

**Zeitschrift:** Geomatik Schweiz : Geoinformation und Landmanagement =  
Géomatique Suisse : géoinformation et gestion du territoire =  
Geomatica Svizzera : geoinformazione e gestione del territorio

**Herausgeber:** geosuisse : Schweizerischer Verband für Geomatik und  
Landmanagement

**Band:** 118 (2020)

**Heft:** 12

**Artikel:** Integration von Geodaten in eine Graphdatenbank und Erstellung einer  
schematischen Darstellung = Intégration de géodonnées dans une BD  
graphe et génération d'une visualisation schématique = Integrazione dei  
geodati in una database a grafo e generazione ...

**Autor:** Vogt, Patrick

**DOI:** <https://doi.org/10.5169/seals-905967>

### **Nutzungsbedingungen**

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. [Siehe Rechtliche Hinweise.](#)

### **Conditions d'utilisation**

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. [Voir Informations légales.](#)

### **Terms of use**

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. [See Legal notice.](#)

**Download PDF:** 05.01.2025

**ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>**

# Integration von Geodaten in eine Graphdatenbank und Erstellung einer schematischen Darstellung

Der Betrieb der Komponenten des Bieler Stromnetzes ist, wie bei jedem anderen Stromnetz auch, sehr komplex. Er erfordert die Mitwirkung zahlreicher auf verschiedene Gebiete spezialisierter Akteure. Die Mitarbeiter von Energie Service Biel/Bienne müssen auf aktuelle, vollständige und genaue Unterlagen zurückgreifen können, damit sie ihre täglichen Aufgaben effizient ausführen können. Der Werkplan bildet mit seinem hohen Detaillierungsgrad das wichtigste Arbeitsinstrument für die Festlegung des Standorts der jeweiligen Objekte. Die Mitarbeiter brauchen jedoch auch eine schematische Darstellung, die das Netzkonzept und die Beziehungen zwischen den Komponenten wiedergibt. Ausserdem gibt diese Darstellung auch einen Überblick über die Stellen, an denen die Kabel nicht übereinanderliegen und als einzige Darstellungslinie abgebildet sind. Dank der schematischen Darstellung sollte es auch möglich sein, im Falle einer Betriebsstörung schnell und effizient einzugreifen, den Ersatz bestimmter Kabel zu planen oder einen neuen Netzabschnitt auszulegen.

P. Vogt

## 1. Fragestellung und Ziele

Die gegenwärtige schematische Darstellung des Niederspannungsnetzes muss ersetzt werden. Die Informationsdichte ist hier zu hoch, insbesondere im Bereich des Stadtzentrums. Es muss folglich eine Alternative für die visuelle Darstellung gefunden werden. Gleichzeitig gilt es, die Bearbeitungszeiten für das zukünftige Produkt auf ein Minimum zu verkürzen.

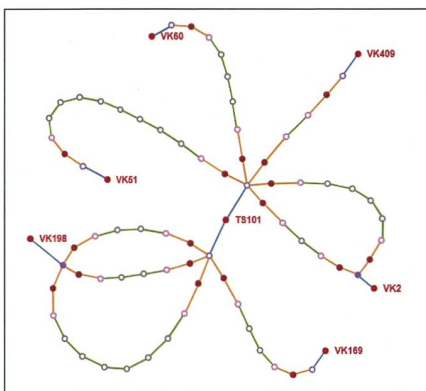


Abb. 1.

Ziele:

- Schaffung eines Visualisierungstools mit einfachem und effizientem Zugang zu den Daten.
- Erstellung einer schematischen Darstellung, die es ermöglicht, das Netzkonzept und die Netzverbindungstechnik schnell zu erfassen.

- Verwendung der Graphdatenbank Neo4j.
- Integration der Simulationsfunktionen der JavaScript-Bibliothek D3, um die Häufung von Kabeln zu vermeiden.

## 2. Umsetzung

### 2.1 Prüfung der Graphdatenbanken

Die Graphdatenbank Neo4j bietet verschiedene Vorteile. Sie bietet die optimale Eingliederung der linearen Topologien unserer unterirdischen Netze dank seiner Knoten- und Beziehungsverwaltung. Die physikalische Modellierung der Beziehungen ermöglicht es, die Joinsmechanismen bei komplexen Abfragen unberücksichtigt zu lassen.

