Developing a CityGML-based Graph Data Model for Utility Infrastructure in Smart Cities

Ensiyeh Javaherian Pour[1], Behnam Atazadeh[1], Abbas Rajabifard[1] and Soheil Sabri[2]

[1] The Centre for Spatial Data Infrastructure and Land Administration, Department of Infrastructure Engineering, The University of Melbourne, Victoria 3010, Australia - (ensiyeh, behnam.atazadeh, abbas.r)@unimelb.edu.au
[2] Urban Digital Twin Lab, School of Modelling Simulation and Training, University of Central Florida, Orlando, USA - soheil.sabri@ucf.edu

Keywords: GSW 2025, Smart City Infrastructure, CityGML, Utility Networks, Graph Data Model

Abstract

Graph data models are essential for the development of smart cities, where interconnected systems such as utility networks, transportation, and IoT devices must function cohesively. The complexity of smart city infrastructure necessitates 3D data structures capable of managing intricate relationships, dynamic environments, and high connectivity across diverse systems. Graph data models are particularly suited for this purpose, as they offer an integrated 3D digital representation of urban complexity and interconnectivity. This study employs the Labelled Property Graph (LPG) framework to develop a 3D graph data model based on the Utility Network Application Domain Extension (ADE) of the CityGML standard. The proposed approach enhances utility network data management, enabling advanced analyses such as connectivity assessment and pathfinding. The developed graph data model is evaluated in terms of constraint preservation, information integrity, and connection realism. Results demonstrate that the model accurately represents real-world utility network structures while preventing data loss and duplication.

1. Introduction

1.1 Challenges of Managing Utility Networks in Smart Cities

Utility networks, such as electricity, gas, oil, water, and sewerage, are typically hidden beneath layers of soil and urban infrastructure (Yani Lai, 2023). This creates significant challenges, particularly in smart city environments, where accurate and real-time infrastructure management is critical (Farid, 2024). Construction projects often encounter difficulties due to limited, outdated, or fragmented information about the precise location of utility networks (Saeidian et al., 2024).

Excavation and drilling without accurate knowledge frequently result in unintended damage, disrupting essential services such as water, gas, and electricity, potentially cutting off entire neighbourhoods. These incidents lead to costly repairs and operational challenges for utility providers while also causing significant delays in construction timelines (Yong-Kang Qiao, 2022). In a smart city context, such disruptions not only impact immediate services but also hinder long-term infrastructure planning and management (Rathore et al., 2021). Consequently, integrated utility network management is essential for minimising accidents and ensuring smoother, safer operations in both construction projects and ongoing urban infrastructure management (Den Duijn et al., 2018).

1.2 Graph Data Models for Utility network Management

In recent years, several 3D spatial data models have been developed to address the challenges of modelling utility networks. One of the most widely used 3D spatial data models for smart cities is CityGML, an international standard recognised by the Open Geospatial Consortium (OGC) (OpenGeospatialConsortium, 2021). An Application Domain Extension (ADE) of CityGML has been developed to provide a foundation for 3D digital management and representation of utility networks, enabling advanced 3D data analysis, simulation, and visualisation (Saeidian et al., 2024).

However, CityGML's data model, traditionally structured as a relational data model, faces significant limitations in managing the highly connected and dynamic nature of utility networks (Hor et al., 2018). Relational data models rely on rigid tabular structures and require costly join operations to manage relationships, leading to inefficiencies when dealing with large-scale urban environments and complex networks such as utility networks (Ilonen, 2023) . These inefficiencies become particularly evident in scenarios requiring real-time queries and continuous updates to both underground and above-ground infrastructure, where performance degradation in relational databases is a common issue (Rathore et al., 2021).

To address the limitations of relational data models, graph data models offer a compelling alternative. Unlike relational models, graph data models are inherently designed to manage complex, interconnected data by representing entities as nodes and relationships as edges (Kankanamge et al., 2017). This structure enables faster querying and traversal of relationships without the need for expensive table joins (Ji, 2020). When implemented in graph databases, graph data models excel at handling the intricate structures and topologies of utility networks, efficiently managing the dynamic relationships between underground components and above-ground infrastructure (Jamkhedkar et al., 2018). Furthermore, graph data models are particularly well-suited for smart cities, where real-time data access, scalability, and efficient infrastructure management are crucial for urban planning, monitoring, and operations.

1.3 Objective and Scope of the Paper

In this paper, we propose a 3D graph data model based on the Utility Network ADE in CityGML. Translating the data elements from the Utility Network ADE into a graph data model requires redefining certain elements to align with the graph data modelling paradigm. Specifically, entities in the relational model are mapped to nodes, while their attributes are represented as properties within the graph. However, in cases where attributes encapsulate significant relationships or complexity, they are transformed into nodes rather than simple properties. This allows for more expressive queries and a more efficient representation of interconnected data.

By converting the Utility Network ADE features into nodes, edges, and properties within a Labelled Property Graph (LPG), the inherent advantages of graph structures can be leveraged for enhanced utility network data management. This approach also aligns with the broader objectives of smart city infrastructure by facilitating advanced analyses, such as connectivity assessment and pathfinding

2. Literature Review

2.1 Graph Theory in Smart City

Smart cities represent the transformation of urban environments through the integration of advanced technologies (Lim et al., 2024). At the core of a smart city is the interconnection of various systems, such as transportation, energy grids, utility networks, public services, and environmental monitoring (Spicer et al., 2023). These interconnected systems, with their many subsystems, inherently lend themselves to graph theory applications. Graph theory excels in managing dynamic models and highly connected systems, where relationships between entities continuously evolve over time (Pierfrancesco Bellini, 2018).

