COMILLAS
UNIVERSIDAD PONTIFICIA
ICAI

MASTER UNIVERSITARIO EN INGENIERÍA DE TELECOMUNICACIONES

TRABAJO FIN DE MASTER

Graph-Based Socio-Technical Information Model for Power Distribution Network Planning and Operations

Autor: Tomás Pérez Gutiérrez

Director académico: Bruce Stephen

Director industrial: Ciaran Higgins

Madrid

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título

Graph-Based Socio-Technical Information Model for Power Distribution Network Planning and Operations

en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el

curso académico 2024/25 es de mi autoría, original e inédito y

no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido

tomada de otros documentos está debidamente referenciada.

Fdo.: Tomás Pérez Gutiérrez Fecha: 13 / 08 / 2025

Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO

Fdo.: Bruce Stephen Fecha: 13 / 08 / 2025

Fdo.: Ciaran Higgins Fecha: 13 / 08 / 2025

AUTORIZACIÓN PARA LA DIGITALIZACIÓN, DEPÓSITO Y DIVULGACIÓN EN RED DE PROYECTOS FIN DE GRADO, FIN DE MÁSTER, TESINAS O MEMORIAS DE BACHILLERATO

1º. Declaración de la autoría y acreditación de la misma.

El autor D. Tomás Pérez Gutiérrez

DECLARA ser el titular de los derechos de propiedad intelectual de la obra: Graph-Based Socio-Technical Information Model for Power Distribution Network Planning and Operations, que ésta es una obra original, y que ostenta la condición de autor en el sentido que otorga la Ley de Propiedad Intelectual.

2º. Objeto y fines de la cesión.

Con el fin de dar la máxima difusión a la obra citada a través del Repositorio institucional de la Universidad, el autor **CEDE** a la Universidad Pontificia Comillas, de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, los derechos de digitalización, de archivo, de reproducción, de distribución y de comunicación pública, incluido el derecho de puesta a disposición electrónica, tal y como se describen en la Ley de Propiedad Intelectual. El derecho de transformación se cede a los únicos efectos de lo dispuesto en la letra a) del apartado siguiente.

3º. Condiciones de la cesión y acceso

Sin perjuicio de la titularidad de la obra, que sigue correspondiendo a su autor, la cesión de derechos contemplada en esta licencia habilita para:

a) Transformarla con el fin de adaptarla a cualquier tecnología que permita incorporarla a internet y hacerla accesible; incorporar metadatos para realizar el registro de la obra e incorporar "marcas de agua" o cualquier otro sistema de seguridad o de protección.
b) Reproducirla en un soporte digital para su incorporación a una base de datos electrónica, incluyendo el derecho de reproducir y almacenar la obra en servidores, a los efectos de garantizar su seguridad, conservación y preservar el formato.
c) Comunicarla, por defecto, a través de un archivo institucional abierto, accesible de modo libre y gratuito a través de internet.
d) Cualquier otra forma de acceso (restringido, embargado, cerrado) deberá solicitarse expresamente y obedecer a causas justificadas.
e) Asignar por defecto a estos trabajos una licencia Creative Commons.
f) Asignar por defecto a estos trabajos un HANDLE (URL *persistente)*.

4º. Derechos del autor.

El autor, en tanto que titular de una obra tiene derecho a:

a) Que la Universidad identifique claramente su nombre como autor de la misma
b) Comunicar y dar publicidad a la obra en la versión que ceda y en otras posteriores a través de cualquier medio.
c) Solicitar la retirada de la obra del repositorio por causa justificada.
d) Recibir notificación fehaciente de cualquier reclamación que puedan formular terceras personas en relación con la obra y, en particular, de reclamaciones relativas a los derechos de propiedad intelectual sobre ella.

5º. Deberes del autor.

El autor se compromete a:

a) Garantizar que el compromiso que adquiere mediante el presente escrito no infringe ningún derecho de terceros, ya sean de propiedad industrial, intelectual o cualquier otro.
b) Garantizar que el contenido de las obras no atenta contra los derechos al honor, a la intimidad y a la imagen de terceros.
c) Asumir toda reclamación o responsabilidad, incluyendo las indemnizaciones por daños, que pudieran ejercitarse contra la Universidad por terceros que vieran infringidos sus derechos e intereses a causa de la cesión.

d) Asumir la responsabilidad en el caso de que las instituciones fueran condenadas por infracción de derechos derivada de las obras objeto de la cesión.

6º. Fines y funcionamiento del Repositorio Institucional.

La obra se pondrá a disposición de los usuarios para que hagan de ella un uso justo y respetuoso con los derechos del autor, según lo permitido por la legislación aplicable, y con fines de estudio, investigación, o cualquier otro fin lícito. Con dicha finalidad, la Universidad asume los siguientes deberes y se reserva las siguientes facultades:

- La Universidad informará a los usuarios del archivo sobre los usos permitidos, y no garantiza ni asume responsabilidad alguna por otras formas en que los usuarios hagan un uso posterior de las obras no conforme con la legislación vigente. El uso posterior, más allá de la copia privada, requerirá que se cite la fuente y se reconozca la autoría, que no se obtenga beneficio comercial, y que no se realicen obras derivadas.
- La Universidad no revisará el contenido de las obras, que en todo caso permanecerá bajo la responsabilidad exclusive del autor y no estará obligada a ejercitar acciones legales en nombre del autor en el supuesto de infracciones a derechos de propiedad intelectual derivados del depósito y archivo de las obras. El autor renuncia a cualquier reclamación frente a la Universidad por las formas no ajustadas a la legislación vigente en que los usuarios hagan uso de las obras.
- La Universidad adoptará las medidas necesarias para la preservación de la obra en un futuro.
- La Universidad se reserva la facultad de retirar la obra, previa notificación al autor, en supuestos suficientemente justificados, o en caso de reclamaciones de terceros.

Madrid, a ……13…….. de ……Agosto……... de …….2025……

ACEPTA

Fdo…Tomás Pérez Gutiérrez……

Motivos para solicitar el acceso restringido, cerrado o embargado del trabajo en el Repositorio Institucional:

COMILLAS
UNIVERSIDAD PONTIFICIA
ICAI

MASTER UNIVERSITARIO EN INGENIERÍA DE TELECOMUNICACIONES

TRABAJO FIN DE MASTER

Graph-Based Socio-Technical Information Model for Power Distribution Network Planning and Operations

Autor: Tomás Pérez Gutiérrez

Director académico: Bruce Stephen

Director industrial: Ciaran Higgins

Madrid

MODELO DE INFORMACIÓN SOCIOTÉCNICO BASADO EN GRAFOS PARA LA PLANIFICACIÓN Y OPERACIÓN DE REDES DE DISTRIBUCIÓN ELÉCTRICA

Author: Pérez Gutiérrez, Tomás
Supervisor: Stephen, Bruce.
Collaborating Entity: Scottish Power Energy Networks (SPEN)

RESUMEN

Este proyecto presenta un sistema de información basado en grafos que integra en un solo modelo la estructura de la red eléctrica y su contexto geográfico y socioeconómico. El sistema permite analizar y visualizar de forma conjunta aspectos técnicos, espaciales y sociales, tanto en arquitecturas de despliegue locales (Neo4j) como en arquitecturas empresariales en la nube (Amazon Neptune). Mediante ejemplos y un caso real, se demuestra su utilidad para identificar áreas vulnerables, evaluar riesgos y planificar la red con criterios de equidad, mejorando la toma de decisiones técnicas y sociales.

Palabras clave: grafo de conocimiento, redes de distribución eléctrica, vulnerabilidad social, planificación de la red.

1. Introducción

La planificación y operación de redes de distribución es cada vez más compleja debido a la integración de energías renovables, la expansión del vehículo eléctrico y los cambios en los patrones de demanda. A ello se suman criterios sociales y medioambientales que influyen cada vez más en la toma de decisiones. Sin embargo, los datos técnicos, geográficos y socioeconómicos suelen almacenarse en sistemas y formatos diferentes, lo que dificulta el análisis integrado y limita la comprensión del contexto espacial y social de la red.

Un enfoque basado en grafos supera estas barreras al vincular datos heterogéneos en un único modelo. Representando activos, ubicaciones y comunidades como nodos y relaciones, se revelan interdependencias que permiten fundamentar mejor las decisiones sobre inversión, mantenimiento y políticas.

2. Definición del proyecto

Este proyecto desarrolla un grafo de conocimiento multinivel para redes de distribución eléctrica, que reúne en un único modelo el inventario de activos, su ubicación geográfica y los indicadores socioeconómicos de las zonas atendidas. El modelo conecta los registros de activos con datos espaciales (direcciones, códigos postales) e incorpora indicadores de vulnerabilidad como la privación social o la clasificación urbano-rural. Un proceso reproducible de tratamiento de datos integra varias fuentes, las limpia y normaliza, y crea nodos y relaciones siguiendo un esquema unificado.

El grafo ofrece un entorno común para que planificadores y analistas evalúen la red considerando tanto las condiciones locales como el contexto comunitario. Entre sus funciones están vincular cada cliente con su activo y dirección, identificar colectivos vulnerables atendidos por determinados equipos y localizar zonas donde problemas de accesibilidad coinciden con un alto riesgo social ante interrupciones. El sistema cuenta con dos arquitecturas de despliegue: local con Neo4j para desarrollo ágil y en la nube privada de SPEN con Amazon Neptune. Sus capacidades se muestran con ejemplos y un caso de estudio.

3. Descripción del modelo

La Ilustración 1 muestra la arquitectura del sistema. Un proceso ETL modular recopila registros de activos de red, datos de incidencias por interrupciones, indicadores socioeconómicos y metadatos espaciales. Este proceso valida la información, unifica formatos y unidades, enlaza los registros

mediante identificadores comunes (por ejemplo, códigos de equipo y códigos postales) y asigna cada campo a una ontología que define las entidades —como transformadores o áreas postales— y las relaciones entre ellas.

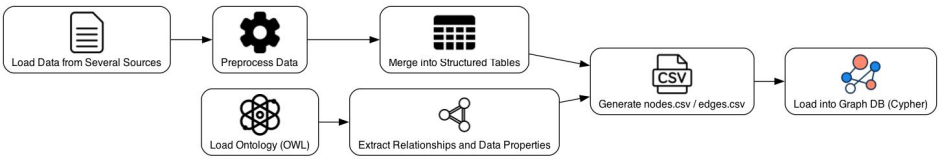


Ilustración *1* - Flujo de procesamiento de datos desde múltiples fuentes*, carga de la* ontología *y creación del grafo.*

Los datos ya armonizados se convierten en nodos y relaciones, con sus correspondientes metadatos (atributos), y se exportan a los archivos nodes.csv y edges.csv. Estos ficheros se cargan en la base de datos de grafos mediante un script específico. Al ser un proceso reproducible e idempotente, el grafo puede regenerarse fácilmente cuando se actualizan las fuentes originales o se modifica la ontología.

La Ilustración 2 muestra las arquitecturas de despliegue. En el entorno de desarrollo, una instancia local de Neo4j en Docker permite explorar el grafo de forma visual y realizar consultas rápidas. En producción, el modelo se aloja en Amazon Neptune dentro de la nube privada de SPEN, con acceso seguro mediante VPN WireGuard. Neptune Graph Explorer ofrece una interfaz web para ejecutar búsquedas, recorrer conexiones y visualizar la información. Ambos entornos utilizan las mismas salidas CSV, garantizando resultados coherentes y reproducibles.

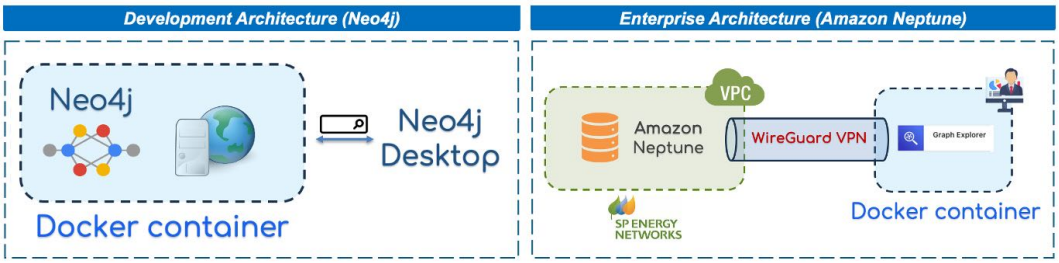


Ilustración 2 - Arquitecturas de desarrollo (Neo4j) y empresarial (Amazon Neptune) para *el despliegue.*

4. Resultados

La implementación del grafo de conocimiento ha logrado su objetivo principal: integrar en un único modelo consultable la estructura de los activos de la red, su contexto geográfico y los indicadores socioeconómicos (véase la Ilustración 3). Esta integración permite realizar análisis conjuntos, tanto técnicos como sociales, que antes resultaban muy limitados cuando los datos se encontraban separados.

Los análisis realizados durante el estudio muestran la versatilidad del sistema. Uno de ellos identificó hogares rurales con una demanda eléctrica muy baja en zonas de alta privación social, lo que apunta a posibles casos de racionamiento energético o problemas de asequibilidad. Otro analizó la accesibilidad a cargadores de vehículos eléctricos a través de la red viaria, detectando comunidades rurales dependientes de un único punto de carga, consideradas prioritarias para reforzar la infraestructura.

Asimismo, el caso de estudio aplicó el modelo para evaluar la vulnerabilidad de toda la red, combinando en una única puntuación la probabilidad de fallo técnico, el impacto de las interrupciones y un índice compuesto de vulnerabilidad social. Los resultados demuestran que variar el peso relativo de los factores técnicos y sociales cambia la clasificación de los activos críticos:

cuando prima la fiabilidad técnica destacan los grandes transformadores, mientras que al priorizar la equidad cobran relevancia activos de distribución más pequeños en comunidades vulnerables.

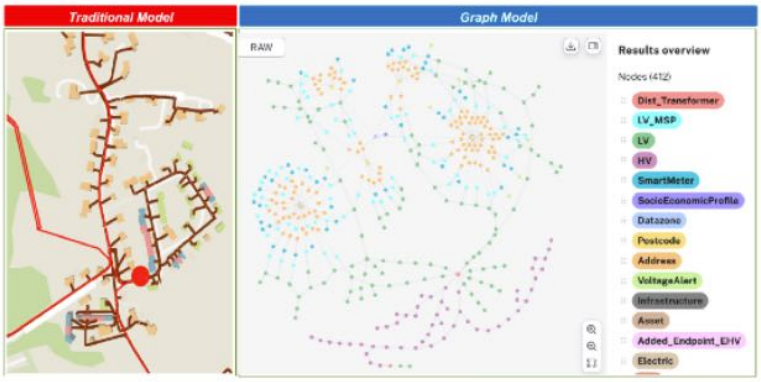


Ilustración 3 - Comparación entre la representación tradicional de la red y el modelo desarrollado basado en grafos.

En resumen, los resultados indican que el grafo propuesto constituye una solución práctica y escalable para detectar solapamientos entre riesgos de ingeniería y vulnerabilidades sociales, habilitando en un mismo entorno tanto evaluaciones de fiabilidad como una planificación orientada a la equidad para operadores, reguladores e investigadores.

5. Conclusiones

Esta tesis demuestra que integrar datos técnicos, espaciales y socioeconómicos en un único grafo multinivel ofrece una visión más completa para analizar y comprender las redes de distribución. El modelo permite realizar consultas entre dominios, explorar los datos de forma visual e identificar zonas donde las limitaciones técnicas coinciden con vulnerabilidades sociales. Se diseñó un proceso ETL reproducible (extracción, transformación y carga) para limpiar, estandarizar y fusionar fuentes heterogéneas, y el grafo resultante se desplegó tanto en Neo4j como en Amazon Neptune, lo que confirma su portabilidad entre entornos de desarrollo y empresariales.

Los casos de estudio ponen de relieve su utilidad para distintos perfiles: los reguladores pueden dirigir la inversión hacia donde genere mayor impacto; los operadores de red pueden mejorar la gestión de activos y la respuesta a incidencias; y la comunidad investigadora puede ampliarlo para nuevos enfoques analíticos y de modelización. Persisten limitaciones, como la ausencia de parámetros eléctricos (p. ej., capacidad) o de datos operativos en tiempo real, así como la dependencia de datos abiertos con resolución e integridad variables. Entre las mejoras futuras se incluyen refinar los límites administrativos y ampliar las pruebas a redes malladas para evaluar la escalabilidad en contextos más complejos. En conjunto, el trabajo sienta una base sólida para investigaciones posteriores, incluido el desarrollo de gemelos digitales de redes eléctricas.

6. Referencias

[1] Neo4j Inc. Getting Started with Neo4j in Docker. https : / / neo4j . com / docs / operations-manual/current/docker/introduction/. Accessed: 2025-08-08. 2025.

[2] Amazon Web Services, Inc. Visualizing Graphs with Neptune Graph Explorer. https://docs.aws.amazon.com/neptune/latest/userguide/visualization-graphexplorer.html. Accessed: 2025-08-08. 2025.

GRAPH-BASED SOCIO-TECHNICAL INFORMATION MODEL FOR POWER DISTRIBUTION NETWORK PLANNING AND OPERATIONS

Author: Pérez Gutiérrez, Tomás
Supervisor: Stephen, Bruce.
Collaborating Entity: Scottish Power Energy Networks (SPEN)

RESUMEN

This thesis presents a knowledge graph for electricity distribution networks, integrating asset topology, geographic data, and socio-economic information into a unified model. The system supports cross-domain analysis and visualisation and is deployed in both a local Neo4j development environment and an enterprise-scale Amazon Neptune cloud platform. Worked examples and a case study demonstrate its applicability to vulnerability assessment and equity-aware network planning.

Keywords: knowledge graph, electricity distribution networks, social vulnerability, power grid planning.

1. Introduction

Distribution network planning and operation are becoming increasingly complex with the integration of renewable generation, the growing adoption of electric vehicles, and the need to respond to changing demand patterns. Planning decisions are also increasingly shaped by social and environmental considerations. However, the underlying data—technical records, geographic information, and socio-economic indicators—are often maintained in separate systems and formats. This separation makes integrated analysis challenging and limits understanding of the network within its broader spatial and social context.

A graph-based approach can bridge these gaps by linking heterogeneous data sources into a single connected model. By representing assets, locations, and communities as nodes and relationships, the graph reveals interdependencies and enables more informed decisions on investment, maintenance, and policy.

2. Project definition

This project develops a multi-layer knowledge graph for electricity distribution networks, integrating asset inventories, geographic context, and socio-economic indicators into a single connected model. It combines electrical asset records with spatial datasets (addresses, postcodes) and vulnerability measures such as social deprivation and urban–rural classification. A reproducible data pipeline ingests multiple sources, cleans and standardises them, and generates graph nodes and relationships aligned with a consistent schema.

The resulting graph provides a shared environment for planners and analysts to assess network conditions in relation to local characteristics and community context. It enables tasks such as tracing customers from asset to address, identifying vulnerable customers served by specific equipment, and pinpointing areas where accessibility issues overlap with outage risk. The system is deployed in two configurations: a local Neo4j setup for rapid development and an Amazon Neptune instance within SPEN's private cloud. Its capabilities and practical value are demonstrated through worked examples and a real-world case study.

3. Model description

Illustration 1 shows the system architecture. A modular ETL pipeline ingests network asset records, outage logs, socio-economic indicators, and spatial metadata. It validates inputs, standardises formats and units, links records via shared identifiers (such as equipment IDs and postcode units),

and maps fields to an ontology that defines entities (e.g., transformers, substations, postcodes) and their relationships.

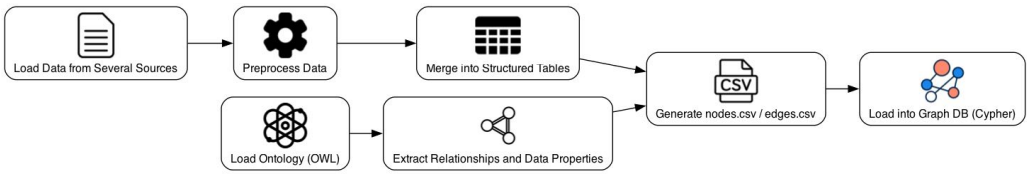


Illustration 1 - Data processing workflow from multiple sources, ontology loading, and graph creation.

The harmonised data is written as nodes and relationships with associated metadata (attributes) and exported as nodes.csv and edges.csv. These files are then loaded into the graph database using an import script. Because the pipeline is reproducible and idempotent, the graph can be rebuilt whenever inputs are updated or the ontology changes.

Illustration 2 showcases the deployment architectures. In development, a local Neo4j graph instance running in Docker enables the visual exploration of the graph and facilitates quick testing of queries. In production, the model is hosted on Amazon Neptune within SPEN's private cloud, accessed securely via WireGuard VPN. Neptune Graph Explorer provides a browser-based interface for running searches, following connections, and visualising the data. Both environments use the same CSV outputs, ensuring aligned and reproducible results.

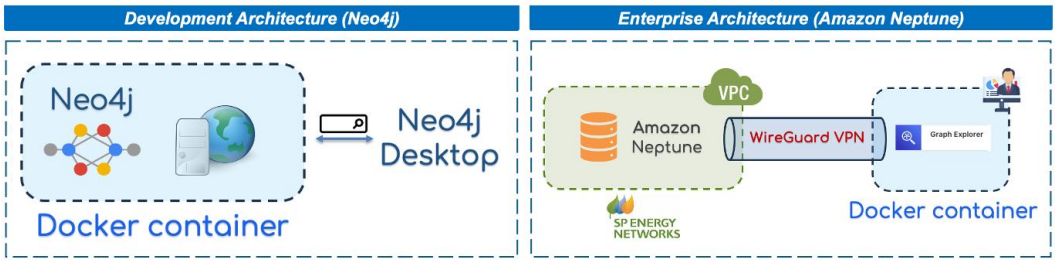
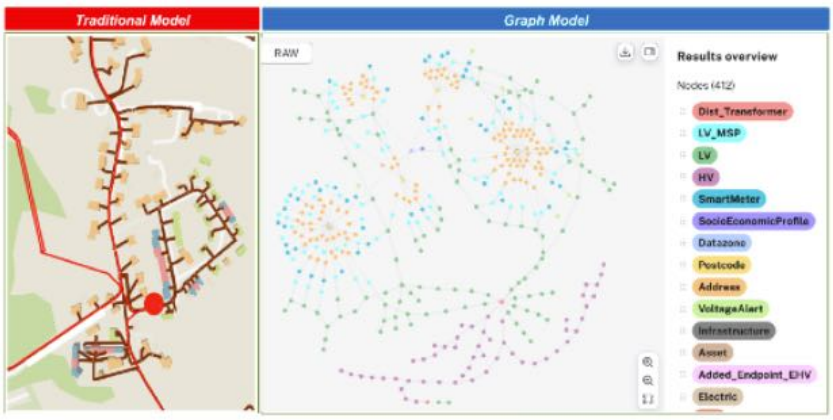


Illustration *2* - Development (Neo4j) and enterprise (Amazon Neptune) architectures for deployment.

4. Results

The implementation of the knowledge graph successfully achieved its primary objective of integrating asset topology, geographic context, and socio-economic indicators into a unified, queryable model (see Illustration 3). This integration enables combined technical and social analyses that were previously impractical with siloed datasets.



Ilustration 3 - Comparison between the traditional network representation and the developed graph-based model.

Worked examples confirmed the system's analytical flexibility. One example identified rural households with very low recorded electricity demand located in high-deprivation areas, suggesting potential cases of energy rationing or affordability constraints. Another example calculated accessibility to electric vehicle chargers based on road network data. This revealed rural communities dependent on a single charging point, highlighting priority areas for infrastructure reinforcement.

The case study applied the model to assess vulnerability across the entire distribution network, combining three elements—likelihood of technical failure, outage impact, and a composite social vulnerability index—into a single asset criticality score. Results showed that altering the balance between technical and social factors changed the ranking of critical assets: large transformers ranked highest when technical reliability was prioritised, while smaller distribution assets in vulnerable communities emerged as most critical when equity was the focus.

Overall, the results show that the proposed knowledge graph offers a practical, portable, and repeatable way to identify overlaps between engineering risks and social vulnerabilities, enabling both reliability assessments and equity-aware planning in one environment for operators, policymakers, and researchers.

5. Conclusions

This thesis demonstrates that combining technical, spatial, and socio-economic data within a single, multi-layer graph model offers a more comprehensive way to analyse and understand electricity distribution networks. The model enables cross-domain queries, visual exploration, and the detection of areas where technical constraints overlap with social vulnerabilities. A reproducible ETL pipeline was designed to clean, standardise, and merge diverse datasets, with the resulting graph deployed in both Neo4j and Amazon Neptune to confirm portability across development and enterprise environments.

The case studies highlight its practical value for different stakeholders: policymakers can use it to target investment where it will have the greatest impact; network operators can apply it to strengthen asset management and outage response; and researchers can build on it for new analytical and modelling approaches. Limitations remain, including the absence of electrical parameters such as capacity or live operational data, and reliance on open datasets with varying resolution and completeness. Further improvements could refine administrative boundaries and extend testing to meshed networks to evaluate scalability in more complex contexts. Overall, the work provides a strong basis for future research, including applications in related areas such as digital twin development for electricity networks.

6. References

[1] Neo4j Inc. Getting Started with Neo4j in Docker. https : / / neo4j . com / docs / operations-manual/current/docker/introduction/. Accessed: 2025-08-08. 2025.

[2] Amazon Web Services, Inc. Visualizing Graphs with Neptune Graph Explorer. https://docs.aws.amazon.com/neptune/latest/userguide/visualization-graphexplorer.html. Accessed: 2025-08-08. 2025.

COMILLAS
UNIVERSIDAD PONTIFICIA
ICAI

MASTER UNIVERSITARIO EN INGENIERÍA DE TELECOMUNICACIONES

TRABAJO FIN DE MASTER

Graph-Based Socio-Technical Information Model for Power Distribution Network Planning and Operations

Autor: Tomás Pérez Gutiérrez

Director académico: Bruce Stephen

Director industrial: Ciaran Higgins

Madrid

Abstract

This thesis presents a multi-layer knowledge graph for electricity distribution networks, integrating asset topology, geospatial context, and socio-economic indicators into a single model. A reproducible ETL pipeline ingests data from multiple sources, standardises them, and maps records to an ontology, producing a connected graph for cross-domain analysis and visualisation. The system is deployed in two configurations: a Dockerised Neo4j instance for development and an Amazon Neptune deployment within Scottish Power Energy Network's (SPEN) private cloud.

Several analyses and a case study are developed to assess the model's capabilities, particularly its ability to link technical network data with social and spatial context. In particular, the model is applied to identify rural households with very low demand in high-deprivation areas, assess electric-vehicle charger accessibility using road-network travel times, and generate asset-criticality scores that balance technical reliability and social vulnerability.

The graph provides a portable, reproducible foundation for equity-aware planning. Limitations include missing electrical parameters and reliance on open datasets. Future work will add operational data, refine boundaries, and test on larger, meshed networks.

Acknowledgements

Thank you to my family for their constant support and patience—I wouldn't be here without you.

Thank you to ScottishPower Energy Networks for the chance to work on a project with real-world impact.

Thank you to Bruce Stephen and Ciaran Higgins for their guidance during the project.

Contents

List of Figures

List of Tables

Listings

1 Introduction

1.1 Objectives

Electricity distribution networks are becoming increasingly complex as they integrate renewable generation, support the uptake of technologies like electric vehicles, and adapt to shifting demand patterns. At the same time, network planning and operation are increasingly influenced by socio-economic and environmental considerations. However, the data underpinning these decisions—technical network records, geographic information, and socio-economic indicators—are often stored in separate formats and systems, making it difficult to analyse them together. This separation limits the ability to assess the network in its broader spatial and social context.

A graph-based approach offers a practical way to bridge these gaps by linking diverse datasets into a single, connected model. Such an integrated representation allows analysts to explore relationships between infrastructure, location, and community characteristics, enabling more informed decisions on investment, maintenance, and policy.

This thesis aims to design and apply a multi-layer property graph that brings together physical network data, spatial information, and socio-economic indicators. The work is organised around the following objectives:

- **Objective 1:** ***Define scope and select data sources*** — Set the focus and boundaries of the study by identifying gaps in existing data, reviewing relevant literature, and discussing priorities with Scottish Power Energy Networks (SPEN). Select a representative distribution network pilot area and gather suitable open datasets, such as electric vehicle (EV) charger locations, road networks, and socio-economic data.
- **Objective 2:** ***Design an integrated graph model*** — Create a flexible property graph schema that represents the distribution network alongside geographic and social layers, linking physical assets, customer locations, and contextual information in one structure.
- **Objective 3:** ***Develop reproducible data processing and loading pipelines*** — Build workflows to clean, standardise, and prepare data from different sources for integration into the graph, keeping identifiers, relationships, and spatial references consistent. The pipelines

must be reproducible to ensure analyses can be repeated and adapted as new data becomes available.

- **Objective 4:** ***Enable cross-domain analytical queries and visualisation*** — Develop queries that combine technical, spatial, and social data to answer multi-layer questions. Make them clear to read and support them with visual outputs for both technical and non-technical audiences.
- **Objective 5:** ***Deploy in local and cloud environments*** — Run the graph database in both a local Neo4j setup and a cloud-based Amazon Neptune instance to verify that the model and queries work consistently across platforms, covering both local development and enterprise configurations.
- **Objective 6:** ***Demonstrate capabilities through case studies*** — Demonstrate the usability and practical value of the integrated graph model by applying it to representative examples and real-world case study scenarios, illustrating how it can support analysis and decision-making in distribution network planning and operation.

1.2 Motivation

Electricity distribution networks are at the forefront of the energy transition. The rapid uptake of electric vehicles, the growth of distributed renewable generation, and evolving patterns of energy demand are placing new and complex requirements on these networks. Meeting these challenges demands not only technical upgrades but also a deeper understanding of how infrastructure interacts with the communities it serves.

In practice, however, network operators often face the problem that relevant information is fragmented across multiple systems. Asset records, geographic datasets, and socio-economic indicators are typically stored in separate formats and maintained by different organisations. This separation makes it difficult to carry out holistic analyses that capture both the physical state of the network and its broader social and spatial context.

Graph-based data models offer a compelling solution to this challenge. By representing assets, locations, and contextual attributes as interconnected nodes and relationships, they provide a natural way to combine diverse datasets into a single, navigable structure. This approach enables the discovery of complex relationships—such as how technical constraints overlap with social vulnerabilities—and supports more informed decision-making in areas like investment planning, asset maintenance, and resilience assessment.

In this context, the motivation for this work is to demonstrate that a multi-layer property graph, integrating technical, spatial, and socio-economic data, can enhance the ability to explore, query, and visualise distribution networks in a way that traditional siloed systems cannot.

1.3 Methodology

This thesis follows a structured approach to bring together multiple datasets into a single, connected graph model of an electricity distribution network. The process is organised into sequential stages, reflecting the project objectives and ensuring that the work is both reproducible and adaptable for future use.

The first stage defines the scope of the study by identifying relevant datasets, reviewing existing literature to understand current practices and limitations, and assessing the availability of open data. This includes technical network records, geographic information, and socio-economic indicators. Any gaps or inconsistencies in the data are recorded for consideration in later analysis.

The second stage focuses on designing a property graph schema that can represent the distribution network alongside its spatial and social context. The schema is kept clear enough for non-technical users to interpret, while remaining flexible enough to incorporate additional datasets or relationship types as the model evolves.

In the third stage, reproducible data processing pipelines are developed. These workflows clean, standardise, and transform the various datasets into a consistent format, ensuring that identifiers, relationships, and spatial references align across sources. This harmonisation is essential for interoperability and for supporting future updates to the model.

Next, the processed data is loaded into two graph database environments: a local Neo4j instance and a cloud-based Amazon Neptune deployment. This dual setup verifies that the model and its queries run consistently across platforms and under different operational conditions.

With the data in place, analytical queries are designed to take advantage of the model's multi-layer structure. These queries bring together information from technical, geographic, and socio-economic layers to answer questions that would be difficult to address using isolated datasets. Where possible, results are paired with visualisations to make the findings accessible to both technical and non-technical audiences.

Finally, the methodology is demonstrated through real-world case study scenarios applied to a chosen pilot region. This step illustrates how the integrated graph model can support practical tasks in a real-world setup.

1.4 Structure

The thesis is organised into six chapters, each building upon the previous to provide a coherent understanding of the work carried out and its significance.

The first chapter, *Introduction*, outlines the context, objectives, motivation, methodology, and

structure of the work. It sets the stage by framing the challenges in analysing electricity distribution networks when technical, spatial, and socio-economic data remain siloed, and introduces the rationale for adopting a graph-based approach.

The second chapter, *Contribution of this thesis*, introduces the key shortcomings in existing approaches and the opportunities for improvement that motivated this research. It also highlights the specific ways in which the thesis seeks to address these gaps.

The third chapter, *Theoretical Framework*, presents the context and concepts necessary for understanding the thesis and its importance. It reviews the relevance of integrating spatial and socio-economic dimensions into network analysis, and the capabilities of graph database technologies for linking and querying diverse datasets. This chapter provides the foundation upon which the rest of the work is developed.

The fourth chapter, *The Graph Model*, details the design and implementation of the multi-layer property graph. It describes the selected data sources, the integration and preprocessing workflows, and the deployment in both local and cloud environments, ensuring reproducibility and adaptability. This chapter also includes example queries and demonstrations of the model's capabilities, alongside a discussion of its limitations and potential improvements.

The fifth chapter, *Case Study: Topological Vulnerability and Socioeconomic Impact of Node Failures*, applies the developed graph model to a selected pilot distribution network region. Through a series of analytical queries and visualisations, it demonstrates how the model can be used to assess network vulnerabilities in their broader spatial and socio-economic context.

The sixth and final chapter, *Conclusions*, summarises the key contributions of the work, discusses its limitations, and outlines recommendations and potential future developments.

2 Contribution of this thesis

Modern electricity distribution networks are expected to deliver reliable technical performance while also addressing the needs and circumstances of the communities they serve. This requires moving beyond a purely engineering-led perspective and incorporating socio-economic and spatial considerations into planning and analysis. However, as shown in Chapter 3 (*Theoretical Framework*) and developed further in Chapter 4 (*The Graph Model*), data from these different domains is typically held in separate systems, making it difficult to carry out integrated assessments.

In response, the present work addresses key research gaps in three areas: integrating technical, spatial, and socio-economic data for distribution networks; applying graph technologies to multi-domain utility planning; and developing methods for running cross-layer queries that reveal both technical vulnerabilities and their social context.

2.1 Identified research gaps

In this thesis, the research sits at the intersection of technical network analysis, spatial modelling, and socio-economic assessment. While each of these areas is studied independently, their integration into a single analytical framework remains rare in both academic and industry practice. In the context of electricity distribution networks, this separation limits the ability to understand how engineering risks and operational constraints overlap with social and geographic factors that affect the communities being served.

Two main gaps are identified. First, there is a persistent disconnect between engineering analysis and socio-economic insight. Infrastructure decisions are often made with a focus on technical performance alone, with limited consideration of social equity or vulnerability. Second, existing data management systems for utilities—typically built around separate geographic information systems (GIS), asset management, and customer information platforms—do not support integrated analysis across multiple domains. Graph database technologies offer a natural way to connect these datasets, but their potential for multi-layered, equity-aware network planning is not yet fully explored.

2.2 Contibutions derived from the research gaps

The present work addresses these gaps by designing and implementing a graph model that unifies technical, spatial, and socio-economic data into a single, queryable structure. The model supports complex, cross-domain analysis and is applied in real-world case studies to demonstrate how this integration can inform more balanced and informed decision-making in distribution network planning. The specific contributions of this thesis are the following:

- **Integration of technical, spatial, and socio-economic data in a unified graph model.** A flexible property graph structure links network assets (feeders, transformers, meters) with spatial datasets (road networks, rural/urban classifications) and socio-economic indicators (e.g., deprivation indices). This enables multi-layered analysis down to postcode and address level.
- **Support for cross-domain analytical queries and vulnerability assessment.** The model enables queries that combine network topology, asset attributes, spatial relationships, and socio-economic factors to identify areas where technical vulnerabilities coincide with high social sensitivity. The inclusion of accessibility data, such as road network connectivity, enriches the context of the analysis.
- **Facilitation of equity-aware network planning.** The integrated approach shows how operational and investment decisions can be guided by both technical priorities and social considerations, for example by identifying areas with both high outage risk and high socio-economic vulnerability.

3 Theoretical Framework

3.1 Socio-economic context of electric networks

3.1.1 Energy poverty and social vulnerability

Electricity infrastructure is a socio-technical system [1] that both shapes and is shaped by the economic and social conditions of the communities it serves. In this context, socio-economic analyses examine how factors such as poverty, demographics, and geography influence energy access, reliability, and affordability. Two key concepts in this domain are:

- **Energy poverty**, defined as the inability of a household to access essential energy services at a reasonable cost [2], and
- **Social vulnerability**, which refers to the heightened risks faced by certain populations during energy disruptions due to underlying social and economic disadvantages [1].

Recent research shows a positive and growing correlation between social vulnerability and outage-related metrics, suggesting that structurally disadvantaged communities are not only more exposed to disruptions but that these disparities have worsened over time [3, 4]. These findings underscore the need for network planning to move beyond purely technical considerations and explicitly incorporate socio-demographic risk profiles.

Geographic classifications, particularly the urban–rural distinction, further shape infrastructure design and performance. Urban areas typically have dense, interconnected networks with multiple delivery paths, whereas rural areas—defined in Scotland as communities with fewer than 3,000 residents [5]—often rely on thinner, more radial systems. These structural differences have a direct impact on the reliability and resilience of electricity services.