Ausserdem erfährt der Modellierungsschritt durch die Verbindung zur realen Welt und zur verwendeten Sprache eine Flexibilisierung und Vereinfachung. Selbst nach der Implementierung der Struktur und der Dateneingabe lässt sich das Datenbankschema ganz einfach weiterentwickeln.

### 2.2 Übertragung der Daten vom RDBMS auf das GDBMS

Für den Datenimport in Neo4j werden in der Regel CSV-Dateien verwendet. Das ist das einfachste Verfahren, um schnell eine Betriebsdatenbank aufzubereiten.

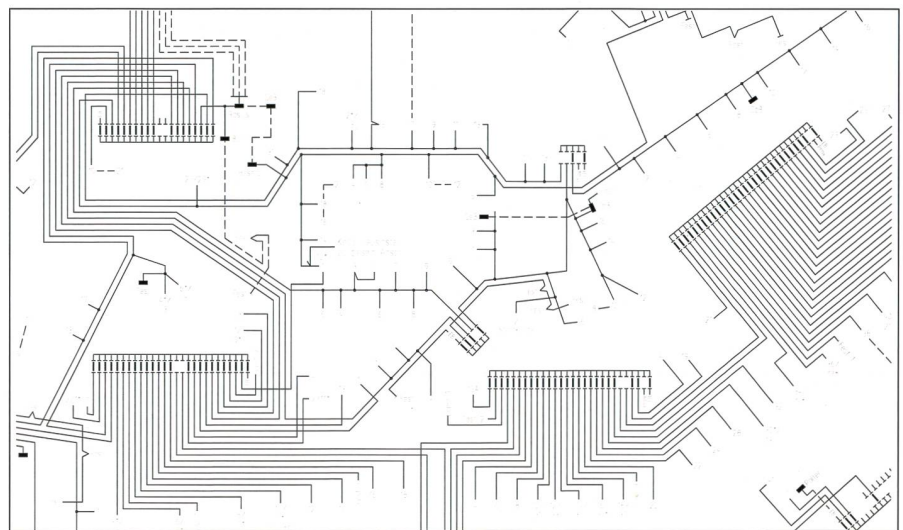


Abb. 2: Auszug aus der schematischen Darstellung des Niederspannungsnetzes.  
 Fig. 2: Extrait de la représentation schématique du réseau basse tension.  
 Fig. 2: Estratto della rappresentazione schematica della rete di bassa tensione.

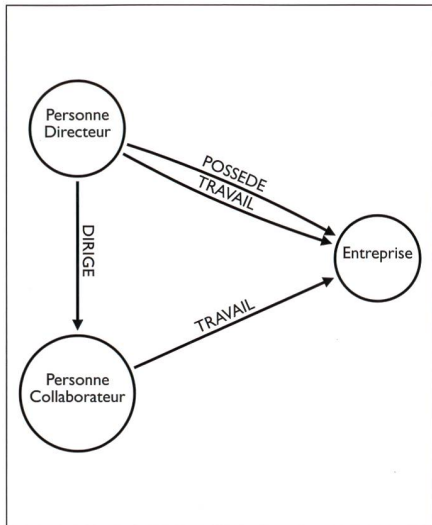


Abb. 3: Beispiel einer Graphmodellierung. Mit Arrow Tool selbst erstellte Abbildung (<http://www.apcjones.com/arrows/>).

Fig. 3: Exemple de modélisation graphe. Illustration personnelle réalisée avec l'outil Arrow Tool (<http://www.apcjones.com/arrows/>)

Fig. 3: Esempio di modellizzazione a grafo. Illustrazione personale realizzata con Arrow Tool (<http://www.apcjones.com/arrows/>).