In mathematical terms, a graph consists of vertices (or nodes) and edges, which represent the relationships between these entities. Formally, a graph is defined as

$$G = (V, E)$$

where:
V: denotes a set of vertices, representing individual entities.
E: represents the set of edges, defining the connectivity relationships between these vertices.

In a smart city, infrastructure can be conceptualized as a graph, where nodes represent entities such as sensors, devices, or infrastructure components, and edges represent relationships or interactions, such as data flow, physical connections, or service dependencies (Gorawski and Grochla, 2019). Among the graph models commonly employed in smart city applications are Resource Description Framework (RDF) graphs and LPGs.

2.2 RDF Graphs in Smart City

RDF is a data model that represents information as triples: subject, predicate, and object. This structure allows RDF to form a graph of interconnected data points, where relationships between entities are explicitly defined. RDF is widely used to establish semantic relationships across diverse datasets, which is particularly valuable in scenarios where data interoperability is essential (Wu et al., 2024). In the context of smart city applications, RDF offers a standardised approach to describing and linking data from various systems, such as Internet of Things (IoT) networks, transportation, and utilities. Its semantic nature facilitates the integration of data across different platforms, making it more efficient to manage and analyse the relationships between various city infrastructures (Bellini et al., 2015). Additionally, RDF's compliance with web standards ensures its effective application in global, interconnected systems (Wu et al., 2024).

RDF's structured format is particularly advantageous for applications that require interoperability between diverse datasets, especially in smart city environments where multiple systems must work in coordinate. Several studies have explored the use of RDF for enhancing smart city data management, each contributing to a more integrated approach to handling complex urban systems. One notable study translates CityGML into RDF, which allows for more sophisticated queries over CityGML data and supports real-time analysis by facilitating the integration of external data sources such as OpenStreetMap. This methodology significantly improves the ability to manage the hierarchical and semantic complexities inherent in urban datasets (Ding et al., 2024).

Building on this, another approach extends the use of RDF by converting Industry Foundation Classes (IFC) into CityGML using RDF graphs. This approach not only enhances smart city data management but also enables more advanced querying capabilities and fosters seamless interaction between various city systems. In increasingly interconnected urban environments, RDF provides the semantic framework necessary to unify these diverse data sources. Such integration is especially critical for managing the relationships between underground utility networks and above-ground infrastructure (Lam et al., 2024).

Moreover, RDF has been applied to create dynamic geospatial knowledge graphs. For example, CityGML data has been transformed into RDF to construct a Semantic 3D City Database, which facilitates flexible and dynamic interactions with large-scale urban datasets. This application further demonstrates RDF's capacity to enable efficient management of the complexities of modern urban environments (Chadzynski et al., 2023)

2.3 LPGs in Smart City

LPGs offer a flexible and efficient data model for managing complex, interconnected systems, making them particularly suitable for smart city applications (Ning et al., 2024). In an LPG, nodes represent entities, while edges represent the relationships between them, both of which can store multiple properties. This allows LPGs to handle rich metadata and facilitate real-time querying, a critical requirement in smart cities where infrastructure systems, utilities, and IoT networks must constantly interact and be analysed (Ferilli et al., 2022).

While research on the direct conversion of CityGML into LPGs is still developing, the approach demonstrates significant promise for managing 3D spatial data, largely due to its scalability and capacity to handle complex, multi-dimensional relationships. A notable methodology involves first converting both IFC and CityGML into RDF, which is then transformed into an LPG database. This process enhances data integration and enables advanced analyses in urban planning, mobility, and infrastructure management within smart cities (Hor et al., 2018a). In parallel, more attention has been devoted to converting IFC data into LPGs, particularly in Building Information Model (BIM) and smart city infrastructure applications. Several graph-based approaches have been proposed to transform IFC models into LPGs, integrating real-time sensor data to improve data management and analysis, which is especially beneficial for monitoring systems within smart cities (Gradišar and Dolenc, 2021). Furthermore, a model-driven approach has been developed to fully convert IFC data into LPGs, significantly improving accessibility and query efficiency for building information. This conversion not only simplifies the discovery of hidden relationships within building data but also eliminates the need for specialised IFC parsers, further streamlining data management (Zhu et al., 2023).

Compared to traditional relational databases, LPGs provide a more intuitive representation of the complex relationships within IFC data. By using LPGs, critical information is preserved, and entities are accurately associated, making it an ideal solution for managing building data in complex construction projects. LPGs also capture intricate relationships between IFC entities with greater efficiency, supporting more effective data analysis and visualisation (Zhao et al., 2020).

Although numerous studies have explored the application of RDF in smart cities, several limitations obstruct its suitability for managing the complex and dynamic data in these environments. RDF, as a semantic web standard, is primarily designed for machine interpretation through ontologies and structured data (Pauwels et al., 2017). While RDF excels in semantic data integration, it often reduces human readability and usability, especially when dealing with the complex data structures of city infrastructure.