3.1.2 Equity and justice in grid planning

Ensuring equity and justice in electricity grid planning involves designing systems in which no community bears an unfair burden and all people have equitable access to reliable and affordable energy [6, 7]. The concept of *energy justice* applies principles from environmental and climate justice to the energy domain and is typically framed around three dimensions [8]:

- **Distributional justice**: how the costs (e.g., outages, rising energy prices, siting of infrastructure) and benefits (e.g., grid upgrades, access to renewables) of energy systems are socially and spatially distributed.
- **Procedural justice**: the degree to which historically marginalised groups are included in decision-making processes related to grid planning.
- **Recognition-based justice**: the need to acknowledge and respect the identities and experiences of systematically overlooked groups—such as indigenous populations, ethnic minorities, the elderly, and people with disabilities—and to ensure their visibility and influence in energy policy.

Empirical studies confirm that low-income and minority populations are disproportionately impacted by outages—not only do they experience longer service interruptions, but their capacity for recovery is also more limited [3, 4, 9]. Nonetheless, utilities have historically prioritised technical criteria—such as restoring service to the largest number of customers—without explicitly addressing social vulnerability. This approach has raised concerns about *infrastructure injustice*, where communities already facing disadvantage are systematically underserved.

In response, regulators are beginning to act. Countries like the UK, the US, and Spain are shifting into the creation of multi-objective planning models that include equity along with traditional goals like cost efficiency, reliability, and sustainability:

1. **United Kingdom**: *Ofgem* requires Distribution Network Operators (DNOs) to maintain *Priority Services Registers* for customers with medical or financial vulnerabilities, and to implement support measures such as targeted investments and resilience interventions [10, 11].
2. **United States**: Equity has become a federal priority through Executive Orders 13985 and 14091, leading to initiatives such as the Department of Energy's *Community Benefits Plan* and the *Justice40* program, which allocates 40% of a number of federal investment benefits to underserved communities [12].
3. **Spain**: The *National Energy and Climate Plan* (INECP) includes strategies to reduce energy poverty. The key initiative is the *Bono Social Eléctrico*, which provides subsidised electricity to low-income households. However, challenges such as administrative hurdles and a lack of digital access have limited access to this support. The INECP also promotes building retrofits and regional equity in the context of the energy transition [13].

Together, these efforts reflect a shift from traditional, technically driven planning models to multi-objective frameworks that incorporate social criteria. The goal is to ensure that the benefits of grid modernisation are shared fairly and that no community is overlooked.

3.1.3 Data landscape for equity analysis in the UK

Effective socioeconomic analysis of electrical infrastructure requires integrating diverse datasets to understand how demographics, geography, and service delivery interact. In the UK, a range of high-resolution, well-structured sources supports this task and helps identify spatial inequalities and at-risk populations:

- **Census data**, provided by the Office of National Statistics (ONS) and its devolved counterparts, offer detailed demographic and housing characteristics at small area levels (e.g., data zones in Scotland, super output areas of the lower layer in England) [14]. Key variables include age structure, income, housing tenure, overcrowding, access to central heating, and car ownership. These indicators help identify communities vulnerable to energy hardship—such as low-income households or individuals with medical needs—and support modelling of localised energy demand.

- **Indices of Multiple Deprivation (IMD)** serve as spatial proxies for social disadvantage, ranking areas based on income, health, education, housing, and access to services [15]. National examples include the Scottish Index of Multiple Deprivation (SIMD), the English IMD, and the Welsh IMD. When combined with infrastructure performance metrics (e.g., outages), these indices reveal where deprivation and poor service provision intersect, enabling planners to prioritise interventions—such as network upgrades or energy efficiency programmes—where they are socially most impactful.

- **Utility and regulatory data** further improve this picture. Ofgem's RIIO-ED2 framework uses metrics like Customer Interruptions (CI) and Customer Minutes Lost (CML) to monitor service quality across regions [16]. Utilities also maintain Customer Relationship Management (CRM) systems, including the Priority Services Register (PSR), which identifies vulnerable users based on age, health, or disability [11]. Data on prepayment meters and payment arrears offer additional insights into financial hardship and potential need for support [17].

- Beyond traditional sources, new data-driven frameworks offer more nuanced spatial analysis. The *Spatial Signatures Framework*, developed by Fleischmann and Arribas-Bel, classifies small areas (datazones) across Great Britain into 16 functional urban types using physical layout, land use mix, population density, and activity patterns derived from open geospatial data [18]. This approach goes beyond the urban–rural binary and supports better understanding of how urban form relates to energy vulnerability and infrastructure resilience. Overlaying grid data with spatial signatures helps identify under-served area types. When combined with socio-economic data, this enables multivariate profiling of neighbourhoods—considering both who lives there and how space is organised and used.

- **Internationally**, comparable tools support energy equity planning. In the United States,

the Social Vulnerability Index (SVI) by the Centers for Disease Control and Prevention (CDC) and the Agency for Toxic Substances and Disease Registry (ATSDR) maps community vulnerability to disasters and infrastructure failures using 16 census-based variables covering income, household composition, minority status, and housing conditions [19]. In the European Union, the Energy Poverty Advisory Hub (EPAH) tracks key indicators—such as inability to keep homes adequately warm or energy bill arrears—to inform targeted policy responses. Globally, the International Energy Agency (IEA) and World Bank monitor access metrics to guide investments in the approximately 750 million people who still lack electricity services worldwide [20].

Together, these datasets provide a solid foundation for identifying where energy systems are failing to meet social needs and for guiding equitable, resilient infrastructure planning.

3.2 Grid performance metrics

Electric power networks are traditionally evaluated using standard reliability metrics that quantify the frequency and duration of supply interruptions. These performance indices form a core part of regulatory oversight and benchmarking in many countries, shaping both investment decisions and customer service obligations. However, while such metrics offer a robust overview of network-level performance, they often neglect the social dimension of power outages—namely, who is affected, how severely, and with what capacity to respond. This section introduces both conventional and emerging socio-technical performance metrics, with a focus on their structure, application, and limitations.

3.2.1 Standard reliability metrics in distribution grids

Reliability indices measure how well electrical distribution systems perform by tracking power outages that affect customers. These metrics show how often outages occur, how long they last, and the overall dependability of the power supply. Table 3.1 lists the measures most commonly required by regulators and reported by distribution companies.

These indices are commonly used for operational comparison, meeting regulations, and planning for the long term. They summarize outage patterns across all customers, helping utilities and regulators assess service performance, spot areas that need improvement, and establish financial rewards or penalties.

Table 3.1: Frequently used reliability indices (IEEE 1366 definitions [21]).

Metric	*Meaning*	*Formula (annual)*
SAIFI	System Average Interruption **Frequency** Index	$\frac{\sum \text{Total Number of Customers Interrupted}}{\text{Total Number of Customers Served}}$
SAIDI	System Average Interruption **Duration** Index	$\frac{\sum \text{Customer Minutes of Interruption}}{\text{Total Number of Customers Served}}$
CAIDI	Customer Average Interruption Duration Index	$\frac{\sum \text{Customer Minutes of Interruption}}{\text{Total Number of Customers Interrupted}} = \frac{\text{CMI}}{\text{CI}}$
MAIFI	Momentary Average Interruption Frequency Index	$\frac{\sum \text{Total Number of Customer Momentary Interruptions}}{\text{Total Number of Customers Served}}$
CAIFI	Customer Average Interruption Frequency Index	$\frac{\sum \text{Total Number of Customers Interrupted}}{\text{Number of Distinct Customers Who Were Interrupted}}$
ENS	Energy Not Supplied	$\sum$ Energy Not Supplied (in MWh)

Notes. "Customer Minutes of Interruption" (CMI) is the sum, over all sustained events, of outage duration in minutes multiplied by the number of customers affected. "Customers Interrupted" (CI) counts customers experiencing at least one sustained (> 3 min) interruption during the reporting period. "Momentary interruptions" are those lasting less than one minute.

3.2.2 UK Quality-of-Service indicators under *RIIO-ED2*

In Great Britain, Ofgem's *RIIO-ED2 Regulatory Instructions and Guidance, Annex F: Interruptions* [22] mandates the following Quality-of-Service (QoS) metrics (see Table 3.2).

Table 3.2: QoS interruption metrics required by RIIO-ED2 Annex F. [22, 23]

Indicator	*Definition (summary)*
Customers Interrupted (CI)	Percentage of customers with $\geq$1 sustained outage (> 3 min) in a year, excluding re-interruptions from the same incident.
Customer Minutes Lost (CML)	Average number of customer-minutes lost per customer per year due to interruptions lasting more than 3 minutes.
Short Interruptions (SI)	Percentage of customers experiencing interruptions lasting between 1 second and 3 minutes during the year.
Customers Re-interrupted (RI)	Number of customers re-interrupted per 100 customers per year.

3.2.3 Limitations of current metrics

However, standard metrics share a common limitation: they treat all customers equally, regardless of their vulnerability or critical energy needs. For example, losing power for two hours has very different implications for a hospital than for a vacant building, yet traditional metrics assign them the same weight. This uniform weighting hides differences and can cause underinvestment in communities that rely more on stable power for social or medical needs. Addressing this gap requires a shift toward metrics and planning frameworks that take into account differentiated impacts and community resilience.

3.2.4 Current developments in socially-responsive grid metrics

In this regard, contemporary research has focused on developing indices or metrics that explicitly combine technical and social vulnerability data.

3.2.4.1 Social Vulnerability Index (SVI) in the United States

In the *U.S.*, a group of researchers developed a *Social Vulnerability Index (SVI)* to identify communities at greater risk during disasters [24]. This index, based on 15 census-derived indicators (e.g., poverty, age, race, housing conditions), has since been applied to energy contexts. For example, in [25], researchers used the SVI to integrate equity into power system planning during extreme events. They developed a two-stage stochastic optimisation model that, in the case of *Winter Storm Uri in Texas*, guided generator winterisation decisions while aiming to minimise the adverse effects experienced by socially vulnerable communities.

3.2.4.2 Socio-technical index in Puerto Rico

In *Puerto Rico*, where energy infrastructure is already fragile, [26] proposed an enhanced *Socio-Technical Vulnerability Index*, combining the SVI with physical accessibility indicators such as road quality, proximity to supermarkets, and hospitals. The index was used to assess where grid upgrades could yield the greatest social benefit [26].

3.2.4.3 PO-RSVI: combining Risk and capacity

Another important contribution is the *Power Outage Risk–Social Vulnerability Index (PO-RSVI)*, introduced by [27]. This index merges outage susceptibility, coping capacity, and access to critical services. Applied to three communities in Texas, the study found that PO-RSVI revealed disparities missed by conventional metrics and offered valuable insights for utility planners. Crucially, it highlighted how the integration of such indices in *network investment planning* can help align resilience spending with community-specific risks and financial capacities.

3.2.4.4 Geospatial targeting in the UK

In the *UK*, similar principles are beginning to be adopted. The *National Grid* and the *Centre for Sustainable Energy (CSE)* have developed *GIS-based tools* that overlay Priority Services Register data, outage history, and census-derived vulnerability indicators to map where support is most needed. They created a dashboard that identifies geographical gaps—or “thin spots”—in support service provision, particularly in areas with limited PSR coverage or underserved vulnerable groups [28].

3.2.4.5 Prospective outlook

Although still in the early stages of deployment, these *socially aware, indicator-based and geospatial approaches* show strong potential for guiding more targeted and equitable infrastructure planning. By linking technical vulnerability with socio-economic risk, they enable better prioritisation of grid reinforcements, the strategic deployment of decentralised energy resources, and the design of fairer emergency response protocols. As data quality and modelling capabilities improve, these methods could become central to future energy planning and regulation.

3.3 Foundations of graph theory

Graph theory provides the mathematical foundation for the graph model developed in this thesis. This section introduces the essential definitions, notation, and structural properties needed to understand the underlying graph principles and their applications in electrical network modelling and analysis.

3.3.1 Basic terminology and notation

A **graph** $G = (V, E)$ is an *ordered pair* in which V is a non-empty, finite set of **vertices** (or *nodes*), and $E \subseteq V \times V$ is a set of **edges** (or *links*) connecting pairs of vertices [29]. Unless stated otherwise, the number of vertices is denoted $|V| = n$, and the number of edges is denoted $|E| = m$.

3.3.1.1 Classification of graphs

Graphs can be classified according to the *structure and properties of their edges* along several key dimensions:

3.3.1.1.1 Directionality

This dimension refers to whether the edges in a graph have a specific orientation or direction, which is relevant in many applications.

- **Undirected graphs**: Each edge is an unordered pair $\{u, v\}$, implying bidirectional connectivity between nodes. Power-system topologies are often modeled this way when only connectivity—rather than direction of flow—is of interest.
- **Directed graphs (digraphs)**: Each edge is an ordered pair (u, v), indicating a directed connection from vertex u to vertex v. This structure is essential when the direction of interaction matters, such as in energy transfer or information flow.

3.3.1.1.2 Edge Weights

This dimension captures whether the edges carry numerical values representing physical or operational attributes.

- **Weighted graphs**: A weight function $w : E \to \mathbb{R}^+$ assigns a scalar value (e.g., impedance, capacity, failure rate) to each edge.
- **Unweighted graphs**: All edges are treated equally, without any associated weights, focusing solely on structural connectivity.

3.3.1.1.3 Edge Multiplicity

This property defines how many edges are allowed between pairs of vertices.

- **Simple graphs**: Allow at most one edge between any pair of distinct vertices and prohibit self-loops (edges from a vertex to itself).
- **Multigraphs**: Permit multiple parallel edges between the same pair of vertices and may include self-loops. These models are useful for representing redundant paths or transformer tap settings.

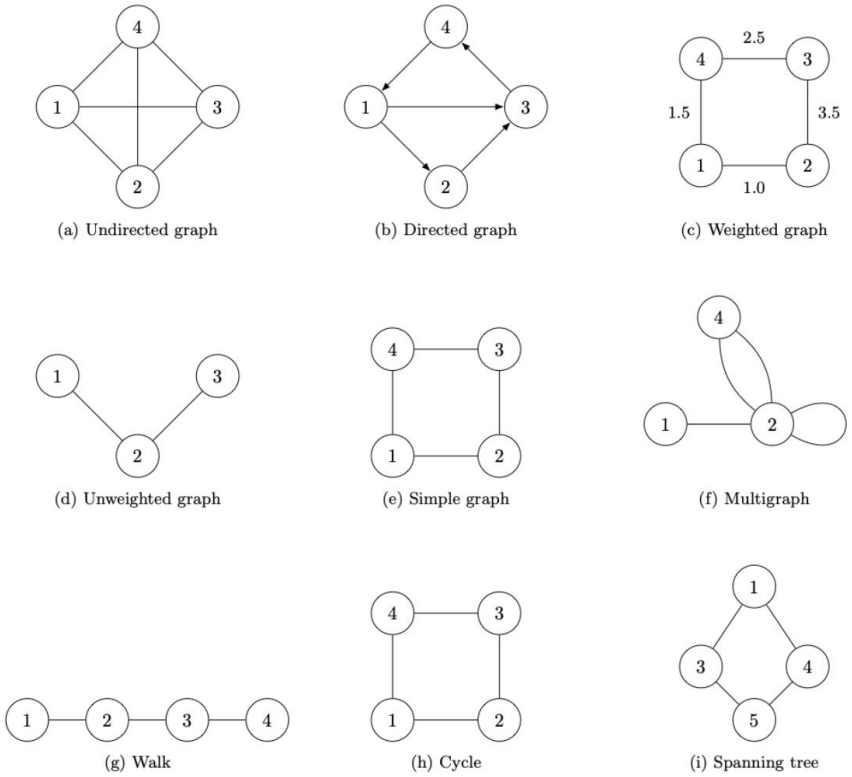


Figure 3.1: Illustration of basic graph types and connectivity concepts.

3.3.1.2 Graph connectivity

Connectivity describes the topological relationships among vertices and how they are linked through paths in the graph.

- A **walk** is a sequence of vertices $v_1, v_2, \ldots, v_k$ such that each consecutive pair $(v_i, v_{i+1}) \in E$ for $1 \leq i < k$.

- A **path** is a walk with no repeated vertices, i.e., $v_i \neq v_j$ for all $i \neq j$.
- A **cycle** is a closed path where $v_1 = v_k$ and all other vertices $v_2, \ldots, v_{k-1}$ are distinct.
- A graph is **connected** if for every pair of vertices $u, v \in V$, there exists a path from u to v.
- A **spanning tree** is a minimal connected subgraph $T = (V, E_T)$ that includes all vertices and contains exactly $n - 1$ edges, ensuring full connectivity without forming any cycles.

3.3.2 Essential Graph Properties

Graphs can be further characterized through a set of structural metrics that quantify how vertices are connected, how efficiently information or flow might propagate through the network, and how localized or global interactions are organized. These properties are fundamental to understanding many applications of graph-based models.

- **Degree of a vertex** $d(v)$: The degree of a vertex $v \in V$ is the number of edges incident to it. For undirected graphs:

  $$d(v) = |\{u \in V \mid \{u, v\} \in E\}|$$

  For directed graphs, one defines the *in-degree* $d^-(v)$ and *out-degree* $d^+(v)$ based on incoming and outgoing edges, respectively.

- **Shortest path** $\mathsf{sp}(u, v)$: The shortest path between two vertices $u, v \in V$ is the path with the minimum number of edges (or minimum total weight, in the case of weighted graphs). The function $\mathsf{sp}(u, v)$ denotes the length of this shortest path:

  $$\mathsf{sp}(u, v) = \min_{p \in P_{uv}} \mathsf{length}(p)$$

  where P_{uv} is the set of all paths from u to v.

- **Average path length** $\ell(G)$: The average length of the shortest paths between all pairs of distinct vertices in the graph:

  $$\ell(G) = \frac{1}{n(n-1)} \sum_{\substack{u,v \in V \\ u \neq v}} \mathsf{sp}(u, v)$$

- **Betweenness centrality** $C^B(v)$: This centrality metric quantifies the extent to which a vertex v lies on shortest paths between other vertex pairs. It is defined as:

  $$C^B(v) = \sum_{\substack{s,t \in V \\ s \neq t \neq v}} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

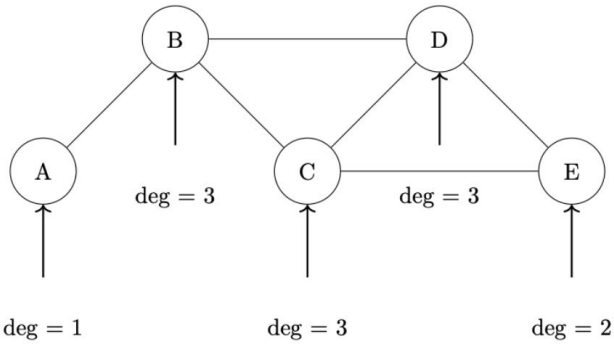
where σ_{st} is the total number of shortest paths between s and t, and $\sigma_{st}(v)$ is the number of those paths that pass through v.

- **Clustering coefficient** $C^L(v)$: The local clustering coefficient of a vertex $v \in V$ measures the tendency of its neighbors to also be connected. It is defined as:

$$C^L(v) = \frac{2T(v)}{d(v)(d(v)-1)}$$

where $T(v)$ is the number of triangles (i.e., sets of three mutually connected vertices) that include v.

These graph properties provide essential tools for analyzing structure, redundancy, and centrality within network models, and will be referenced throughout this work.



Essential Graph Properties:
Degree: B, C, D = 3 (hub nodes)
Shortest Path: sp(A, E) = 3 via A–B–C–E
Average Path Length: $\ell(G) \approx 2.3$
Betweenness Centrality (approx.): $C = 0.67$, $A = E = 0$
Clustering Coefficient (local): $C_L(B) = 0.33$, $C_L(C) = 0.67$, $C_L(D) = 0.67$

Figure 3.2: Example graph annotated with essential metrics: degree, centrality, shortest paths, and clustering.

3.4 Applications of Graphs

3.4.1 Graphs as real-world models

Graphs offer a powerful abstraction for representing real-world infrastructures, including electric power systems, transportation grids, communication networks, and beyond. Their intuitive structure, composed of nodes (or vertices) and edges (or links), makes them ideal for modelling systems where the relationships or flows between entities are as relevant as the entities themselves.

In such models, nodes represent physical components such as substations, routers, power plants, or road intersections. Edges capture the physical or logical connections between them—power

lines, fiber links, or road segments, respectively. These nodes and connections can represent the exchange of electricity, data, vehicles, or other types of flow, with graph models naturally accommodating diverse data and metadata through attributes such as direction, capacity, cost, reliability, or latency. Therefore, graphs capture both the structure and, with suitable attributes, the dynamics of complex real-world systems, allowing the model to describe not only fixed infrastructure, but also changing operating conditions.

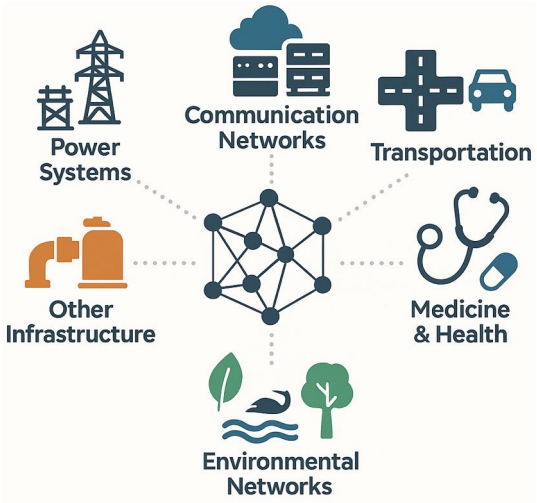


Figure 3.3: Typical domains that can be modelled with graph abstractions.

The potential of graph-based representations resides in their flexibility. The same mathematical framework can describe systems at different scales, from microgrids to national power networks, or from local street networks to global airline systems. It can also integrate multiple domains, such as cyberphysical systems that combine IT and operational infrastructure. With the increase in high-resolution data, graph models are gaining popularity for analysis, simulation, and optimization of complex systems.

3.4.2 Applications of Graphs in the Electricity Sector

Electrical power systems can be naturally represented as graphs, where *vertices* correspond to physical components such as buses, substations, transformers, and loads, and *edges* represent the electrical connections—transmission lines, distribution feeders, or cables—that enable the flow of both energy and operational information. This abstraction provides a powerful, flexible foundation for modelling, analysis, and optimisation of the grid.

Graph-based methods are increasingly being adopted across the electricity sector because they allow the integration of structural topology, physical laws, and operational data into a single analytical framework. This enables applications ranging from vulnerability assessments to real-time digital twins, supporting planning, operations, and asset management in more adaptive and data-driven ways. The following subsections outline several key application areas.

3.4.2.1 Vulnerability and Resilience Analysis

Understanding how the network responds to failures is critical for both operational security and long-term planning. Early studies modelled grids as unweighted graphs and applied metrics such as average path length, betweenness centrality, and clustering coefficient to assess robustness against N–k contingencies—the simultaneous failure of k components [30]. These analyses revealed that real grids are far more sensitive to targeted attacks on a few strategically important nodes or lines than to random failures.

However, purely topological models ignore the physics of electricity flows. To address this, the *net-ability* index [31] incorporates electrical distance and line capacity into edge weights, yielding more realistic rankings of critical components. Later work [32, 33] integrates weighted graphs with probabilistic cascading-failure models, capturing how local outages can propagate through the system and identifying components whose failure would cause disproportionate disruption.

3.4.2.2 Optimal Power Flow and Network Operation

The Optimal Power Flow (OPF) problem seeks the most cost-effective way to dispatch generation and route electricity while satisfying demand and adhering to physical and operational constraints. Expressing the network as a graph transforms OPF into a constrained optimisation problem over that graph. While the full AC formulation is non-linear and computationally demanding, convex relaxations such as semidefinite, chordal, or second-order cone programming often produce exact or near-exact solutions, particularly for radial distribution feeders [34].

Recently, graph neural networks (GNNs) have been applied to learn OPF mappings directly from historical data [35] or through unsupervised optimisation models [36]. These models can deliver fast, approximate solutions suitable for real-time control, contingency screening, and interactive scenario analysis.

3.4.2.3 Asset Management and Maintenance Planning

Electric utilities face the challenge of prioritising inspection, repair, or replacement of assets under constrained budgets. Graph models make it possible to capture not only the health of individual components but also their interdependencies and their role within the wider network. This allows utilities to assess the systemic impact of a potential failure.

Knowledge graphs extend this principle by integrating heterogeneous datasets—sensor measurements, inspection records, maintenance logs—into a unified, queryable structure. For example, RDF-based equipment graphs [37] enable engineers to trace all operational dependencies of a high-voltage transformer in a single search. Other work merges knowledge graphs with big-data analytics to create unified platforms for asset risk assessment and lifecycle management [38].

3.5 Graph-based data models for semantic integration and analysis

3.5.1 From basic to contextual graphs

Earlier sections discussed general graph models and their use in operational systems. However, in these "basic" graphs, the meaning of nodes and edges is not stored within the graph itself. Instead, it is defined in the queries and programs that use the graph. As a result, anyone unfamiliar with the system must first figure out this hidden logic before they can begin meaningful analysis. This dependence on external knowledge slows down exploration and becomes a significant barrier if the original designers are no longer available [39].

These limitations point to the need for more contextual graphs, where structure such as labels, relationship types, properties, or ontological terms is embedded within the graph itself. Making the graph self-descriptive shifts essential context from external code into the data itself, making analysis easier, reducing maintenance, and preserving knowledge over time.

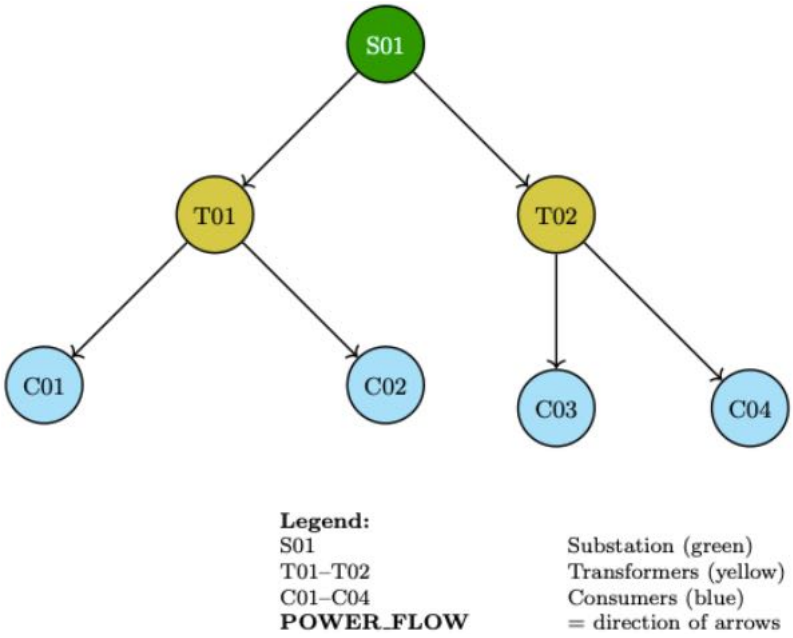


Figure 3.4: Example of basic graph

3.5.2 Main graph data models in practice

In practice, most graph database implementations in industry use one of two main models: *RDF triple graphs* and *property graphs* [40]. Each has different strengths depending on the use case.

3.5.2.1 RDF triple graphs

RDF (Resource Description Framework) is a W3C standard designed for sharing data across the web [41]. It represents information as *triples* in the form:

$$<\text{subject, predicate, object}>$$

- The *subject* is the starting node,
- The *predicate* is the relationship or edge,
- The *object* is the connected node or a value.

All parts of the triple are usually identified with URIs to ensure global consistency. RDF is practical in scenarios requiring *interoperability*, *semantic reasoning*, or *formal ontologies*, such as in academic publishing, industrial automation, or knowledge integration across organisations.

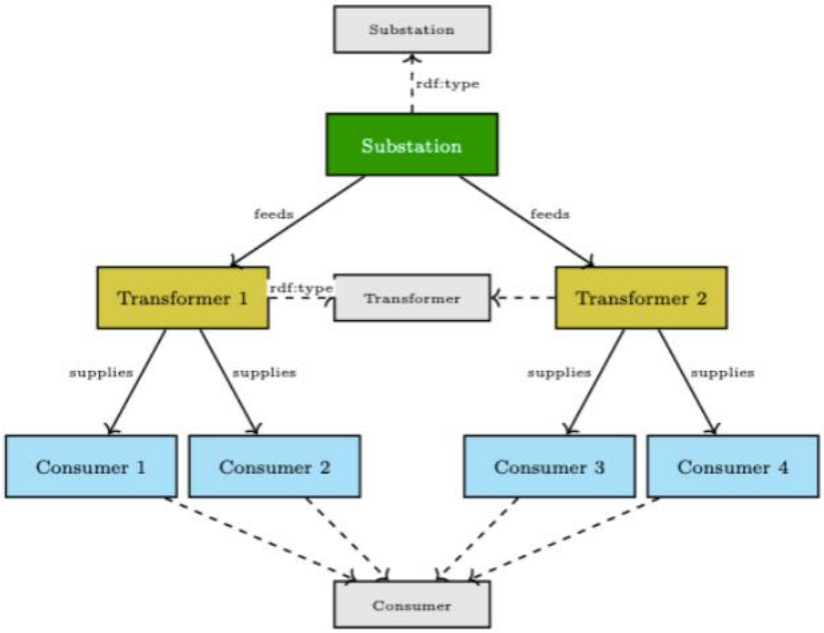


Figure 3.5: Example of a RDF graph

3.5.2.2 Property graphs

Property graphs, on the other hand, are designed for flexibility and speed. They represent data as *nodes* and *relationships*, just like RDF, but include additional structure: each node and relationship can have *key–value properties* as attributes, and nodes can have multiple *labels* [40]. This allows property graphs to mirror how we intuitively think about connected data.

Property graphs support powerful and more interpretable query languages such as Cypher [42], Gremlin [43], and GSQL [44], which are declarative and optimised for pattern matching rather than reasoning. This makes them especially effective for real-time applications like recommen-

dation engines, network analysis, or fraud detection, where performance and clarity matter more than strict ontological formality.

Although there is no universal standard for property graphs, the upcoming *GQL (Graph Query Language)* standard [45], alongside widely adopted implementations like Cypher, is shaping a common approach across platforms.

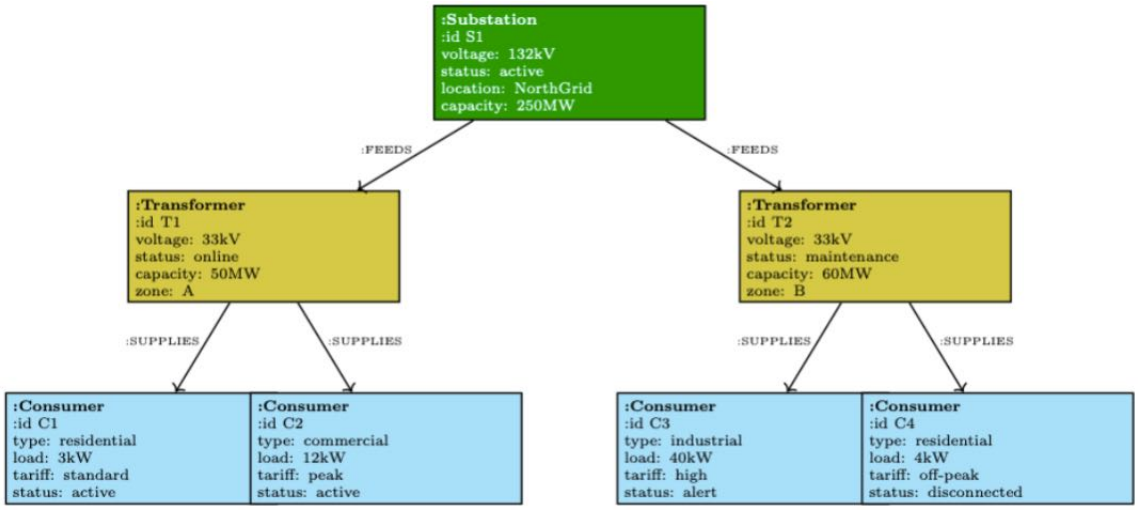


Figure 3.6: Example of a property graph

3.5.3 Knowledge graphs

A knowledge graph is a graph-based model that interlinks entities, relationships, and metadata to represent complex knowledge in a flexible, schema-light format [40]. It supports structured, semi-structured, and unstructured data, making it well-suited for representing knowledge in complex domains [38]. By embedding semantics directly into the graph, knowledge graphs provide a powerful framework for understanding, reasoning, and decision-making in areas like power systems, healthcare, and logistics.

3.5.3.1 Foundations of knowledge graphs

While property graphs bring structure to data through labels, relationship types, and properties, they still fall short of capturing the deeper meaning or intent behind those connections [39]. *Knowledge graphs* take the next step by enriching graph data with *semantic context*—using taxonomies and ontologies to express not just *how* things are related, but *what* those relationships mean in a real-world or business setting.

3.5.3.1.1 Taxonomies: hierarchical classification in graphs

A taxonomy is a hierarchical classification system that organizes concepts using broader-narrower (parent-child) relationships [39]. Items with similar characteristics are grouped under common categories, connected through relationships like *is_a* or *subcategory_of*, enabling reasoning at

different levels of abstraction.

In a power systems knowledge graph, for instance, entities such as *LV Transformer*, *MV Transformer*, and *HV Transformer* may fall under the parent category *Transformer*, which itself belongs to *Electrical Equipment*. This supports semantic queries (e.g., ”all transformers”) and reasoning about substitution or compatibility across classes.

Taxonomies are valuable in domains requiring semantic similarity, structured navigation, and classification-based reasoning—especially in search, recommendation, and data integration. They also support multi-dimensional classification, where an entity can belong to multiple hierarchies (e.g., by voltage, asset type, or priority), allowing flexible, domain-specific insights.

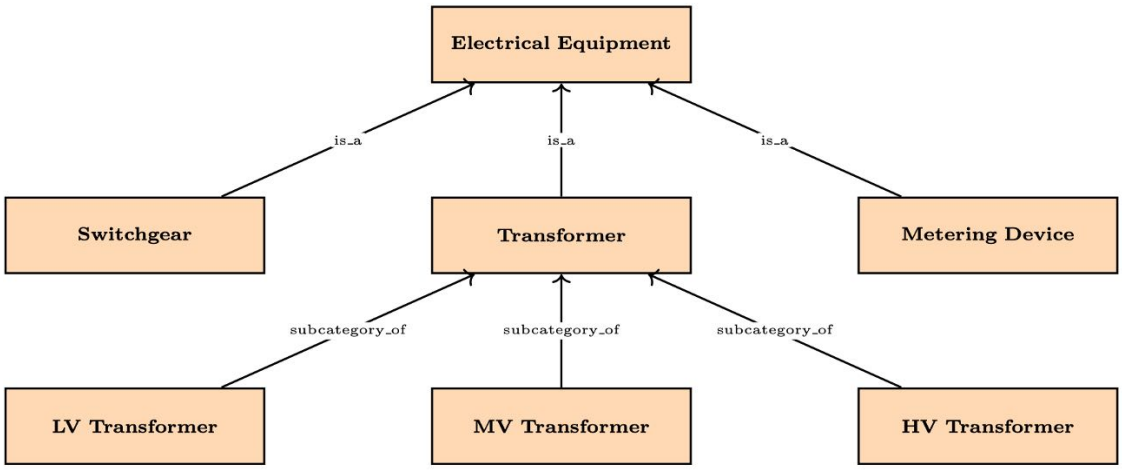


Figure 3.7: Hierarchical taxonomy of electrical equipment. The diagram illustrates how specific components like LV, MV, and HV Transformers are organized as subtypes under the broader class *Transformer*, which itself is part of the category *Electrical Equipment*.

3.5.3.1.2 Ontologies: semantic meaning beyond hierarchies

While taxonomies organize concepts hierarchically, ontologies go further by formally defining concepts, relationships, and rules in a machine-readable format [40]. Unlike taxonomies, ontologies capture richer semantics [39], such as: *part_of* (e.g., a voltage regulator is part of a substation) *compatible_with* (e.g., a transformer type works with a sensor) *requires*, *monitors*, *causes*, etc.

In knowledge graphs, ontologies provide the semantic structure—linking nodes and edges to domain meanings and ensuring data consistency. For example, they can enforce rules like “a Substation must connect to at least one Transformer” or “a Smart Meter only monitors Consumers.”

Ontologies also help align different datasets or taxonomies, making them essential in complex systems where technical, financial, and operational data need to work together.

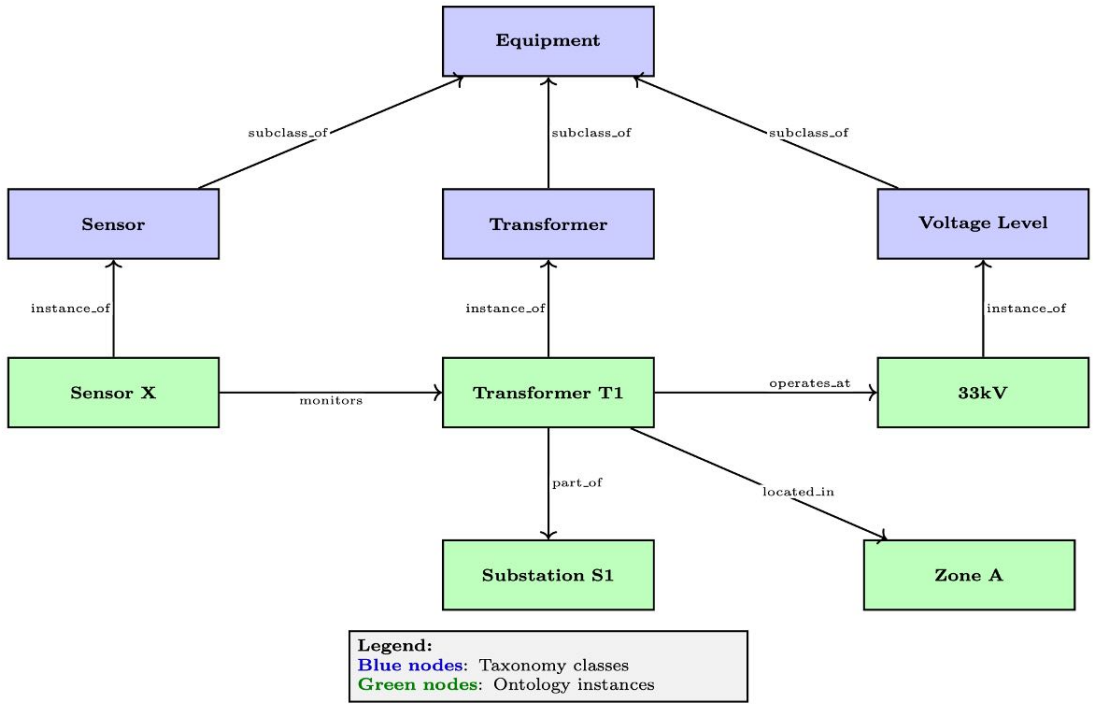


Figure 3.8: Integration of taxonomy and ontology in a knowledge graph for power systems. **Blue nodes** represent taxonomy classes organized hierarchically (e.g., *Transformer* is a subclass of *Equipment*). **Green nodes** denote ontology instances that inherit semantic meaning from their corresponding classes and are connected via domain-specific relationships such as *monitors*.