Um den Aufwand für die Datenkonsolidierung auf ein Minimum zu reduzieren, mussten zwei Schritte direkt in der relationalen Datenbank durchgeführt werden. Zunächst musste eine Funktion eingerichtet werden, die alle notwendigen Attribute auf iterative Weise übernimmt. Anschliessend waren zwei Ansichten notwendig, die diese Funktion verwenden und deren Ergebnisse die Mittel für die Bearbeitung in FME liefern. In der Arbeitsoberfläche FME findet man eingangs die beiden Ansichten und die Tabelle, die die expliziten Beziehungen des Stromkabelnetzes modelliert. Die Idee ist hier, zwei Verarbeitungskanäle zu schaffen. Der erste Kanal speichert die Daten der verschiedenen Quellen im CSV-Format. Der zweite bereitet die Metadaten vor und integriert auch die den Importbefehlen entsprechende Syntax.

## 2.3 Modellierung der Daten in Neo4j

Im Bereich der relationalen Datenbanken

ist es eher unüblich, Mehrfachverknüpfungen zwischen zwei Tabellen oder für Objekte auf verschiedenen Ebenen zu erstellen. Bei den Graphdatenbanken sieht es anders aus. Nach dem Import der Knoten und der Modellierung der initialen Beziehungen wurden weitere zusätzliche Verbindungen zum Modell hinzugefügt, um die Abfrage und die Datenbereitstellung zu erleichtern.

Ab diesem Zeitpunkt erfolgte die Entwicklung des Modells in Neo4j parallel zur Umsetzung und Einrichtung der Visualisierungsoberfläche. Diese Schritte waren eine grosse Herausforderung. Einerseits ging es um die Festlegung der Attribute, die verwendet werden, um die Suchfunktion zu füllen. Andererseits musste die Abfrage definiert werden, die die Formatierung des Ergebnisses ermöglicht und dieses direkt in der Visualisierungsbibliothek nutzbar macht.

## 2.4 Wahl der Visualisierungsbibliothek

Obwohl die Graphdatenbank Neo4j ein in einem Webbrowser verwendbares Visualisierungstool umfasst, beschränkt sich ihre Verwendung auf Personen, die über die entsprechenden Kenntnisse verfügen, um die Oberfläche zu bedienen und Abfragen mit Cypher (SQL der GDBMS) zu erstellen. Zum Glück gibt es effiziente Alternativen für den unternehmensgerechten Einsatz und insbesondere für Applikationen, die für die verschiedenen Nutzergruppen besser zugänglich sind. Im Rahmen dieses Projektes hat sich meine Wahl auf eine Anwendungsarchitektur gerichtet, die die JavaScript-Bibliothek D3 miteinbezieht. Sie integriert das D3-Force-Modul, das eine grosse Flexibilität sowohl bei den visuellen als auch bei den Verhaltenskonfigurationen bietet.

## 2.5 Umsetzung der Visualisierung und Integration der Funktionen der «D3»-Bibliothek

Nach der Bereitstellung der Anwendungsarchitektur mussten verschiedene Anpassungen vorgenommen werden: auf der Clientseite bei der JavaScript-Datei und auf der Serverseite bei der API. Diese Ände-

rungen haben es insbesondere ermöglicht, das zuvor als JSON formatierte Datenset abzurufen, um die Knoten und Beziehungen zur D3-Bibliothek zu erzeugen. Mit dem D3-Force-Modul soll eine Simulation mit den Knoten und den entsprechenden Beziehungen erstellt werden, die Verhaltensparameter wie Anziehung, Abstossung, Geschwindigkeit oder Umgang mit Kollisionen mitberücksichtigt. Die Anpassung dieser Parameter ermöglicht dann anschliessend, das Verhalten der Knoten zu beeinflussen und die anfänglich übereinanderliegenden Netzelemente zu «entwirren».