One of RDF’s main limitations lies in its triple-based structure, which can restrict its efficiency when managing rich properties directly on nodes and edges, as required for CityGML’s multi-dimensional city data (Zhu et al., 2023). Furthermore, RDF’s query language, SPARQL, though highly expressive, is notably more complex and less intuitive than Cypher, which is commonly used in LPGs. This added complexity can slow down real-time querying, particularly in large-scale smart city environments where fast data access is essential (Angles et al., 2019).

Another drawback of RDF is its reliance on proprietary tools or third-party solutions to improve performance. While these tools can enhance RDF’s efficiency, they are often not fully optimised or scalable for the real-time data demands typically generated in smart city infrastructures. As a result, RDF struggles to meet the needs of smart cities that require continuous data updates and real-time analytics across interconnected systems such as transportation, utilities, and IoT networks (Alocci et al., 2015).

In contrast, LPGs offer a more flexible and intuitive structure for managing the complex relationships inherent in CityGML. LPGs allow both nodes and edges to carry multiple properties, which provides a more natural representation of interconnected city systems. This makes LPGs better suited for real-time querying and handling rich metadata, enabling faster performance when processing large-scale, dynamic datasets (Zhu et al., 2023).

LPGs are also more suited for graph traversal and pathfinding operations, which are critical for smart city applications like urban planning, infrastructure monitoring, and mobility management (Jamkhedkar et al., 2018).

The inherent scalability and high performance of LPGs make them a more practical solution for converting CityGML into graph data models. LPGs offer a more efficient way to represent spatial relationships and support faster, more effective querying, positioning them as the preferred choice for managing 3D city models in smart city infrastructures.

3. Methodology

In this study, we propose the conversion of the relational data model of the Utility Network ADE in CityGML into a graph data model using the LPG framework. An LPG consists of three core components: nodes (entities), edges (relationships), and properties (attributes), which together represent the data structure in a highly connected network. Each node and edge can have a label that defines its role or type, while properties store additional information that describes the nodes and edges in more detail.

3.1 Nodes

In graph data models, nodes represent entities or objects within a utility network, and they can be derived from two sources in the Utility Network ADE: Directed Nodes and Undirected Nodes.

3.1.1 Directed Nodes: Directed nodes originate from FeatureTypes in the Utility Network ADE, which represent key components within the utility network, such as Network, AbstractNetworkFeature, or NetworkGraph. In converting the data to a graph model, each FeatureType becomes a node, with the node’s label reflecting the name of the FeatureType (e.g., "Network"). This label categorises the nodes and defines their function in the utility network. Directed nodes act as the core entities within the graph, representing the fundamental structure of the utility network.

3.1.2 Undirected Nodes: Undirected nodes are derived from properties within a FeatureType that reference other DataTypes or FeatureTypes. In the relational data model, certain properties of a FeatureType may be linked to external entities with their own attributes and relationships. When converting this structure into a graph data model, these referenced entities are treated as separate nodes to maintain the integrity of the relationships.

As shown in Figure *1*, within the Network entity, the property *relatedParty* is connected to the RelatedParty data type. This entity, in turn, includes a property named *party*, which is linked to another FeatureType, Party. Additionally, within the Party FeatureType, the property *pointOfContact* refers to the ContactType data type. This example illustrates how a single property, *relatedParty*, in the Network entity results in the creation of multiple nodes (RelatedParty, Party, and ContactType).

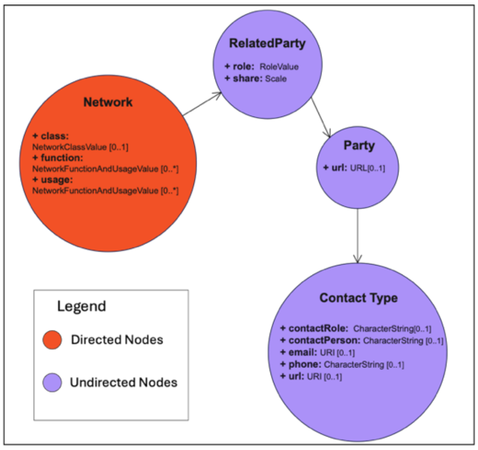

Figure 1. Directed and undirected nodes in the Network entity

3.2 Edges

In the graph data model, edges represent the relationships between nodes (both directed and undirected). When translating the relational data model from the Utility Network ADE CityGML to a graph data model, existing relationships are transformed into edges with distinct labels and attributes. Depending on the nature of the relationship in CityGML, the edge can either be directed or self-Loop.

https://doi.org/10.5194/isprs-annals-X-G-2025-405-2025 | © Author(s) 2025. CC BY 4.0 License.

3.2.1 Directed Relationships: Directed relationships represent connections where one entity is superior or related hierarchically to another, such as composition, inheritance, or associations. In these cases, the edge is directed from the superordinate node (the parent entity) to the subordinate node (the child entity). The edge is labelled according to the type of relationship (e.g., inheritance, composition) as defined in the Utility Network ADE, with its properties capturing the specifics of that relationship.

