureSurface.

An example for buildings and a bridge in the test area is given in Fig. 17.

Although in the test dataset no tunnels were contained, a theoretical mapping was carried out based on the CityGML Tunnel module. However, as already mentioned, a ClosureSurface class is indeed already available for the Tunnel module. Thus, in order to support software which uses the tunnel module, the LandUseClosureSurface can be xLinked to the TunnelClosureSurface.

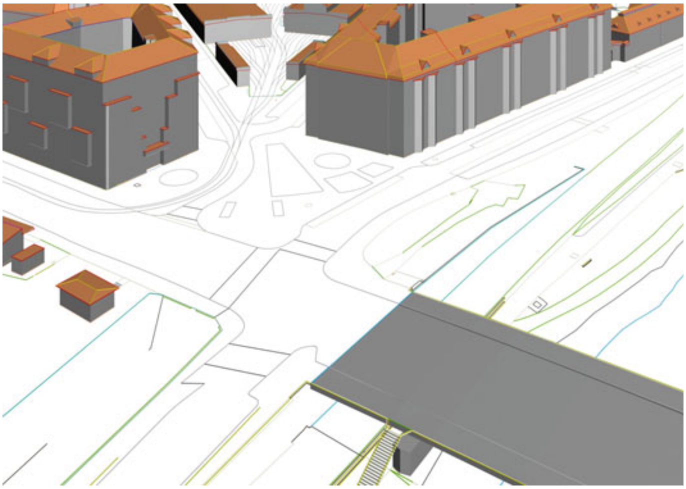

Fig. 17 Example of objects in the test area, modelled using classes from Building and Bridge modules

4.5 Core Module

The CityGML core module contains base classes that are then used (or inherited) by the depending thematic modules (see Fig. 18). It is therefore also the most logical module where to add ADE elements and properties that will be then likewise inherited by all subclasses (see Fig. 19). This applies in particular to the abstract class CityObject. In the case of Vienna, the core module was used to extend the CityObject class by a number of properties, mostly due to the need to store metadata of each object in terms of temporal and geometrical accuracy information. These metadata accompanying the original surveyed data (e.g. points and lines) which are then aggregated separately and associated to the resulting cityobject. In terms of temporal quality information, this is expressed as timestamps when a certain horizontal and vertical value was surveyed, and by means of codelists that define vertical and horizontal accuracy values. Such codelists contain values used in the data processing pipeline, but can be linked to external look-up tables.

Finally, metadata are also stored with regard to the last temporal interval check carried out on the constituent geometries of the cityobject, and finally the timestamp associated to the creation of the 3D geometry of the cityobject.

Finally, the SurfaceSealing property is used to characterize the sealing type of the surface, and it is associated with a codelist, too. This attribute is meant to be used by all modules containing "terrain objects", i.e. LandUse, Transportation and WaterBody.

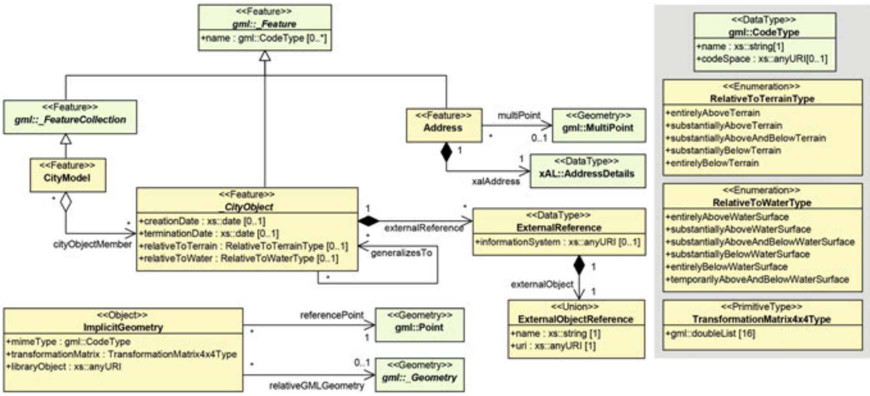

Fig. 18 Class diagram of the CityGML 2.0 core module. Image source: Gröger et al. (2012)