## 2.6 Erstellte Visualisierung

Die Visualisierung ermöglicht es dem Benutzer, eine einmalige oder mehrfache Infrastruktursuche auszuführen. Sie bietet die Möglichkeit, die Position der Knoten zu verwalten und einen Knopf für den Datenexport im JSON-Format. Die Simulation liefert mithilfe der Hintergrundkarte einen ersten bereinigten Datensatz. Danach unterstützt sie den Benutzer bei der Platzierung von Objekten. Das aktuelle Produkt umfasst nicht alle Netzpunkte. Die Verbindungen zwischen den Muffen und Gebäudeanschlüssen sowie die der direkt an den Ausgangspunkten angeschlossenen Kabel in den Infrastrukturen wurden noch nicht integriert. Der nächste Entwicklungsschritt wird darin bestehen, diese Knoten und Beziehungen zur visuellen Umgebung hinzuzufügen. Nachstehend ein Überblick über die gegenwärtige Web-Schnittstelle:



Abb. 7: Screenshot der Web-Schnittstelle für die Visualisierung der Daten.

### 3. Ausblick und Fazit

Die Graphdatenbank bietet eine optimale Eingliederung der linearen Topologien unserer üblichen RDBMS und öffnet die Tür für neue Geschäftsapplikationen. Andererseits ermöglicht die entwickelte Web-Schnittstelle auch, den Erfordernissen im Zusammenhang mit der Erstellung eines neuen schematischen Plans nachzukommen. Ergänzende Tests haben jedoch gezeigt, dass die SVG-Technologie, die für das Rendern der Elemente einge-

setzt wird, in Sachen Leistungsfähigkeit an ihre Grenzen stösst, wenn es darum geht, eine Simulation zu erstellen und eine grosse Anzahl von Knoten darzustellen. Gegenwärtig wird ein neuer Ansatz geprüft. Es handelt sich um eine Bibliothek, die die bereits in D3-Force vorhandenen Funktionen übernimmt. Diesmal stützt sich jedoch die Visualisierung auf die WebGL-Technologie. Diese ermöglicht es, die Hardwarebeschleunigung des Grafikprozessors am Endgerät zu nutzen und folglich bei der Ausführung der Si-

mulation die Rechenkapazität beträchtlich zu erhöhen.

Patrick Vogt  
Geomatiktechniker FA  
Verantwortlicher für die  
Netzdokumentation  
Energie Service Biel/Bienne  
Gottstattstrasse 4  
Postfach  
CH-2501 Biel/Bienne  
patrick.vogt@esb.ch

## Intégration de géodonnées dans une BD graphe et génération d'une visualisation schématique

Le réseau électrique de la ville de Bienne, comme tout réseau, demande une gestion pointue de ses composants. Cette dernière nécessite l'intervention de nombreux acteurs spécialisés dans différents domaines. Pour effectuer leurs tâches quotidiennes de manière efficace, les collaborateurs chez Energie Service Bienne doivent avoir la possibilité de s'appuyer sur une documentation actuelle, complète et précise. Le plan d'ouvrage, avec son niveau de détail élevé, constitue le principal outil de travail pour définir l'emplacement des objets. Cependant, les collaborateurs ont également besoin d'une représentation schématique qui exprime la logique du réseau. Celle-ci doit permettre de visualiser les relations entre les composants et offrir une vue où les câbles ne sont pas superposés et représentés par une ligne de tracé unique. De plus, elle doit permettre d'effectuer une intervention rapide et efficace en cas de panne, de planifier le remplacement de certains câbles ou encore de dimensionner une nouvelle section du réseau.

*P. Vogt*

### 1. Problématique à résoudre et objectifs

La représentation schématique actuelle de la basse tension du réseau électrique doit être remplacée. La densité d'informations y est trop élevée, spécialement dans le centre urbain. Il faut donc trouver

une alternative à cette représentation visuelle tout en minimisant le temps de gestion qui devra être alloué au futur produit.

Objectifs:

- Fournir un outil de visualisation avec un accès simple et efficace aux données.
- Générer une représentation schématique permettant une compréhension

rapide de la logique et connectique du réseau électrique.

- Utiliser la base de données graphe Neo4j.
- Intégrer les capacités de simulation de la librairie JavaScript D3 afin d'éviter les superpositions de câbles.

### 2. Réalisation

#### 2.1 Étude des bases de données graphes

L'utilisation de Neo4j offre différents avantages. Cette BD orientée graphe permet l'intégration optimale des topologies linéaires de nos réseaux souterrains grâce à sa gestion des nœuds et relations. La modélisation physique des relations permet de faire abstraction des mécanismes de jointures lors des requêtes complexes.