For instance, the AbstractLink entity has an association relationship with the Node FeatureType. In the relational data model, this association is represented by a property called *startNode*. When converting this into the graph data model, an edge is created from the AbstractLink node to the Node node. This edge is labelled "Association Relationship," and the property *startNode* is retained to preserve the original relationship's definition (Figure *2*).

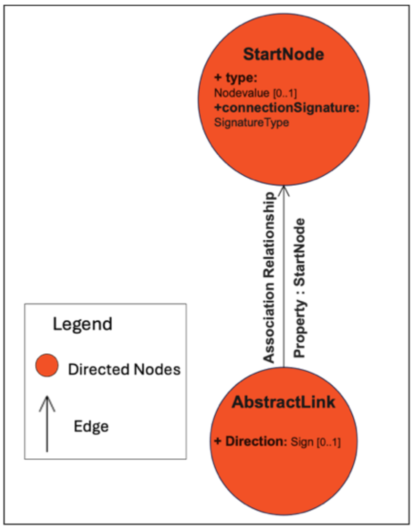

Figure 2. Association Relationships with property *StartNode*

3.2.2 Self-Loop Relationships: In some instances, an entity may have a relationship with itself, represented as a self-loop in the graph data model. For example, in the Utility Network ADE, a subnetwork has an aggregation relationship with the Network FeatureType. This relationship is depicted by an edge that starts and ends at the same Network node, forming a self-loop. In this case, the edge is labelled "Self-Loop Relationship," and its properties capture the aggregation type as defined by the Utility Network ADE.

3.3 Properties (Attributes)

In the Utility Network ADE, each entity (FeatureType) is associated with a set of attributes that describe its characteristics and details. When converting the relational data model to a graph data model, these attributes are translated into properties for the corresponding nodes and edges in the graph. Properties provide additional descriptive information that is crucial for managing and querying the network effectively.

For each node, the properties derived from the original attributes in Utility Network ADE are attached to the node as key-value pairs. These properties can include specific details such as dimensions, material types, operational status, and other relevant metadata. Similarly, properties can be applied to edges to describe the nature of the relationship between nodes, such as start and end points, capacity, or directional flow.

If additional properties are required for specific project needs or dataset requirements, the graph data model offers flexibility in defining new properties. It is important, however, to carefully specify the data type, value range, and whether these attributes should be represented as an enumeration, a code list, or a single value. This careful definition ensures consistency and clarity throughout the data model.

4. Result

In this section, we demonstrate the conversion of Utility Network ADE models from CityGML into a graph data model, specifically using LPGs. To evaluate the feasibility and effectiveness of this approach, we selected several reference utility network models from the Utility Network ADE study (Becker et al., 2011) to illustrate how these networks can be represented as graphs.

To avoid making the graph data models overly complex, certain unnecessary properties and some geometry nodes are excluded for simplicity. However, the full model includes all the required properties, nodes, and relationships. This approach helps maintain readability while still preserving all essential relationships and constraints.

4.1 Exterior and Interior Nodes in LPGs

We provide a sample utility network extracted from study on the Utility Network ADE (Figure 3) (Becker et al., 2011). Figure 3 represents a network of pipes located within the same FeatureGraph and connected through an InteriorFeatureLink. The sample highlights an important challenge of two distinct types of nodes (exterior and interior) within a single FeatureGraph.

The exterior nodes (represented as red circles) denote the endpoints of the pipes, while the interior nodes (represented as green diamonds) signify connection points where multiple pipes converge within the FeatureGraph. Managing the interaction between these different node types is essential for accurately modelling the flow and connectivity of the utility network in a graph database built on the proposed graph data model.

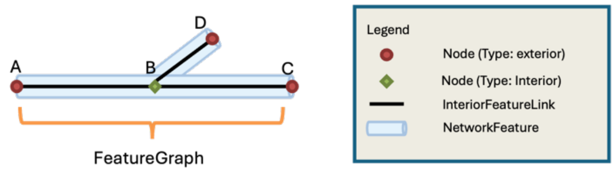

Figure 3. FeatureGraph sample including exterior and interior nodes

Figure 4 represents the result of translating the FeatureGraph in Figure 3 into a graph data model. In the graph data model (Figure 4):

- The yellow nodes represent the pipe from start point A to end point B.
- The blue nodes represent the pipe from start point B to end point C.
- The purple nodes represent the link from start point B to end point D.

In Figure 3, the three InteriorFeatureLinks are part of the same FeatureGraph. This constraint is respected in the graph data model (Figure 4), where all three InteriorFeatureLinks (yellow, blue, purple) originate from a single FeatureGraph node.

All InteriorFeatureLinks are connected to an AbstractLink in the graph model, representing the direction of each pipe. The GM_Curve node stores the geometry of each InteriorFeatureLink, ensuring that the spatial and topological information is preserved within the graph.

Each InteriorFeatureLink via an AbstractLink includes two nodes, one for the start and one for the end. The type of each node (whether it is exterior or interior) is stored as a property on the nodes. This provides detailed information about the role of each node in the network.