3.5.3.2 Constructing a knowledge graph

As described in [38], constructing a knowledge graph typically begins with *entity and relationship extraction*, where relevant terms and links are identified in the source data—for example, power lines, substations, sensors, and their operational statuses. This is followed by *knowledge fusion*, which resolves inconsistencies across multiple data sources. Techniques like *entity alignment* ensure that different representations of the same object are unified.

Next, *entity linking* connects these aligned entities into a meaningful structure—such as associating a sensor with the equipment it monitors. Once built, the graph can power *intelligent search*, allowing users to explore and query complex, interconnected data more intuitively.

Additionally, knowledge graphs support *data auditing* by validating consistency and tracking changes, and *data resource management* by monitoring the flow, access, and lifecycle of information [38]. This makes them especially valuable for critical domains like smart power system design, where data must be both accurate and actionable across systems and stakeholders.

4 The Graph Model

4.1 Introduction to the graph model

This chapter introduces the design and implementation of a knowledge graph system for electricity distribution networks. The system integrates data from multiple domains—technical, social, and geographic—into a unified graph structure that enables rich, context-aware analysis of infrastructure performance and planning.

At its core, the model links physical network assets (such as substations, transformers, and meters) with indicators of customer vulnerability, urban–rural classifications, and road infrastructure connectivity. The graph structure allows querying of both technical and social attributes, enabling planners and analysts to explore how engineering risks intersect with social disadvantage.

The development is guided by the contributions outlined in the introduction, particularly the need for a unified, cross-domain framework that supports equity-aware decision-making in utility planning.

4.1.1 Related works

Knowledge graphs (KGs) are gaining interest in smart grids to organise and analyse heterogeneous data. Acting as semantic layers, they connect entities—such as equipment, events, and customers—and their relationships, improving interoperability, search, and reasoning. A typical "big knowledge graph" (BKG) architecture includes layers for data acquisition, graph construction, computation and management, and application services (e.g., customer support, dispatch, operations), often implemented with platforms such as Neo4j, RDF/SPARQL, and distributed processing [46].

One example is the BKG framework described in [46], which integrates internal utility systems with external data sources such as meteorology and GIS. It extracts and models entities and relationships, storing them in big-data or graph databases (e.g., Spark/Hadoop with Neo4j/HBase) to enable semantic search and decision-making. A related review [47] describes a similar workflow—covering knowledge extraction, ontology modelling, reasoning, and regular

updates—which reflects common practice in the field.

In general, reported applications generally fall into four areas :

1. **Search and information services** – improving retrieval of grid data and operational procedures;
2. **Dispatch and fault handling** – organising rules, case histories, and procedures for operator support;
3. **Maintenance and fault diagnosis** – integrating defect logs, condition data, and specifications for diagnosis and planning;
4. **Customer service and decision support** – linking events and assets to service processes.

Construction methods typically involve entity and relation extraction, ontology learning, embedding-based inference, and quality control processes to ensure accuracy and timely updates [46, 47]. Pilot projects demonstrate how KGs can break down "information islands" by aligning equipment data and their interconnections for visualisation and analysis [48].

Despite these advances, reviews still describe the field as early-stage. Challenges remain in keeping knowledge up to date, deepening reasoning capabilities, and integrating data from heterogeneous sources [47]. Furthermore, most current implementations focus on technical and operational data, with limited inclusion of socio-demographic information or links to other infrastructures.

This thesis extends existing approaches by linking network assets with socioeconomic indicators and spatial context (e.g., rurality, road access). In doing so, it enables cross-domain queries that combine technical network data with measures of social vulnerability, supporting analyses that move beyond purely operational concerns and address areas identified in the literature as underexplored.

4.1.2 General aims of the model

The graph model is designed with a clear set of goals that reflect real-world challenges in distribution network management. It acts as a flexible and extensible tool for infrastructure modeling, impact analysis, and planning support. The five main objectives are described in Table 4.1.

Table 4.1: General aims of the graph model

Objective	Purpose	Expected Outcome
O1. Represent infrastructure as a connected graph	Organise network components (such as substations, cables, and meters) into a structured graph that reflects their physical and logical relationships.	Supports network analysis based on structure and connectivity.
O2. Integrate relevant contextual data	Link infrastructure with additional information such as demographic, social, or spatial indicators where available.	Allows for more comprehensive assessments that include local context.
O3. Support scenario-based planning	Enable users to explore and compare the possible effects of planned changes or disruptions in the network.	Helps prioritise actions and assess potential impacts.
O4. Provide a flexible and reusable data structure	Create a consistent format that can be updated with new data and used in different analysis tasks.	Supports future expansion and use across multiple applications.
O5. Improve clarity and communication	Produce outputs that help explain decisions to both technical and non-technical audiences.	Supports better collaboration and engagement with external stakeholders.

4.1.3 Motivation for the model

The transition to a low-carbon, data-rich energy system is reshaping the responsibilities of Distribution Network Operators (DNOs) across the UK. Organizations like Scottish Power Energy Networks must now plan, operate, and invest in increasingly complex low-voltage infrastructures while meeting societal expectations for fairness, transparency, and resilience.

At the same time, regulatory frameworks (such as RIIO-ED2) and stakeholder engagement requirements are pushing utilities to demonstrate social impact awareness alongside technical performance. Traditional tools and siloed databases fall short in capturing these multi-layered realities.

In response, this work proposes a graph-based socio-technical data model that bridges engineering metrics with social context. Graph databases offer a natural way to represent infrastructure components and their interdependencies—while also embedding semantic links to the communities they serve.

By aligning technical infrastructure data with contextual indicators, the model promotes a more equitable, evidence-based approach to network planning. In particular, the proposed system allows distribution network operators (DNOs) to:

- Identify critical infrastructure serving vulnerable populations.
- Simulate disruption scenarios and their social impact.
- Support investment decisions that are technically justified and socially fair.

4.1.4 Industrial applications of the model

The model's utility extends across several operational and strategic domains within DNOs (see Table 4.2).

Table 4.2: Industrial applications of the graph model for DNOs

Impact Area	Value Proposition	Outcomes
Strategic	Embeds social responsibility into long-term network planning, reinforcing corporate reputation and stakeholder trust.	*Data-driven investment roadmaps that visibly prioritise deprived communities; improved engagement with local authorities and community groups.*
Regulatory	Provides auditable evidence that asset interventions are both technically justified and socially fair, easing compliance with evolving Ofgem requirements.	*Clear documentation of how socioeconomic indicators influence project prioritisation; stronger submissions for RIIO price-control reviews.*
Operational	Supports day-to-day decisions by linking fault risk, customer vulnerability and logistics in a single view.	*Dynamic work-order scheduling that sends crews to faults with highest combined technical and social impact; proactive inventory planning to prevent material shortages.*
Internal Collaboration	Serves as a shared data backbone for planners, analysts and field engineers.	*Reduced data silos; faster cross-team analyses for EV-charging roll-outs, climate-resilience studies and outage simulations.*

By linking different types of data into one connected model, the system helps DNOs shift their focus from just assets to the people and communities those assets serve. It supports fairer, more transparent decisions while also improving how teams work and how companies respond to regulatory demands.

4.2 System architecture

The implementation of the graph model relies on a modular system architecture designed to support data ingestion, transformation, graph construction, and deployment across multiple environments. This architecture was developed with flexibility and scalability in mind, allowing for both local prototyping using Neo4j and cloud-based deployment on Amazon Neptune.

At its core, the system ingests technical and contextual datasets from various sources—including network asset records, outage logs, socioeconomic indicators, and geographic metadata—and transforms them into a consistent, ontology-informed graph schema. This data processing pipeline is complemented by a deployment layer that enables users to interact with the graph using modern database tools and visual interfaces.

The system supports two primary deployment configurations:

- A local Neo4j environment, accessible via Docker and suited for development, testing, and visualization using Cypher queries.
- A secure, cloud-hosted deployment on Amazon Neptune, connected via a WireGuard VPN tunnel to ensure compliance with SPEN's privacy and network policies.

The remainder of this section provides a description of the architecture, covering the data processing workflow, database design, and the technical components involved in each deployment scenario.

4.2.1 Graph construction methodology

The development of the graph model follows a structured pipeline that transforms raw data and semantic definitions into a graph-based representation of the electricity distribution network and its surrounding context. This methodology is illustrated in Figure 4.1.

As shown in the diagram, the methodology consists of six stages:

1. **Loading data from several sources**
   Multiple heterogeneous data sources are ingested, including infrastructure asset registries, outage event logs, socioeconomic indicators and spatial metadata.
2. **Preprocessing and standardisation**
   Raw data is cleaned, validated, and normalised. This includes steps such as standardizing geographic references, filtering for relevant voltage levels, and resolving missing or ambiguous entries.
3. **Merging into structured tables**
   Preprocessed datasets are joined and aligned using shared keys such as equipment IDs or postcode units. This results in a set of intermediate tabular datasets that consolidate both technical and contextual attributes.
4. **Ontology-based semantic mapping**
   In parallel, an ontology is loaded to define domain semantics. This ontology specifies the relevant classes (e.g., `Transformer`, `SmartMeter`, `Postcode`) and relationships (e.g., `CONNECTED_TO`, `SERVES`) that will govern the graph schema.
5. **Extraction of nodes and relationships**
   Based on the ontology and structured data, the system extracts nodes and edges representing entities and their interconnections. Each node and relationship is annotated with relevant metadata (e.g., voltage level, deprivation index, location coordinates).

6. **CSV generation and graph loading**
   The extracted data is exported as two CSV files—`nodes.csv` and `edges.csv`—following the format required by the target graph database. These are then ingested into the database using Cypher scripts, forming the final graph model.

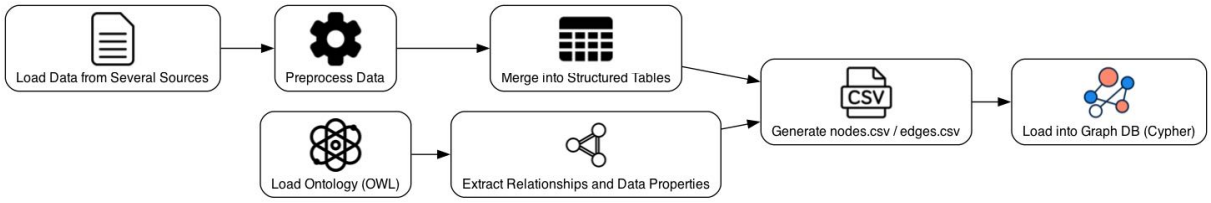


Figure 4.1: End-to-end methodology for graph model construction.

This modular and ontology-driven approach ensures that the resulting graph is not only topologically valid but also semantically meaningful, supporting both intuitive querying and scalability as new datasets or analytical requirements emerge. Section 4.3 describes each step of the pipeline in greater detail.

4.2.2 Local deployment: Neo4j

The initial deployment configuration used for the graph system was a local Neo4j environment. This configuration aims at supporting faster prototyping and iterative development without incurring infrastructure costs or requiring cloud access. This setup is illustrated in Figure 4.2.

4.2.2.1 Motivations behind the use of Neo4j

The two main motivations for adopting this approach during the development phase are as follows:

- **Rapid, non-disruptive prototyping**: Developers need a flexible environment in which the graph model can be tested, updated, and queried without depending on external systems or enterprise infrastructure. By running Neo4j locally via Docker, the development process can proceed independently and with minimal setup time.
- **Proof of concept for Neo4j as a candidate for enterprise deployment**: While the final decision for production deployment was to use Amazon Neptune (as described in Section 4.4.2.1), this local setup served as an initial validation of Neo4j's capabilities, particularly in terms of graph query functionality, semantic structuring, and ease of use.

4.2.2.2 Advantages of Using Neo4j

Neo4j offers several strengths that make it a suitable choice for this project. One of its key advantages is the Cypher query language [42], which provides an expressive and intuitive way

to perform graph pattern matching. This makes it easier to explore complex relationships within the data.

Another important feature is Neo4j's native support for the property graph model. This allows both nodes and relationships to carry rich metadata, enabling more informative representations of the data.

Neo4j also benefits from a well-developed open-source ecosystem, which includes a wide range of tools for data import, export, visualization, and automation. These resources contribute to a more flexible and scalable development workflow.

In addition to its technical strengths, Neo4j Desktop [49] provides excellent visualization capabilities. Its user-friendly interface enables users to explore the graph structure visually and interact with query results without the need for programming. This makes it especially valuable in collaborative environments, where different stakeholders—such as data scientists, engineers, and decision-makers—need to engage with the graph data in accessible ways.

4.2.2.3 Neo4j configuration and licensing information

In this project, the Neo4j Community Edition was used. This edition is fully featured and licensed under the GNU General Public License v3 (GPL v3) [50], which permits free use in both open-source and closed-source projects, whether on personal devices, internal servers, or cloud infrastructure. Its permissive licensing made it ideal for academic and research use during the thesis development phase.

> *"Neo4j Community Edition is a fully featured, best-in-class graph database that uses the GPL v3 license. We chose GPL because it means that Neo4j Community Edition can be used for free with your project: whether in the cloud or behind the firewall."*
> — **Neo4j Documentation**

This configuration proved especially effective for early-stage schema design, relationship exploration, and query development. Once the model was validated and refined locally, it could be exported (in CSV form) and loaded into the enterprise graph engine used in the production deployment.

The detailed configuration steps, including Docker setup, CSV import scripts, and example queries, will be discussed further in Section 4.4.1.

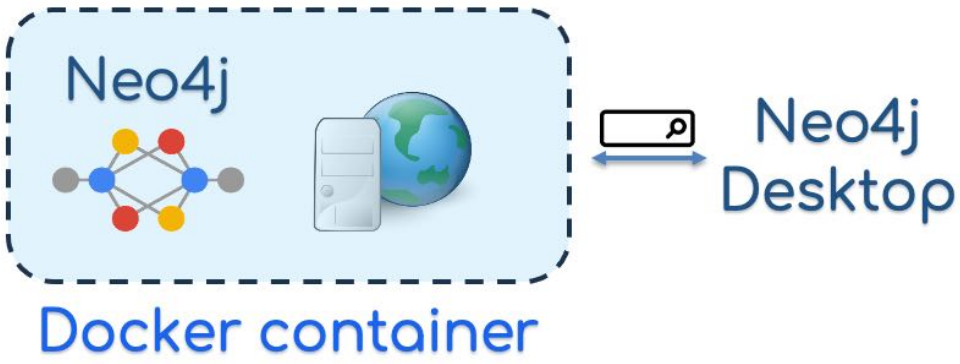


Figure 4.2: Local development architecture using Neo4j and Docker.

4.2.3 Private cloud deployment: Amazon Neptune

For enterprise deployment, the graph system was set up on Amazon Neptune, running inside the Virtual Private Cloud (VPC) environment managed by SPEN. As shown in Figure 4.3, this cloud-based setup provides a secure and scalable platform that's well-suited to production use.

Neptune was chosen largely because it fits well within SPEN's existing Amazon Web Services (AWS) infrastructure. It integrates easily with services like AWS Lambda and supports a multi-language approach, allowing queries to be run using Gremlin, SPARQL, and OpenCypher within the same database instance [51].

> *"Neptune supports the popular property-graph query languages Apache TinkerPop Gremlin and Neo4j's openCypher, and the W3C's RDF query language, SPARQL. This enables you to build queries that efficiently navigate highly connected datasets."* — AWS Neptune Documentation

This flexibility enables teams with diverse backgrounds to collaborate more effectively. Developers can stick with the query languages they know, and existing code from other platforms, such as Neo4j, can often be reused with minimal changes. For this project, OpenCypher was utilized throughout, which facilitated a seamless transition from earlier development work conducted in Neo4j.

To allow users to explore and query the graph visually in the cloud, the Neptune Graph Explorer tool [52] was deployed as a Docker container on users' workstations. This tool offers a web-based interface that serves as a counterpart to Neo4j Desktop, enabling users to run queries and explore the graph structure in a more intuitive manner.

> *"Graph Explorer supports visualizing both property graphs and RDF graphs."* — Graph Explorer Documentation

Access to the Neptune instance is protected using a WireGuard VPN tunnel, which allows external users to connect securely without exposing the database publicly. This setup meets

SPEN's internal cybersecurity requirements while still supporting collaboration across teams.

In addition to its security and compatibility, Neptune offers strong scalability. It can handle growing datasets and multiple users without much manual effort. Features like integration with AWS Lambda also open the door for automation—whether it's regular data imports, triggering alerts, or running parts of an analytics pipeline.

Details about the technical setup, including VPN access, data import routines, and tool configuration, are provided in Section 4.4.2.

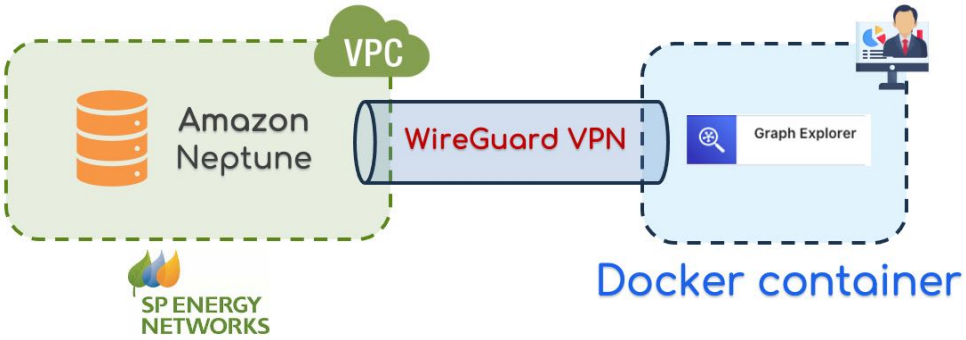


Figure 4.3: Private cloud enterprise architecture using Amazon Neptune and Docker.

4.3 Implementation

The graph model was developed through a structured pipeline designed to convert raw, heterogeneous datasets into a semantically enriched graph suitable for both querying and analysis. As outlined in Figure 4.1, the pipeline consists of six stages, starting with data ingestion and ending with the final loading into the graph database.

The subsections that follow provide a detailed explanation of each stage in the pipeline, beginning with the input data sources.

4.3.1 Input data sources and formats

The graph model brings together datasets from a range of domains to capture the technical, social, and spatial aspects of the electricity distribution network. The datasets used in the model can be grouped into four main categories:

1. **Technical infrastructure data** provided by SPEN, which includes detailed information on distribution network assets, customer connection points, and smart meter telemetry.

2. **Socioeconomic indicators** that help assess levels of community vulnerability, deprivation, and demographic characteristics across different areas.

3. **Mobility and transport data** related to electric vehicle (EV) charging infrastructure and the road network, used to evaluate spatial accessibility and readiness for electrification.

4. **Geospatial classification schemes**, which enable postcodes and regions to be categorised according to population density, accessibility, and characteristics of the built environment.

Table 4.3: Overview of datasets used in the graph model

Category	Dataset Name	Description	Format(s)	Source
Technical Infrastructure	SPEN Asset Registry	Inventory of distribution infrastructure assets (transformers, fuses, protection devices, etc.), including asset IDs, types, spatial coordinates, and network-related metadata.	CSV / XLSX	SPEN
	SPEN Line and Link Data	Spatial and logical topology of cables and junctions connecting assets across the network.	CSV	SPEN
	Customer Address Mapping	Address and postcode information of the customer premises and smart meters.	CSV	SPEN
	Smart Meter Voltage Alerts	One week of half-hourly voltage readings from customer smart meters, including minimum voltage, maximum voltage, and demand.	CSV	SPEN
	Smart Meter Fault Alerts	Event-based logs of fault conditions at the smart meter level (e.g. supply loss).	CSV	SPEN
Socioeconomic Context	Scottish Index of Multiple Deprivation (2020)	A relative measure of deprivation across 6,976 small areas (data zones) in Scotland. Provides postcode-level scores across seven domains: income, employment, health, education, geographic access, housing, and crime. Includes both decile and quintile rankings.	XLSX	[53]
Mobility and Transport	ChargePlace Scotland EV Infrastructure	Dataset of public electric vehicle (EV) charging stations across Scotland, including locations, connector types, power ratings (kW), network operators, and operational status.	CSV / XLSX	[54]
	OS Open Roads	National road network dataset for Great Britain, including geometries, road classifications, and road segment linkages. Used for spatial accessibility and transport network analysis.	GPKG	[55]
Geospatial Classification	Urban–Rural Classification (2022)	Standard classification from the Scottish Government that defines rurality based on settlement size (under 3,000 people) and remoteness using drive-time from settlements of 10,000 or more.	CSV / XLSX	[5]
	Spatial Signatures Framework	Typology of over small areas (datazones) in Great Britain based on built form, land use, density, and functional characteristics.	CSV / GPKG	[18]

Together, these datasets form the foundation of the semantic graph model. By integrating these diverse sources, the model can represent relationships between infrastructure assets, communities, and spatial features in a structured and queryable format. A summary of the key datasets used is provided in Table 4.3.

4.3.2 Data preprocessing

Before building the knowledge graph, a data preprocessing pipeline was created to prepare each dataset for integration. This step ensured that data—regardless of its source, format, or domain—was clean, consistent, and compatible with the graph architecture developed later in the pipeline.

Given the range of data involved, preprocessing was carried out in several stages:

- **General cleaning and standardisation**, applied across all datasets. This included tasks such as field name standardisation, spatial reference alignment, and the correction of common encoding issues.
- **Dataset-specific processing**, tailored to the unique characteristics of each input.
- **Postcode completion**, ensuring that every spatial record was associated with a valid postcode. This step was critical for enabling spatial joins and adding contextual metadata.
- **Geographic filtering**, used to restrict the dataset to the defined study area and remove records outside the area of interest.

The following subsections describe each stage in more detail, beginning with the general preprocessing procedures applied across all input files.

4.3.2.1 General preprocessing

The general preprocessing stage was implemented in Python, utilising the `pandas`, `geopandas`, and `datetime` libraries. This stage was responsible for cleaning and standardising all input datasets to ensure consistency and interoperability across diverse data domains. The preprocessing involved the following steps:

1. **Data Cleaning and Normalisation.** The first step involved data cleaning and normalisation. Datasets were checked for missing values in critical fields—such as asset identifiers, postcodes, and spatial coordinates—which were either removed or flagged, depending on their importance. Duplicate records were dropped based on unique identifiers to prevent redundancy. Column names were standardised across all datasets using a consistent naming convention (e.g., `asset_id`, `asset_type`, `postcode`), supporting automated parsing and integration. Measurement units were verified and harmonised, with voltages expressed in volts, electric power figures converted to kilowatts (kW) and durations in minutes.
2. **Spatial Harmonisation.** Spatial harmonisation was applied to all georeferenced data. Geographic coordinates were validated and reprojected to a common coordinate reference system—WGS84 (`EPSG:4326`)—to ensure compatibility with mapping tools and external

spatial datasets. Postcode fields were cleaned by uppercasing, trimming whitespace, and removing any non-standard characters, making them suitable for spatial joins.

3. **Temporal Parsing and Aggregation.** Temporal fields, which appeared across several datasets, also required standardisation. In many cases, dates and times were stored as strings or in inconsistent formats. These were parsed into proper `datetime` objects using the `datetime` module, ensuring full support for time-based operations such as filtering, sorting, and aggregation. This applied to a variety of time-stamped data, including charging events, smart meter readings, and other event-based logs. Where relevant, summary statistics such as durations or averages were computed to simplify downstream processing while retaining key temporal insights. Temporal values were also aligned across datasets to support consistent joins and comparisons.

4. **Format Standardisation and Export.** Finally, the cleaned and standardised datasets were exported using a uniform structure. All files were saved as UTF-8 encoded CSVs, with quoted string fields and standardised column headers. These outputs served as the input to the next stage of the pipeline—intermediate table structuring and merging—as described in Section 4.3.3.

4.3.2.2 Specific preprocessing

4.3.2.2.1 Linking MSP nodes, customer addresses and smart-meter data

Each `LV_MSP` (low-voltage metering service point) in the asset register represents the point at which electricity is delivered from the network to a building. In many cases, a single MSP serves multiple premises—such as individual flats or business units—which are recorded in the corporate address book as separate entries, each with its own Unique Property Reference Number (UPRN), street address, and postcode.

As part of the preprocessing workflow, these two datasets were linked: for every `LV_MSP`, all associated addresses referencing the same MSP identifier were retrieved, allowing each network node to be enriched with a full list of the premises it supplies.

Not all addresses are equipped with smart meters, so the next step was to check which of the linked premises also appear in the smart meter register. Where a match existed, the corresponding service-point node was further enriched with meter-level details—such as an additional postcode, regional tags, and voltage statistics. Addresses without a smart meter were retained, but their meter-related fields were left blank.

By the end of this process, each service-point node contained:

- A list of the individual premises it supplies, identified by their UPRNs and postcodes.
- Information on whether each premise has a live smart meter, and if so, the voltage

conditions recorded at that location.

4.3.2.2.2 Re-indexing the Urban–Rural Classification

The Scottish Government's 2022 Urban–Rural Classification is published on Data Zone 2022 codes, whereas the Scottish Index of Multiple Deprivation (`SIMD 2020`) and most of our other contextual layers use Data Zone 2011.

To bring both hierarchies onto the same spatial reference, a postcode-based translation was performed. From the cleaned postcode tables generated in the earlier preprocessing step, every postcode was already linked to both its 2011 and its 2022 data zone codes. Using those linkages as a bridge, each urban–rural record (in 2022 codes) was mapped to all postcodes that fall inside that zone. Those same postcodes were then looked up in the 2011 directory, giving us the corresponding Data Zone 2011 for each record.

The result is a one-to-one table that expresses the official Urban–Rural classes on the 2011 geography, allowing them to be merged directly with `SIMD` scores and any other dataset that relies on Data Zone 2011 identifiers.

4.3.2.2.3 Assigning spatial-signature types to postcodes

To capture more detailed built-environment context than a basic urban–rural split, the project uses the *Spatial Signatures Framework*, which classifies places across Great Britain into sixteen types—such as dense urban core, peri-urban fringe, or dispersed rural. The data is provided as a GeoPackage (`.gpkg`) with polygon layers covering the entire country.

Because analysis is carried out at postcode level, each postcode in the dataset needed to be assigned a spatial signature type. This was accomplished using a custom Python tool that builds and maintains a persistent postcode-to-signature mapping table. The process includes the following steps:

1. **Coordinate Resolution:** For any postcodes not yet seen, the script uses the `postcodes.io` [56] bulk API to fetch coordinates. Both WGS84 (latitude/longitude) and British National Grid (eastings/northings) references are retrieved.

2. **Geometry Projection:** Coordinates are transformed to match the coordinate system used in the spatial signatures dataset (EPSG:27700) and converted into geometry points.

3. **Point-in-Polygon Join:** Each postcode point is spatially joined to the signature polygons using a standard containment check. If a point lies within a polygon, the corresponding signature type is assigned.

4. **Snapping for Unmatched Points:** If a point falls outside all polygons (e.g. due to being located on a road centre-line), the script uses nearest-neighbour snapping to find and assign the closest polygon. This ensures full coverage.

5. **Persistent Storage:** All postcode-to-signature pairs are saved in a local Parquet file. If the file or its directory does not exist, it is created automatically. Future runs process only genuinely new postcodes, making the operation efficient and incremental.

6. **In-Place Enrichment:** Optionally, the tool can add a signature column directly to a given `DataFrame` or `GeoDataFrame`, making it easy to integrate into preprocessing workflows.

The result is a complete and reusable mapping between postcodes and spatial signature types.

4.3.2.3 Postcode filling

Many of the datasets used in the graph model included spatial coordinates (latitude and longitude), but several were missing postcode information. For instance, the SPEN infrastructure registry gives exact locations for each asset but does not include postcodes. Similarly, the OS Open Roads node data had no direct link to postcode units.

Postcodes are important for connecting spatial data with contextual information—like rurality, deprivation, or transport accessibility. They make it possible to run queries such as *"transformers in rural areas with high social vulnerability"* or *"EV chargers in areas with good transport links."* To enable this, a reverse geocoding step was added to assign postcodes to all records with coordinates, but missing postcodes.

4.3.2.3.1 Postcode boundary access constraints

Ideally, postcode assignment would rely on official boundary data from Ordnance Survey. However, datasets like *Code-Point with Polygons* are not openly licensed and are only available to organisations covered by the Public Sector Geospatial Agreement (PSGA) [57].

Because of these restrictions, a custom three-step reverse geocoding process was built using open-source alternatives. It combined public APIs, open boundary datasets, and proximity matching based on postcode centroids to assign postcodes to all records. This approach ensured complete coverage while keeping accuracy and reproducibility at acceptable levels.

4.3.2.3.2 Reverse postcode geocoding pipeline

To link every spatial record to a valid UK postcode, a three-step reverse geocoding pipeline was implemented. Each method was applied in order—if one failed to return a match, the next was used—ensuring full coverage without sacrificing spatial accuracy.

1. **Primary Method:** `postcodes.io` **API**
   The first method used `postcodes.io`, an open UK postcode API based on Ordnance Survey data [56]. It supports bulk reverse geocoding of geographic coordinates and is widely recognised for its reliability and accuracy.

   This approach successfully resolved the majority of location records. However, approxi-

mately 15% of coordinates returned no results and were passed to the secondary method.

2. **Secondary Method: Longair Postcode Boundaries**
   The second method relied on an open dataset, which includes polygon geometries for postcode units across Great Britain [58]. The data was converted into Parquet format and indexed using a spatial index via the `GeoPandas` and `Shapely` Python libraries.

   For each unresolved point, a spatial join was used to assign the postcode of the polygon containing that point. Although unofficial, this dataset proved accurate—manual validation of 200 sample records against Google Maps confirmed high consistency. As a result, results from this method were accepted.

3. **Fallback Method: Centroid-Based Code-Point Lookup**
   For the remaining 1–2% of points that remained unmatched, a final fallback method was used. Postcodes were assigned based on proximity to the nearest postcode centroid, using the *Code-Point Open* dataset provided by Ordnance Survey under the Open Government Licence [59].

   While this centroid-based method is less precise than polygonal boundaries, it provided a practical and reliable fallback to ensure that no spatial record was left without postcode metadata.

4.3.2.4 Region selection

As described earlier in Section 5.2.1.1, a **pilot region** in South Scotland was chosen to support the development and testing of the graph model. While some datasets—such as the smart meter measurements or asset registries—were already scoped to this area, others, particularly the *national road network* (OS Open Roads) and spatial typology datasets, covered the entirety of Great Britain.

To reduce unnecessary processing time, memory usage, and storage requirements, a custom **interactive visual tool** was developed. This allowed users to define a geographic boundary of interest directly on a map, ensuring that only relevant records from large national datasets were retained for graph construction.

4.3.2.4.1 Interactive spatial filtering tool.

The tool was implemented as an interactive web-based map (see Figure 4.4 where users can:

- View the full electric network nodes,
- Click on points to reveal metadata (such as ID, type, voltage level),
- Draw a rectangular or polygonal area defining the region of interest.

Once drawn, the selected shape is exported in `GeoJSON` format. The geometry is then used as a spatial mask: datasets containing latitude and longitude coordinates were filtered to include only the features *within* the selected region. The result is a clean, reusable `GPKG` (GeoPackage) file containing just the subset of interest.

4.3.2.4.2 Advantages of this approach

This filtering approach offers several benefits:

- **Efficiency:** Avoids processing millions of irrelevant records.
- **Flexibility:** Enables rapid redefinition of the study region by simply redrawing the boundary.
- **Visual validation:** Provides a map-based interface to confirm that the selection aligns with real-world infrastructure clusters or administrative zones.

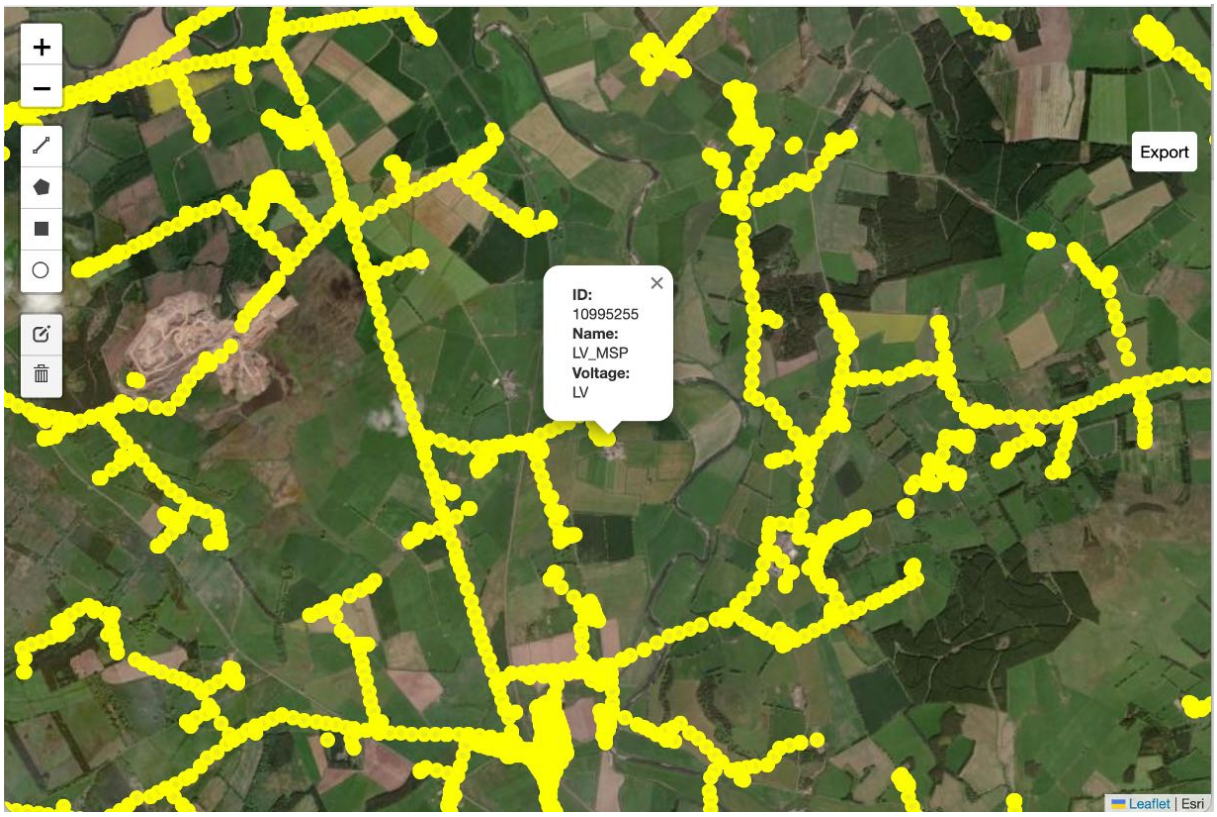


Figure 4.4: Interactive map view showing network infrastructure nodes in the selected region. Metadata such as `ID`, `Name`, and `Voltage` is displayed when hovering or clicking on each node. This feature helps users visually inspect and verify the assets included in the graph model.

Figure 4.5 illustrates the selection process. Yellow points represent distribution network nodes, and the blue rectangle defines the selected area used throughout the analysis.

Figure 4.5: Interactive map-based selection of the study region using the developed filtering tool.

4.3.3 Merging into structured tables

After preprocessing, the cleaned datasets were combined into a set of structured intermediate tables. These tables were designed using a relational data model (see Figure 4.6), where each dataset is separated into logically distinct tables—similar to how information is stored in a traditional enterprise database.

Each table represents a clear entity or concept—such as smart meters, EV charging sessions, electrical assets, or postcode lookups—and is linked to other tables through well-defined relationships using foreign keys. This structure helps maintain data integrity, avoids duplication, and makes it easier to scale the data for use in downstream pipelines or integration into knowledge graphs.

Organising the data in this way also supports consistent data management across different parts of an organisation. By working from the same structured schema, teams in operations, planning, and regulatory roles can access and report on the data in a unified and consistent manner.

The structured tables serve two main purposes:

- They bring together information from different sources into a single, consistent schema (e.g., merging smart meter voltage alerts with their geographic and administrative context).
- They provide a normalised format that is easier to load into graph or relational systems for querying, analysis, or visualisation (e.g., joining EV charging sessions with charger

metadata and postcode-level attributes).

Figure 4.6 shows the full relational schema used to organise the processed data.