De plus, l'étape de modélisation est flexible et simplifiée par son apparentement au monde réel ainsi qu'au langage utilisé. Même après l'implémentation de la structure et l'injection des données, il est facilement possible de faire évoluer le schéma de la base de données.

#### 2.2 Transfert des données du SGBD-R vers le SGBD-G

L'import des données dans Neo4j se fait en principe par le biais d'un fichier CSV. C'est la manière la plus simple d'obtenir rapidement une BD opérationnelle. Dans le but de minimiser la procédure de

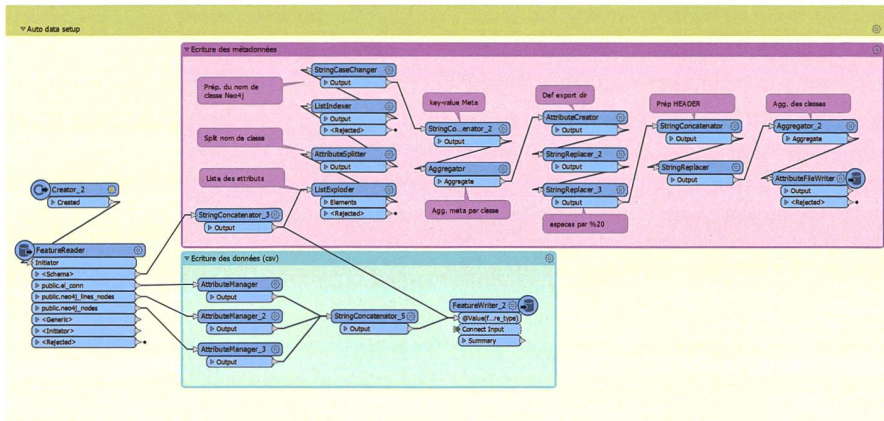


Fig. 4: Workspace FME pour la préparation des fichiers d'imports.

Abb. 4: FME-Workspace für die Vorbereitung der Importdateien.

Fig. 4: Workspace FME per la preparazione dei file d'importazione.

consolidation des données, deux opérations ont dû être réalisées directement dans la base de données relationnelle. Tout d'abord une fonction récupérant tous les attributs nécessaires de manière itérative, puis deux vues utilisant cette fonction et dont les résultats fournissent les ressources pour le traitement dans FME. Dans l'espace de travail FME, on retrouve en entrée les deux vues ainsi que la table modélisant les relations explicites du réseau de câbles électriques. L'idée ici est de créer deux canaux de traitement. Le premier écrit les données des différentes sources au format CSV. Le second prépare les métadonnées et intègre également la syntaxe correspondant aux commandes d'import.

### 2.3 Modélisation des données dans Neo4j

Créer des jointures multiples entre deux tables ou sur des entités de différents niveaux n'est pas monnaie courante dans le monde des bases de données relationnelles. Il en est autrement dans celui des BD graphes. Après l'import des nœuds et la modélisation des relations initiales, d'autres liaisons complémentaires ont été ajoutées au modèle en vue de faciliter l'interrogation et la mise à disposition des données.

À partir de ce stade, le développement du modèle dans Neo4j a été réalisé parallèlement à la mise en place et à la configuration de l'interface de visualisation.

Ces étapes ont constitué deux défis majeurs.

D'une part il a fallu choisir les attributs qui seraient utilisés pour renseigner la fonction de recherche. D'autre part il a fallu définir la requête qui permettrait de formater le résultat et de le rendre directement exploitable par la librairie de visualisation.

### 2.4 Choix de la librairie de visualisation

Bien que Neo4j intègre un outil de visualisation exploitable dans un navigateur web, son utilisation est limitée aux personnes ayant des connaissances pour piloter l'interface et générer des requêtes avec le langage Cypher (SQL des SGBD-G). Heureusement il existe des alternatives efficaces pour un déploiement en entreprise et surtout avec des applications plus accessibles aux différents groupes d'utilisateurs. Dans le cadre de ce projet, mon choix s'est orienté vers une architecture applicative qui sollicite la librairie JavaScript D3. Celle-ci intègre le module d3-force qui offre une grande flexibilité au niveau des configurations, tant visuelles que comportementales.