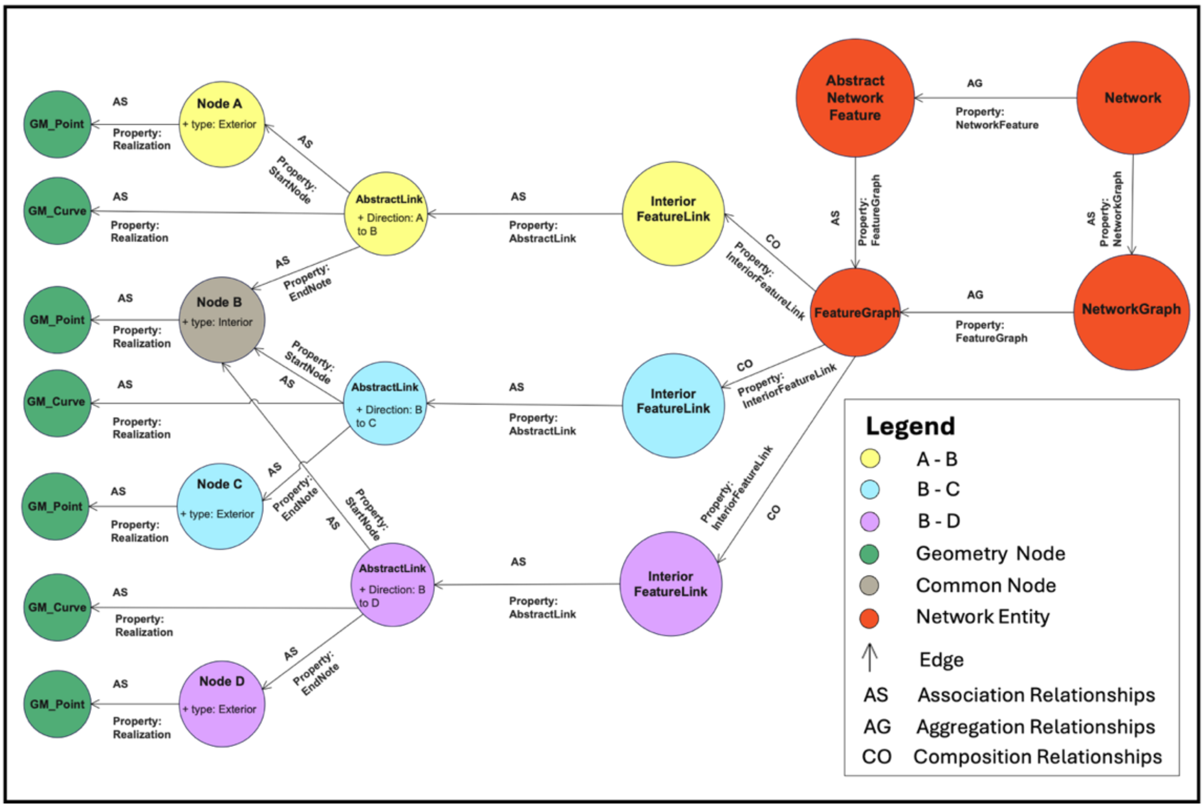

Figure 4. Graph data model from single FeatureGraph including exterior and interior nodes

To avoid duplication, the graph data model ensures that point B, which connects to point A, point C, and point D in Figure 3, is represented as a single node (B). This common node retains its connections to all three points, preserving the network structure.

Finally, for each node, the geometry is stored in a separate GM_Point. This separation allows for efficient querying and management of spatial data while preserving the integrity of the original utility network structure.

4.2 InterFeatureLink in LPGs

In this section, we provide another important sample from the Utility Network ADE study (Becker et al., 2011). The diagram in Figure 5 represents the connection between two FeatureGraphs within the same NetworkGraph. This complexity arises when multiple FeatureGraphs coexist in a single network, and their interactions need to be modelled while preserving all relationships and constraints.

The graph data model (Figure 6) allows us to accurately represent this connection between FeatureGraphs through a combination of InteriorFeatureLinks and InterFeatureLinks.
In Figure 6 graph data model:

- The purple nodes represent FeatureGraph 1.
- The blue nodes represent FeatureGraph 2.
- The orange nodes represent the InterFeatureLink connecting FeatureGraph 1 and FeatureGraph 2.

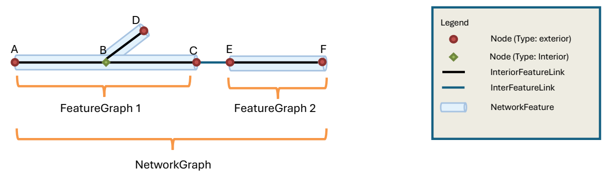

Figure 5. NetworkGraph sample including InteriorFeatureLink and InterFeatureLink

Both FeatureGraphs are part of the same NetworkGraph, as illustrated in Figure 5. The graph data model in Figure 6 captures two distinct FeatureGraphs originating from the same NetworkGraph.

FeatureGraph 1 follows the same structure as in the previous graph model (Figure 4), consisting of both interior and exterior nodes connected by three InteriorFeatureLinks.
FeatureGraph 2 is simpler than FeatureGraph 1, containing only two exterior nodes without any interior nodes, connected by a single InteriorFeatureLink.

The key complexity in this model lies in the presence of the InterFeatureLink, represented by the orange nodes, which signifies the connection between the two FeatureGraphs. The InterFeatureLink connects node C in purple (FeatureGraph 1) with node E in blue (FeatureGraph 2). Both nodes are connected to the InterFeatureLink via an AbstractLink, ensuring that the direction and flow of the connection are maintained, along with the geometry of the InterFeatureLink line.