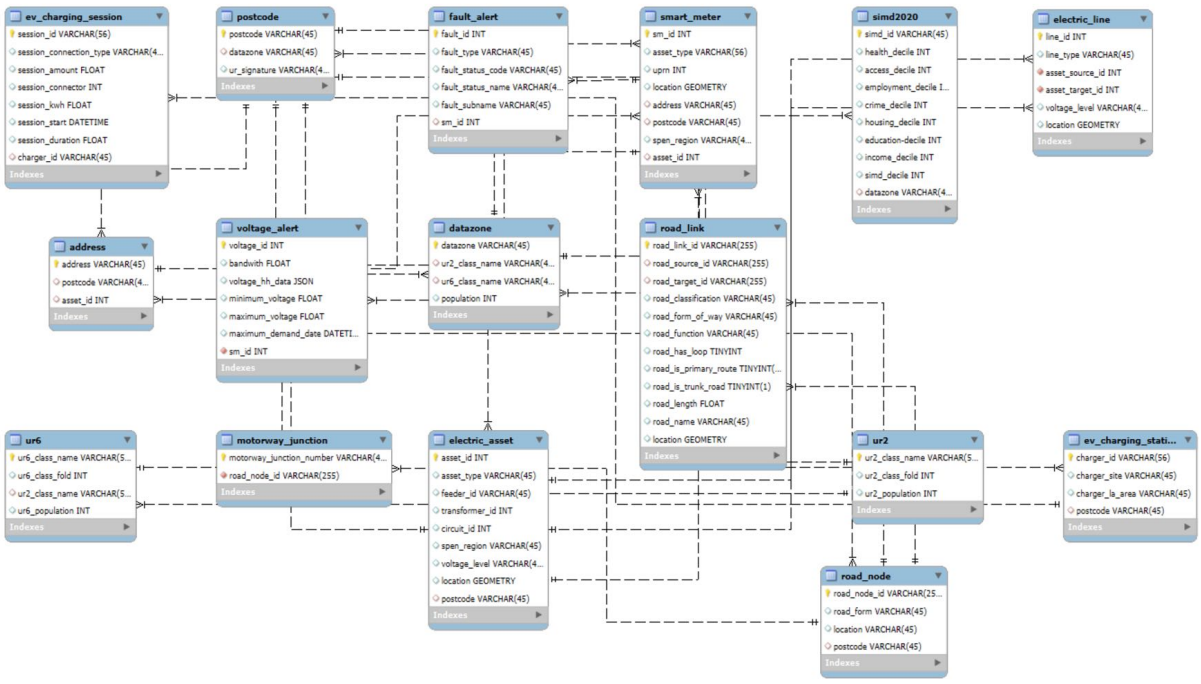


Figure 4.6: Relational data model used to structure the cleaned datasets into intermediate tables.

4.3.4 Ontology-based semantic mapping

While a relational schema organises data efficiently, it does not explain what each entity means or how it should relate to others. To make the structure and meaning of the data explicit, a domain ontology was developed using Protégé [60].

The ontology defines:

- What entities exist in the system (as classes and subclasses),
- How those entities are related (through object properties), and
- What attributes describe each class (as data properties).

The final ontology is exported as a Turtle (`.ttl`) file, which is used by later stages of the pipeline to automatically generate nodes and relationships in the graph.

4.3.4.1 Taxonomy: Object class definition

The class hierarchy (shown in Figures 4.7 and 4.8) was designed with two main goals in mind: *differentiation* and *scalability*.

4.3.4.1.1 Differentiation

Each class represents a distinct concept—such as a type of asset (e.g., `LV_MSP`, `Primary_Transformer`), a spatial unit (`Postcode`, `Datazone`), or an event or measurement (`VoltageAlert`). This separation supports targeted and interpretable queries. For example:

> "How many `LV_Fuses` protect a `Busbar` that feeds rural postcodes in the lowest health decile?"

The overall class structure is divided into several branches. Figure 4.7 shows the taxonomy of spatial, demographic, and infrastructure-related classes, while Figure 4.8 presents the hierarchy for electrical and operational concepts.

4.3.4.1.2 Scalability

The class structure is flexible enough to support future extensions. New equipment types, alert categories, or spatial layers can be added under existing parent classes without changing the overall structure. For instance:

- A new `LV_MonitorAlert` can be added under `Alert` → `MeasurementAlert`.
- Additional transport modes (e.g., cycling or rail) can be added as siblings to existing `TransportNode` or `TransportLink` classes.

This structure balances detail and flexibility, keeping the model both expressive and adaptable to evolving data.

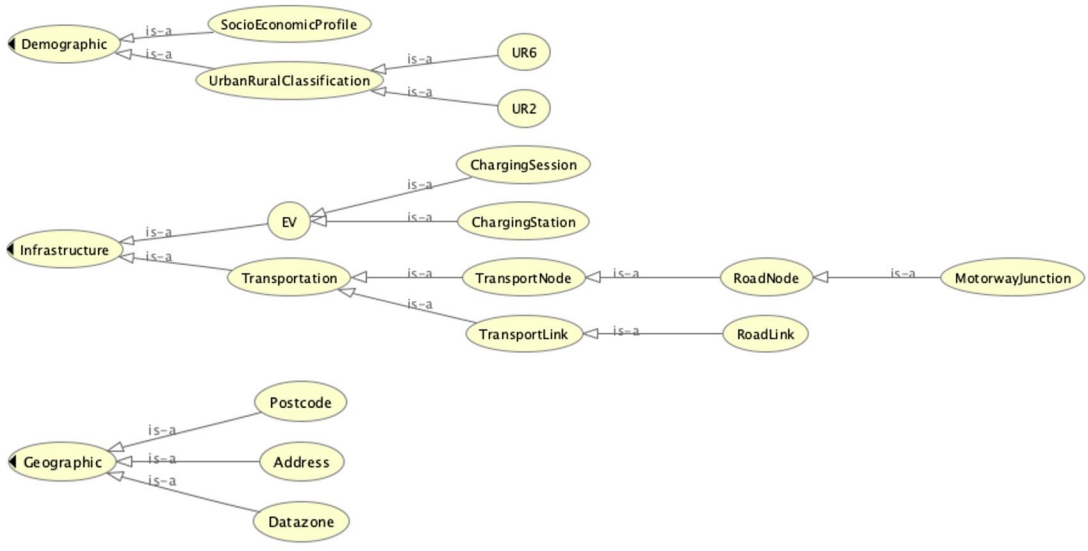


Figure 4.7: Top-level taxonomy for non-electrical domains.

Figure 4.8: Electric-infrastructure taxonomy.

4.3.4.2 Relationships: Object properties definition

Object properties were defined to reflect real-world operational and physical relationships, making graph queries semantically meaningful and aligned with how the system works in practice.

Examples include:

- A transformer *supplies* a feeder
- A smart meter *generates* a voltage alert

These relationships help ensure that graph queries reflect real processes—such as tracing the flow of electricity through the network or linking alerts back to monitoring devices.

In addition, foreign keys from the structured tables (see Figure 4.6) were used to identify how entities are naturally connected in the data. These links informed the design of object properties, ensuring consistency between the ontology and the underlying relational schema.

Where the relationship is inherently undirected—such as a cable linking two joints—the corresponding property was modelled as *symmetric* (e.g., `INTERCONNECTS`, `CONNECTS_TO`), allowing bidirectional traversal during graph analysis.

All class-to-class relationships are summarised in Table 4.4, which provides a complete reference for how entities are connected within the ontology.

Subject Entity	Object Entity	Relationship
Postcode	Datazone	BELONGS_TO
RoadNode	RoadNode	CONNECTED_TO
ChargingStation	ChargingSession	HAS_CHARGING_SESSION
SmartMeter	FaultAlert	HAS_FAULT_ALERT
SmartMeter	VoltageAlert	HAS_VOLTAGE_ALERT
LV_MSP	SmartMeter	HAS_SMART_METER
Datazone	SocioEconomicProfile	HAS_SOCIOECONOMIC_PROFILE
Address	Postcode	IN_POSTCODE
SmartMeter, ChargingStation	Address	LOCATED_AT
LV_MSP	Address	SERVES_ADDRESS
Dist_Transformer	Intersection_Point_HV, Added_Endpoint_HV	SUPPLIES
Intersection_Point_HV, Added_Endpoint_HV	Dist_Transformer	SUPPLIED_BY
HV_MSP, LV_MSP, Unmetered_SP	ElectricAsset	SERVED_BY
ElectricAsset	HV_MSP, LV_MSP, Unmetered_SP	SERVES
LV_Fuse, Protective_Device, Dynamic_Protective_Device	ElectricAsset	PROTECTS
ElectricAsset	LV_Fuse, Protective_Device, Dynamic_Protective_Device	PROTECTED_BY
Electric switches, joints, links, intersections	Same types	INTERCONNECTS
Added_Endpoint_*	Intersections and Joints	CONNECTS_TO

Table 4.4: Ontology object properties linking subject and object entities.

4.3.4.3 Data Properties: assigning attributes to classes

For each class in the ontology, relevant attributes were defined based on the structured tables. These data properties match the field names and data types defined in the relational data model (see Figure 4.6), allowing the graph loader to instantiate individuals directly from table rows—without the need for manual field mapping.

4.3.5 Entities/relationships extraction and CSV generation

The final stage of the pipeline involves converting the cleaned tables and ontology-derived semantics into two plain-text files: `nodes.csv` and `edges.csv`. These files are formatted to support bulk loading into Neo4j and are also fully compatible with Amazon Neptune's Open-Cypher bulk loader format, as described in the AWS documentation [61]. No manual adjustments are required—both platforms accept the output as-is.

Step 1 — Build an ontology reference. The process begins by parsing the ontology (in Turtle format) to extract a reference sheet for each class. For every class defined, the script identifies:

- **Data properties** — literal attributes such as voltages, names, and timestamps.
- **Object properties** — valid relationships to other classes.

Inheritance is automatically applied. This means that any property declared for a superclass (e.g., `Asset`) is also recognised for its subclasses (e.g., `LV_MSP`, `Protective_Device`), ensuring consistency across all entity types.

Step 2 — Generate `nodes.csv`. Each structured table representing a real-world entity—such as postcodes, addresses, smart meters, transformers, or charging stations—is processed to produce graph nodes.

What is done	Why it matters
A globally unique ID is created for each class instance(e.g., `simd_222bfe283b`).	Ensures each node has a stable, unambiguous identifier.
Any column name that matches a declared data property in the ontology is included as-is.	Guarantees that CSV property names match ontology labels.
Nodes are labelled with all applicable ontology labels (e.g., `Asset`, `LV`, `LV_MSP`).	Enables flexible querying at different levels of abstraction.

Table 4.5: Node generation logic from structured entity tables

Each row is written in a format ready for import, such as:

```
ID,:LABEL,asset_type,voltage_level,postcode,...
12088991,"Asset;Electric;LV;LV_MSP",LV_MSP,LV,ML118NT,...
```

Step 3 — Generate `edges.csv` Graph edges are created from two sources:

- **Inline foreign keys**: If a table includes a column referencing another entity (e.g., `smart_meter_id`), an edge is generated using the corresponding object property from the ontology (e.g., `HAS_SMART_METER`).
- **Dedicated link tables**: Datasets that represent connections—such as road links or electrical joints—are converted directly into edges. For undirected connections, two edges (one in each direction) are written to ensure consistent traversal behaviour in both Neo4j and Neptune.

Example edge:

```
:ID,:START_ID,:END_ID,:TYPE,...
e68047,DG14BQ,S01007616,BELONGS_TO,...
```

Duplicates are automatically removed, and any edge referencing a missing source or target node is excluded.

4.3.6 Graph Loading: from CSVs to graph databases

With `nodes.csv` and `edges.csv` prepared in OpenCypher-compatible format, the final step is to load this data into a graph database for querying and exploration. This project supports two targets: Neo4j and Amazon Neptune. Both pipelines rely on the same export format, ensuring compatibility and interoperability across systems.

4.3.6.1 Loading into Neo4j

A custom, performance-optimised Python loader was developed to efficiently ingest large volumes of graph data into Neo4j. The loader uses batched Cypher queries to insert both nodes and relationships, significantly reducing round-trips to the database and ensuring stable execution.

The process begins by importing nodes from `nodes.csv`. Each row represents a real-world entity—such as a postcode, electric asset, smart meter, or customer address. Each node is labelled according to its ontology-derived class and subclass structure (e.g., `Postcode;Region`, `ElectricAsset;LV;LV_MSP`). These labels support flexible queries at multiple levels of abstraction.

To ensure data integrity:

- Each node is assigned a globally unique business ID.
- All labels and properties are aligned with the domain ontology.
- A uniqueness constraint is applied (if not already present) to prevent duplicate insertions.
- Nodes are grouped by label combinations and loaded in large batches for efficiency.

Once all nodes are inserted, the loader processes the `edges.csv` file to create graph relationships. Relationships are grouped by type (e.g., `LOCATED_IN`, `HAS_METER`, `CONNECTED_TO`) and created using `MERGE` operations to maintain uniqueness. Any edges that reference missing nodes are skipped and logged for review.

The loader is fully configurable: batch sizes can be adjusted, node or edge stages can be selectively skipped, and optional validation reports can be generated. This provides a robust and scalable mechanism for loading Neo4j graphs aligned with the project's semantic model.

4.3.6.2 Loading into Amazon Neptune

Amazon Neptune supports bulk ingestion using CSV files in OpenCypher format, provided the files are stored in an S3 bucket [62]. Although the data export from this project is fully compatible, S3 access was not available during development. Nonetheless, preparing for production-scale deployment on Neptune is strongly recommended—particularly for large datasets such as full electric network models.

To validate the graph structure and data correctness in a Neptune-like environment, a custom loader was developed. This loader mirrors the Neo4j logic and connects to a Neptune instance using a Bolt-compatible interface. While this method is slower than S3-based ingestion, it allows for testing of:

- ID integrity and uniqueness
- Ontology-compliant labels and properties
- Relationship resolution between entities

This approach made it possible to load and test small amounts of data directly on the Neptune instance, without needing to use the S3-based bulk loader during development.

4.3.6.3 Loading process summary

Both loading strategies rely on the same ontology-aligned CSV files. Neo4j was used for local exploration and validation, while the custom Neptune loader confirmed compatibility and correctness. When scaling to production, Amazon Neptune's S3-based bulk loader remains the preferred option due to its ability to handle parallel, high-volume ingestion with minimal setup.

4.4 System deployment

In Section 4.2, the overall architecture of the graph-based system is introduced, outlining both the local and cloud-based deployment models. This section provides a more detailed explanation of how these deployments were actually configured and run in practice. The focus is on the specific tools and methods used to set up the environments, connect with the graph databases, and make the system operational for both development and production purposes.

4.4.1 Local deployment configuration

To streamline deployment and ensure consistency across devices, Neo4j was run using Docker. Docker is a containerisation platform that packages software and its dependencies into portable units called containers [63]. This makes it especially useful for deploying applications in collaborative or multi-device research settings, where consistent configuration is essential.

4.4.1.1 Why Use Docker for Neo4j?

Using Docker for Neo4j offers several advantages:

- **Consistency:** The same container image and settings can be used across different machines without version mismatches.
- **Portability:** Containers can be easily transferred or replicated across environments.
- **Isolation:** Neo4j runs in a self-contained environment, avoiding conflicts with other local services or libraries.
- **Ease of setup:** There is no need to install Neo4j or manage dependencies manually on the host system.

4.4.1.2 Setting up Neo4j with Docker

This project used Neo4j Community Edition version 2025.5.1, launched via the official Neo4j Docker image, following best practices from the Neo4j Operations Manual [64]. The following command was used to start the container:

```
docker run \
  -p 7474:7474 -p 7687:7687 \
  --name neo4j-ce \
  --env NEO4J_ACCEPT_LICENSE_AGREEMENT=eval \
  --env NEO4J_AUTH=neo4j/password \
  --env NEO4J_apoc_export_file_enabled=true \
  --env NEO4J_apoc_import_file_enabled=true \
  --env NEO4J_apoc_import_file_use__neo4j__config=true \
  --env NEO4J_PLUGINS='["apoc", "apoc-extended"]' \
```

```
10    neo4j:2025.05
```

This setup exposes both the browser interface (port 7474) and the Bolt protocol (port 7687), enabling connections via Neo4j Desktop, web browsers, or Python scripts. It also sets authentication credentials, activates APOC libraries, and accepts the required license agreement.

4.4.1.2.1 APOC and APOC Extended

APOC (Awesome Procedures On Cypher) is a standard extension for Neo4j that adds a wide range of useful procedures, including tools for graph traversal, data transformation, import/export, and date/time handling [65].

The `apoc-extended` module provides further functionality, including enhanced file I/O, advanced import/export capabilities, and additional utility procedures. Both were enabled during container setup.

4.4.1.2.2 Visual management via Docker Desktop

Although the container runs in the background, it can be monitored and managed through Docker Desktop, a graphical tool for working with containers. With Docker Desktop, users can:

- Start or stop the Neo4j container
- Monitor memory, CPU, and port usage
- View logs and inspect configuration
- Open a terminal directly into the container

An active Neo4j container will show port 7474 as open, confirming that the browser interface and Bolt connection are both available (see Figure 4.9).

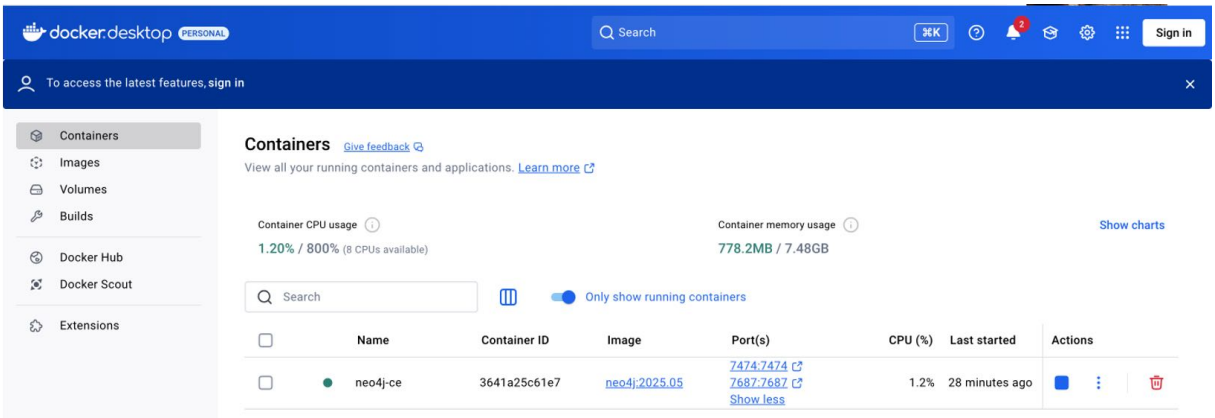


Figure 4.9: Neo4j container running in Docker Desktop.

4.4.1.2.3 Accessing the Neo4j terminal

For advanced operations, the Neo4j shell can be accessed from within the Docker container. This enables direct interaction with the system database for administrative tasks such as defining constraints, creating indexes, or managing user accounts. The shell can be launched using the following command from inside the container:

```
cypher-shell -u neo4j -p password --database system
```

This command connects to the system database, from where administrative commands can be run (see Figure 4.10).

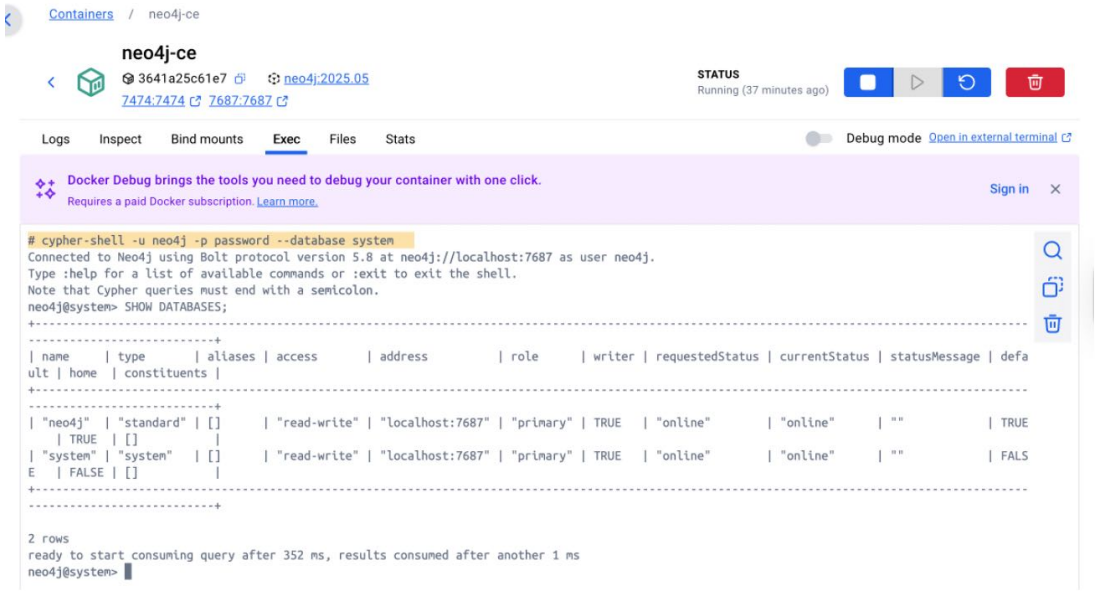


Figure 4.10: Accessing the Neo4j system database through the container terminal using `cypher-shell`. The command enables administrative operations such as inspecting available databases, as shown by the output of `SHOW DATABASES`.

4.4.1.3 Connecting the Neo4j container to Neo4j Desktop

Once the Neo4j container is running locally, it can be connected to Neo4j Desktop—a graphical development environment that supports interaction with both local and remote graph databases [49]. This tool enables users to run queries, inspect schema elements, and explore graph data in a visual and intuitive manner.

To establish a connection, a new **Remote connection** must be created in Neo4j Desktop using the following configuration (see Figures 4.11 and 4.12):

- **Protocol:** `bolt://`
- **Connection URL:** `localhost:7687`

- **Database user:** `neo4j`
- **Password:** the one specified during Docker setup (e.g., `password`)
- **Connection name:** a label such as `TFM` for easy identification

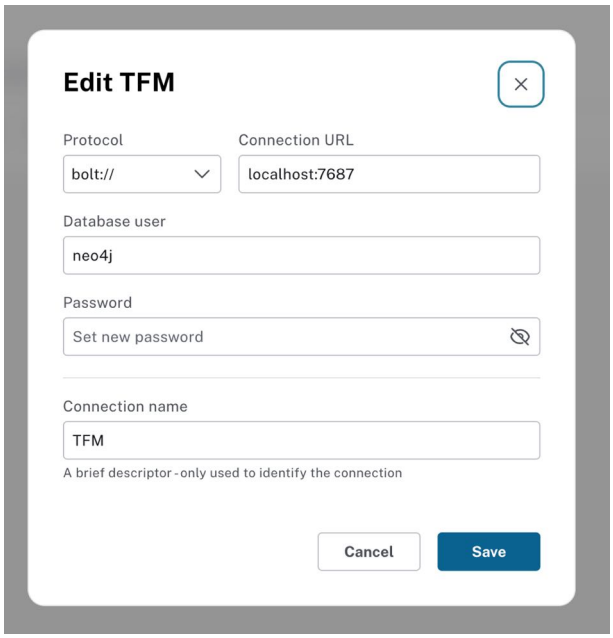


Figure 4.11: Connection setup for the local Neo4j container via Bolt protocol.

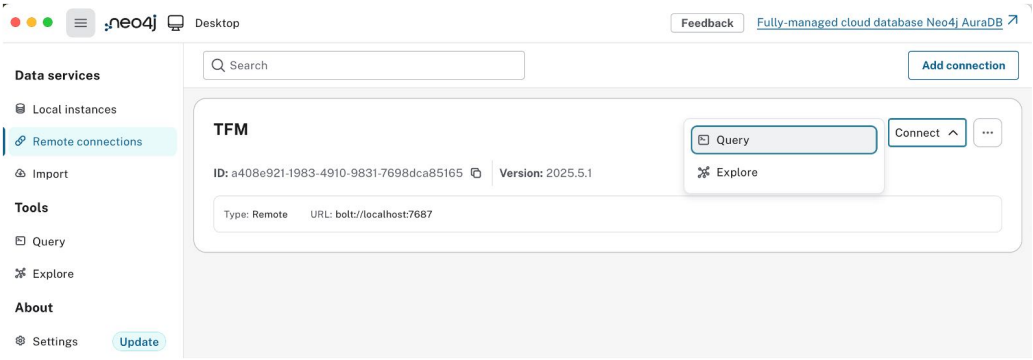


Figure 4.12: Remote connection to the Docker-hosted Neo4j instance registered in Neo4j Desktop.

Once connected, the user interface provides several key views (see Figure 4.13):

- The **Query panel** supports Cypher queries, returning results in graph, table, or raw JSON format.
- The **Database information panel** shows all loaded node labels, relationship types, and a live count of graph elements.
- The **Schema and Indexes views** offer insight into the structural organisation of the data.

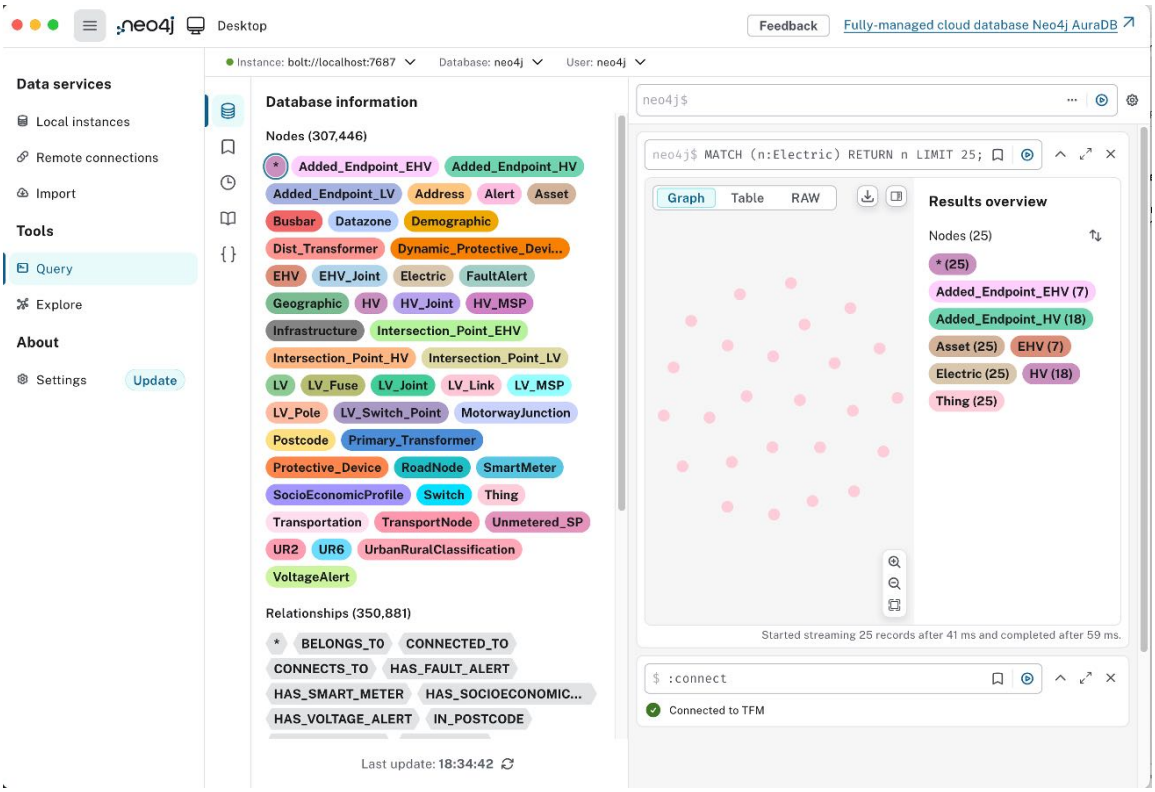


Figure 4.13: Querying and inspecting the local Neo4j instance using Neo4j Desktop.

While Neo4j Desktop includes an `Explore` tab, it is important to note that this is powered by Neo4j Bloom, a premium visualisation tool available only in the Enterprise Edition of Neo4j [66]. As this project uses the Community Edition, Bloom features are not accessible. Nevertheless, the available query and graph views in Neo4j Desktop provide sufficient functionality for interactive development and graph inspection.

This desktop-based environment was key for enabling fast feedback loops during development, especially when validating graph schema design, checking ontology mappings, and running early-stage analytics queries.

4.4.2 Cloud deployment: Amazon Neptune (VPC)

The production deployment of the graph system was carried out using **Amazon Neptune**, a fully managed graph database service hosted within SPEN's private **Virtual Private Cloud (VPC)** on AWS. This setup ensures that the infrastructure remains secure and compliant with SPEN's internal IT policies.

4.4.2.1 Deployment inside SPEN's VPC

The Neptune instance was deployed as part of SPEN's internal VPC, which restricts external access and routes all traffic through controlled, encrypted channels. This configuration provides:

- **High security:** Only authenticated devices inside the VPC—or connected through approved VPN tunnels—can access the database.
- **Isolation:** The instance is not exposed to the public internet, minimising attack surface and safeguarding sensitive data.

4.4.2.1.1 Dual endpoints for read and write operations

Amazon Neptune offers separate network endpoints for reading and writing operations:

- `<cluster-name>.cluster-ro.<region>.neptune.amazonaws.com` — Read-only endpoint, intended for executing queries without modifying data.
- `<cluster-name>.cluster.<region>.neptune.amazonaws.com` — Writer endpoint, used for updates, imports, and write operations.

Both endpoints are accessible over HTTPS on port `8182`. Cypher queries can be executed via POST requests to the `/openCypher` path. For example:

```
POST https://<read-endpoint>:8182/openCypher
Content-Type: application/json

{
  "query": "MATCH(m:SmartMeter) RETURN m LIMIT 10"
}
```

This flexibility allows different teams (e.g., those using Gremlin or Cypher) to share the same infrastructure without maintaining separate databases.

4.4.2.1.2 Secure Access via WireGuard VPN

Access to Neptune from outside the VPC requires a secure connection via a **WireGuard VPN tunnel**. WireGuard is a modern, lightweight VPN protocol that provides encrypted peer-to-peer communication.

Once the VPN is established, the user's machine operates as if it were part of SPEN's private network—enabling access to Neptune and other internal AWS resources.

The VPN profile is configured with:

- A unique public/private key pair.
- A static internal IP address within the VPC.
- DNS settings and port mappings for correct routing.

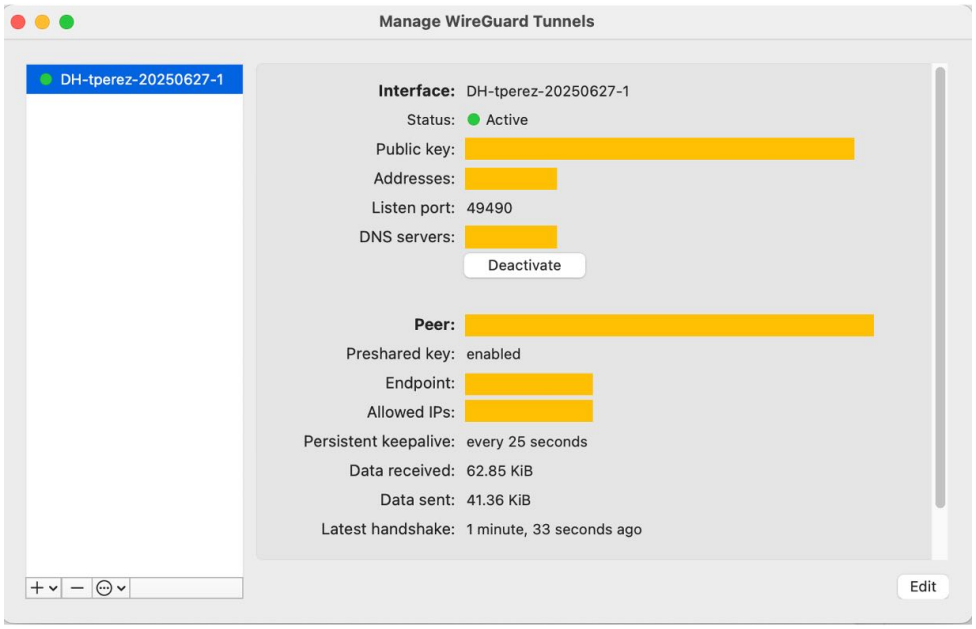


Figure 4.14: Active WireGuard VPN tunnel used to securely access Neptune from outside the VPC.

After establishing the tunnel, any script or application on the local machine can securely query or update the Neptune database.

4.4.2.1.3 Checking instance status

To verify that the Neptune instance is running and reachable, the following status endpoint can be accessed via a browser:

```
https://<cluster-endpoint>:8182/status
```

If the service is healthy, this endpoint returns a JSON response:

```
{
  "status": "healthy"
}
```

This check confirms that both the VPN and Neptune instance are operational before proceeding with data ingestion or analysis.

4.4.2.2 Setting up and connecting to neptune graph explorer

To provide a user-friendly, visual interface for interacting with the Amazon Neptune graph database, the **Neptune Graph Explorer** was deployed locally using Docker. This tool enables users to inspect nodes, run queries, and explore the property graph similarly to Neo4j Desktop, but adapted for the Neptune environment.

The Graph Explorer is especially useful in collaborative scenarios where analysts, developers, or

stakeholders need an interactive front-end to browse the graph or test OpenCypher queries. While the core Neptune engine is cloud-hosted and accessed through API endpoints, the Graph Explorer provides a complementary graphical interface that runs locally and connects securely to the cloud backend.

4.4.2.2.1 *Deployment via docker*

The tool is available as a Docker image maintained by AWS [67]. To deploy it, the official instructions from the Graph Explorer GitHub repository were followed [52]. The container was launched using Docker Desktop, where it appeared as a running service named `graph-explorer`.

Once active, the interface is accessible at:

`http://localhost/explorer/`

Figure 4.15 shows the container running inside Docker Desktop.

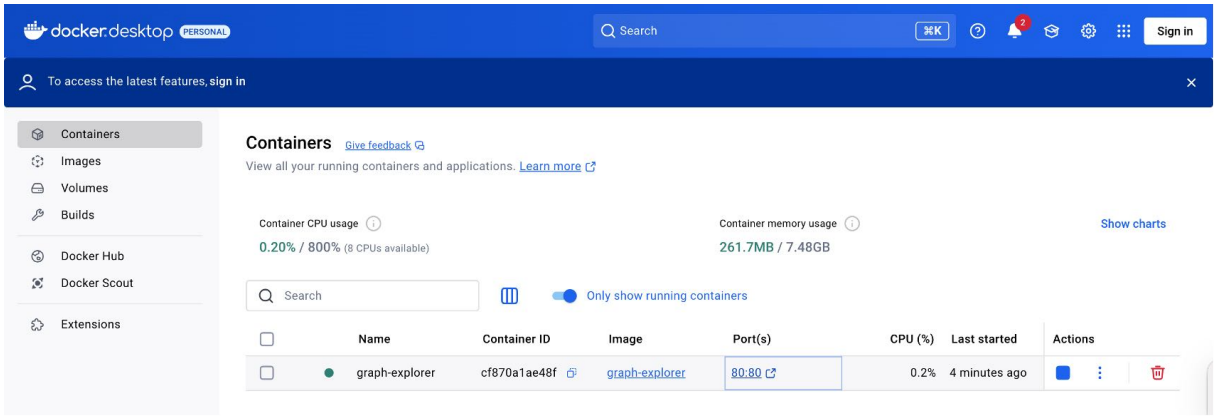


Figure 4.15: Graph Explorer running locally in Docker.

4.4.2.2.2 *Configuring the connection*

To connect the Graph Explorer to the Neptune instance, a connection must be configured via the application's interface. This includes specifying:

- A name for the connection.
- The graph type (in this case, `OpenCypher - PG (Property Graph)`).
- The URL of the Neptune read endpoint (typically the `cluster-ro` endpoint).
- Proxy server usage (`http://localhost`) to allow traffic routing through the local VPN.

Figure 4.16 shows an example configuration.

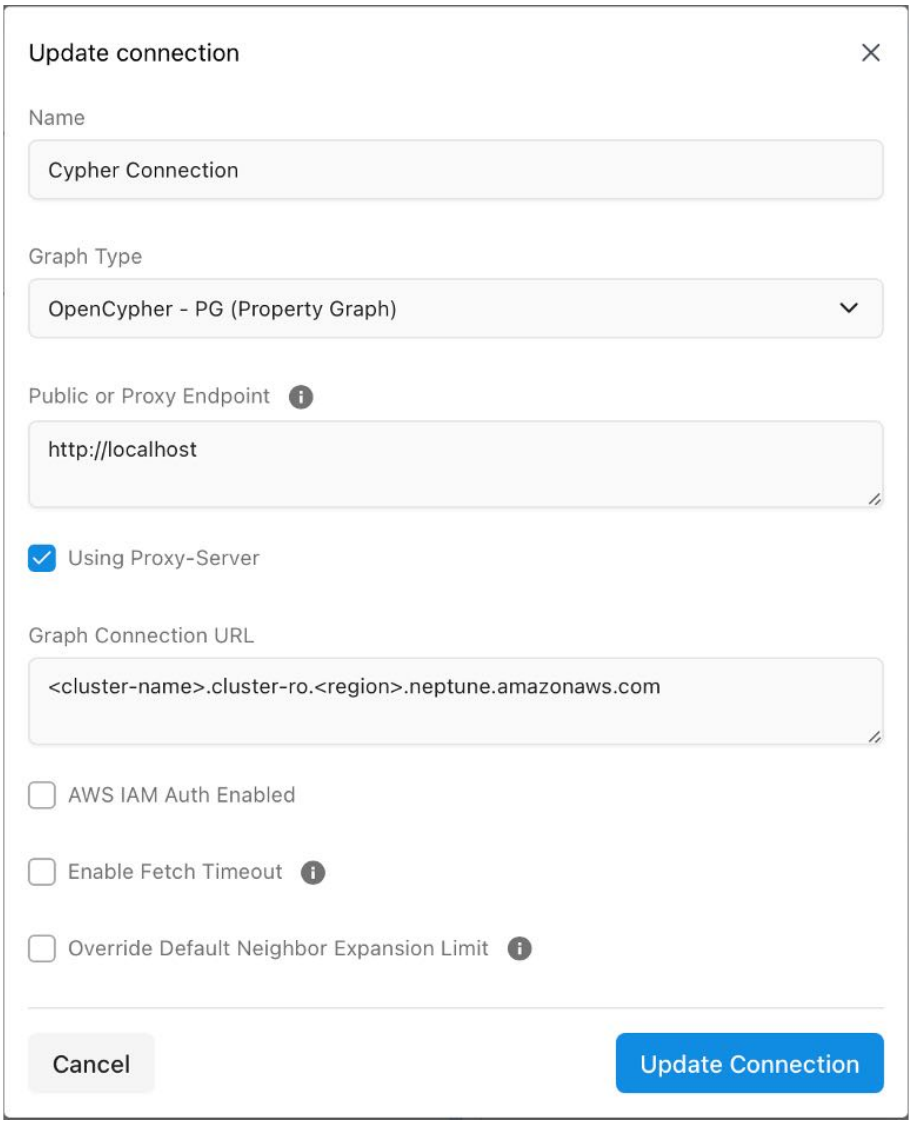


Figure 4.16: Configuration screen for connecting Graph Explorer to Neptune.