### 2.5 Mise en place de la visualisation et intégration des fonctions de la librairie «D3»

Après le déploiement de l'architecture applicative, différentes adaptations ont été nécessaires côté client au niveau du

code JavaScript et également côté serveur au niveau de l'API. Ces modifications ont notamment permis de récupérer le jeu de données formaté précédemment en JSON afin de générer les nœuds et relations avec la librairie D3.

Le module d3-force a pour but de générer une simulation avec les nœuds et leurs relations en y intégrant des paramètres comportementaux tels que l'attraction, la répulsion, la vélocité ou encore la gestion des collisions. L'ajustement de ces paramètres a ensuite permis d'influencer le comportement des nœuds et ainsi de «démêler» les éléments du réseau initialement superposés.

### 2.6 Visualisation produite

La visualisation permettant à l'utilisateur d'effectuer une recherche d'infrastructure, unique ou multiple, intègre une gestion de la position des nœuds et offre un bouton pour l'export des données au format JSON. La simulation fournit un premier rendu épuré avec l'appui de la carte placée en arrière-plan. Elle accompagne ensuite l'utilisateur dans le choix du placement des objets.

Le produit actuel ne comprend pas tous les points du réseau. Les liaisons entre manchons et raccordements de bâtiments ainsi que celles des câbles connectés directement aux départs dans les infrastructures ne sont pas encore intégrées. Ces nœuds et relations doivent être ajoutés à l'environnement visuel et constitueront la prochaine étape de développement.

Voici un aperçu de l'interface web actuelle:



Fig. 7: Capture d'écran de l'interface web pour la visualisation des données.

### 3. Perspectives et conclusion

La base de données graphe intègre parfaitement les topologies linéaires de nos SGBD-R habituels et son utilisation ouvre la porte à de nouvelles applications métier. D'autre part, l'interface web développée permet de couvrir les besoins inhérents à la génération d'un nouveau plan schématique. Cependant, des tests complémentaires ont démontré que la technologie

SVG utilisée pour générer le rendu des éléments atteint ses limites en termes de performances lorsqu'il s'agit de simuler et de représenter un grand nombre de nœuds. Une nouvelle piste est actuellement explorée. Il s'agit d'une librairie reprenant les fonctionnalités déjà présentes dans d3-force. Mais cette fois, la visualisation s'appuie sur la technologie WebGL. Celle-ci permet d'exploiter l'accélération matérielle du processeur graphique au niveau du terminal et donc d'augmenter

considérablement les capacités de calcul lors de l'exécution de la simulation.

Patrick Vogt

Technicien en géomatique avec brevet fédéral

Responsable documentation réseaux  
Energie Service Biel/Bienne

Rue de Gottstatt 4

Case postale

CH-2501 Biel/Bienne

patrick.vogt@esb.ch

## Integrazione dei geodati in un database a grafo e generazione di una visualizzazione schematica

La rete elettrica della città di Bienne, come ogni rete, richiede una gestione attenta dei suoi componenti. Questo presuppone l'intervento di numerosi attori specializzati in campi diversi. Per poter svolgere i loro compiti quotidiani in modo efficiente, i collaboratori di Energie Service Bienne devono disporre di una documentazione completa, precisa e aggiornata. Il piano dell'opera, con il suo alto livello di dettaglio, è il principale strumento di lavoro per definire la posizione degli oggetti. Tuttavia, i collaboratori devono anche avere a disposizione una rappresentazione schematica che rispecchi la logica della rete. Tale schema dovrebbe mostrare l'interconnessione tra i componenti e fornire una prospettiva su cui siano indicati i cavi non sovrapposti e rappresentati da un unico tracciato di percorso. Inoltre, lo schema deve consentire di intervenire in modo rapido ed efficiente in caso di guasto o quando si tratta di pianificare la sostituzione di alcuni cavi o addirittura di dimensionare una nuova sezione della rete.