5. Evaluation and Discussion

The evaluation of the proposed graph data model is divided into three key areas: constraint preservation, information integrity, and connection realism. These aspects ensure that the graph model aligns with the structure and rules defined in the CityGML Utility Network ADE while maintaining the completeness and accuracy of the data. By verifying the predefined constraints, checking for missing or duplicated information, and assessing whether the graph accurately reflects real-world relationships, this evaluation aims to confirm the feasibility and effectiveness of the graph data model in managing utility networks within smart city infrastructures.

5.1 constraint preservation

In any data model translation process, especially one involving complex systems like the CityGML Utility Network ADE, it is essential to ensure that the integrity of the original model's constraints is preserved. These constraints serve a critical function in maintaining the logical relationships and hierarchical structures between various network components, such as nodes, and links. The preservation of constraints is particularly important for maintaining the accuracy of real-world representations, especially in utility networks where misrepresentation of connections or mismanagement of node properties could lead to errors in network analysis or operational inefficiencies.

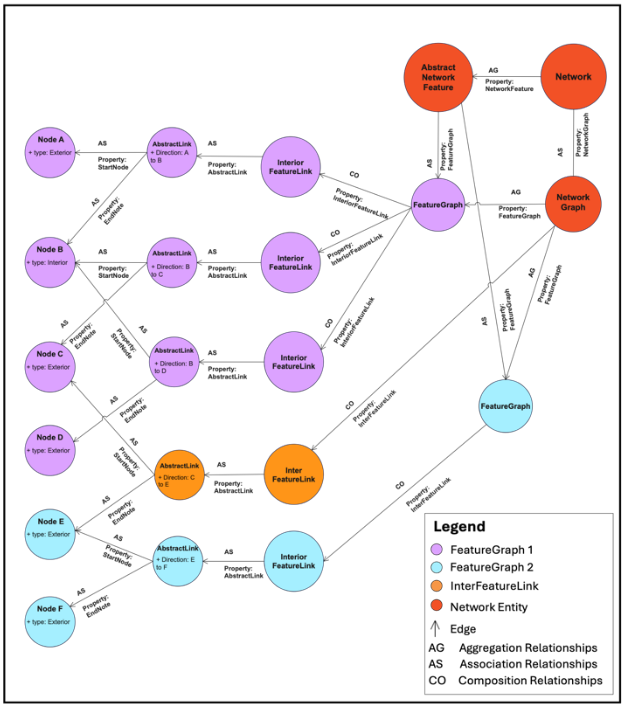

Figure 6. NetworkGraph sample including InteriorFeatureLink and InterFeatureLink

5.1.1 NetworkGraph Constraints

The NetworkGraph entity in the CityGML Utility Network ADE imposes a constraint stating, *"When the representation is defined, all start and end nodes of the related feature graphs and inter feature links must be part of the same schematic type."* This ensures that the nodes and links maintain a consistent schematic representation within a network.

In the graph data model of FeatureGraph with exterior and interior nodes (Figure 4), all the InteriorFeatureLinks originate from the same FeatureGraph, which is directly connected to the NetworkGraph. Therefore, all the InteriorFeatureLinks, along with their associated nodes, maintain a single, unified representation within the NetworkGraph, satisfying the constraint.

In a more complex scenario involving two connected FeatureGraphs within a NetworkGraph (Figure 6), the InterFeatureLink connects two distinct FeatureGraphs. All three entities, the two FeatureGraphs and the InterFeatureLink, originate from the same NetworkGraph. As a result, the shared representation across the network ensures that the constraint is respected, maintaining consistency across the entire system.

5.1.2 InteriorFeatureLink Constraints

The InteriorFeatureLink entity in CityGML Utility Network ADE has a key constraint: *"Both nodes must belong to the same FeatureGraph."* This ensures that the start and end nodes of an InteriorFeatureLink are consistently placed within the same feature graph, maintaining the integrity of internal connections.

When modelling one FeatureGraph containing both exterior and interior nodes (Figure 4), all start and end nodes of the InteriorFeatureLinks connect directly to the related FeatureGraph. In this case, the constraint is satisfied as all nodes involved in the links are part of the same FeatureGraph, ensuring a cohesive structure.

In the case of two FeatureGraphs (Figure 6), FeatureGraph 1 follows the same rule, maintaining the constraint that the nodes belong to the same graph. FeatureGraph 2 similarly connects its start and end nodes to its corresponding InteriorFeatureLink, which adheres to the same FeatureGraph.

5.1.3 InterFeatueLink Constraints

The InterFeatureLink entity in CityGML Utility Network ADE imposes several constraints to ensure proper connections between different FeatureGraphs. The constraints state:

1. *Each Node must belong to a different FeatureGraph.*
2. *Each Node type must be exterior.*
3. *The connectionSignature of both Nodes must be compatible.*
4. *Both Nodes must either belong to the same Network or to two Networks of the same commodity and hierarchy level.*

The presence of the InterFeatureLink is crucial in cases where two FeatureGraphs are involved. As illustrated in Figure 6, the start node originates from FeatureGraph 1 (represented in purple), while the end node comes from FeatureGraph 2 (represented in blue). Both nodes are of exterior type, satisfying the first and second constraints.