4.4.2.2.3 Graph explorer interface

After a successful connection, the interface displays a dashboard summarising the graph schema—listing node labels, relationship types, and element counts (see Figure 4.17). The interface displays thousands of nodes and relationships, all routed through the VPN tunnel.

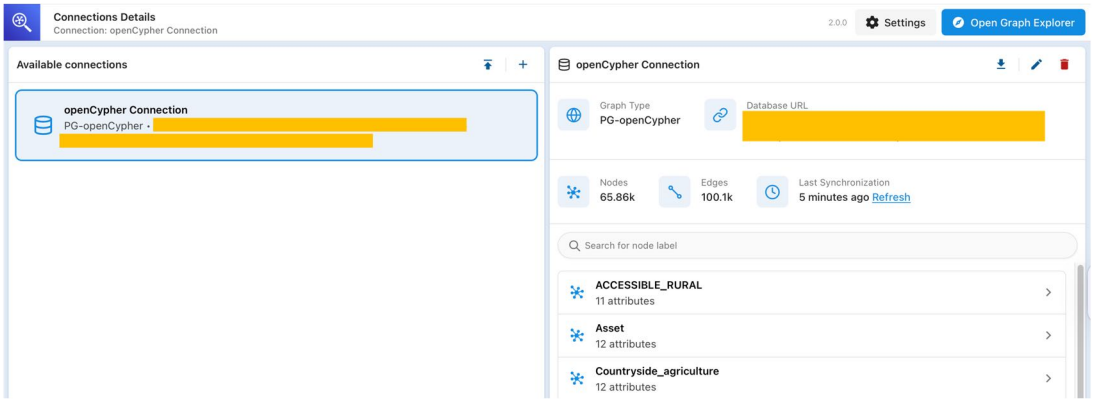


Figure 4.17: Graph Explorer UI showing a live connection to a Neptune graph.

Furthermore, as shown in Figure, 4.18 the Graph Explorer interface includes several key components:

- **Graph View** — A central canvas with layout options (e.g., force-directed), used to visualise clusters of nodes and their connections.
- **Query Panel** — Located on the right-hand side, where users can write and execute OpenCypher queries (e.g., `MATCH (n) RETURN n LIMIT 100;`).
- **Table View** — Positioned at the bottom, displaying returned nodes along with their IDs, labels, attributes, and neighbour counts.
- **Side Panel** — Provides options to filter nodes and relationships by label or type.
- **Interaction Tools** — Allow users to add or remove nodes from the view and export query results.

Queries can be run interactively, with results shown both as visual networks and raw data tables. This setup effectively mirrors the experience of Neo4j Desktop, enabling visual navigation and debugging of large-scale knowledge graphs in Neptune.

For further information, consult the official AWS documentation on Neptune Graph Explorer [67]

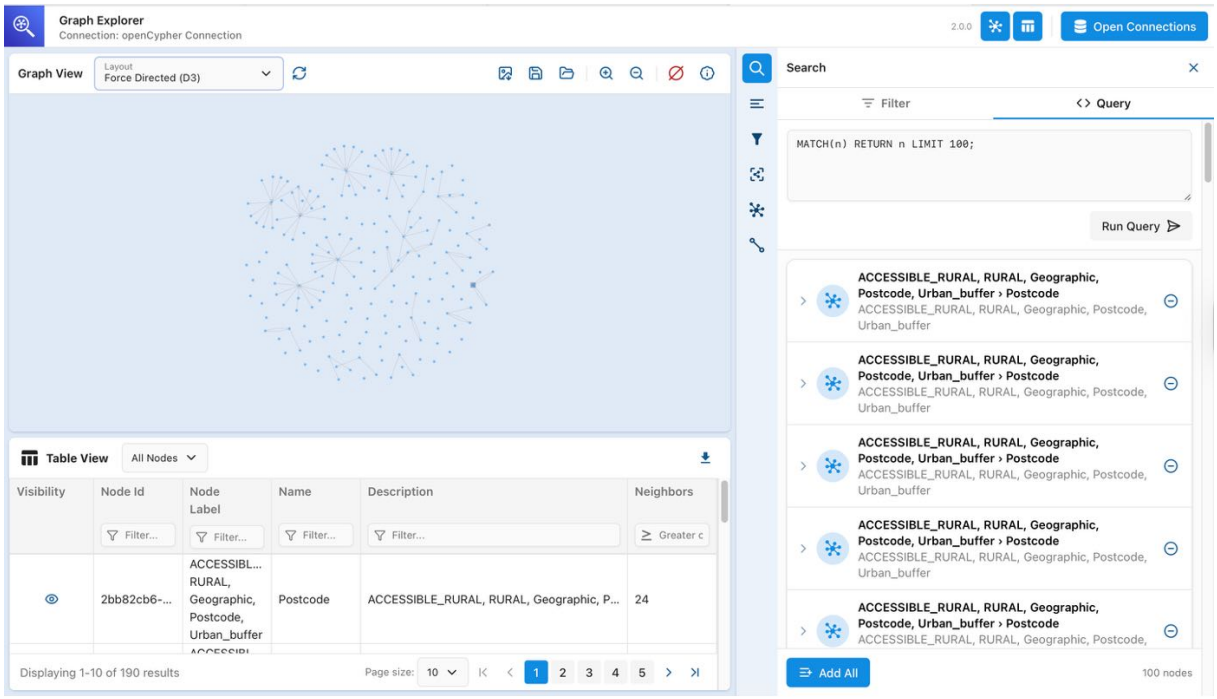


Figure 4.18: Graph Explorer UI showing the results of a query to the Neptune graph..

4.5 Querying the graph

4.5.1 Query Languages Used

Two primary query languages were available in this project's graph deployments: **Cypher** and **Gremlin**. Both can be used to traverse and manipulate property graphs, but their syntax and underlying philosophy differ significantly.

4.5.1.1 Cypher

A declarative graph query language originally developed for Neo4j and now adopted by other platforms such as Amazon Neptune. Cypher focuses on pattern matching, using an ASCII-art style syntax to describe relationships between nodes. For example:

```
MATCH (m:LV_MSP {id: '12026491'})
      -[:SERVES_ADDRESS]->(a:Address)
      -[:IN_POSTCODE]->(:Postcode)
      -[:BELONGS_TO]->(:Datazone)
      -[:HAS_SOCIOECONOMICPROFILE]->(p:SocioEconomicProfile)
RETURN m.asset_id AS lv_msp_id,
       a.address  AS customer_address,
       p.simd_decile AS simd_decile
```

Query 4.1: Retrieve customers (addresses) served by a given LV_MSP and their associated SIMD decile with Cypher

This style visually mirrors the underlying graph structure, making it intuitive for domain experts who are not necessarily graph database specialists. Its declarative approach allows the user to specify *what* pattern to retrieve without prescribing *how* to traverse the graph.

4.5.1.2 Gremlin

A traversal-based query language defined by Apache TinkerPop. Gremlin uses an imperative style, expressing queries as a sequence of function calls or steps. For example:

```
g.V().hasLabel('LV_MSP').has('id', '12026491').
  as('m').
  out('SERVES_ADDRESS').hasLabel('Address').as('a').
  out('IN_POSTCODE').
  out('BELONGS_TO').hasLabel('Datazone').
  out('HAS_SOCIOECONOMICPROFILE').hasLabel('SocioEconomicProfile').
     as('p').
  select('m','a','p').
    by('asset_id').
    by('address').
    by('simd_decile')
```

Query 4.2: Retrieve customers (addresses) served by a given LV_MSP and their associated SIMD decile with Gremlin

This offers fine-grained control over the traversal process but can be more verbose and less immediately interpretable for stakeholders unfamiliar with programming constructs.

4.5.1.3 Choice of querying language

For this project, Cypher was preferred as the primary query language for the following reasons:

- **Interpretability** — Cypher queries closely resemble the conceptual model of the graph, making them easier to read and explain to non-technical collaborators such as planners, engineers, and policymakers.
- **Reduced verbosity** — Most analytical queries could be expressed in fewer lines, which is advantageous for quick prototyping and debugging.
- **Cross-platform availability** — Cypher is the default language in Neo4j and supported in Amazon Neptune's OpenCypher endpoint, ensuring consistent syntax across both local and cloud deployments.

Gremlin remained available for cases requiring advanced traversal optimisations or compatibility with other TinkerPop-compliant tools, but it was not the main focus during development.

4.6 Graph exploration

This section presents three examples that illustrate the structure of the knowledge graph and some of its potential uses, showing how technical, geographic, and socio-economic data can be combined for analysis and visualisation.

4.6.1 Example 1: Exploring a distribution transformer and its subgraph

To illustrate the potential of the graph model, a specific distribution transformer (`id = 9078228`) was selected and its associated subgraph retrieved. This view does not only include the physical electrical infrastructure but also integrates other contextual layers such as smart meters, voltage alerts, customer addresses, datazones, and associated socioeconomic indicators like SIMD scores.

The Cypher query shown below starts from a distribution transformer node and expands through its connected network up to a maximum depth of 15 hops, avoiding cycles and including all relationships relevant to the extended data model. The procedure `apoc.path.subgraphAll` from the APOC library is used to return both the nodes and relationships within this enriched subgraph:

```
MATCH (dt:Dist_Transformer {id: "9078228"})
WHERE dt.location IS NOT NULL

CALL apoc.path.subgraphAll(dt, {
  maxLevel:15,
  bfs: false,
  filterStartNode: true
}) YIELD nodes AS electric_nodes, relationships AS
   electric_relationships

RETURN
  dt.id AS transformer_id,
  dt,
  electric_nodes,
  electric_relationships
```

Query 4.3: Find a certain Distribution Transforer and extract its subgraph

Figure 4.19 shows, on the left, the traditional geospatial network model and, on the right, its equivalent in the graph model. The central node of the subgraph (in red) represents the transformer, from which the physical infrastructure and additional linked datasets radiate. In this section, progressive “zoom-ins” are performed on different parts of the subgraph to examine how these diverse layers—ranging from engineering assets to customer-level socioeconomic data—are connected in a unified and queryable structure.

4.6.1.1 Zooming into the distribution transformer

The first level of zoom focuses on the distribution transformer node at the centre of the subgraph. In the graph visualisation (Figure 4.20), the transformer is shown in red and is directly connected to two main categories of physical assets: Low Voltage (LV) nodes (green) and High Voltage (HV) nodes (purple).

This arrangement makes it possible to inspect both downstream and upstream connectivity from the transformer in a single view. LV nodes represent the low-voltage side of the network that feeds customer connections, while HV nodes represent the high-voltage side supplying the transformer from upstream substations or feeders.

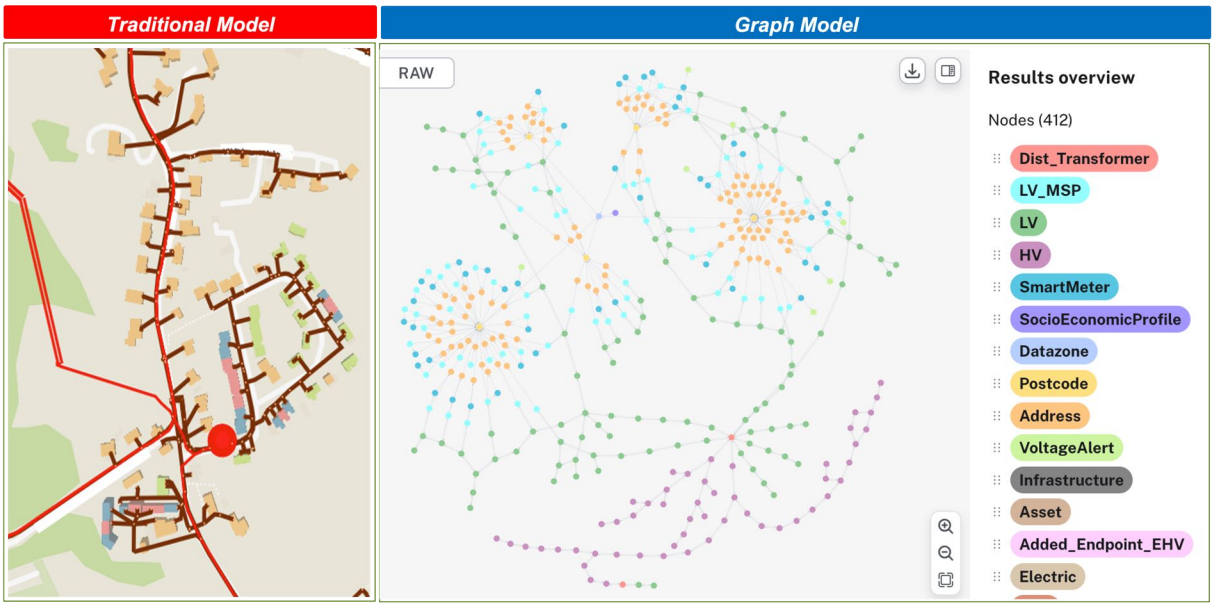


Figure 4.19: Distribution transformer subgraph. Comparison between the traditional geospatial view (left) and the graph-based visualisation (right) for the transformer. The graph model combines physical network assets with other linked datasets, enabling exploration from the transformer down to customer-level indicators.

In the traditional geospatial model (left), these connections are shown as lines and points on a map, offering geographic context but limited interactivity for integrating additional data. In the graph model (right), the same connections are represented as relationships between nodes, enabling immediate expansion into related layers such as metering data, voltage events, and socioeconomic indicators.

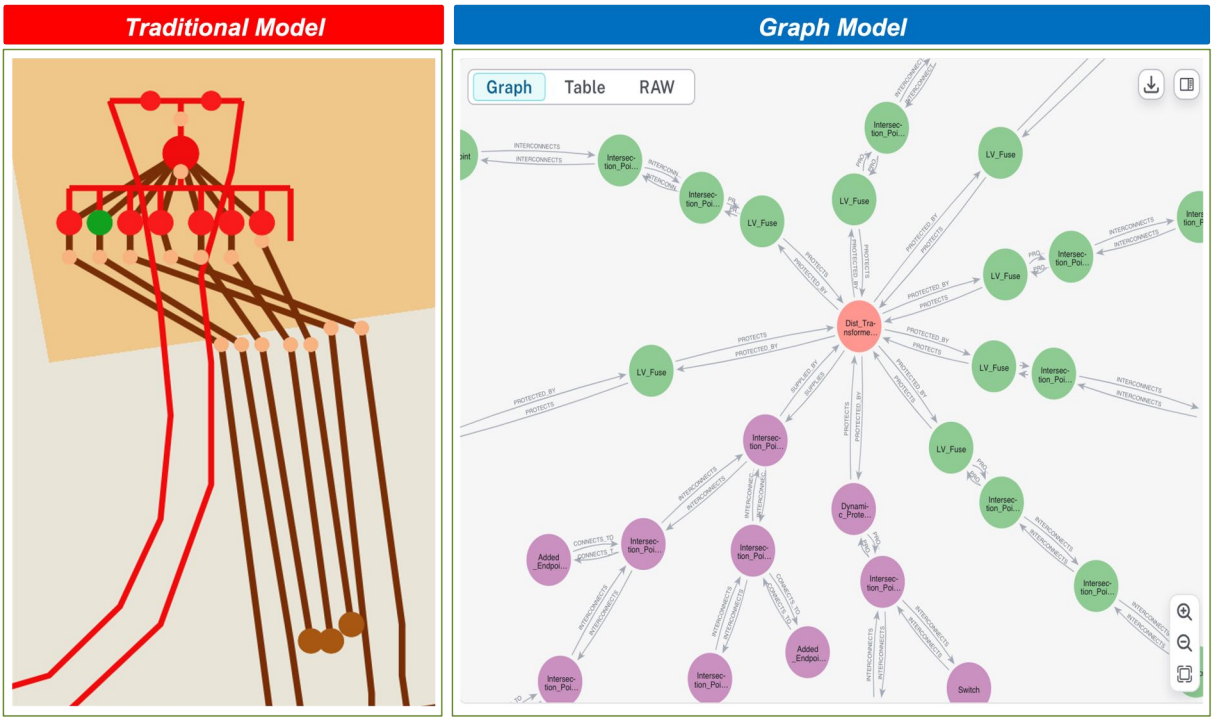


Figure 4.20: Zoom into the distribution transformer. The central red node represents the transformer, connected to LV nodes (green) and HV nodes (purple), showing the bidirectional connectivity within the electrical network.

4.6.1.2 *Zoom into LV joints and address-level granularity*

A closer inspection of the network highlights the role of Low Voltage (LV) joints in the local topology. In Figure 4.21, the green nodes represent `LV_Joint` elements branching out to several `LV_MSP` nodes (cyan).

In traditional schematics, this stage of the network is shown as part of the physical layout but without revealing the address-level detail. The graph model, in contrast, makes explicit which *specific* addresses are served by each joint, which of those addresses have smart meters, and how they connect back to postcode units. This enables downstream analysis—such as socioeconomic profiling, vulnerability scoring, or targeted operational planning—from a single, unified visualisation.

Figure 4.22 takes the zoom further, focusing on two `LV_MSP` nodes. Here, the relationships between each MSP, its served addresses, and the associated smart meters are clearly visible. Each address is also linked to its postcode node. These multi-layer links (asset $\rightarrow$ address $\rightarrow$ smart meter / postcode) are unmanageable to extract from traditional schematics but are inherent in the graph representation.

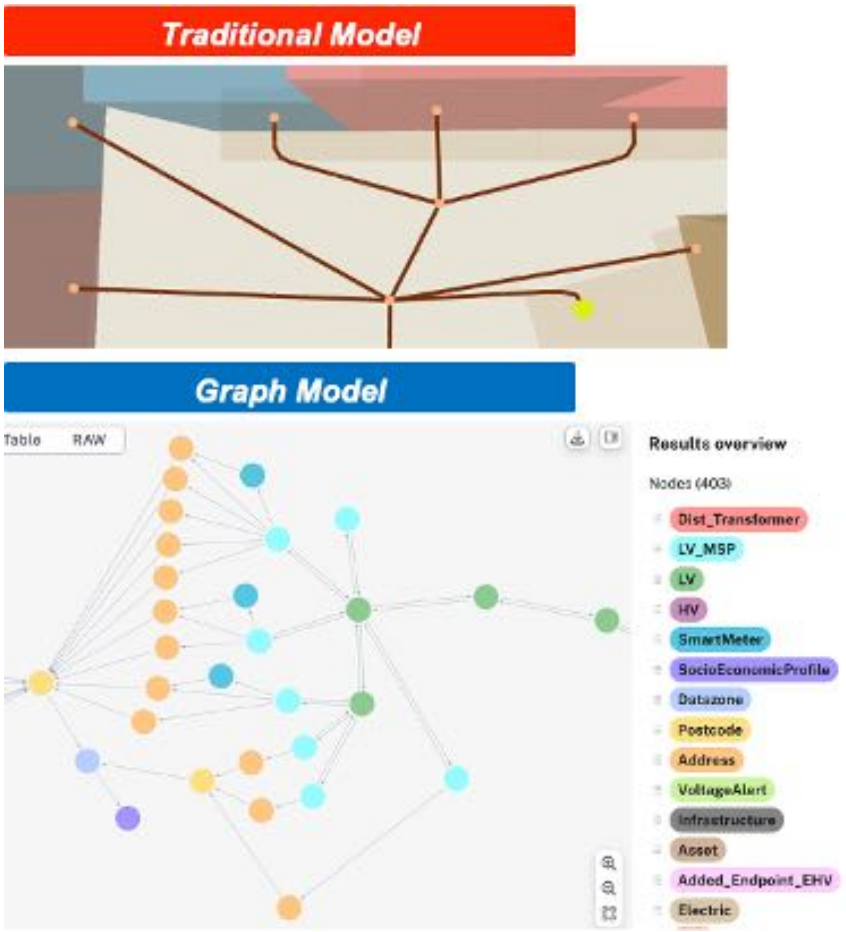


Figure 4.21: LV joints (green) branching to multiple `LV_MSP` nodes (cyan), with downstream address and smart-meter context. The traditional view (top) shows the layout; the graph view (bottom) adds address-level semantics.

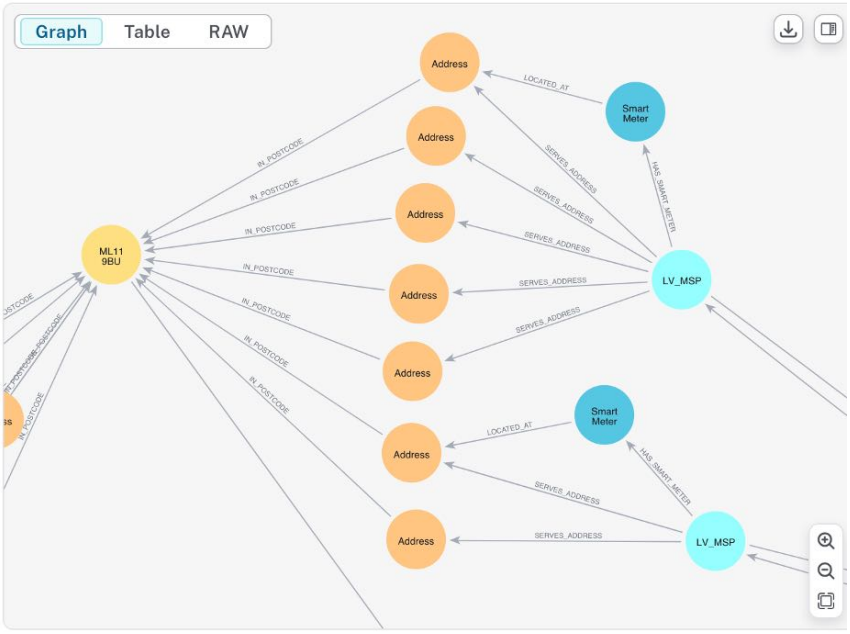


Figure 4.22: Detail of two `LV_MSP` nodes showing explicit links to served addresses, smart meters, and their postcodes.

4.6.1.3 Smart meters linked to sensor-originated alerts

The graph also captures sensor-level events. Figure 4.23 shows a `SmartMeter` node connected to a `VoltageAlert` via the `HAS_VOLTAGE_ALERT` relationship. The alert node stores the time-stamped, half-hourly voltage series (`voltage_hh_data`), as well as derived features such as `maximum_demand` and `maximum_voltage`. In the same way, meters can link to `FaultAlert` events when available. This design keeps raw measurements and their summaries close to the asset that generated them, enabling queries that jump from field events to affected customers, postcodes, and transformers without intermediate joins.

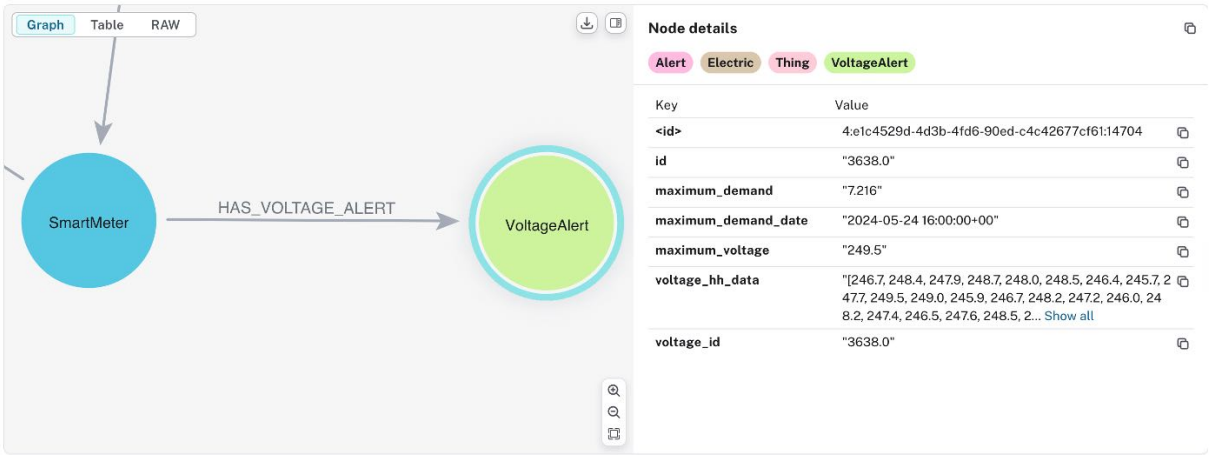


Figure 4.23: A `SmartMeter` related to a `VoltageAlert` (`HAS_VOLTAGE_ALERT`). The alert node retains the half-hourly voltage trace and key statistics, enabling end-to-end tracing from sensor events to network and customer context.

4.6.1.4 Linking network assets to postcodes, datazones, and socioeconomic context

The graph model links the electrical network to its geographic and demographic context. Each `Address` connects to a `Postcode` via `IN_POSTCODE`, and each `Postcode` connects to a `Datazone` via `BELONGS_TO`. Datazones store attributes such as `population`, the urban–rural class (UR2 and UR6), and are also linked to a `SocioEconomicProfile` node (`HAS_SOCIOECONOMICPROFILE`) containing indicators like SIMD deciles.

This integration enables the network to be analysed in its social context. It becomes possible to quantify how many `LV_MSPs`—and the households they serve—fall within high-deprivation deciles, compare outage exposure across rural and urban areas, or prioritise reinforcement where technical criticality overlaps with social vulnerability. In practice, the graph allows to easily pivot from a transformer to the postcodes it serves, to their datazones and SIMD ranks—without changing tools or performing manual joins.

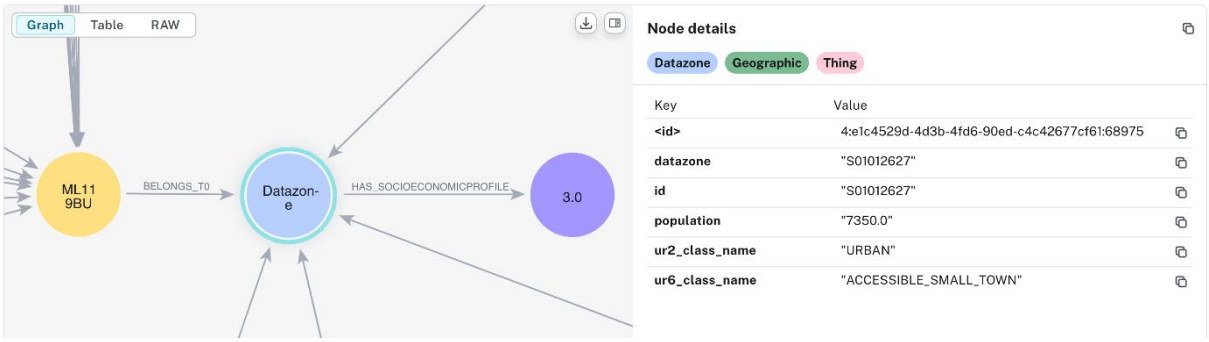


Figure 4.24: Geographic and socioeconomic linkage. A `Postcode` (left) `BELONGS_TO` a `Datazone`, which `HAS_SOCIOECONOMIC_PROFILE` (purple). The datazone node carries population, UR2/UR6 (urban/rural) class, enabling analyses that relate network characteristics to rurality and social deprivation.

4.6.2 Example 2 — Identifying rural customers close to an EV charger

The purpose of this analysis is to identify rural electricity customers who are located within a short driving distance, measured along the road network, from an electric vehicle (EV) charging station.

By combining road network topology, geographic information, and EV infrastructure data, the analysis highlights rural areas where residents already have relatively good access to charging facilities. This insight can support planning decisions, such as focusing outreach campaigns to encourage EV adoption in these areas or reallocating infrastructure investment towards regions with poorer accessibility.

4.6.2.1 Parameter configuration

The query is structured so that all key thresholds and filters are defined as parameters at the beginning, making them easy to adjust without modifying the rest of the logic. The main parameters are:

- The maximum road-network distance to a charger (`maxKm`).
- The number of nearby road nodes to consider for each `LV_MSP` (low-voltage feeder).
- The number of candidate road nodes per charger postcode.
- An optional social context filter, for example limiting results to rural areas (`UR2 = RURAL`).

This design makes it straightforward for analysts and planners to adjust the definition of "nearby", apply different geographic or social focus criteria, or widen/narrow the scope of the analysis, without altering the core query.

```
:param ur2Class => 'RURAL';
:param maxKm => 5;
:param kOrigins => 3;
```

```
:param kTargetsPerPC => 3;
```

Query 4.4: Setting example 2 query parameters

4.6.2.2 Query approach

The query identifies rural customers who have at least one EV charger within the specified maximum road distance. The process follows these steps:

1. For each `LV_MSP` (representing a set of customer connections), find the nearest road network node to use as the starting point.
2. For every EV charger, identify its postcode and find road nodes within that postcode that are closest to its centroid.
3. Calculate road-network distances from each `LV_MSP` starting node to each candidate charger road node, using the recorded `road_length` values.
4. Retain only the matches where the shortest route is within the `maxKm` threshold.
5. Create a virtual `NEAR_BY_ROAD` relationship between the customer's `Address` and the `ChargingStation`, storing the calculated distance in kilometres.

This approach measures accessibility based on the actual road network rather than straight-line distances, providing a more realistic assessment of charger proximity for rural customers.

```
// 1) LV and anchor candidates (k origins)
MATCH (lv:LV_MSP)
WHERE lv.location IS NOT NULL

CALL (lv){
  WITH lv
  MATCH (c:RoadNode)
  WHERE c.location IS NOT NULL
  WITH lv, c
  ORDER BY point.distance(c.location, lv.location) ASC
  LIMIT coalesce($kOrigins, 5)                    //     multiple
     origins
  // calculate degree (number of neighbors) to skip cul-de-sacs
  OPTIONAL MATCH (c)-[:CONNECTED_TO]-(:RoadNode)
  WITH lv, c, count(*) AS deg
  RETURN collect({node: c, deg: deg}) AS originCandidates
}

WITH lv, originCandidates

// 2) Prefer non-leaves; if none, use cul-de-sacs anyway
```

```
WITH lv,
     [oc IN originCandidates WHERE oc.deg > 1 | oc.node] AS junctions,
     [oc IN originCandidates WHERE oc.deg = 1 | oc.node] AS leaves
WITH lv, CASE WHEN size(junctions) > 0 THEN junctions ELSE leaves END AS origins

// 3) Social context (if needed for visualization)
OPTIONAL MATCH (lv)-[:SERVES_ADDRESS]->(addr:Address)
OPTIONAL MATCH (addr)-[:IN_POSTCODE]->(pc:Postcode)
OPTIONAL MATCH (pc)-[:BELONGS_TO]->(dz:Datazone)
OPTIONAL MATCH (dz)-[:HAS_SOCIOECONOMICPROFILE]->(prof:SocioEconomicProfile)
WITH lv, origins, addr, pc, dz, prof
WHERE dz.ur2_class_name = $ur2Class

// 4) Destination candidates by charger postcode (multiple)
CALL (lv){
  WITH lv
  MATCH (cs:ChargingStation)
  WHERE cs.postcode IS NOT NULL
  OPTIONAL MATCH (pc2:Postcode {postcode: cs.postcode})
  // take several RNs from the postcode, prioritizing closeness to the centroid if it exists,
  // otherwise, any with a location
  MATCH (rnPC:RoadNode)
  WHERE rnPC.postcode = cs.postcode AND rnPC.location IS NOT NULL
  WITH lv, cs, pc2, rnPC
  ORDER BY CASE WHEN pc2.centroid IS NULL THEN 1 ELSE 0 END,
           CASE WHEN pc2.centroid IS NULL THEN 0 ELSE point.distance(rnPC.location, pc2.centroid) END ASC
  LIMIT coalesce($kTargetsPerPC, 3)                //     multiple destinations per postcode
  RETURN cs, collect(rnPC) AS targets
}
WITH lv, origins, addr, pc, dz, prof, cs, targets

// 5) All origin destination combinations and pick the best cost
UNWIND origins AS origin
UNWIND targets AS t
CALL apoc.algo.dijkstra(origin, t, 'CONNECTED_TO', 'road_length', 9e12) YIELD weight
WITH lv, addr, pc, dz, prof, cs, origin, t, weight/1000.0 AS km
// (optional) penalize leaf origins to discourage cul-de-sacs:
WITH lv, addr, pc, dz, prof, cs, km
WHERE km <= $maxKm
WITH lv, addr, pc, dz, prof, cs, min(km) AS bestKm   // best route
```

```
    per charger
// keep ALL cs that match (no LIMIT 1)
WITH lv, addr, pc, dz, prof, collect({cs: cs, km: bestKm}) AS
    nearChargers

// 6) (Optional) virtual relationships for visualization
UNWIND nearChargers AS nc
WITH lv, addr, pc, dz, prof, nc.cs AS cs, nc.km AS km
CALL apoc.create.vRelationship(addr, 'NEAR_BY_ROAD', {km: km}, cs)
    YIELD rel AS r_addr_cs

// 7) Re-match context edges so they render
OPTIONAL MATCH (lv)-[r_lv_addr:SERVES_ADDRESS]->(addr)
OPTIONAL MATCH (addr)-[r_addr_pc:IN_POSTCODE]->(pc)
OPTIONAL MATCH (pc)-[r_pc_dz:BELONGS_TO]->(dz)
OPTIONAL MATCH (dz)-[r_dz_prof:HAS_SOCIOECONOMICPROFILE]->(prof)

// 8) Return (no DISTINCT hammer unless needed)
RETURN
  lv, addr, pc, dz, prof, cs,
  r_lv_addr, r_addr_pc, r_pc_dz, r_dz_prof, , r_addr_cs
ORDER BY addr.id;
```

Query 4.5: Nearest charging stations for rural low-voltage customers

4.6.2.3 Results and interpretation

The query returned a total of 4,344 nodes, including 2,330 `Address` nodes linked to three `ChargingStation` nodes via `NEAR_BY_ROAD` relationships. The visualisations in Figures 4.25 and 4.26 reveal two distinct clusters:

- A large cluster (top left) containing the majority of addresses but served by only **one** EV charger.
- A smaller cluster (bottom) served by **two** EV chargers, offering relatively greater charger availability per customer.

This distribution suggests that the top-left cluster could be a candidate for EV infrastructure reinforcement, as a single charger serves a large rural population. In contrast, the bottom cluster already benefits from higher charger density.

By linking these proximity results with additional layers—such as socioeconomic indicators, vehicle ownership patterns, or projected EV uptake—it becomes possible to prioritise investment in areas where access to charging facilities is most constrained, supporting evidence-based network reinforcement and transport–energy planning.

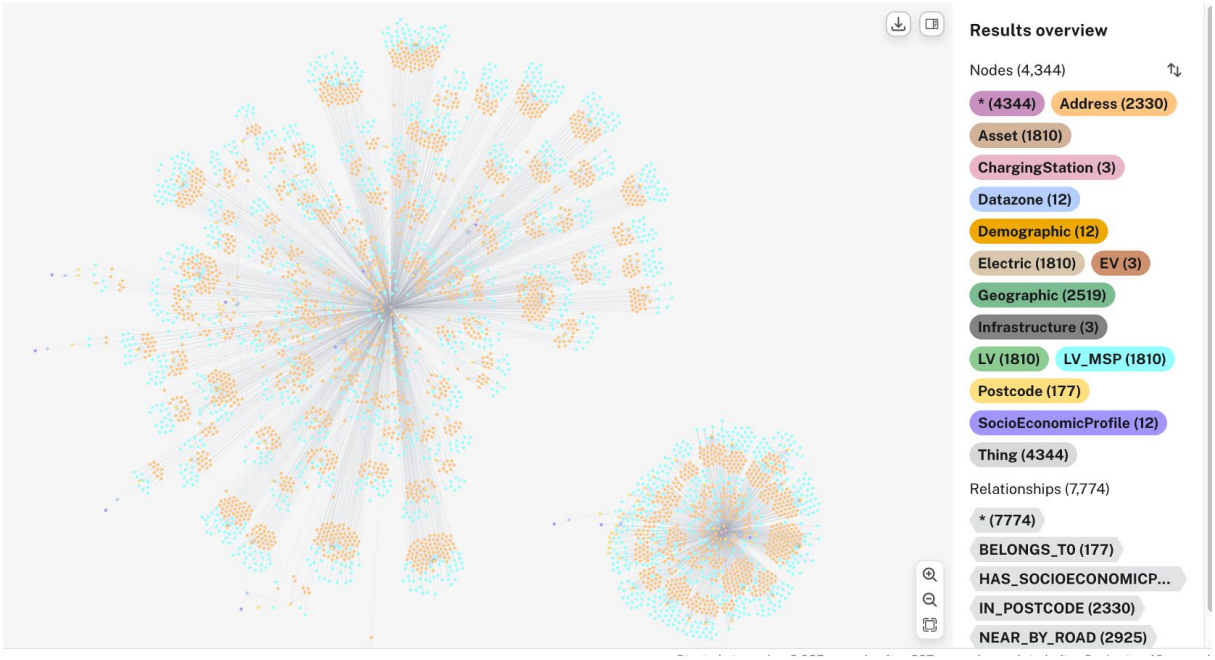


Figure 4.25: Overview of rural customers and their proximity to EV chargers. Orange nodes are `Address` entities, cyan nodes are `LV_MSP`s, and pink nodes are `ChargingStations`, connected via `NEAR_BY_ROAD` relationships. Two main clusters are visible: a large one (top left) served by a single charger, and a smaller one (bottom) served by two chargers.