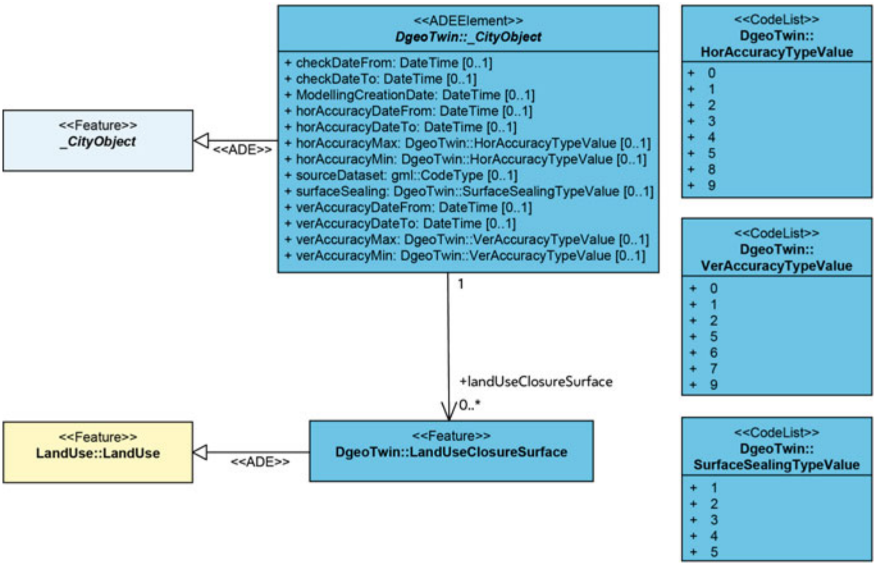

Fig. 19 Additional classes of the core module for the Digital geoTwin data model

5 Evaluation of the Results

Once the 3DCityDB has been extended with the ADE, all test data have been transformed and imported into the database. The results so far allow us to say that:

- In general, the developed data model allows to satisfy the requirements defined at the beginning of the study; the profile based on CityGML is a very valuable starting point upon which to develop and "add" a relatively small ADE;

- All data from the case study area can be successfully converted without data losses using the developed FME workbenches. All objects have been imported "as is", or have been successfully transformed and enriched as planned;
- We have encountered some limitations in terms of 3DCityDB: there is currently no support for Implicit Geometries in an ADE, therefore we have temporarily used the native LoD4 in lieu of the ADE LoD0 for Vegetation and CityFurniture objects (these limitations have been also acknowledged by the 3DCityDB development team).

Furthermore, when it comes to the major "changes" regarding CityGML:

- There have been no issues associated with the decision to "drop" the Relief module. As originally envisioned, all terrain objects could be mapped to the target classes in the respective modules (WaterBody, Transportation, etc.) with the LandUse module taking up (for the most part) its role. This decision allows, among the rest, to support fully 3D meshes also for terrain objects, instead of 2.5D TINs;
- The LandUseClosureSurface class has been used and it works as planned to "seal" the terrain in places where a 3D object (like a building or a bridge) is removed but preserving a water-tight surface;
- The required metadata for temporal and geometrical accuracy information regarding the geometry objects can be stored successfully.

Figures 20 and 21 show an example of 3D urban objects mapped to and integrated into the semantic 3D city model of Vienna based on the data model presented in this paper.

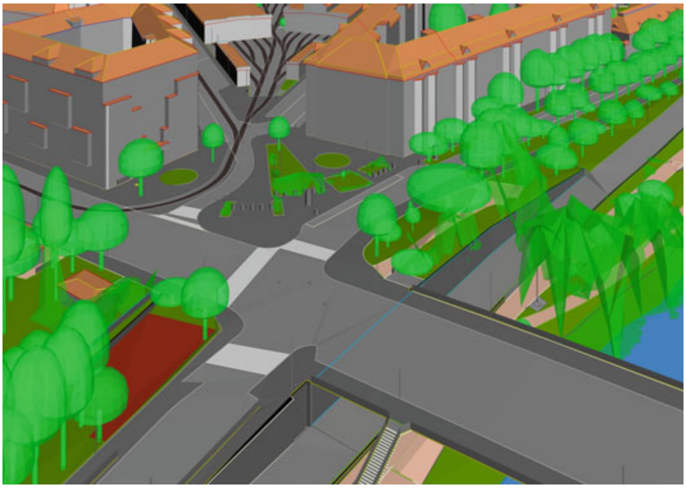

Fig. 20 Detail view of the heterogeneous objects in the test area, modelled using the Digital geoTwin data model developed for Vienna