Furthermore, the connectionSignature of both nodes is compatible, which can be reflected in the connectionSignature property of the graph data model, ensuring the third constraint is met. Additionally, both nodes are part of the same NetworkGraph, ensuring that the nodes either belong to the same network or have compatible hierarchy, satisfying the fourth constraint (Figure 6).

https://doi.org/10.5194/isprs-annals-X-G-2025-405-2025

5.2 Information Integrity

Ensuring information integrity during the translation from the Utility Network ADE to the graph data model is crucial for maintaining the original structure and meaning of the network data. In the graph data model, all entities from the Utility Network ADE are preserved. Also, certain properties are translated into nodes to facilitate easier querying and retrieval of interconnected information (Figure *1*). This translation does not lead to information duplication but rather enhances accessibility by converting properties into nodes, making it simpler to manage and query complex relational data. All attributes are carefully preserved and assigned to their corresponding properties, ensuring no data is lost during the translation. Additionally, the types of relationships between entities are maintained by labelling them appropriately in the graph model.

This avoidance of duplication and preservation of all information is mirrored in the results as well. For instance, in the case of node B (Figure 4), which connects to three InteriorFeatureLinks (AB, BC, and BD), the graph model ensures that node B is not duplicated but is referenced by each of the other nodes and links.

5.3 Connection Realism

In this section, we assess the realism of connections by comparing the structures in Figure 3 and Figure 5 with their corresponding graph data models in Figure 4 and Figure 6. The degree of a node in the graph data model reflects the number of connections in the real-world utility network. Since Figure 3 is part of the larger structure shown in Figure 5**Errore. L'origine riferimento non è stata trovata.**, we focus on Figure 5 for a comprehensive assessment.

Node	Real-world connections	Node degree
A	1	1
B	3	3
C	2	2
D	1	1
E	2	2
F	1	1

Table 1. Noce connections comparison

The degree of connectivity in the graph data model matches the real-world utility network structure, confirming that the model accurately captures the relationships in the data.

6. Conclusion and Future Works

In this paper, we demonstrated that a graph data model based on CityGML can be used to represent utility networks. Graph data models offer significant advantages for smart cities by enhancing connectivity, and pathfinding. Graph data models have capability to represent complex relationships and connections between various city infrastructure components. In addition, for pathfinding and route optimization, graph data models prove especially effective, particularly for tasks such as routing maintenance within the utility networks. Also, it has ability to map complex relationships between nodes and dynamic route planning.

Traditional relational data models require separate conceptual, logical, and physical phases. In contrast, graph data models improve this process by combining the conceptual and logical layers, allowing for direct implementation into graph databases once the data model is prepared.

Transitioning from relational databases to graph databases for managing large volumes of highly connected, dynamic data is essential as smart cities develop. Graph databases are equipped to handle the complexities of modern city infrastructure compared to traditional, schema-bound relational databases.

As cities become increasingly complex, the flexibility and scalability of graph databases will be essential for handling the growing data in smart city infrastructure.

While the proposed graph data model successfully translates the CityGML Utility Network ADE into an LPG, there are several limitations to consider.

- The model has been tested on specific utility networks and may require further validation to generalise across different types of urban infrastructures.
- The performance of the graph model in handling large-scale networks and real-time applications has not been extensively evaluated, particularly in scenarios with frequent updates and high query demands, which are common in smart city environments.
- The representation of complex utility networks and their hierarchical models needs further investigation. It is essential to explore whether the graph data model accurately reflects the structure and relationships at various hierarchical levels within utility networks.

Future research can build upon the foundation established in this paper. While our focus was on the core of Utility Network ADE, this ADE also contains several additional packages, including those for Material, Functional Characteristics, Network Components, Geometry of Network Components, Network Properties, and the Electrical Network package. These packages should also be translated into a graph data model and evaluated to assess the full functionality of the utility networks in a graph data model.

After modelling the graph data model, further investigation is needed to implement it in graph databases. This will allow for an in-depth evaluation of the model's feasibility through querying and analysis, providing clearer insights into its practical applications.

Additionally, utility networks represent just one aspect of CityGML. Extending this approach to other CityGML modules or ADEs, such as buildings and roads, could further enhance the integration and management of infrastructures, advancing the development of smart cities.

Acknowledgements

This research was supported by Australian Research Council [grant numbers: DE220100094, IH210100048]. The authors acknowledge the support of industry partners: South East Water and Emerson. The authors emphasize that the views expressed in this article are the authors' alone.

References

Alocci, D., Mariethoz, J., Horlacher, O., Bolleman, J. T., Campbell, M. P. & Lisacek, F. 2015. Property graph vs RDF triple store: A comparison on glycan substructure search. *PloS one,* 10**,** e0144578.

Angles, R., Thakkar, H. & Tomaszuk, D. 2019. RDF and Property Graphs Interoperability: Status and Issues. *AMW,* 2369**,** 1-11.

Becker, T., Nagel, C. & Kolbe, T. H. 2011. Integrated 3D modeling of multi-utility networks and their interdependencies for critical infrastructure analysis. *Advances in 3D geo-information sciences***,** 1-20.

Bellini, P., Nesi, P. & Pantaleo, G. Benchmarking RDF stores for smart city services. 2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity), 2015. IEEE, 46-49.