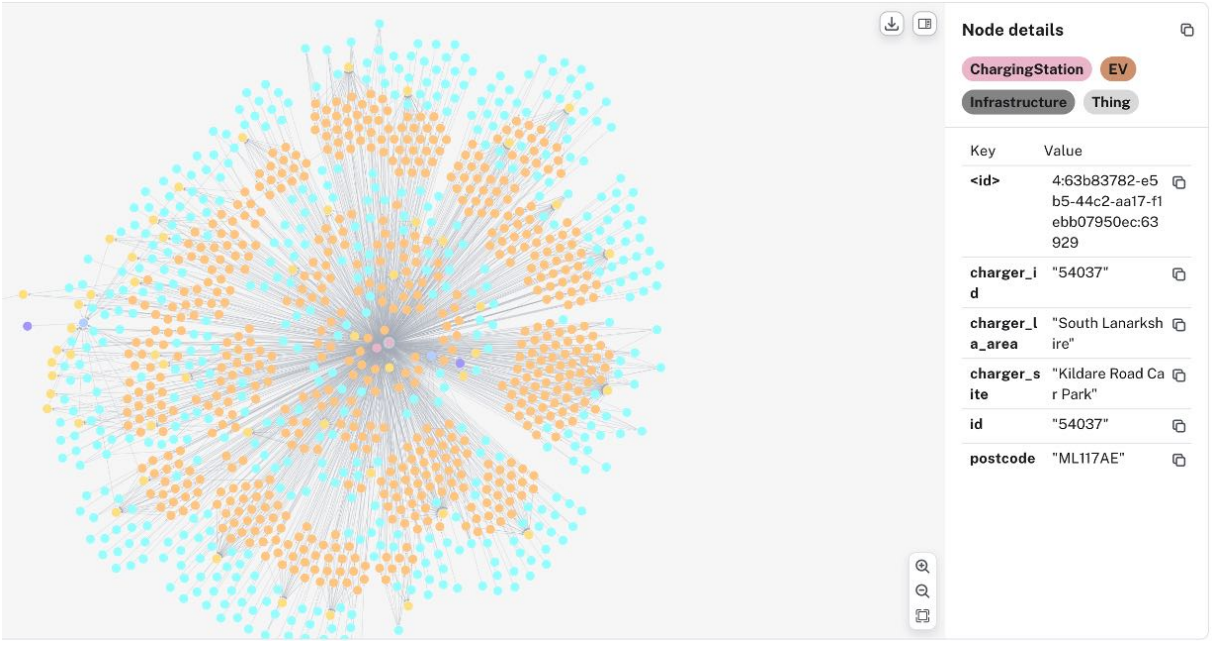


Figure 4.26: Zoom into the bottom cluster from Figure 4.25, showing the two `ChargingStations` (pink) and their connected customers. This area has higher charger availability compared to the large top-left cluster.

4.6.2.4 Limitations of this analysis

While useful, the analysis has several limitations:

- **Postcode-level charger locations:** Charging station positions are approximated using the centroid of their postcode. This is a practical workaround given the lack of exact coordinates, but it introduces some location error.
- **LV_MSP to road node assignment:** Each `LV_MSP` is linked only to its nearest road node, without considering alternative nodes that might offer shorter travel times. This can occasionally overestimate distances.
- **Road network data limitations:** Distances are calculated purely from road lengths. Speed limits and travel times are not available, so accessibility cannot be measured in terms of travel duration.

4.6.3 Example 3: Identifying potentially vulnerable low-demand rural customers

This example shows how the graph can link technical and social data to identify customers who may be at risk of energy poverty. For this case, a customer is considered:

- **Low demand:** Smart-meter record with `maximum_demand` ≤ 5 kWh/day.
- **Vulnerable area:** SIMD decile ≤ 3 (most deprived).
- **Rural:** `ur2_class_name` = `"RURAL"`.

Note: This requires a SmartMeter linked to an LV_MSP and at least one VoltageAlert (or other meter telemetry) to measure demand. Households without smart meters can still be located in vulnerable areas via their Datazone, but their individual demand cannot be confirmed.

```
:param maxDemand => 5;
:param vulnDecile => 3;
:param ur2Class => 'RURAL';
```

Query 4.6: Setting example 3 query parameters

```
MATCH (lv:Electric {asset_type:'LV_MSP'})-[:HAS_SMART_METER]->(sm:
   SmartMeter)
MATCH (sm)-[sv:HAS_VOLTAGE_ALERT]->(va:VoltageAlert)
MATCH (sm)-[:LOCATED_AT]->(addr:Address)
      -[:IN_POSTCODE]->(pc:Postcode)
      -[:BELONGS_TO]->(dz:Datazone)
      -[:HAS_SOCIOECONOMIC_PROFILE]->(profile:SocioEconomicProfile)
WHERE toFloat(va.maximum_demand) <= $maxDemand
  AND toInteger(profile.simd_decile) <= $vulnDecile
  AND dz.ur2_class_name = $ur2Class
WITH lv, sm, va, addr, pc, dz, profile, sv
MATCH path = (lv)-[:HAS_SMART_METER]->(sm)
                -[:LOCATED_AT]->(addr)
```

```
                -[:IN_POSTCODE]->(pc)
                -[:BELONGS_TO]->(dz)
                -[:HAS_SOCIOECONOMIC_PROFILE]->(profile)
RETURN nodes(path) AS nodes,
       relationships(path) AS relationships,
       va AS voltage_alerts,
       sv AS voltage_relationships;
```

Query 4.7: Rural low-demand customers in deprived areas

4.6.3.1 *Interpretation of results*

Figure 4.27 presents the results of a query that traces LV_MSPs connected to smart meters, links them to their associated addresses, postcodes, and datazones, and then retrieves the corresponding socio-economic profiles. The results are filtered to include only rural locations with a SIMD decile of 3 or lower and a recorded maximum demand of 5 kWh/day or less.

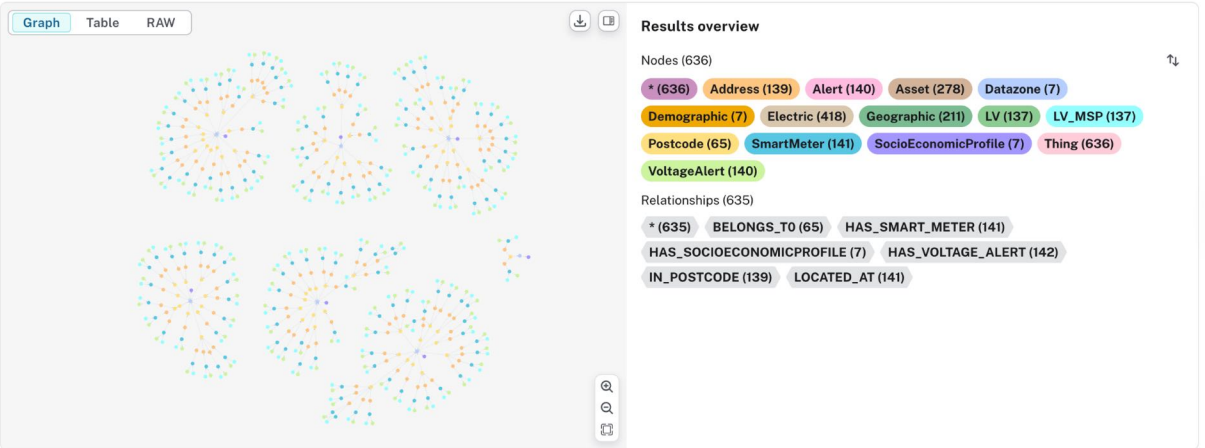


Figure 4.27: Rural low-demand customers in deprived areas. Results of the Cypher query showing LV_MSPs linked to SmartMeters, Addresses, Postcodes, Datazones, and SIMD profiles. Filters: `maximum_demand` $\leq$ 5 kWh/day, `SIMD decile` $\leq$ 3, and `UR2` = `RURAL`.

Figure 4.28 focuses on a single household. This household is in a SIMD decile 1 datazone (most deprived), and its smart meter records around 3–4 kWh/day—significantly below the UK average of ≈7–10 kWh/day [68].

While low demand by itself does not confirm hardship, when combined with high deprivation it may justify further investigation, such as checking for self-rationing, pre-payment constraints, or an unoccupied property.

4.6.3.2 *Relevance of this analysis*

The graph combines network topology with household context in one step, helping teams to:

- Target interventions where technical risk and social vulnerability overlap.
- Flag cases for investigation where low demand and high deprivation coincide.
- Adapt the query easily—for example, changing demand thresholds, rurality class, or focusing on urban areas.

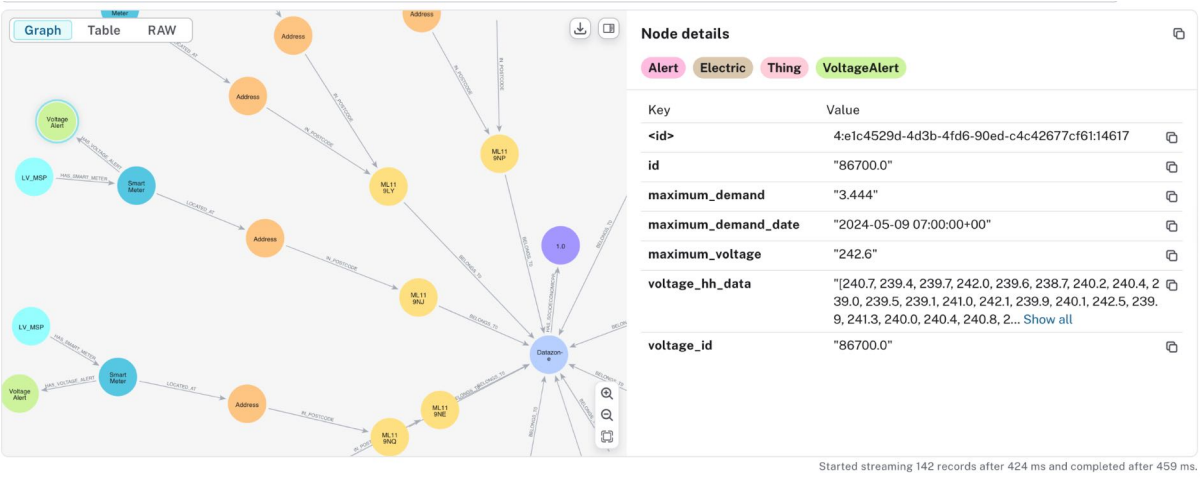


Figure 4.28: Single-household example: SmartMeter with attached `VoltageAlert`, Address → Postcode → Datazone (SIMD decile 1). The meter records ≈3–4 kWh/day, below typical UK usage (7–10 kWh/day), suggesting a case for outreach or further checks.

4.7 Achievements, limitations, and next steps

4.7.1 Summary of key achievements

This work has shown that an electrical distribution network can be effectively represented as a multi-layer, knowledge graph. By semantically aligning all source datasets to a unified ontology, it was possible to achieve a consistent language and relationship structure across heterogeneous data sources.

The resulting model integrates physical network assets—such as transformers, joints, service points, and smart meters—with spatial layers including addresses, postcodes, and road networks, as well as socio-economic data from datazones and the Scottish Index of Multiple Deprivation. This combination allows for a richer and more context-aware representation than would be possible with any single dataset.

A series of parameterised queries demonstrated how the graph can support advanced, multi-dimensional analyses. Examples included identifying vulnerable rural customers with low energy demand and finding customers in proximity to electric vehicle chargers using the road network. These queries were designed so that operational thresholds—such as maximum distance or demand levels—can be adjusted without altering the underlying Cypher logic, making them more accessible to non-technical users.

From an operational standpoint, the graph was successfully deployed both in a local Dockerised Neo4j environment and in the cloud via Amazon Neptune within a secure VPC. A reusable ingestion pipeline was implemented to ensure that new datasets can be integrated through updates to the ontology mapping alone, without the need to restructure the loading process.

4.7.2 Extensibility and limitations

The model was designed to be extensible in several dimensions: it can accommodate new datasets that align with the ontology, adapt to alternative or enriched ontologies if required, and be deployed on multiple graph database platforms thanks to a shared export format.

Nonetheless, some important limitations remain. The electrical modelling currently lacks certain operational details, such as transformer capacities, phase configurations, and real-time load flows—attributes that would be essential for a fully functional digital twin. The reliance on open data also means that some desirable information, such as precise asset coordinates, is missing. Spatial modelling is simplified, as higher-level administrative boundaries were not included due to the pilot area being entirely within a single local authority. Moreover, the model has only been validated on radial networks, leaving its performance on meshed configurations untested. Finally, it operates in a static data environment, without mechanisms for real-time updates or event-driven ingestion.

4.7.3 Recommendations and future work

Future developments should aim to enrich the model with operational electrical parameters, expand the spatial hierarchy to include regional and cross-boundary relationships, and test the framework on meshed network topologies. Integrating real-time data streams from smart meters, voltage monitors, and EV charging stations would enable near-live monitoring and more responsive analysis.

Fusing the model with further internal utility datasets—subject to privacy and governance rules—could address some of the current gaps, while refining the granularity of certain relationships might open new analytical possibilities without adding complexity. There is also scope to explore edge analytics, processing some queries closer to the data source to reduce latency and network load.

Taken together, these enhancements would significantly increase the model's value as a decision-support system, enabling it to serve not only as a static analytical tool but also as a dynamic platform for operational monitoring, strategic planning, and policy evaluation.

5 Case Study 1: Topological Vulnerability and Socioeconomic Impact of Node Failures

5.1 Introduction to the case study

Access to an uninterrupted supply of electricity is fundamental for the proper operation of modern society, from industry and commerce to residential life and critical public services. Even localised power outages can result in severe social and economic outcomes, especially in areas with vulnerable populations [4][69] . The failure of critical components within transmission and distribution networks can have far-reaching consequences, including widespread blackouts. Therefore, utility operators must identify and protect critical infrastructure in order to guarantee reliable and continuous power supply.

This section introduces a case study on the vulnerability analysis of an electrical distribution network in the UK. It outlines the objectives of the study, highlights the importance of assessing the vulnerability of the power grid, and reviews prior work related to the field. Particular emphasis is placed on understanding vulnerability through the lens of direct customer impact and underlying social factors.

5.1.1 Related works

In recent decades, research has applied complex network theory and other analytical methods to understand the vulnerabilities of the power grid. A common approach in the literature is to model power grids as a graph and apply topological metrics- especially node degree, betweenness centrality, and degree distribution- to flag vulnerable nodes and lines [70]. Classic studies showed that removing a few elements of high degree or high betweenness can fragment the network or reduce its global performance, while random failures tend to have milder effects [70] [71] . Following this line, many vulnerability assessments classify components according to these topological indices and then quantify robustness through system-level indicators, such as loss of global efficiency or degradation of connectivity after simulated attacks [70] [72] [73].

However, relying solely on topological indicators is problematic when assessing real customer impacts. For example, the failure of a high-betweenness node, which would cause a large amount

of unserved energy, may in practice be less critical if it primarily serves affluent, well-supported residents than the loss of a lower-load node supplying a socially deprived or otherwise vulnerable population.

Addressing these gaps, recent studies have begun to incorporate impact-based metrics such as customer minutes lost (CML), interruption cost, and social vulnerability indices into the analysis of power outages in order to capture their real-world effects more accurately [4] [27] [74].This shift allows for a more holistic assessment of the tangible consequences of outages and promotes a more all-inclusive approach to prioritising network reinforcement: identifying who is affected, for how long, and with what social and economic repercussions.

Building on this line of research, the present case study demonstrates the value of the proposed graph-based model in going beyond conventional electrical impact metrics. It shifts the focus towards understanding the direct effects of component failures on customers as social entities, linking potential network disruptions to concrete outage consequences. This perspective seeks to support more informed decisions about where to strengthen the network and allocate resources in a way that contributes more meaningfully to social welfare.

5.1.2 Objectives of the case study

To address the limitations of traditional network analysis approaches and to showcase the potential applications of the graph-based model developed in this thesis, the case study applies the model to a section of SPEN's distribution network (to be defined in detail throughout the case study). The aim is to assess the criticality of each network node by combining (i) its probability of failure and (ii) its estimated downtime, with (iii) the expected impact for the customers it serves.

By combining network topology analysis with customer-focused indicators such as Customer Minutes of Interruption (CMI) and a social vulnerability metric derived from SIMD 2020 and the Urban/Rural Classification Framework, the study produces a composite customer impact score for each node. This score captures both the scale of the disruption in terms of electricity supply downtime and the social context of the area affected. Building on this framework, the objectives of the case study are the following:

1. **Quantify node failure likelihood:** estimate the failure probability of each node using historical data to assess risk.
2. **Estimate customer outage downtime:** for every customer in the study area, calculate the annual Customer Minutes of Interruption by simulating failures at each network node and tracing the resulting loss of supply.
3. **Develop a social vulnerability metric:** design a weighted index using SIMD 2020 and Urban Rural Classification to reflect the relative social priority and energy dependence of

different communities.

4. **Derive a composite node impact score:** integrate failure probability, technical impact and social vulnerability into a single metric to rank the criticality of each node.
5. **Identify critical assets:** pinpoint nodes where failures would have the greatest technical and social impact, and propose targeted reinforcements or contingency strategies.

These objectives are designed to guide the development of a practical decision-making tool for network operators, helping to identify not only where prolonged outages are most likely to occur, but also where they would have the most severe real-world consequences.

5.2 Methodology

5.2.1 Scenario definition and subnetwork selection

5.2.1.1 Rationale for scenario choice and selection criteria

The implementation of large-scale vulnerability assessments in distribution networks presents considerable computation and data integration challenges. National-scale distribution network implementations consist of millions of interconnected nodes and edges, which involve substations, transformers, smart meters, and other components. While node-by-node failure simulation at this scale is technically feasible, it can be highly inefficient during early-stage development and testing. Therefore, for proof-of-concept purposes, it is often more practical to work with smaller, but still representative, sections of the network. This approach allows for faster iteration, easier testing and the ability to refine methods before scaling up to the full network.

As a result, a smaller portion of the network was selected to serve as a testbed for developing and testing the proposed methodology. The goal was to identify a region that met the following criteria:

- It is **small enough** to allow for fast processing, simulation and iterative testing.
- It is a **representative example of the wider network**. It contains a diverse set of network features such as multiple voltage levels (LV, HV, EHV), a mix of urban, peri-urban and rural zones, and areas with varying socioeconomic profiles.
- It has **rich data availability**, including asset geolocation, customer connection points, and postcode-level socioeconomic indicators.
- It has a **radial configuration**. While the SPEN Manweb is mostly meshed, this design is unusual in the UK, where most distribution networks- including SPEN SPD region-

follow a radial configuration [75] [76]. To ensure broader relevance and simplify initial implementation, the methodology was tested on a radial network first.

Based on these criteria, the distribution network surrounding the town of Lanark in South Lanarkshire, South Scotland, was selected as the scenario area (see Figure 5.1). This subnetwork includes hundreds of low-voltage, medium-voltage, and high-voltage feeders, and covers a geographically diverse region that includes the towns of Lanark, Biggar, and nearby rural communities. It features a mix of low-density rural areas, small settlements, and more densely populated town centers, offering a representative snapshot of the diverse conditions typically found across the UK.

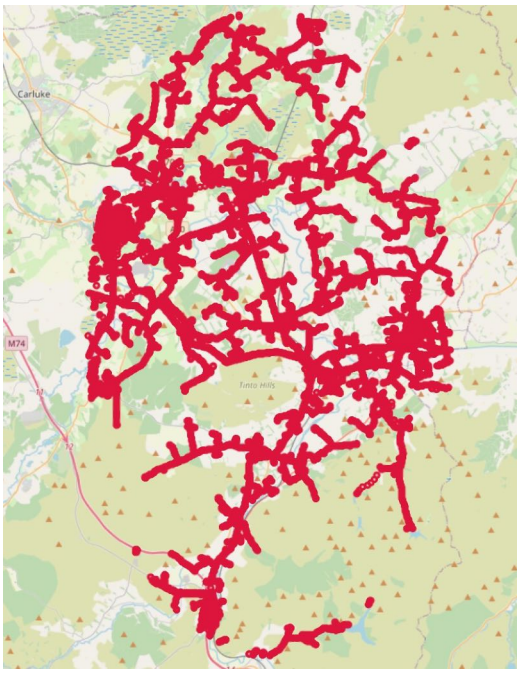


Figure 5.1: Case study network in South Scotland

5.2.1.2 *Exploratory analysis of the selected subnetwork*

To evaluate the alignment of the selected subnetwork with established criteria, an exploratory analysis was conducted on its spatial, demographic, and infrastructural characteristics. The following findings validate Lanark's selection as an appropriate pilot region for this study.

5.2.1.2.1 Urban-Rural composition and spatial diversity

The region contains a heterogeneous mix of urban and rural households. According to the 2-fold Urban/Rural Classification, 61% of households are located in rural areas (7,303 households), while 39% (4,598 households) are classified as urban (Figure 5.2, top left). The more detailed 6-fold classification reveals that most households are located in *Accessible Rural* zones (61.3%), followed by *Accessible Small Towns* (38.6%), with a minimal proportion in *Remote Rural* areas (0.1%) (Figure 5.2, top right).

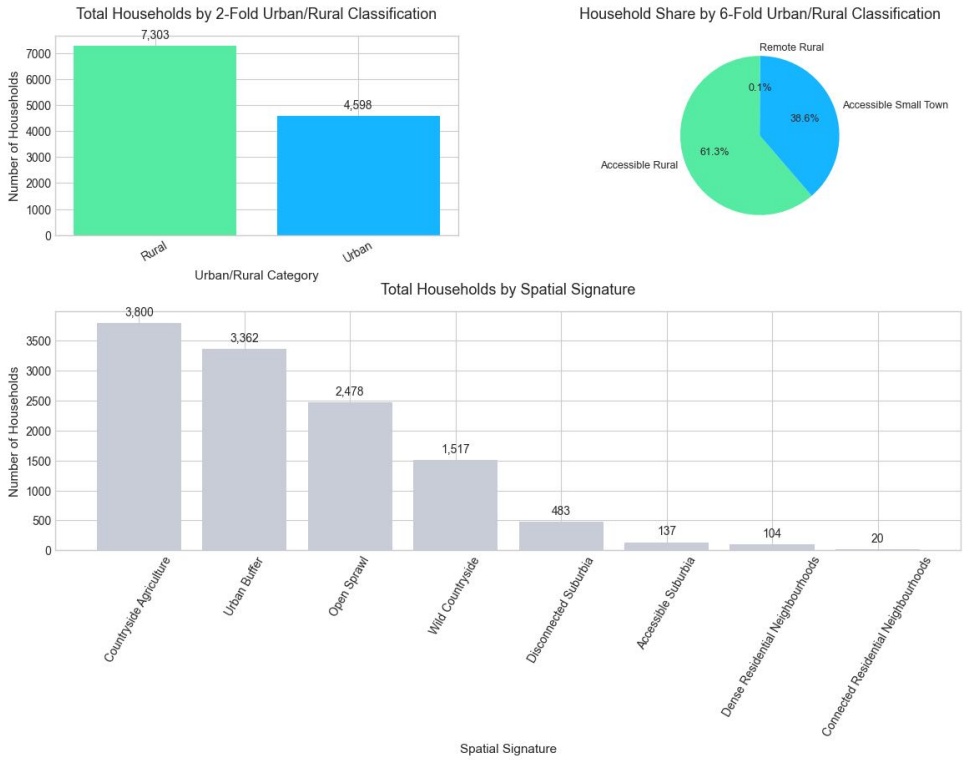


Figure 5.2: Distribution of households by urban classification and spatial signature in the selected subnetwork.

Although dense urban areas would broaden the scope of the analysis, this region provides an appropriate balance between moderately dense urban zones and diverse rural settlements. Furthermore, the Spatial Signature framework discovers several settlement patterns (Figure 5.2, bottom), different in characteristics and spatial layout. The dominant types include *Countryside Agriculture* (3,800 households), *Urban buffer* (3,362), and *Open Sprawl* (2,478). More complex urban forms, including *Dense Residential Neighbourhoods* and *Connected Residential Neighbourhoods*, are present in smaller concentrations. This combination of population densities, accessibility levels, and settlement patterns ensures that the methodology can be tested in various contexts, supporting the generalizability of the findings to other types of settlement in the UK.

5.2.1.2.2 Socioeconomic profile

The geographical distribution of households by SIMD 2020 deciles shows substantial differences between socioeconomic segments (Figure 5.3). As expected, the most common deciles fall within the middle ranges (deciles 4-8), representing the majority of the population distribution. However, while very wealthy households (deciles 9-10) are few in number, there is a significant proportion of socially deprived households in the lower deciles (1-3).

The spatial pattern reveals that isolated communities, particularly in the northern and south-western areas, tend to exhibit higher levels of deprivation (Figure 5.3, top left). In contrast,

urban areas, such as Lanark, display the characteristic socioeconomic heterogeneity typical of most cities. The most affluent areas are concentrated in the southeastern region around Biggar, an area recognised for its high quality of life within Lanarkshire. This distribution pattern makes the region particularly suitable for testing how social vulnerability indices can complement technical assessments on response to outages and prioritisation of network reinforcements, ensuring that the social dimension of power systems is captured in the analysis.

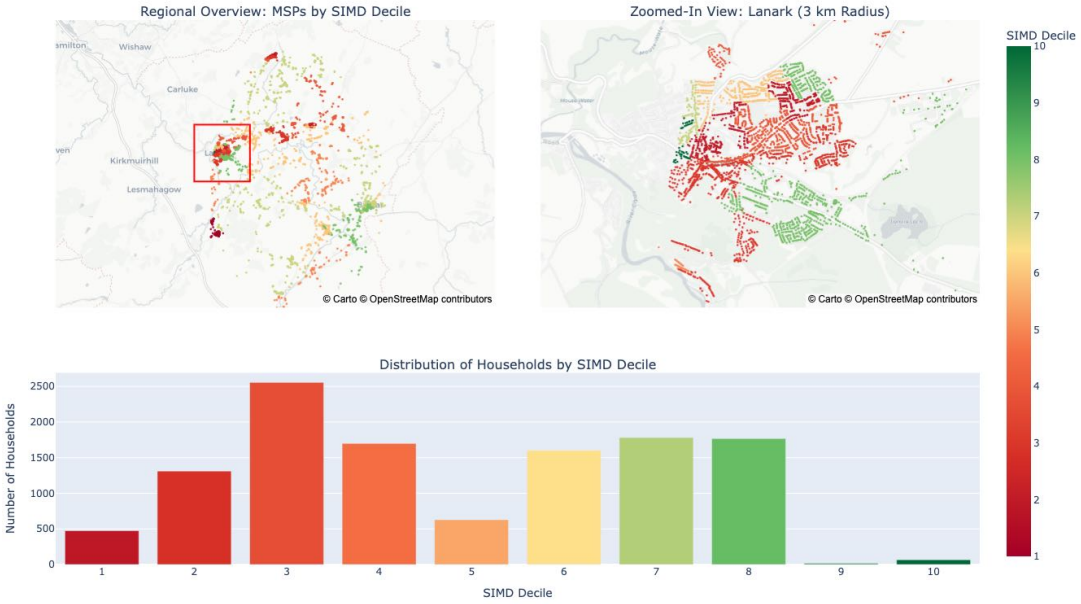


Figure 5.3: Distribution of households by SIMD in the selected subnetwork and the Lanark area.

5.2.1.2.3 Infrastructure coverage

The chosen subnetwork provides a detailed and coherent representation of the hierarchical structure typical of SPEN's distribution network. Power flows begin at the *Primary Transformer*, where voltage is stepped down from the transmission system. From there, electricity is routed through a *Busbar*, which distributes the supply to multiple *Distribution Transformers* for further voltage reduction.

Downstream of the *Distribution Transformer*, the *Low Voltage (LV)* network delivers electricity directly to end users. This segment includes a range of asset types such as *LV Intersection Points*, *LV Joints*, *LV Links*, *LV Fuses*, *LV Endpoints*, and *LV MSPs*, as well as specific termination components like the *Unmetered Supply Point (Unmetered SP)*. These ultimately connect to individual customers, including homes, businesses, and *EV Chargers*.

Upstream of the *Distribution Transformer*, the *High Voltage (HV)* network carries power through a combination of *HV Intersection Points*, *HV Joints*, *HV Switches*, *HV Protective Devices*, *HV MSPs*, and *HV Endpoints*. This structure supports radial power flow from the *Primary Transformer* to downstream consumers while incorporating redundancy and protection at key stages.

While *Extra High Voltage (EHV)* assets—such as *EHV Intersection Points*, *EHV Joints*, *EHV Switches*, and *EHV Endpoints*—are present in the dataset, these are situated upstream of the *Primary Transformer* and form part of the transmission-distribution interface. As such, they are excluded from the scope of this vulnerability assessment, which is limited to the distribution-level network comprising the HV and LV segments.

As shown in Figure 5.4, the dataset includes 13,104 *LV Joints*, 10,719 *LV MSPs*, and more than 2,500 *LV Intersection Points*, as well as several hundred *Distribution Transformers*. On the HV side, over 4,700 *HV Intersection Points* and more than 2,100 additional *HV components* are recorded. Although the *EHV* layer includes 103 *EHV Intersection Points*, these are not considered in the analysis.

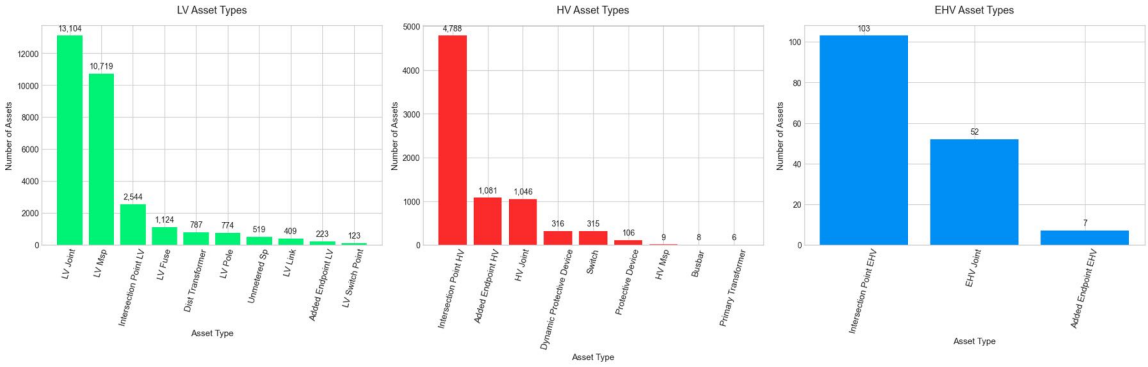


Figure 5.4: Distribution of asset types by voltage level in the selected subnetwork.

5.2.2 Customer downtime simulation following network node failures

This section discusses the methodology for estimating the technical impact of individual node failures within the pilot network. The approach incorporates historical fault data with graph simulations to model the propagation of outages and the effects on end users.

The process involves three main steps. First, each node in the network is assigned a failure probability and an expected repair time based on its asset classification. Second, network graph simulations emulate the failure of individual nodes to trace supply loss by identifying customers disconnected from upstream sources. Third, total *Customer Minutes of Interruption (CMI)* is calculated for each failure scenario, generating node-level indicators that quantify the technical severity of an outage relating to its duration and population affected.

This analysis forms the technical foundation of the case study and will be combined in the next section with social vulnerability metrics to develop a composite measure of node criticality.

5.2.2.1 Outage dataset selection and rationale

To estimate the annual probability of failure for electrical assets in the Scottish Power Distribution (SPD) network, this study uses historical fault data from Electricity North West (ENWL). The

analysis is based on the *Unplanned Outages* dataset available through *ENWL's Open Data Portal* [77], which provides structured records of power interruptions dating back to 2000, including the time, location, cause, duration, and affected asset category.

ENWL was selected as a proxy data source for several reasons:

- **Geographic similarity:** ENWL operates in North West England [78], a region with environmental and geographic conditions reasonably comparable to Scotland. As a result, observed failure patterns in ENWL's network are assumed to be reasonably transferable to SPD's operational context.
- **Data quality:** ENWL provides one of the most comprehensive and publicly accessible outage datasets among UK DNOs, with sufficient granularity to support asset-level analysis by cause, equipment type, and voltage level.
- **Operational convergence:** Following the acquisition of ENWL by Iberdrola [79], both networks now operate under the same corporate structure as SPD. This alignment may promote convergence in day-to-day operations, data handling practices, and overall network management.

Together, these factors justify the use of historical ENWL data to infer failure probabilities for the SPD network in the absence of directly equivalent public data.

5.2.2.2 Outage data preprocessing

To preserve analytical rigour, the ENWL outage data underwent a series of preprocessing steps, detailed in the following subsections.

5.2.2.2.1 Timeframe selection

Only outage events from 2015 onward were considered. Limiting the dataset to the post-2015 period, when *RIIO-ED1* introduced a single reporting standard, guarantees uniform equipment coding and directly comparable data across all UK DNOs. In addition, old entries may not be representative of today's modern, increasingly digitalised grid. Therefore, focusing on the most recent ten years strikes a balance between providing a sufficiently large sample for reliable estimates while ensuring closer alignment with current network conditions.

5.2.2.2.2 Voltage level normalisation

Data was standardised and grouped into two categories based on voltage level:

- **Low Voltage (LV):** $\leq$ 11 kV
- **High Voltage (HV):** $>$ 11 kV

This classification aligns with the operational structure in SPEN, where distribution transformers

mark the boundary between the high-voltage and low-voltage networks.

5.2.2.2.3 Harmonisation of asset types

To ensure compatibility with the SPD network model, equipment categories from the ENWL dataset were standardised. Table 5.1 illustrates the mapping between ENWL equipment categories and the corresponding SPD asset types found in the pilot network (see Section 5.2.1.2.3).

Table 5.1: Mapping of Network Node Types to ENWL Equipment Categories for Unplanned Outage Analysis.

Node Type	Mapped Category	Justification
Added Endpoint EHV	POWER CABLES	Terminates or connects EHV underground cables.
Added Endpoint HV	POWER CABLES	Terminates or connects HV underground cables.
Added Endpoint LV	POWER CABLES	Terminates or connects LV underground cables.
Busbar	SWITCHGEAR, FUSEGEAR AND BUSBARS	Central switching element in substations.
Dist Transformer	POWER TRANSFORMERS, REACTORS ETC.	Steps voltage down from HV to LV.
Dynamic Protective Device	PROTECTION EQUIPMENT	Intelligent fuse, relay or recloser unit.
EHV Joint	POWER CABLES	Connects two EHV cable sections.
HV Joint	POWER CABLES	Connects two HV cable sections.
Intersection Point EHV	POWER CABLES	Logical EHV cable junction in the model.
Intersection Point HV	POWER CABLES	Logical HV cable junction in the model.
Intersection Point LV	POWER CABLES	Logical LV cable junction in the model.
LV Fuse	SWITCHGEAR, FUSEGEAR AND BUSBARS	Protection component on LV circuits.
LV Joint	POWER CABLES	Connects two LV cable segments.
LV Link	SWITCHGEAR, FUSEGEAR AND BUSBARS	Link box or similar interface equipment.
LV Pole	OVERHEAD LINES	Structural support for LV overhead lines.
LV Switch Point	SWITCHGEAR, FUSEGEAR AND BUSBARS	Switching location on LV circuits.
Primary Transformer	POWER TRANSFORMERS, REACTORS ETC.	Bulk supply transformer at primary level.
Protective Device	PROTECTION EQUIPMENT	General fault detection and isolation device.
Switch	SWITCHGEAR, FUSEGEAR AND BUSBARS	Manual or automated switching unit.

5.2.2.3 Estimation of annual failure rates

Previous research has employed various stochastic modeling approaches to predict power system asset failures, including Poisson regression, Monte Carlo simulation and Bayesian models [80, 81, 82, 83]. These models are effective in capturing the stochastic nature of discrete rare events such as faults. However, all such approaches require knowledge of the underlying asset population (or exposure time) to normalize event counts into per-unit failure rates. Since the ENWL unplanned outages dataset lacks asset inventory data at each point in time, asset-level failure rate estimation cannot be performed using these methods.

To address this limitation, a proxy-based approach is adopted. Historical fault data from the ENWL unplanned outages dataset (post-2015) is used to calculate the average annual number of faults. These events are disaggregated by three key dimensions: *asset category* (a), *fault cause* (c), and *voltage level* (v).

Since the total number of ENWL assets is unknown, the resulting fault rates cannot be directly translated into per-unit probabilities. Instead, the annual fault counts are scaled to the Scottish Power Distribution (SPD) network using the ratio of customer base between the two utilities—more precisely, using the number of customers in the SPD pilot region relative to

ENWL's total customer base. This approach assumes that asset count is proportional to the number of customers served in each network. As a result, it enables the estimation of the expected number of faults per year in the pilot region for each category (a, c, v).

Finally, to obtain a per-asset failure probability, these scaled fault counts are normalised by the number of SPD assets in each category within the pilot area. The resulting metric represents the estimated probability that an individual asset of type a, affected by fault cause c, and operating at voltage level v, will experience an unplanned outage in a given year.

This methodology provides a practical way to infer failure probabilities in the absence of full asset registries, while preserving technical and contextual distinctions across fault categories.

Disclaimer. This approach assumes that: (i) fault events are *independent* of one another (e.g., a protective-device failure is treated as unrelated to a potential subsequent transformer fault); (ii) no cascading or correlated failures are modelled; and (iii) spatial or climatic dependencies are ignored. Consequently, the resulting probabilities represent first-order failure risks only, and may understate compound or cascading outage scenarios.

5.2.2.3.1 Classification of fault events

Unplanned outages were first grouped by three dimensions:

- **Asset Category** (a): Standardised to align with SPD asset types used in the network model (see Table 5.1).
- **Fault Cause** (c): Using the `Direct Cause Category` field from the ENWL dataset, following definitions in the *RIIO-ED1 Regulatory Instructions and Guidance: Annex F* [84].
- **Voltage Level** (v): Grouped into low-voltage or high-voltage according to the classification in Section 5.2.2.2.2.