Fig. 21 Overview of the heterogeneous objects in the test area, modelled using the Digital geoTwin data model developed for Vienna

6 Conclusions

The creation, management and usage of a (geospatially enabled) digital twin is a complex process and challenge that many cities in the world are starting to learn from and to deal with. On the one hand, the process is strongly dependent on the context and the intended goals, the underlying IT infrastructure, and the intended applications, on the other hand, geodata play in any case a definitely relevant role.

Specifically geodata (modelling, storage, management) was the core of the project Digital geoTwin in Vienna. One of its goals has been the definition of a suitable data model to deal with (vector-based) spatial data of all most relevant urban objects generally available and surveyed in a city, in order to create a semantic (3D) virtual model of the city.

The purpose of this data model is to provide a coherent and integrated source of semantically enriched geospatial data covering all spatial entities that will be part of the Digital geoTwin of Vienna.

Part of the project was to test whether, how and to which extent the international standard CityGML version 2.0 could be adopted (and if needed, modified) for the needs of the Municipality of Vienna. 3D modelling of all city objects was carried out in a test area of the City of Vienna. These results were used to develop and test the data model.

At the end of the project, results and the experiences collected allow us to say that CityGML can be used as a reference data model for the Digital geoTwin of the City of Vienna, from which a profile has been extracted and to which a relatively small ADE has been added. At least to the best of the author's knowledge, there are no other examples in literature where such an extensive work of data and workflow analyses, 3D modelling, mapping and transformation has been carried out in the context of CityGML-based city models at national or international level.

In conclusion, we can say that CityGML is a very rich, but rather complex data model which can be configured and extended in order to comply with specific needs. However, it requires a good level of knowledge, not only in terms of theory, but also about current software availability, functionalities and limitations. From the point of view of geodata producers CityGML represents a good starting point for the development of a semantic 3D city model. From the point of view of the data users, CityGML still suffers from the insufficient support from the side of commercial software. For example, some software products only support the Building module, as this reflects probably the current usage of CityGML in "real life", as already pointed out in Chap. 3.1. This is also reflected in the definition of the LoD concept, which is quite well defined and established for buildings, but still rather unclear, or poorly documented for other modules. Nevertheless, if support from commercial software might be sometimes a bit behind, it must be noted that support from open-source initiatives has been boosting in the recent years, however with all pros and cons tied to it (especially from the point of view of a department of a municipality). Particularly welcome is a recently released new open-source plugin for QGIS that allows to interact with the CityGML-based model stored in the 3DCityDB directly from QGIS (Agugiaro et al. 2023).

6.1 Outlook

Although the results obtained are satisfactory, there are still some topics that need particular attention. Some will be listed here as points of future investigation.

In general, the Digital geoTwin project has been carried out using CityGML version 2.0 and trying to avoid potential incompatibilities with the (back then) upcoming version 3.0. In fact, CityGML 3.0 comes with some interesting changes and additions that are expected to contribute to the reduction of the customization level required by the current CityGML profile plus DigitalgeoTwin ADE. To start with, LoD0 geometries are now supported by all CityObjects, therefore part of the ADE would not be needed anymore.

The new "space concept" brings new possibilities for modelling legal and logical spaces such as clearance gauges or planning schemas. This does not affect the Digital geoTwin but is relevant for a digital urban twin and, thus, will be assessed in the future.

Finally, using the Versioning module of CityGML 3.0 it should be possible to model the time-dependent changes of a city. Since changes play a relevant role both in past and future it is crucial to understand and predict (or plan) the urban evolution trends.