Chadzynski, A., Li, S., Grišiūtė, A., Chua, J., Hofmeister, M., Yan, J., Tai, H. Y., Lloyd, E., Tsai, Y. K. & Agarwal, M. 2023. Semantic 3D city interfaces—Intelligent interactions on dynamic geospatial knowledge graphs. *Data-Centric Engineering,* 4**,** e20.

Den Duijn, X., Agugiaro, G. & Zlatanova, S. 2018. Modelling below-and above-ground utility network features with the CityGML Utility Network ADE: Experiences from Rotterdam. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences,* 4**,** 43-50.

Ding, L., Xiao, G., Pano, A., Fumagalli, M., Chen, D., Feng, Y., Calvanese, D., Fan, H. & Meng, L. 2024. Integrating 3D City Data through Knowledge Graphs. arXiv 2023. *arXiv preprint arXiv:2310.11555.*

Farid, A. M. 2024. A Hetero-functional Graph Resilience Analysis for Convergent Systems-of-Systems. *arXiv preprint arXiv:2409.04936.*

Ferilli, S., Redavid, D. & Di Pierro, D. LPG-based Ontologies as Schemas for Graph DBs. SEBD, 2022. 256-267.

Gradišar, L. & Dolenc, M. 2021. IFC and Monitoring Database System Based on Graph Data Models. *Advances in Civil Engineering,* 2021**,** 4913394.

Hor, A. H., Sohn, G., Claudio, P., Jadidi, M. & Afnan, A. 2018. A SEMANTIC GRAPH DATABASE FOR BIM-GIS INTEGRATED INFORMATION MODEL FOR AN INTELLIGENT URBAN MOBILITY WEB APPLICATION. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.,* IV-4**,** 89-96.

Ilonen, J. 2023. A Case Study on Transitioning from Relational Data models to Graph Data models in an Industrial Context.

Jamkhedkar, P., Johnson, T., Kanza, Y., Shaikh, A., Shankaranarayanan, N. & Shkapenyuk, V. A graph database for a virtualized network infrastructure. Proceedings of the 2018 International Conference on Management of Data, 2018. 1393-1405.

Ji, Q. 2020. *Geospatial Inference and Management of Utility Infrastructure Networks.* Newcastle University.

Kankanamge, C., Sahu, S., Mhedbhi, A., Chen, J. & Salihoglu, S. Graphflow: An active graph database. Proceedings of the 2017 ACM International Conference on Management of Data, 2017. 1695-1698.

Lam, P.-D., Gu, B.-H., Lam, H.-K., Ok, S.-Y. & Lee, S.-H. 2024. Digital Twin Smart City: Integrating IFC and CityGML with Semantic Graph for Advanced 3D City Model Visualization. *Sensors,* 24**,** 3761.

Lim, Y., Edelenbos, J. & Gianoli, A. 2024. What is the impact of smart city development? Empirical evidence from a Smart City Impact Index. *Urban Governance,* 4**,** 47-55.

Ning, Y., Liu, H., Wang, H., Zeng, Z. & Xiong, H. 2024. UUKG: unified urban knowledge graph dataset for urban spatiotemporal prediction. *Advances in Neural Information Processing Systems,* 36.

Opengeospatialconsortium 2021. OGC City Geography Markup Language (CityGML) Part 1: Conceptual Model Standard.

Pauwels, P., Zhang, S. & Lee, Y.-C. 2017. Semantic web technologies in AEC industry: A literature overview. *Automation in construction,* 73**,** 145-165.

Pierfrancesco Bellini, P. N. 2018. Performance assessment of RDF graph databases for smart city services. *Journal of Visual Languages & Computing,* 45**,** 24-38.

Rathore, M. M., Attique Shah, S., Awad, A., Shukla, D., Vimal, S. & Paul, A. 2021. A Cyber-Physical System and Graph-Based Approach for Transportation Management in Smart Cities. *Sustainability,* 13**,** 7606.

Saeidian, B., Rajabifard, A., Atazadeh, B. & Kalantari, M. 2024. Managing underground legal boundaries in 3D-extending the CityGML standard. *Underground Space,* 14**,** 239-262.

Spicer, Z., Goodman, N. & Wolfe, D. A. 2023. How ‘smart’are smart cities? Resident attitudes towards smart city design. *Cities,* 141**,** 104442.

Wu, D., Wang, H.-T. & Tansel, A. U. 2024. A survey for managing temporal data in RDF. *Information Systems,* 122**,** 102368.

Yani Lai, Y. W., Jing Cheng, Xiangsheng Chen, Quan Liu, 2023. Review of constraints and critical success factors of developing urban underground space. *Underground Space,* 12**,** 137-155.

Yong-Kang Qiao, F.-L. P., Yong-Peng Luan, Xiao-Lei Wu 2022. Rethinking underground land value and pricing: A sustainability perspective. *Tunnelling and Underground Space Technology,* 127.

Zhao, Q., Li, Y., Hei, X. & Yang, M. 2020. A Graph-Based Method for IFC Data Merging. *Advances in civil engineering,* 2020**,** 8782740.

Zhu, J., Wu, P. & Lei, X. 2023. IFC-graph for facilitating building information access and query. *Automation in Construction,* 148**,** 104778.