5.2.2.3.2 Computation of annual fault rates in ENWL

After classification, faults were aggregated by year and averaged across the full observation period (2015 onwards). For each unique combination (a, c, v):

$$\lambda_{a,c,v}^{(\text{ENWL})} = \frac{N_{a,c,v}^{(\text{ENWL})}}{T} \tag{5.1}$$

Where:

- $N_{a,c,v}^{(\text{ENWL})}$ is the total number of faults observed in ENWL for group (a, c, v).
- T is the number of years of observation.

5.2.2.3.3 Customer-based scaling of number of faults to the pilot region

As the total number of assets in ENWL is unknown, a scaling factor is applied based on customer base. Let:

- C_{SPD} = number of customers in the SPD pilot region.
- C_{ENWL} = number of customers in the ENWL network.

Then, the scaling factor is:

$$s = \frac{C_{\text{SPD}}}{C_{\text{ENWL}}} \tag{5.2}$$

The estimated number of faults per year in SPD is:

$$\lambda_{a,c,v}^{(\text{SPD})} = \lambda_{a,c,v}^{(\text{ENWL})} \cdot s \tag{5.3}$$

This yields a proxy fault rate for SPD based on ENWL's data and the relative network size.

5.2.2.3.4 Calculation of annual failure probability per unit (asset)

Finally, the number of assets per classification in the pilot region is used to calculate failure probability per unit. Let:

- $A_{a,c,v}^{(\text{SPD})}$ = number of SPD assets of type (a, c, v) in the pilot area.

Then, the annual probability of failure per unit is:

$$p_{a,c,v} = \frac{\lambda_{a,c,v}^{(\text{SPD})}}{A_{a,c,v}^{(\text{SPD})}} \tag{5.4}$$

Note. This procedure assumes that all assets of the same type have equal failure probability, regardless of location. This simplification ignores local differences (e.g., loading, topology), but enables an estimation based on asset class, voltage level, and fault cause. For the calculations, ENWL is assumed to have 2.4 million customers according to [78].

5.2.2.4 Typical duration of individual outage events

To estimate the expected downtime per node, it is first necessary to determine the typical duration of individual outage events. This is achieved by analysing historical incident data recorded in the ENWL unplanned outages dataset, using the same preprocessing steps described in Section 5.2.2.2.

For each unique combination (a, c, v), outage records are grouped, and the following statistics are computed:

- Number of incidents
- Mean and median duration
- Standard deviation
- Minimum and maximum duration
- 75$^{\text{th}}$ percentile duration $D^{75}_{a,c,v}$

The 75$^{\text{th}}$ percentile is selected as the expected repair time, as it provides a conservative estimate that captures the majority of events while minimising the influence of extreme values.

$$D^{75}_{a,c,v} = \text{75th percentile of observed incident durations for group } (a, c, v) \tag{5.5}$$

Note. Durations are measured in minutes. Only complete and valid records are included to ensure data reliability.

5.2.2.5 Estimation of the expected annual downtime per asset type and voltage level

Once the annual failure probability $p_{a,c,v}$ and the typical repair duration $D^{75}_{a,c,v}$ have been estimated for each combination of asset type a, fault cause c, and voltage level v, the total expected downtime for assets of type a at voltage level v is calculated by aggregating across all fault causes.

Let $\mathcal{C}_{a,v}$ be the set of all fault causes associated with asset type a and voltage level v. The expected downtime is given by:

$$\text{Downtime}_{a,v} = \sum_{c \in \mathcal{C}_{a,v}} p_{a,c,v} \times D^{75}_{a,c,v} \tag{5.6}$$

This metric, expressed in minutes per year, represents the average annual unavailability of an individual asset of type a operating at voltage level v, due to all considered failure scenarios.

5.2.2.6 Total Customer Minutes of Interruption (CMI)

Assessing the true impact of network failures on customers requires taking into account not only how long an outage lasts, but also how many customers it disrupts. Using this information,

the annual **Customer Minutes of Interruption** (CMI) for each network node is determined in two steps:

1. **Identify disconnected customers.** For every candidate node, its failure is simulated in the network model. Supply points that lose connectivity to a primary transformer—meaning no alternative power-flow paths remain, as defined in Section 5.2.1.2.3—are flagged as *interrupted*. The size of this set equals the number of customers, denoted $N_{\text{cust},\,n}$, who would be without power if node n fails.

2. **Compute the node's annual CMI.** Each node belongs to an asset category a and operates at voltage level v. Using the expected annual downtime for that group, $\text{Downtime}_{a,v}$ (derived previously), the annual customer-minutes of interruption for node n is

$$\text{CMI}_{n,a,v} \;=\; N_{\text{cust},\,n} \;\times\; \text{Downtime}_{a,v}.$$

5.2.2.6.1 *Impact analysis of node failures on customer connectivity*

The customer impact analysis begins by extracting the electrical subgraph from the graph database (see Query 5.1 and 5.2) and preparing it for local processing using the `NetworkX` library in Python [85].

While the graph database is well-suited for persistently storing and querying large-scale network data, `NetworkX` provides a more efficient environment for executing complex topological algorithms and allows greater control over in-memory traversal logic. The following design considerations are applied during this process:

1. **Filtering non-electrical relationships.** Some relationships in the graph represent geographic or demographic metadata rather than physical electrical connectivity. These are excluded to prevent inaccuracies in the connectivity graph. The excluded relationship types are defined in the variable `EXCLUDED_REL_TYPES`:

   `{SERVES_ADDRESS, HAS_SMART_METER, LOCATED_AT, IN_POSTCODE, BELONGS_TO, HAS_SOCIOECONOMIC_PROFILE}`

2. **Voltage-level encoding.** Each node's nominal voltage label is mapped to an integer level ℓ (see Table 5.2, which is used by the path-validation algorithm to enforce monotonic voltage descent—that is, supply paths are expected to move consistently from higher to lower voltage levels.

   In specific cases, limited upward transitions may be allowed to account for real-world scenarios, such as re-routing through a distribution transformer followed by reconnection to an alternative high-voltage path and eventually returning to low voltage. To capture this behavior, the algorithm permits upward transitions if they pass through nodes explicitly

listed in the ALLOWED_TRANSITION_NODES set:

$$\{\texttt{Dist\_Transformer}\}$$

Table 5.2: Voltage-level encoding used for path validation.

Voltage Label	Typical Range	Assigned Level ℓ
EHV	$\gtrsim 33$ kV	3
HV	11–33 kV	2
LV	< 11 kV	1

3. **Level-specific source sets.** A valid supply path must begin at a level-3 source, pass through a level-2 node, and terminate at a level-1 transformer before reaching a low-voltage meter service point (MSP). The permitted node types at each level are shown at Table 5.3.

Table 5.3: Permitted node types by voltage level.

Level	Allowed Node Types
L3	`Primary Transformer`
L2	`Busbar`
L1	`Distribution Transformer`

4. **Local graph construction with `NetworkX`.** After filtering, node and edge lists are imported into `NetworkX`. Executing breadth-first search (BFS) locally is significantly faster than issuing repeated Cypher queries to the graph database. It also enables custom logic—such as voltage-aware path validation and critical-node analysis.

5. **Path enumeration and critical-node detection.** The process begins with IsPathValid (Algorithm 1), which verifies whether a path complies with the voltage-level hierarchy. It enforces a monotonic voltage descent—i.e., transitions must proceed from higher to lower voltage levels—except where upward transitions are explicitly allowed through specific node types defined in `ALLOWED_TRANSITION_NODES`, such as `Distribution Transformer`.

Algorithm 1: Check if a Path is Valid

Input: Path (`path`), node attributes: `node_type`, `level`, sets `LEVEL_3_SOURCES`, `LEVEL_2_SOURCES`, `LEVEL_1_SOURCES`, and set `ALLOWED_TRANSITION_TYPES`

Output: Boolean (True if path is valid)

typesInPath $\leftarrow$ [node_type(n) for n in path]

`// 1. Presence of required source levels`

hasL3 $\leftarrow$ any($t \in$ `LEVEL_3_SOURCES` for t in typesInPath)

hasL2 $\leftarrow$ any($t \in$ `LEVEL_2_SOURCES` for t in typesInPath)

hasL1 $\leftarrow$ any($t \in$ `LEVEL_1_SOURCES` for t in typesInPath)

if *not* *(hasL3* ***and*** *hasL2* ***and*** *hasL1)* **then**

  return False

`// 2. Source type ordering (L3→L2→L1)`

idxL3 $\leftarrow$ first index matching `LEVEL_3_SOURCES`

idxL2 $\leftarrow$ first index matching `LEVEL_2_SOURCES`

idxL1 $\leftarrow$ first index matching `LEVEL_1_SOURCES`

if *not* *(idxL3* $<$ *idxL2* $<$ *idxL1)* **then**

  return False

`// 3. No invalid upward transitions`

for $i \leftarrow 0$ **to** $|path| - 2$ **do**

  $u \leftarrow$ path[i]; $v \leftarrow$ path[i+1]

  if *level(v)* $>$ *level(u)* ***and*** *node_type(u)* $\notin$ *ALLOWED_TRANSITION_TYPES* **then**

    return False

return True

Next, EnumeratePaths (Algorithm 2) traverses the electrical graph from each valid primary source, collecting all supply paths to low-voltage meter service points (MSPs) that satisfy the level constraints enforced by IsPathValid.

Algorithm 2: Enumerate Valid MSP Paths via BFS

Input: Graph G, lists `level_3_nodes`, `lvMsps`, integer `MAX_DEPTH`, and function `IsPathValid`

Output: Maps `validPathsPerMsp` and `nodesInValidPaths`

```
validPathsPerMsp ← empty map (MSP → list of valid paths)
nodesInValidPaths ← empty map (node → set of MSPs)
foreach sourceNode ∈ level_3_nodes do
    queue ← initialize queue with (sourceNode, [sourceNode])
    while queue not empty do
        (currentNode, currentPath) ← dequeue(queue)
        if |currentPath| > MAX_DEPTH then
            continue
        foreach neighborNode adjacent to currentNode do
            if neighborNode ∈ currentPath then
                continue
            newPath ← currentPath + [neighborNode]
            if |newPath| > MAX_DEPTH then
                continue
            if neighborNode ∈ lvMsps and IsPathValid(newPath) then
                validPathsPerMsp[neighborNode].append(newPath)
                foreach internalNode in newPath (excluding last) do
                    nodesInValidPaths[internalNode].add(neighborNode)
            else
                enqueue(queue, (neighborNode, newPath))
totalPaths ← total number of valid paths collected
```

Then, ComputeCriticalNodeImpact (Algorithm 3) determines whether a node is *critical* for any MSP by checking whether the node appears in *all* valid supply paths to that MSP. The number of such uniquely dependent MSPs is recorded as $N_{\text{MSP},\,n}$, representing the number of low-voltage service points that would be affected by a failure at node n. The corresponding number of impacted customers ($N_{\text{cust},\,n}$) is computed later based on the customer count per MSP.

Algorithm 3: Compute Critical Node Impact on MSPs

Input: Maps `validPathsByMsp` (MSP → valid paths),
`nodesInValidPaths` (node → MSPs where node appears)
Output: Sorted table of critical nodes and affected MSPs
criticalNodesList ← empty list
foreach *node in nodesInValidPaths* **do**
  affectedMsps ← nodesInValidPaths[node]
  criticalMsps ← empty list
  foreach *msp in affectedMsps* **do**
    validPaths ← validPathsByMsp[msp]
    if *validPaths not empty* ***and*** *node appears in* ***every*** *path in validPaths* **then**
      append msp to criticalMsps
  if *criticalMsps not empty* **then**
    Add to criticalNodesList:
      - node identifier
      - node type
      - count of critical MSPs (length of criticalMsps)
      - list of critical MSPs affected
nodeImpactTable ← create table from criticalNodesList

Note: This analysis is a simplified approximation based solely on topological connectivity. It does not account for power flow constraints, load distribution, or capacity limits that might arise when a node fails. Furthermore, only low-voltage MSPs are considered in the customer impact calculation. Although in reality an LV MSP failure could affect downstream customers (e.g., in a cascaded setup), this work focuses exclusively on failures of network infrastructure elements, treating LV MSPs as terminal endpoints for the purposes of fault analysis.

```
MATCH (n:Electric)
WHERE n.voltage_level IS NOT NULL AND n.asset_type IS NOT NULL
RETURN n.id AS id,
       n.asset_type AS asset_type,
       n.voltage_level AS voltage_level
```

Query 5.1: Extract electric nodes with voltage and type metadata.

```
MATCH (a:Electric)-[r]-(b:Electric)
WHERE a.voltage_level IS NOT NULL AND b.voltage_level IS NOT NULL
  AND NOT type(r) IN EXCLUDED_REL_TYPES
WITH a, b, type(r) AS rel_type
WITH CASE WHEN a.id < b.id THEN a ELSE b END AS n1,
     CASE WHEN a.id < b.id THEN b ELSE a END AS n2,
     rel_type
```

```
RETURN DISTINCT n1.id AS source,
                n2.id AS target,
                rel_type
```

Query 5.2: Extract valid electrical edges, excluding non-electrical relationships defined in EXCLUDED_REL_TYPES.

5.2.2.6.2 *Compute the annual Customer Minutes of Interruption per node*

To estimate the total annual CMI caused by the failure of a network node, it is necessary to translate the number of impacted (LV_MSPs) into actual household counts. Since an LV_MSP typically serves multiple customers, Query 5.3 is performed on the graph database to retrieve the number of households connected to each affected MSP.

```
MATCH (msp:Electric {asset_type: 'LV_MSP'})
OPTIONAL MATCH (msp)-[:SERVES_ADDRESS]->(a:Address)
OPTIONAL MATCH (postcode:Postcode {id: msp.postcode})-[:BELONGS_TO
   ]->(datazone:Datazone) -[:HAS_SOCIOECONOMICPROFILE]->(profile:
   SocioEconomicProfile)
WITH msp, count(a) AS num_households, postcode, datazone, profile
RETURN msp.id AS msp_id,
       coalesce(msp.postcode,'') AS node_postcode,
       coalesce(toFloat(msp.latitude)) AS latitude,
       coalesce(toFloat(msp.longitude)) AS longitude,
       coalesce(datazone.ur2_class_name,'')    AS ur2_class,
       coalesce(datazone.ur6_class_name,'')    AS ur6_class,
       coalesce(postcode.ur_signature,'') AS ur_signature,
       num_households,
       coalesce(toInteger(profile.income_decile),0)      AS
          income_decile,
       coalesce(toInteger(profile.employment_decile),0)  AS
          employment_decile,
       coalesce(toInteger(profile.education_decile),0)   AS
          education_decile,
       coalesce(toInteger(profile.health_decile),0)      AS
          health_decile,
       coalesce(toInteger(profile.crime_decile),0)       AS
          crime_decile,
       coalesce(toInteger(profile.housing_decile),0)     AS
          housing_decile,
       coalesce(toInteger(profile.access_decile),0)      AS
          access_decile,
       coalesce(toInteger(profile.simd_decile),0)        AS
          overall_decile
ORDER BY num_households DESC
```

Query 5.3: Extract MSP locations, household counts, and socioeconomic profiles.

Additionally, the Query 5.3 extracts the socioeconomic profile associated with each LV_MSP, based on its geographic location (e.g., postcode). This includes metrics such as income, employment, education, and access to services, which will be used in the following section to construct a social vulnerability index. Given:

- $\text{Downtime}_{a,v}$: the expected annual downtime (in minutes) for assets of type a and voltage level v,
- $N_{\text{msp},\,n}$: the set of LV_MSPs that depend critically on node n,
- H_m: the number of households served by each LV_MSP $m \in N_{\text{msp},\,n}$,

the total number of customers affected by node n is computed as:

$$N_{\text{cust},\,n} = \sum_{m \in N_{\text{msp},\,n}} H_m$$

Then, the annual Customer Minutes of Interruption for node n, given its asset type a and voltage level v, is computed as:

$$\text{CMI}_{n,a,v} = N_{\text{cust},\,n} \times \text{Downtime}_{a,v}$$

The resulting value provides a technical measure of service disruption due to the failure of a given node. It forms one of the components of the composite metric developed in the following sections.

5.2.3 Construction of a scoial vulnerability metric

Beyond assessing failure impact from a technical standpoint, a social vulnerability metric was developed as an additional measure of the relative inequities experienced by different populations from power outages. This indicator is constructed through a two-stage methodology:

1. **Rescaling the weights of socioeconomic deciles** to reflect their relative importance in terms of vulnerability to outages and energy poverty. Not all dimensions contribute equally, and some may have nonlinear effects on exposure and recovery capacity.
2. **Constructing a unified social vulnerability index** by combining weighted scores from all relevant indicators. To ensure objectivity, the aggregation employs Pareto dominance ranking through `pymoo.NonDominatedSorting()` in Python [86].

Each of these steps is explained in detail below.

5.2.3.1 Rescaling of socioeconomic weights

Based on the data retrieved from Query 5.3, the socioeconomic profiles and UR6 classifications were obtained for each LV MSP. The socioeconomic profiles include decile scores (from 1 to 10) for categories such as *income*, *employment*, *education*, *health*, *crime*, *housing*, and *access*. In all cases, a lower decile indicates higher vulnerability.

The UR6 classification, which ranges from 1 (most urban) to 6 (most rural), was also considered to reflect the increased exposure of rural areas to power outages. Remote locations typically have lower network redundancy and slower recovery times, which makes them more vulnerable in case of supply interruptions.

However, not all indicators affect outage vulnerability in the same way. For example, poor health conditions can significantly increase the risk during outages due to dependency on medical equipment, while income by itself might not directly determine vulnerability unless combined with other aspects such as housing quality or geographic remoteness.

To account for these differences, each indicator was rescaled using a custom vulnerability curve (see Figure 5.5).

These curves map the original decile scores to continuous weights between 0 and 1, with non-linear shapes that reflect the specific influence of each indicator on vulnerability. In most cases, the lower deciles were assigned higher weights to represent increased risk. The justification for each rescaling curve is presented in Table 5.4.

Table 5.4: Justification for the Shape of Rescaling Curves by Indicator

Indicator	Justification for Curve Shape
Health	Outages pose serious risks to vulnerable individuals (e.g., elderly or those with medical devices). Curve emphasizes low deciles.
Education	Low education may reduce preparedness during outages. Curve decays quickly.
Access	Limited transport/access to services worsens outage effects. Strong early decile weight.
Income	Low income often implies poor housing or heating.
Employment	May signal vulnerability to energy insecurity.
Crime	Outages increase crime risk in unsafe areas.
Housing	Poor housing increases outage impact (e.g., cold, unsafe heating).
UR6 (rurality)	Rural areas face longer restoration. Curve rises with rurality.

5.2.3.2 Social vulnerability index based on multi-criteria Pareto ranking

To avoid assigning arbitrary weights to the different socioeconomic indicators, each LV MSP was classified through a Pareto dominance ranking algorithm to identify areas that are simultaneously vulnerable in multiple dimensions.

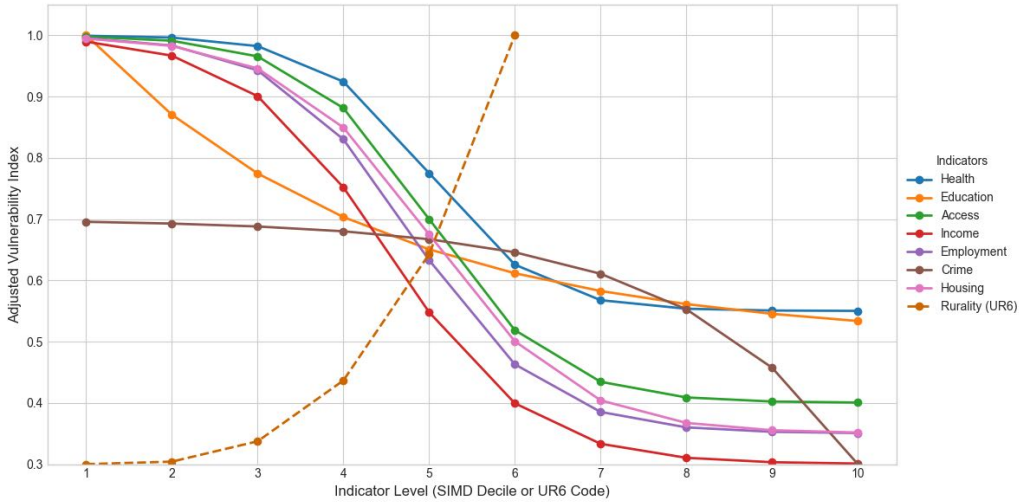


Figure 5.5: Adjusted Social Vulnerability Index across socioeconomic deciles and rurality levels. Each curve represents how different indicators—such as health, education, access, income, and rurality—contribute to increased vulnerability to power outages. Lower deciles (1 = most deprived) and higher rurality codes are associated with higher vulnerability.

This method, widely applied in resilience studies [4, 87], enables a *multi-objective ranking* of LV MSPs based on how many others dominate them across all vulnerability indicators. The algorithm was implemented using `pymoo.NonDominatedSorting()` in Python [86]. As a result, each MSP is assigned a *vulnerability rank*: MSPs that are not dominated by any other are considered *highly vulnerable* across all dimensions.

However, since a single electrical node can affect many LV MSPs, each with distinct locations and social characteristics, it is necessary to aggregate the vulnerability information at the *node level*. To do so, the following method was applied:

1. Each node's set of impacted MSPs was expanded so that each MSP is treated independently.
2. The *Pareto rank* and *number of households* served by each MSP were retrieved.
3. For each node, a *weighted average Pareto score* was computed, weighting the vulnerability of each MSP by its number of households.

The result is a *social vulnerability score per node* that reflects not only the vulnerability of affected areas but also the scale of the impact in terms of population.

5.2.4 Unified criticality score based on technical and social dimensions

For the purpose of ranking network nodes according to both the technical impact of outages and the customers' social vulnerability, an overall *criticality score* was developed. This unified score integrates the Customer Minutes of Interruption from the technical assessment (see Section 5.2.2.6.2) with the social vulnerability index (SVI) derived from the socioeconomic indicators

(see Section 5.2.3).

Then, both components undergo Min-Max normalization [88] to ensure they are on the same scale and comparable. This prevents one metric from disproportionately influencing the final result.

The criticality score is then calculated as a weighted sum of the two normalised scores:

$$\text{Criticality Score} = \alpha \times \text{norm\_CMI} + \beta \times \text{norm\_SVI} \tag{5.7}$$

Where:

- `norm_CMI` is the normalised Customer Minutes of Interruption, representing technical impact.
- `norm_SVI` is the normalised social vulnerability index.
- α and β are the weights assigned to each component, with $\alpha + \beta = 1$.

This approach enables network operators to adjust the weights according to their strategic goals or regulatory obligations. For example, placing more emphasis on vulnerable populations might justify increasing the weight of the social component.

Future work should explore optimisation algorithms that determine optimal weight configurations to maximise broader objectives such as social welfare or customer satisfaction. Such algorithms could also incorporate feedback from regulators or end-users to ensure that prioritisation aligns with equity and resilience goals.

5.3 Discussion of results

This section outlines the conclusions drawn from the criticality analysis of the distribution network. By weighing social vulnerability against technical impact, the approach has allowed for the identification of nodes with the potential for the greatest disruption. Two main points are discussed: (i) the use of graph visualisations to illustrate critical dependencies, and (ii) the effect of varying weight settings on the final criticality score.

5.3.1 Network vulnerability analysis via graph analytics

5.3.1.1 Creating CriticalityScore nodes in the graph database

While the criticality score is initially calculated in Python using the procedure described in Section 5.2.4, it is important for network operators to be able to access and explore the results without needing to recalculate the metrics each time. Since the criticality score remains valid as

long as the network topology and weight parameters remain unchanged, storing it in the graph database ensures better usability and persistent access for analysts.

To this end, a new node type, `CriticalityScore`, is created for each network asset (e.g., busbars, transformers). This node stores the following attributes associated with asset failure:

- The number of LV_MSPs affected,
- The total number of households impacted,
- The total Customer Minutes of Interruption (CMI),
- The aggregated Social Vulnerability Index (SVI), and
- The unified criticality score.

This structure allows analysts to query and visually inspect network vulnerabilities directly through the graph interface, eliminating the need for external computations. Moreover, since all individual components of the criticality score are stored, it becomes possible to recalculate the final index dynamically by issuing queries with different weight parameters.

Figure 5.6 provides an example of a `CriticalityScore` node linked to a Busbar, illustrating the type of insights that can be extracted from this structure. Although this Busbar would affect a large number of customers in the event of a failure — with 553 MSPs and 636 households impacted — its overall criticality score remains relatively low (0.178).

Figure 5.6: Graph representation of an electric node (Busbar) and its associated `CriticalityScore` node. The `HAS_CRITICALITY_SCORE` relationship links the electric element to its computed impact on MSPs, including the total number affected and their unique identifiers. The criticality score shown was calculated using a weight of $\alpha = 0.7$ for technical impact and $\beta = 0.3$ for social vulnerability.

This is mainly due to two factors. First, the CMI is modest (215 minutes) compared to other components, which likely reflects a *lower probability of failure*. Second, the *social vulnerability score*

of the affected population is close to the average (0.52), indicating a moderate socioeconomic impact.

Together, these factors contribute to a lower criticality score, despite the high number of customers exposed. This example underscores the value of a *multidimensional approach*: rather than relying solely on exposure, the method balances (i) *technical impact*, (ii) *failure likelihood*, and (iii) *social vulnerability*.

5.3.1.2 Visual exploration of critical nodes in the network

To identify the elements in the distribution network whose failure would impact the largest number of customers, this section presents a visual exploration using graph analytics.

The objective is to visualise each critical node—such as Busbars or Primary Transformers—along with the LV MSPs that would lose supply in the event of their failure. This enables operators to observe *dependency clusters* around key components and identify which nodes concentrate the most risk.

5.3.1.3 Creating temporary nodes and relationships in the graph

In the current graph data model, the relationship between a critical asset and the MSPs it affects is not stored explicitly. To enable visual analysis, temporary MSP nodes are created based on the `affected_msps` attribute in each `CriticalityScore` node.

The process begins by matching each `Electric` node (e.g., a transformer or busbar) to its associated `CriticalityScore` node. From there, the list of affected MSPs is used to generate temporary MSP nodes and corresponding `CRITICALLY_AFFECTS` relationships. This ensures that each impacted MSP is visually represented in the graph, facilitating a more intuitive understanding of node dependencies (see Query 5.4).

```
MATCH (e:Electric)-[:HAS_CRITICALITY_SCORE]->(c:CriticalityScore)
UNWIND c.affected_msps AS msp_id
MERGE (msp:MSP:Temp {id: msp_id})
MERGE (e)-[r:CRITICALLY_AFFECTS]->(msp)
SET r.temp = true
```

Query 5.4: Create temporary MSP nodes and criticality relationships

5.3.1.4 Visualising critical clusters in the network

Once this temporary structure is created, analysts can execute visualisation queries to identify the most critical clusters. As shown in Table 5.5, Busbars and Primary Transformers are typically among the nodes with the highest number of dependent MSPs.

Focusing the visualisation on these asset types may reveal important structural and vulnerability patterns within the network (see Query 5.5).

```
MATCH (e:Electric)
WHERE e.node_type IN ['Primary_Transformer', 'Busbar']
MATCH (e)-[s:HAS_CRITICALITY_SCORE]->(c:CriticalityScore)
MATCH (e)-[r:CRITICALLY_AFFECTS]->(m:MSP:Temp)
WHERE r.temp = true
RETURN e, c, s, r, m
```

Query 5.5: Visualize critical MSP dependencies

Table 5.5: Ten network nodes with the highest potential impact on MSPs in case of failure.

Node Type	# MSPs Affected
Busbar	553
Busbar	545
Dynamic Protective Device	545
Primary Transformer	545
Primary Transformer	479
Dynamic Protective Device	479
Busbar	466
Switch	376
Intersection Point HV	376
Dynamic Protective Device	376

Figure 5.7 presents a high-level view of the resulting clusters. Red nodes represent Busbars, blue nodes represent Primary Transformers, and orange nodes represent MSPs. The size and density of each cluster offer an immediate visual cue of a node's criticality—the more MSPs directly linked to a component, the greater the risk posed by its failure.

Zooming into a specific area (Figure 5.8) highlights differences in the potential impact of equipment failures. On the right-hand side, the Busbar (red) is connected, via a Primary Transformer (blue), to all MSPs shown in the figure. A failure in this Busbar would therefore disrupt the entire set of MSPs, indicating very high criticality. In contrast, the Busbar on the left serves only the MSPs in its own smaller cluster, so a failure there would affect fewer customers. Before considering socio-economic factors, this suggests that the right-hand Busbar holds greater systemic importance and could be prioritised for reinforcement or redundancy ahead of the left-hand one.

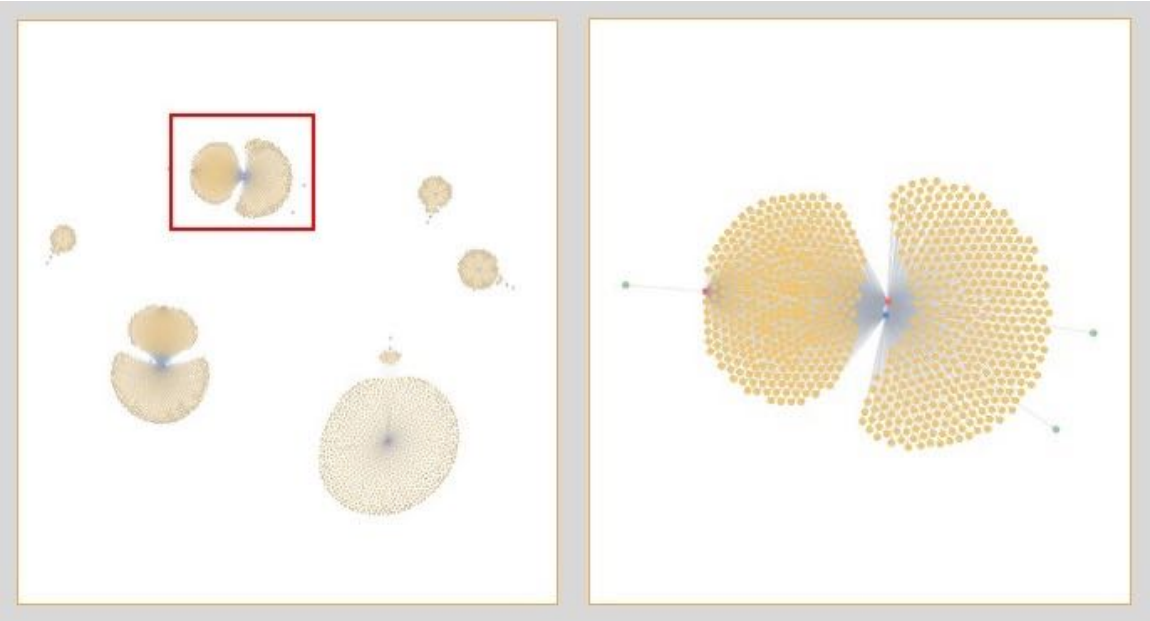


Figure 5.7: Critical Network Clusters Based on MSP Dependency. Left: Overview of clusters formed by Primary Transformers or Busbars and their affected MSPs. Right: Zoomed view of a highly critical cluster where two central nodes impact a large number of MSPs.

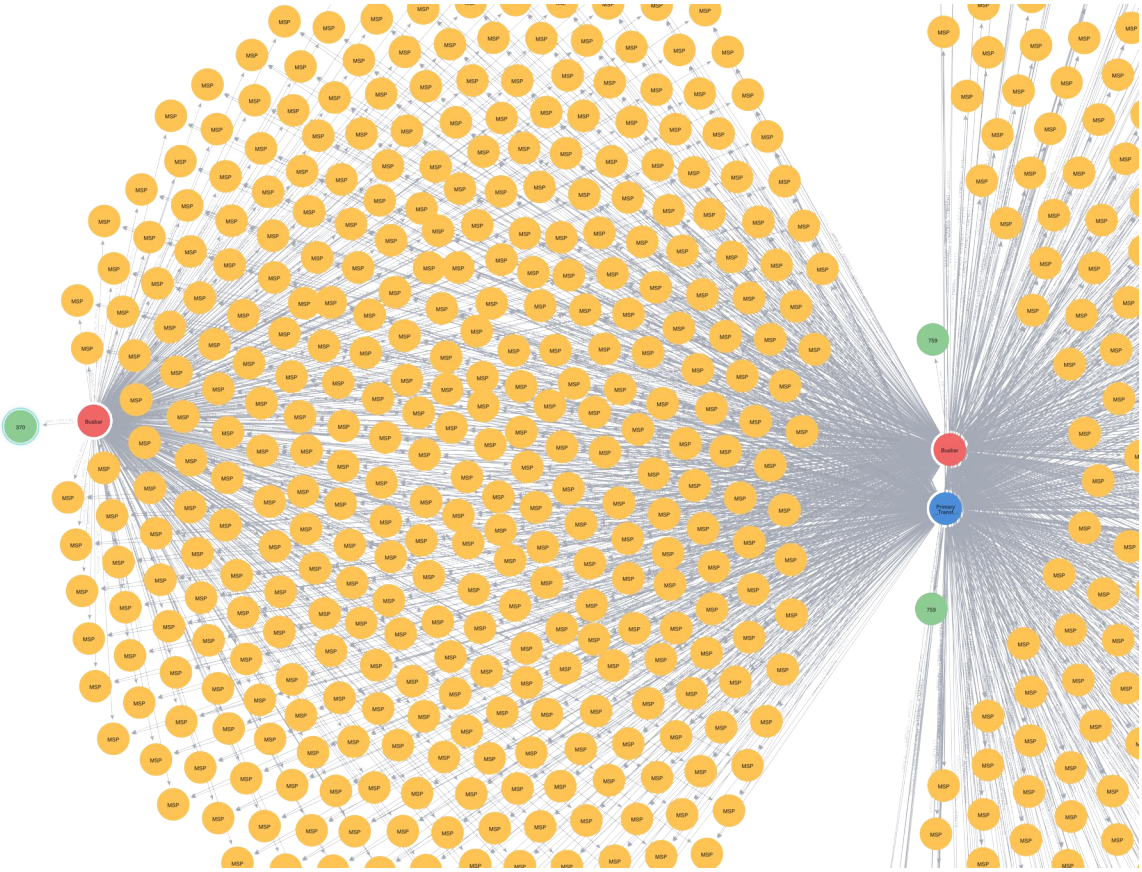


Figure 5.8: Zoomed view of MSPs linked to Busbars and a Primary Transformer. Two Busbars (red) and a Primary Transformer (blue) are shown with their dependent MSPs (yellow), illustrating the wide potential impact of their failure. Green nodes represent the number of MSPs affected by each critical component.

Beyond individual examples, this visualisation approach enables operators to scan the network and quickly identify other critical nodes—particularly those with many surrounding MSPs forming dense dependency clusters.

5.3.1.5 Removing the temporary nodes and relationships

After the analysis, all temporary MSP nodes and `CRITICALLY_AFFECTS` relationships are deleted to maintain a clean graph (see Query 5.6). This workflow enables analysts to gain advanced insights without permanently altering the network model.

```
MATCH ()-[r:CRITICALLY_AFFECTS]->(:MSP:Temp)
WHERE r.temp = true
DELETE r;

MATCH (m:MSP:Temp)
DELETE m;
```

Query 5.6: Remove temporary MSP nodes and relationships

5.3.2 Effect of weight parameters on criticality score outcomes

This section examines how different weight settings impact criticality scores by comparing two opposite scenarios:

- **Scenario A:** $\alpha = 0.7$, $\beta = 0.3$. This scenario places more importance on technical indicators like Customer Minutes of Interruption.
- **Scenario B:** $\alpha = 0.3$, $\beta = 0.7$. This scenario focuses more on the social vulnerability of the affected population.

5.3.2.1 Differences in critical asset rankings

The top five most critical nodes in each scenario highlight a clear difference in priorities. In Scenario A, where technical factors take precedence, the most critical assets are usually *Primary Transformers*. These assets have a wide customer reach and high outage durations (see Table 5.6).

Table 5.6: Top 5 most critical nodes in the network under weight parameters $\alpha = 0.7$ (technical) and $\beta = 0.3$ (social).

#	Node ID	Asset Type	Households Affected	CMI	SVI Score	Criticality Score
1	19100562	Primary Transformer	628	7545.73	0.524	0.857
2	19100534	Primary Transformer	421	5058.52	0.527	0.627
3	9051871	Dist Transformer	185	2222.87	0.750	0.431
4	9057790	Dist Transformer	34	408.53	0.971	0.329
5	9071696	Dist Transformer	62	744.96	0.847	0.323

In Scenario B, where social vulnerability carries more weight, the top-ranked nodes shift to smaller *Distribution Transformers* and *LV Joints* (see Table 5.7). Although these components serve fewer households, they are located in areas with high vulnerability scores of 1.0.

Table 5.7: Top 5 most critical nodes under weight parameters $\alpha = 0.3$ (technical) and $\beta = 0.7$ (social).

#	Node ID	Asset Type	Households Affected	CMI	SVI Score	Criticality Score
1	9021142	Dist Transformer	13	156.20	1.000	0.706
2	9055348	Dist Transformer	9	108.14	1.000	0.704
3	9057890	Dist Transformer	9	108.14	1.000	0.704
4	9060696	Dist Transformer	8	96.12	1.000	0.704
5	14449696	LV Joint	41	95.19	1.000	0.704

This contrast shows how the weighting scheme can greatly change what is considered critical. Scenario A highlights large operational risks, while Scenario B reveals components that may have less technical impact but pose a higher social risk if they fail.

5.3.2.2 Spatial distribution patterns of critical nodes

The criticality maps in Figures 5.9 and 5.10 clearly illustrate how weighting parameters affect geographic prioritization across the network.