*P. Vogt*

### 1. Problematica da risolvere e obiettivi

Bisogna sostituire la rappresentazione schematica attuale della bassa tensione della rete elettrica. La densità di informazioni è troppo grande, soprattutto nel

centro urbano. È quindi necessario trovare un'alternativa a questa rappresentazione viva, riducendo al minimo i tempi di gestione imputabili al prodotto futuro.

Obiettivi:

- Fornire uno strumento di visualizzazione con un accesso semplice ed efficiente ai dati.
- Generare una rappresentazione schematica che consenta una rapida com-

prendere della logica e dei collegamenti della rete elettrica.

- Utilizzare il database a grafo Neo4j.
- Integrare le capacità di simulazione della libreria JavaScript D3 per evitare sovrapposizioni di cavi.

### 2. Realizzazione

#### 2.1 Studio dei database a grafi

L'utilizzo di Neo4j offre vari vantaggi. Grazie alla gestione dei nodi e delle interconnessioni, questo database a grafo permette l'integrazione ottimale delle topologie lineari delle nostre reti sotterranee. La modellazione fisica delle relazioni permette di ignorare i meccanismi di articolazione durante interrogazioni complesse.

In aggiunta, la fase di modellazione è flessibile e semplificata dalla sua somiglianza con il mondo reale e il linguaggio utilizzato. Lo schema del database può essere facilmente adattato anche dopo l'implementazione della struttura e l'inserimento dei dati.

#### 2.2 Transfer di dati dal RDBMS al GDBMS

L'importazione dei dati in Neo4j avviene di norma attraverso un file CSV.

Questo è il modo più semplice per ottenere rapidamente un database operativo. Per ridurre a un minimo la procedura di consolidamento dei dati è stato necessario effettuare due operazioni direttamen-

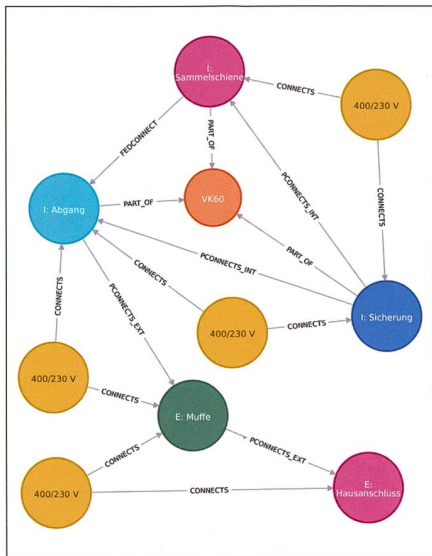


Fig. 5: Screenshot dei dati del grafo in Neo4j.

Abb. 5: Screenshot der Diagrammdaten in Neo4j.

Fig. 5: Capture d'écran des données du graphe dans Neo4j.

te nel database relazionale. La prima operazione consisteva nell'approntare una funzione che recupera tutti gli attributi necessari in modo interattivo, mentre la seconda ha riguardato l'allestimento di due prospettive che utilizzano questa funzione e i cui risultati forniscono le risorse per l'elaborazione in FME. Nell'area di lavoro di FME in entrata si trovano le due prospettive e la tabella che modella le relazioni esplicite della rete di cavi di alimentazione. L'idea è quella di creare due canali di elaborazione: il primo scrive i dati delle diverse fonti nel formato CSV,

mentre il secondo prepara i metadati e integra anche la sintassi corrispondente ai comandi di importazione.

### 2.3 Modellizzazione dei dati in Neo4j

Nel mondo dei database relazionali non è consuetudine creare giunzioni multiple tra due tabelle o su entità a diversi livelli. Questa consuetudine è invece data nel mondo dei database a grafo. Dopo l'importazione dei nodi e la modellazione delle relazioni iniziali, al modello sono stati aggiunti altri legami complementari per facilitare l'interrogazione e la messa a disposizione dei dati.

Da questo punto in poi, lo sviluppo del modello in Neo4j è stato effettuato in parallelo con l'implementazione e la configurazione dell'interfaccia di visualizzazione.