The linking of heterogeneous existing, external datasets plays another major role. CityGML allows to link to external resources via ExternalReference objects. However, linking heterogeneous systems or management processes can be more complex, considering their temporal aspect and their possible validity interval.

Preliminary studies are already on-going, but no final results have been obtained so far.

From the user perspective, the possibility to exploit a digital twin to perform urban planning analyses and simulations opens the door to many new applications. Still, different tools, different set of geometries and attributes are generally used for specific analyses, and another level of complexity is given by the generation of additional alternative scenarios that accompany the “status-quo”. Dealing with such richness and complexity of data (formats, size, scales, resolutions…) is surely within the envisioned “playground” of an urban Digital geoTwin, but still subject of current research worldwide. To start with, at least from the point of view of the (Vienna) municipality, there must be an integrated approach underlying distinct processes for data generation and management between “real and surveyed” data, and simulated, synthetic ones.

References

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, 3:2. SpringerOpen, United Kingdom (2018)

Agugiaro G, Pantelios K, León-Sánchez C, Yao Z, Nagel C (2023) Introducing the 3DCityDB-tools plug-in for QGIS (this same conference proceedings)

Biljecki F, Ledoux H, Stoter J (2016) An improved LOD specification for 3D building models. Comput Environ Urban Syst 59:25–37. https://doi.org/10.1016/j.compenvurbsys.2016.04.005

Biljecki F, Kumar K, Nagel C (2018) CityGML application domain extension (ADE): overview of developments. Open Geospatial Data Softw Stand 3(1):1–17

Durão LFCS, Haag S, Anderl R, Schützer K, Zancul E (2018) Digital twin requirements in the context of industry 4.0, product lifecycle management to support industry 4.0. Springer International Publishing, pp 204–214

Gröger G, Kolbe TH, Nagel K, Häfele K-H (2012) OGC city geography markup language (CityGML) encoding standard

Kolbe TH, Kutzner T, Smyth CS, Nagel C, Roensdorf C, Heazel C (2022) OGC city geography markup language (CityGML) part 1: conceptual model standard

Kumar K, Labetski A, Ohori KA, Ledoux H, Stoter J (2019) Harmonising the OGC standards for the built environment: a CityGML extension for Landinfra. ISPRS Int J Geo Inf 8(6):246. https://doi.org/10.3390/ijgi8060246

Lehner H, Dorffner L (2020) Digital geoTwin Vienna: towards a digital twin city as Geodata Hub. Deutsche Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation (DGPF) 88:63–75. https://doi.org/10.1007/s41064-020-00101-4

Löwner M-O, Benner J, Gröger G, Gruber U, Häfele K-H, Schlüter S (2012) CityGML 2.0—Ein internationaler standard für 3D-Stadtmodelle, Teil 1: Datenmodell. ZFV—Zeitschrift für Geodäsie, Geoinformation und Landmanagement, pp 340–349

Lutz B (2019) Data excellence—IKT-Teilstrategie, Magistratsdirektion der Stadt Wien, Geschäftsbereich Organisation und Sicherheit. Prozessmanagement und IKT-Strategie. https://digitales.wien.gv.at/wp-content/uploads/sites/47/2019/03/Data-Excellence.pdf

Rosen R, Von Wichert G, Lo G, Bettenhausen KD (2015) About the importance of autonomy and digital twins for the future of manufacturing. In: 15th IFAC symposium on information control problems in manufacturing, vol 48, pp 567–572

Van den Brink L, Stoter J, Zlatanova S (2013) UML-based approach to developing a CityGML application domain extension: UML-based approach to developing a CityGML application domain extension. Trans GIS 17(6):920–942

Yao Z, Nagel C, Kunde F, Hudra G, Willkomm P, Donaubauer A, Adolphi T, Kolbe TH (2018) 3DCityDB—a 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML. Open Geospatial Data Softw Stand 3:5

Ying Y, Koeva M, Kuffer M, Zevenbergen J (2022) Toward 3D property valuation—a review of urban 3D modelling methods for digital twin creation. ISPRS Int J Geo Inf 12(1):2