- **Social weighting** ($\alpha = 0.3$, Figure 5.9): Criticality scores concentrate in neighborhoods with documented social vulnerability. The heatmap shows *localised red zones* that have relatively low technical impact but high social significance. These areas would typically receive lower priority in conventional engineering assessments.
- **Technical weighting** ($\alpha = 0.7$, Figure 5.10): Critical areas are more widely spread, especially around major infrastructure and high-demand zones. The most critical regions correspond with the network's primary technical components, mirroring standard utility planning priorities.

The detailed view of Lanark further emphasizes these differing patterns. With social weighting, peripheral areas become highly critical (red zones) despite serving fewer customers, due to elevated vulnerability scores. In contrast, under technical weighting, these same areas appear as low priority (green zones), showing how the choice of parameters fundamentally shifts spatial priorities.

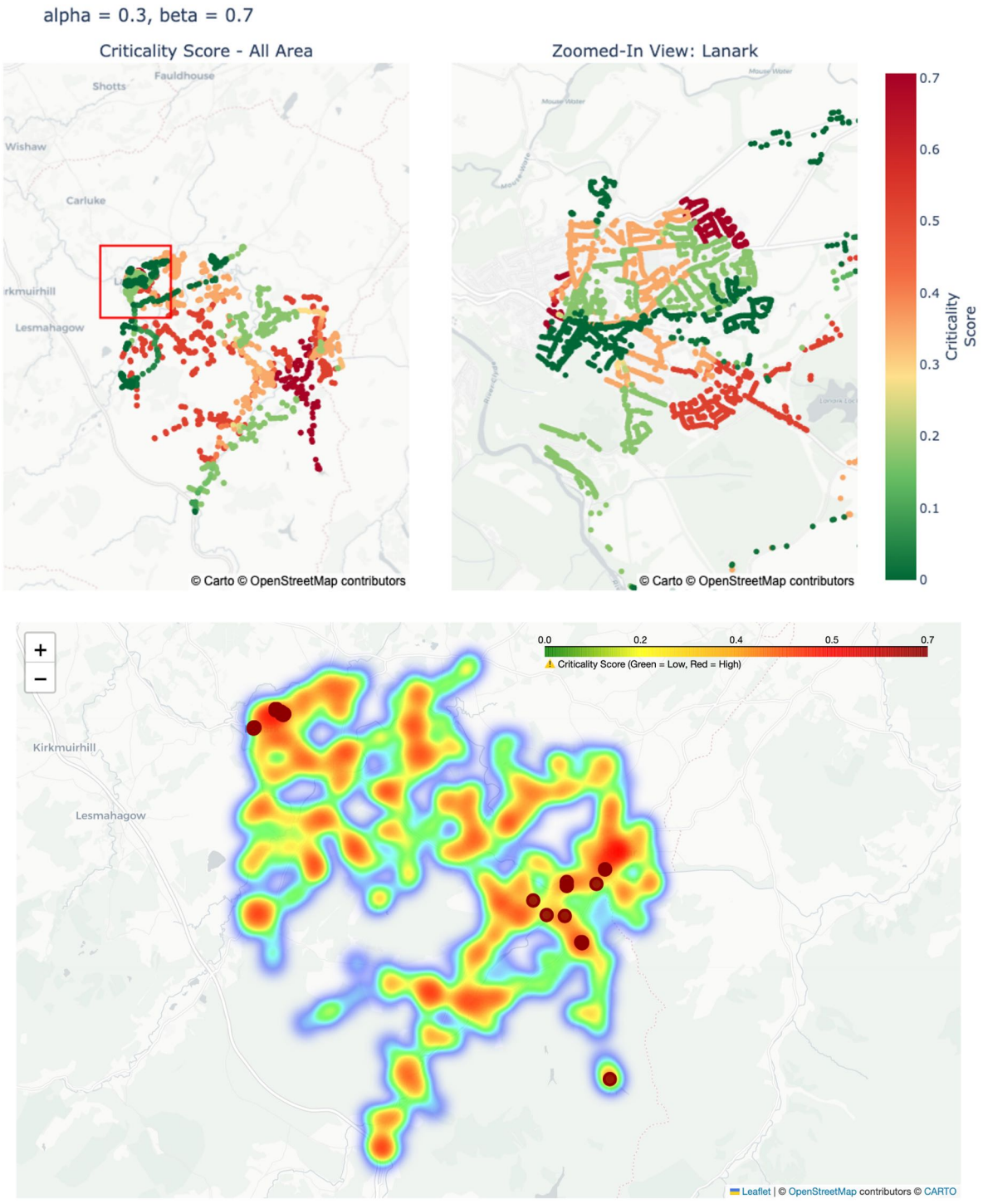


Figure 5.9: Spatial distribution of criticality scores with socially-focused weighting ($\alpha = 0.3$, $\beta = 0.7$).

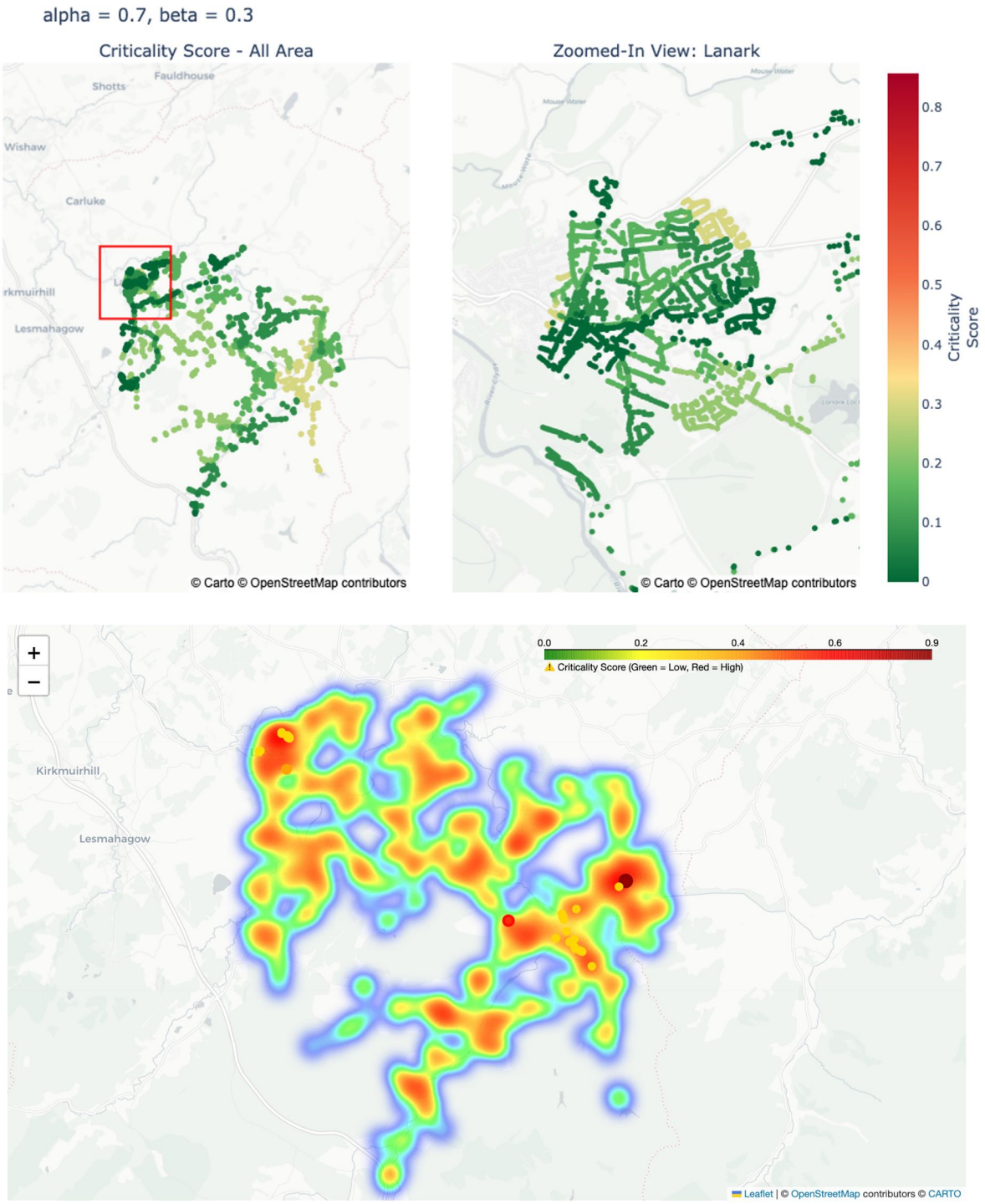


Figure 5.10: Spatial distribution of criticality scores with technical-focused weighting ($\alpha = 0.7$, $\beta = 0.3$).

5.3.2.3 *Strategic takeaways*

These findings show the importance of including both technical and social aspects in criticality assessments. A purely technical approach may miss at-risk populations. On the other hand, a solely social approach might underestimate systemic network risks. The weighting scheme thus becomes a strong tool for shaping infrastructure priorities. By adjusting α and β, decision-makers

can match the analysis with broader policy goals, whether those goals focus on operational reliability, social equity, or a mix of both.

Beyond manual tuning, optimization methods can be used to identify the most effective weighting scheme based on specific planning objectives. For example, multi-objective optimization could support:

- Maximizing network resilience while ensuring fair service across regions.
- Improving customer satisfaction by focusing on areas with a history of complaints or service disruptions.
- Unlocking regulatory incentives connected to performance metrics or the protection of vulnerable populations.
- Improving cost-effectiveness by balancing technical risk reduction with social impact.

These approaches offer a flexible and data-driven basis for connecting infrastructure decisions with various operational and policy targets.

6 Conclusions

The present thesis highlights the importance of integrating technical, spatial, and socio-economic perspectives in electricity distribution network analysis. It emphasises that the future of resilient, efficient, and fair energy systems cannot be achieved through purely technical optimisation, but instead requires a holistic view that incorporates the communities and contexts in which these networks operate. By linking diverse data sources into a unified graph-based model, this work demonstrates how infrastructure planning can be informed by both engineering performance and social equity considerations.

The study underscores the value of graph database technologies in bridging the long-standing separation between engineering and socio-economic domains. Through the development of a multi-layer property graph, the thesis establishes a framework that allows electricity network assets, geographic information, and socio-economic indicators to coexist and interact within a single analytical environment. This integrated structure enables the execution of complex cross-domain queries, the visual exploration of multi-layer relationships, and the identification of vulnerabilities that emerge when technical weaknesses overlap with social sensitivity.

A key part of this work is the development of reproducible data-processing pipelines capable of standardising and integrating heterogeneous open datasets. While the approach has been applied to publicly available data, the methodology is designed to be extensible to additional or proprietary datasets, paving the way for richer analyses. Deployment across both local Neo4j and cloud-based Amazon Neptune environments further demonstrates the portability and adaptability of the model, ensuring its applicability to different organisational and technical contexts.

The practical value of the approach is illustrated through targeted case studies, including the analysis of transformer subgraphs, the tracing of customer relationships to socio-economic profiles, the detection of vulnerable low-demand customers, and the spatial accessibility of electric vehicle charging stations in rural areas. These examples show how the model supports not only technical decision-making but also broader policy objectives such as addressing energy poverty, improving accessibility, and ensuring equitable service provision.

This work also recognises its limitations. The current model does not incorporate electrical

parameters such as capacity, phase allocation, or real-time power flows—factors that would be essential for developing a true digital twin of the network. The reliance on open datasets, while valuable for accessibility, also imposes constraints in terms of completeness, resolution, and accuracy. Furthermore, the relationships between administrative areas could be further refined (e.g., extending from Datazone to Local Authority to Region), and future testing in meshed network contexts, such as those of SP Manweb, would help assess scalability and adaptability to more complex topologies.

From a practical standpoint, the contributions of this work are relevant to both the public and private sectors. For the public sector, the integrated model offers a transparent and flexible tool for planning interventions that account for both infrastructure resilience and social fairness. Policymakers can leverage its ability to visualise and quantify overlaps between technical and socio-economic factors, leading to better-targeted investments and community-focused programmes. For the private sector, particularly distribution network operators, the model provides a means to optimise asset management, anticipate demand changes, and design interventions that strengthen both network reliability and stakeholder trust.

In conclusion, this thesis shows that graph-based modelling is an effective way to bring together and analyse the varied datasets that support modern electricity distribution networks. It offers a solid foundation for more detailed, multi-layered analysis and promotes a fairer, more context-aware approach to infrastructure planning. By allowing stakeholders to examine the links between engineering, geography, and society, the model supports the development of energy systems that are smarter, fairer, and more resilient.

Bibliography

[1] Smitha Rao et al. “Power outages and social vulnerability in the U.S. Gulf Coast: multilevel Bayesian models of outage durations amid rising extreme weather”. In: *Humanities and Social Sciences Communications* 12.1 (June 2025), p. 912. issn: 2662-9992. doi: 10.1057/s41599-025-05274-0. url: https://doi.org/10.1057/s41599-025-05274-0.

[2] George Jiglau et al. “Looking back to look forward: Reflections from networked research on energy poverty”. In: *iScience* 26.3 (2023), p. 106083. issn: 2589-0042. doi: https://doi.org/10.1016/j.isci.2023.106083. url: https://www.sciencedirect.com/science/article/pii/S2589004223001608.

[3] Bo Li et al. *Recent Decade's Power Outage Data Reveals the Increasing Vulnerability of U.S. Power Infrastructure*. 2024. arXiv: 2408.15882 [physics.soc-ph]. url: https://arxiv.org/abs/2408.15882.

[4] Jesse Dugan, Dahlia Byles, and Salman Mohagheghi. “Social vulnerability to long-duration power outages”. In: *International Journal of Disaster Risk Reduction* 85 (2023), p. 103501. issn: 2212-4209. doi: https://doi.org/10.1016/j.ijdrr.2022.103501. url: https://www.sciencedirect.com/science/article/pii/S2212420922007208.

[5] Scottish Government. *Scottish Government Urban Rural Classification 2022*. Tech. rep. Current version as of December 2024, based on Census Day 2022 data; published via gov.scot. Scottish Government, Dec. 2024. url: https://www.gov.scot/publications/scottish-government-urban-rural-classification-2022/.

[6] Initiative for Energy Justice. *Section1– Defining Energy Justice: Connections to Environmental Justice, Climate Justice, and the Just Transition*. https://iejusa.org/section-1-defining-energy-justice/. Accessed 28 July 2025. 2019.

[7] Benjamin K. Sovacool and Michael H. Dworkin. “Energy justice: Conceptual insights and practical applications”. In: *Applied Energy* 142 (2015), pp. 435–444. issn: 0306-2619. doi: https://doi.org/10.1016/j.apenergy.2015.01.002. url: https://www.sciencedirect.com/science/article/pii/S0306261915000082.

[8] Kirsten Jenkins et al. “Energy justice: A conceptual review”. In: *Energy Research Social Science* 11 (2016), pp. 174–182. issn: 2214-6296. doi: https://doi.org/10.1016/j.erss.2015.10.004. url: https://www.sciencedirect.com/science/article/pii/S2214629615300669.

[9] Idowu Ajibade et al. “Who bears the burden? An assessment of vulnerability and resilience to consecutive disasters in the Portland metro region”. In: *Environmental Research Letters* 20.8 (2025), p. 084006. doi: 10.1088/1748-9326/ade459.

[10] Ofgem (Office of Gas and Electricity Markets). *Equality and diversity*. Web page. Accessed July 28, 2025. 2025. url: https://www.ofgem.gov.uk/about-us/working-ofgem/equality-and-diversity.

[11] Ofgem (Office of Gas and Electricity Markets). *Join your supplier's Priority Services Register*. Web page. Accessed July29,2025. 2025. url: https://www.ofgem.gov.uk/join-your-suppliers-priority-services-register.

[12] U.S. Department of Energy. *2023 Equity Action Plan and Update to 2022 Plan*. Equity Action Plan pursuant to Executive Order 14091. Performance.gov publication, update issued January 5, 2024. U.S. Department of Energy, 2024. url: https://assets.performance.gov/cx/equity-action-plans/2023/EO_14091_DOE_EAP_2023.pdf.

[13] Government of Spain / Ministry for the Ecological Transition and Demographic Challenge. *Integrated National Energy and Climate Plan (INECP) 2021–2030*. Public official document. English language version. 2020. url: https://energy.ec.europa.eu/system/files/2020-06/es_final_necp_main_en_0.pdf.

[14] Office for National Statistics. *Census*. Web page. Accessed July29,2025. 2025. url: https://www.ons.gov.uk/census.

[15] Geographic Data Service (GeoDS). *Index of Multiple Deprivation (IMD) datasets in the UK*. Web dataset collection. Aggregated IMD data for England, Wales, Scotland, and Northern Ireland under UK Open Government Licence. 2025. url: https://data.geods.ac.uk/dataset/index-of-multiple-deprivation-imd.

[16] Ofgem (Office of Gas and Electricity Markets). *RIIO-2 Electricity Distribution Annual Report 2023 to2024*. Annual regulatory report under RIIO-ED2. Published7April2025; covers April2023–March2024 under ED2 price control. Ofgem, Apr. 2025. url: https://www.ofgem.gov.uk/sites/default/files/2025-04/RIIO-2%20Electricity%20Distribution%20Annual%20Report%202023%20to%202024.pdf.

[17] Ofgem (Office of Gas and Electricity Markets). *Debt and arrears indicators*. Web page. Accessed July29,2025; Data as of 27June2025. 2025. url: https://www.ofgem.gov.uk/data/debt-and-arrears-indicators.

[18] Martin Fleischmann and Daniel Arribas-Bel. “Geographical characterisation of British urban form and function using the spatial signatures framework”. In: *Scientific Data* 9.1 (Sept. 2022), p. 546. issn: 2052-4463. doi: 10.1038/s41597-022-01640-8. url: https://doi.org/10.1038/s41597-022-01640-8.

[19] Centers for Disease Control and Prevention and Agency for Toxic Substances and Disease Registry. *Social Vulnerability Index (SVI)*. Web page (Place and Health, Geospatial Research, Analysis Services Program). Accessed July29,2025. 2025. url: https://www.atsdr.cdc.gov/place-health/php/svi/index.html.

[20] International Energy Agency (IEA). *Access to electricity*. Web page (SDG7: Data and Projections report series). Accessed July 29, 2025. 2024. url: https://www.iea.org/reports/sdg7-data-and-projections/access-to-electricity.

[21] "IEEE Guide for Electric Power Distribution Reliability Indices". In: *IEEE Std 1366-2022 (Revision of IEEE Std 1366-2012)* (2022), pp. 1–44. doi: 10.1109/IEEESTD.2022.9955492.

[22] Office of Gas and Electricity Markets (Ofgem). *RIIO-ED2 – Annex F: Interruptions (v1.1)*. Regulatory Instructions and Guidance Annex v1.1. Updated direction issued February 1 2024 effective from March 1 2024. Ofgem, Oct. 2023. url: https://www.ofgem.gov.uk/sites/default/files/2023-10/RIIO-ED2%20-%20Annex%20F%20Interruptions%20v1.1.pdf.

[23] Ofgem – The Office of Gas and Electricity Markets. *RIIO-ED2 Regulatory Instructions and Guidance – Annex A: Glossary (v1.1)*. Tech. rep. © Crown copyright 2025; RIGs document reference OFG1163; accessed 2025-07-29. London, UK: Ofgem, Feb. 2025. url: https://www.ofgem.gov.uk/sites/default/files/2025-02/RIIO-ED2%20-%20Annex%20A%20Glossary%20v1.1.pdf.

[24] Barry Flanagan et al. "A Social Vulnerability Index for Disaster Management". In: *Journal of Homeland Security and Emergency Management* 8 (Jan. 2011). doi: 10.2202/1547-7355.1792.

[25] Barış Bilir et al. "Enhancing power grid resilience to winter storms via generator winterization with equity considerations". In: *Sustainable Cities and Society* 114 (2024), p. 105736. issn: 2210-6707. doi: https://doi.org/10.1016/j.scs.2024.105736. url: https://www.sciencedirect.com/science/article/pii/S2210670724005614.

[26] Juan P. Montoya-Rincon et al. "A socio-technical approach for the assessment of critical infrastructure system vulnerability in extreme weather events". In: *Nature Energy* 8.9 (Sept. 2023), pp. 1002–1012. issn: 2058-7546. doi: 10.1038/s41560-023-01315-7. url: https://doi.org/10.1038/s41560-023-01315-7.

[27] Farzane Ezzati et al. "Power outage-risk integrated social vulnerability analysis highlights disparities in small residential communities". In: *Communications Earth Environment* 6 (Apr. 2025). doi: 10.1038/s43247-025-02278-1.

[28] Centre for Sustainable Energy. *NGED Horizon Scanning 2023*. Tech. rep. Accessed: 2025-07-29. National Grid Electricity Distribution, Dec. 2023. url: https://www.nationalgrid.co.uk/downloads-view-reciteme/653130.

[29] Reinhard Diestel. *Graph Theory*. 6th ed. Vol. 173. Graduate Texts in Mathematics. Springer, Berlin, Heidelberg, 2025. isbn: 978-3-662-70106-5. doi: 10.1007/978-3-662-70107-2.

[30] Åke J. Holmgren. "Using Graph Models to Analyze the Vulnerability of Electric Power Networks". In: *Risk Analysis* 26.4 (2006), pp. 955–969. doi: https://doi.org/10.1111/j.1539-6924.2006.00791.x. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1539-6924.2006.00791.x. url: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1539-6924.2006.00791.x.

[31] Ettore Bompard, Roberto Napoli, and Fei Xue. “Analysis of structural vulnerabilities in power transmission grids”. In: *International Journal of Critical Infrastructure Protection* 2.1 (2009), pp. 5–12. issn: 1874-5482. doi: https://doi.org/10.1016/j.ijcip.2009.02.002. url: https://www.sciencedirect.com/science/article/pii/S1874548209000031.

[32] P. Hines et al. “The Topological and Electrical Structure of Power Grids”. In: *2010 43rd Hawaii International Conference on System Sciences*. 2010, pp. 1–10. doi: 10.1109/HICSS.2010.398.

[33] Enrico Zio and Giovanni Sansavini. “Modeling Interdependent Network Systems for Identifying Cascade-Safe Operating Margins”. In: *IEEE Transactions on Reliability* 60.1 (2011), pp. 94–101. doi: 10.1109/TR.2010.2104211.

[34] Subhonmesh Bose et al. “Equivalent Relaxations of Optimal Power Flow”. In: *IEEE Transactions on Automatic Control* 60.3 (2015), pp. 729–742. doi: 10.1109/TAC.2014.2357112.

[35] Damian Owerko, Fernando Gama, and Alejandro Ribeiro. “Optimal Power Flow Using Graph Neural Networks”. In: *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2020, pp. 5930–5934. doi: 10.1109/ICASSP40776.2020.9053140.

[36] Damian Owerko, Fernando Gama, and Alejandro Ribeiro. “Unsupervised Optimal Power Flow Using Graph Neural Networks”. In: *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2024, pp. 6885–6889. doi: 10.1109/ICASSP48485.2024.10446827.

[37] Yachen Tang et al. “Enhancement of Power Equipment Management Using Knowledge Graph”. In: *2019 IEEE Innovative Smart Grid Technologies - Asia (ISGT Asia)*. 2019, pp. 905–910. doi: 10.1109/ISGT-Asia.2019.8881348.

[38] Shuai Huang, Qian Hu, and Yubao Li. “Optimization and Development Methods for Data Asset Management in Power Systems Based on Knowledge Graphs”. In: *2024 5th International Symposium on New Energy and Electrical Technology (ISNEET)*. 2024, pp. 411–416. doi: 10.1109/ISNEET64164.2024.10956034.

[39] Jesús Barrasa and Jim Webber. *Building Knowledge Graphs: A Practitioner’s Guide*. Sebastopol, CA: O’Reilly Media, 2023. isbn: 9781098127107.

[40] Carlos Ramonell, Rolando Chacón, and Hector Posada. “Knowledge graph-based data integration system for digital twins of built assets”. In: *Automation in Construction* 156 (Oct. 2023). doi: 10.1016/j.autcon.2023.105109.

[41] Guus Schreiber and Yves Raimond. *RDF 1.1 Primer*. W3C Working Group Note. June 2014. url: https://www.w3.org/TR/rdf11-primer/.

[42] Neo4j, Inc. *Getting Started with Cypher*. Accessed: 31 July 2025. 2025. url: https://neo4j.com/docs/getting-started/cypher/.

[43] Apache TinkerPop Project. *Gremlin – The Apache TinkerPop Graph Traversal Language*. Accessed: 31 July 2025. 2025. url: https://tinkerpop.apache.org/gremlin.html.

[44] TigerGraph, Inc. *GSQL: Graph Query Language*. Accessed: 31 July 2025. 2025. url: https://www.tigergraph.com/gsql/.

[45] Neo4j, Inc. *GQL Conformance Appendix*. Accessed: 31 July 2025. 2025. url: https://neo4j.com/docs/cypher-manual/current/appendix/gql-conformance/.

[46] Jian Wang et al. "A survey on the development status and application prospects of knowledge graph in smart grids". In: *IET Generation, Transmission Distribution* 15 (3 2021), pp. 383–407. doi: 10.1049/gtd2.12040. eprint: https://digital-library.theiet.org/doi/pdf/10.1049/gtd2.12040. url: https://digital-library.theiet.org/doi/abs/10.1049/gtd2.12040.

[47] Han Ding et al. "A Review of the Construction and Application of Knowledge Graphs in Smart Grid". In: *2021 IEEE Sustainable Power and Energy Conference (iSPEC)*. 2021, pp. 3770–3775. doi: 10.1109/iSPEC53008.2021.9736038.

[48] Haichao Huang et al. "Constructing Knowledge Graph from Big Data of Smart Grids". In: *2019 10th International Conference on Information Technology in Medicine and Education (ITME)*. 2019, pp. 637–641. doi: 10.1109/ITME.2019.00147.

[49] Neo4j Inc. *Download Center*. https://neo4j.com/download/. Accessed: 2025-08-08. 2025.

[50] Neo4j, Inc. *Neo4j Licensing*. Neo4j. url: https://neo4j.com/licensing/ (visited on 08/10/2025).

[51] *What Is Amazon Neptune?* Amazon Neptune User Guide. Amazon Web Services. url: https://docs.aws.amazon.com/neptune/latest/userguide/intro.html (visited on 08/10/2025).

[52] Amazon Web Services, Inc. *Graph Explorer: Getting Started Guide (README)*. https://github.com/aws/graph-explorer/blob/main/additionaldocs/getting-started/README.md. Accessed: 2025-08-08. 2025.

[53] Scottish Government. *Scottish Index of Multiple Deprivation 2020: Introduction*. Online booklet. Scottish Government, Jan. 2020. url: https://www.gov.scot/publications/scottish-index-multiple-deprivation-2020/.

[54] Scottish Government / Transport Scotland. *ChargePlace Scotland*. https://chargeplacescotland.org. Accessed 7 August 2025. 2025.

[55] Ordnance Survey. *OS Open Roads*. https://www.ordnancesurvey.co.uk/products/os-open-roads. Data updated every six months; accessed 7 August 2025. 2025.

[56] Ideal Postcodes. *Postcodes.io: UK Postcode Geolocation API (Overview)*. https://postcodes.io/docs/overview/. Accessed 7 August 2025. 2025.

[57] Ordnance Survey. *Code-Point with Polygons*. https://www.ordnancesurvey.co.uk/products/code-point-polygons. Quarterly updates; accessed 7 August 2025. 2025.

[58] Mark. *Open Data GB Postcode Unit Boundaries*. https://longair.net/blog/2021/08/23/open-data-gb-postcode-unit-boundaries/. Accessed 7August2025. Aug. 2021.

[59] Ordnance Survey. *Code-Point Open*. https://www.ordnancesurvey.co.uk/products/code-point-open. Updated quarterly (February, May, August, November); accessed 7 August 2025. 2025.

[60] Mark A. Musen. "The Protégé Project: A Look Back and a Look Forward". In: *AI Matters* 1.4 (June 2015), pp. 4–12. doi: 10.1145/2757001.2757003. url: https://doi.org/10.1145/2757001.2757003.

[61] Amazon Web Services. *Load format for openCypher data*. https://docs.aws.amazon.com/neptune/latest/userguide/bulk-load-tutorial-format-opencypher.html. Amazon Neptune User Guide – accessed 8 Aug 2025. 2025.

[62] Amazon Web Services. *Using the Amazon Neptune bulk loader to ingest data*. Amazon Neptune User Guide. Amazon Web Services, Inc. url: https://docs.aws.amazon.com/neptune/latest/userguide/bulk-load.html (visited on 08/12/2025).

[63] Docker Inc. *Docker: Empowering App Development for Developers*. https://www.docker.com. Accessed: 2025-08-08. 2025.

[64] Neo4j Inc. *Getting Started with Neo4j in Docker*. https://neo4j.com/docs/operations-manual/current/docker/introduction/. Accessed: 2025-08-08. 2025.

[65] Neo4j Inc. *APOC Documentation: Awesome Procedures On Cypher*. https://neo4j.com/labs/apoc/. Accessed: 2025-08-08. 2025.

[66] Neo4j Inc. *Neo4j Bloom*. https://neo4j.com/product/bloom. Accessed: 2025-08-08. 2025.

[67] Amazon Web Services, Inc. *Visualizing Graphs with Neptune Graph Explorer*. https://docs.aws.amazon.com/neptune/latest/userguide/visualization-graph-explorer.html. Accessed: 2025-08-08. 2025.

[68] Louise Horscroft. *How much electricity does the average home use per day in the UK?* AquaSwitch blog post; accessed 2025-08-09. 2023. url: https://www.aquaswitch.co.uk/blog/average-electricity-usage/.

[69] Adam X Andresen et al. "Understanding the social impacts of power outages in North America: a systematic review". In: *Environmental Research Letters* 18.5 (May 2023), p. 053004. doi: 10.1088/1748-9326/acc7b9. url: https://dx.doi.org/10.1088/1748-9326/acc7b9.

[70] Momhammad Ali Saniee Monfared, Mahdi Jalili, and Zohreh Alipour. "Topology and vulnerability of the Iranian power grid". In: *Physica A: Statistical Mechanics and its Applications* 406 (2014), pp. 24–33. issn: 0378-4371. doi: https://doi.org/10.1016/j.physa.2014.03.031. url: https://www.sciencedirect.com/science/article/pii/S037843711400226X.

[71] Réka Albert, István Albert, and Gary L. Nakarado. "Structural vulnerability of the North American power grid". In: *Physical Review E* 69.2 (Feb. 2004). issn: 1550-2376. doi: 10.1103/physreve.69.025103. url: http://dx.doi.org/10.1103/PhysRevE.69.025103.

[72] Paolo Crucitti, Vito Latora, and Massimo Marchiori. “A topological analysis of the Italian electric power grid”. In: *Physica A: Statistical Mechanics and its Applications* 338.1 (2004). Proceedings of the conference A Nonlinear World: the Real World, 2nd International Conference on Frontier Science, pp. 92–97. issn: 0378-4371. doi: https://doi.org/10.1016/j.physa.2004.02.029. url: https://www.sciencedirect.com/science/article/pii/S0378437104002249.

[73] Banghua Xie et al. “The Vulnerability of the Power Grid Structure: A System Analysis Based on Complex Network Theory”. In: *Sensors* 21.21 (2021). issn: 1424-8220. doi: 10.3390/s21217097. url: https://www.mdpi.com/1424-8220/21/21/7097.

[74] Alexys H Rodríguez A, Abdollah Shafieezadeh, and Alper Yilmaz. “System outage fragility for power systems: A robust data-driven framework for disparity analysis using multiple hurricane events”. In: *International Journal of Disaster Risk Reduction* 118 (2025), p. 105240. issn: 2212-4209. doi: https://doi.org/10.1016/j.ijdrr.2025.105240. url: https://www.sciencedirect.com/science/article/pii/S2212420925000640.

[75] SP Distribution plc. *LongTerm Development Statement*. Technical Report. Glasgow, United Kingdom: SP Energy Networks, Nov. 2024. url: https://www.spenergynetworks.co.uk/userfiles/file/SPD-Long-Term-Development-Statement-Nov-2024.pdf.

[76] SP Manweb (SP Energy Networks). *Network Development Plan – Parts 1 & 2: Capacity and Development Report*. Technical Report. Glasgow, United Kingdom: SP Energy Networks, 2024. url: https://www.spenergynetworks.co.uk/userfiles/file/SPM%20NDP%20-%20Parts%201%20and%202%20-%20Capacity%20and%20Development%20Report%20v1.pdf.

[77] Electricity North West. *Unplanned Outages*. https://electricitynorthwest.opendatasoft.com. Accessed: 2025-07-25. 2025. url: https://electricitynorthwest.opendatasoft.com/explore/dataset/unplanned-outages/information/?disjunctive.main_equipment_involved_1&disjunctive.direct_cause_category&disjunctive.district_name&disjunctive.primary_substation&disjunctive.voltage&disjunctive.incident_reporting_year.

[78] Electricity North West. *Section 2: Company Overview*. Business Plan Annex. Accessed: 2025-07-25. Electricity North West, 2023. url: https://www.enwl.co.uk/globalassets/about-us/regulatory-information/documents/business-plan-annexes/individual-sections/section2-companyoverview.pdf.

[79] Electricity North West. *UK CMA Clears Iberdrola's Acquisition of Electricity North West*. Accessed: 2025-07-25. Electricity North West Newsroom. 2024. url: https://news.enwl.co.uk/news/uk-cma-clears-iberdrolas-acquisition-of-electricity-north-west.

[80] Shuai YANG et al. “Failure probability estimation of overhead transmission lines considering the spatial and temporal variation in severe weather”. In: *Journal of Modern Power*

Systems and Clean Energy 7.1 (Jan. 2019), pp. 131–138. issn: 2196-5420. doi: 10.1007/s40565-017-0370-4. url: https://doi.org/10.1007/s40565-017-0370-4.

[81] Y. Zhou, A. Pahwa, and S.-S. Yang. "Modeling Weather-Related Failures of Overhead Distribution Lines". In: *IEEE Transactions on Power Systems* 21.4 (2006), pp. 1683–1690. doi: 10.1109/TPWRS.2006.881131.

[82] Abdurrahman Ünsal and Mumyakmaz Serdar. "PREDICTING THE FAILURES OF TRANSFORMERS IN A POWER SYSTEM USING THE POISSON DISTRIBUTION: A CASE STUDY". In: (Dec. 2005).

[83] O. Alizadeh Mousavi, R. Cherkaoui, and M. Bozorg. "Blackouts risk evaluation by Monte Carlo Simulation regarding cascading outages and system frequency deviation". In: *Electric Power Systems Research* 89 (2012), pp. 157–164. issn: 0378-7796. doi: https://doi.org/10.1016/j.epsr.2012.03.004. url: https://www.sciencedirect.com/science/article/pii/S0378779612000727.

[84] Office of Gas and Electricity Markets (Ofgem). *RIIO-ED1 Regulatory Instructions and Guidance: Annex F – Interruptions*. Regulatory guidance, v1.0. Ofgem, June 2015. url: https://www.ofgem.gov.uk/sites/default/files/docs/2020/04/riio-ed1_regulatory_instructions_and_guidance_annex_f_-_interruptions.pdf.

[85] Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. "Exploring Network Structure, Dynamics, and Function using NetworkX". In: *Proceedings of the 7th Python in Science Conference*. Ed. by Gaël Varoquaux, Travis Vaught, and Jarrod Millman. Pasadena, CA USA, 2008, pp. 11–15.

[86] J. Blank and K. Deb. "pymoo: Multi-Objective Optimization in Python". In: *IEEE Access* 8 (2020), pp. 89497–89509.

[87] Lisa Rygel, David O'sullivan, and Brent Yarnal. "A Method for Constructing a Social Vulnerability Index: An Application to Hurricane Storm Surges in a Developed Country". In: *Mitigation and Adaptation Strategies for Global Change* 11.3 (May 2006), pp. 741–764. issn: 1573-1596. doi: 10.1007/s11027-006-0265-6. url: https://doi.org/10.1007/s11027-006-0265-6.

[88] PhD Loukas Serafeim. "Everything you need to know about Min-Max normalization: A Python tutorial". In: *Towards Data Science (TDS Archive)* (2020). May 28, 2020; last published Feb 3, 2025. url: https://towardsdatascience.com/everything-you-need-to-know-about-min-max-normalization-in-python-b79592732b79/.

A Alignment with the United Nations Sustainable Development Goals

This project supports several of the United Nations Sustainable Development Goals (SDGs), particularly in the areas of energy access, sustainable infrastructure, and reducing inequality. By improving how distribution network operators (DNOs) and distribution utilities plan and manage their low- and medium-voltage networks—especially in areas facing social vulnerability—the project helps translate broad sustainability principles into concrete, local action.

A.1 SDG 7: Affordable and Clean Energy

By integrating data on social deprivation and public infrastructure into planning decisions, the approach enables DNOs to identify and prioritise support for households at risk of energy poverty. This promotes more equitable access to reliable and affordable electricity, while helping optimise network upgrades to serve those with the greatest need. The result is a smarter, fairer distribution system that advances the clean-energy transition without leaving vulnerable communities behind.

A.2 SDG 9: Industry, Innovation and Infrastructure

Using a graph-based model improves how infrastructure data are linked and shared across teams. It supports innovation in how DNOs understand their assets, respond to faults, and plan for future scenarios such as electric-vehicle uptake or climate-related stresses. The model encourages resilient, inclusive, and data-driven infrastructure management.

A.3 SDG 10: Reduced Inequalities

By explicitly including deprivation and vulnerability indicators in network planning, the method enables targeted interventions in underserved areas. This helps reduce disparities in service quality and resilience, ensuring that investment and operational decisions benefit those who need them most.

A.4 SDG 11: Sustainable Cities and Communities

By helping DNOs identify underserved neighbourhoods and align investment with local needs, the approach contributes to stronger, more resilient communities. It supports collaboration with local authorities and public stakeholders to address gaps in infrastructure and services, improve living conditions, and reduce energy-related hardship. The model's insights can also inform sustainable urban-development strategies as cities plan for electrification, decarbonisation, and climate resilience.