Questi passi hanno rappresentato due grandi sfide: da una parte, si sono dovuti scegliere gli attributi che sarebbero stati utilizzati per informare la funzione di ricerca. D'altra è stato necessario definire l'interrogazione che avrebbe consentito di formattare il risultato e renderlo direttamente utilizzabile dalla libreria di visualizzazione.

### 2.4 Scelta della libreria di visualizzazione

Sebbene Neo4j integri uno strumento di visualizzazione che può essere usato in un browser web, il suo uso è limitato alle persone che hanno le conoscenze adeguate per gestire l'interfaccia e generare

una query con il linguaggio Cypher (SQL del GDBMS). Fortunatamente, esistono alternative efficaci per l'implementazione in un'azienda e soprattutto con applicazioni più accessibili ai diversi gruppi di utenti.

Per questo progetto ho scelto un'architettura applicativa che utilizza la libreria JavaScript D3. Questo integra il modulo d3-force che offre una grande flessibilità in termini di configurazioni sia visive che comportamentali.

### 2.5 Attuazione della visualizzazione e integrazione delle funzioni di libreria «D3»

Dopo l'implementazione dell'architettura dell'applicazione, da parte del cliente si sono resi necessari vari adattamenti sia del codice JavaScript che del server nell'API. Queste modifiche hanno permesso di recuperare il set di dati precedentemente formattato in JSON per generare i nodi e le relazioni con la libreria D3.

Lo scopo del modulo d3-force è quello di generare una simulazione con i nodi e le loro relazioni integrando parametri comportamentali come l'attrazione, la repulsione, la velocità o la gestione delle collisioni. L'adeguamento di questi parametri ha poi permesso di influenzare il comportamento dei nodi e quindi di «distribuire» gli elementi della rete inizialmente sovrapposti.

### 2.6 Visualizzazione generata

La visualizzazione permette all'utente di effettuare una ricerca singola o multipla dell'infrastruttura, integra la gestione della posizione del nodo e offre un pulsante per l'esportazione dei dati in formato JSON. La simulazione fornisce un primo rendering essenziale con il supporto della mappa posta in background e assiste successivamente l'utente nella scelta del posizionamento degli oggetti. Il prodotto attuale non comprende tutti i punti della rete. I collegamenti tra i manicoti e le connessioni degli edifici e quelli dei cavi collegati direttamente agli alimentatori nelle infrastrutture non sono ancora stati integrati. Questi nodi e queste relazioni devono essere aggiunti

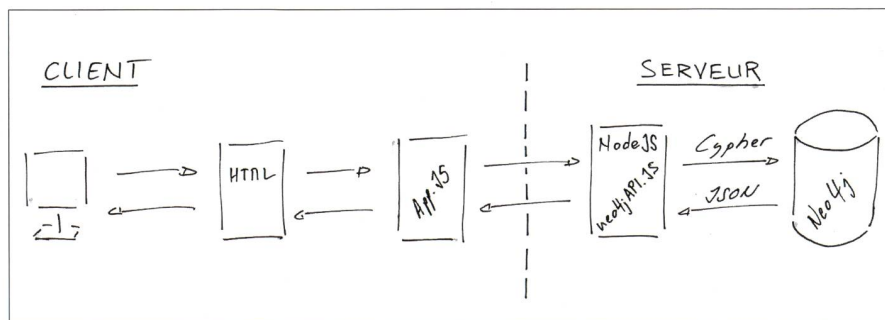


Fig. 6: Panoramica dell'architettura applicativa (Fonte: illustrazione personale).

Abb. 6: Überblick über die Anwendungsarchitektur (Quelle: selbst erstellte Abbildung).

Fig. 6: Aperçu de l'architecture applicative (Source: Illustration personnelle).

all'ambiente visivo e costituiranno la prossima fase di sviluppo. Ecco uno spaccato dell'attuale interfaccia web:



Fig. 7: Screenshot dell'interfaccia web per la visualizzazione dei dati.

### 3. Prospettive e conclusione

Il database a grafo integra perfettamente le topologie lineari dei nostri consueti RDBMS e il suo utilizzo apre le porte a nuove applicazioni aziendali. D'altra parte, l'interfaccia web sviluppata permette di soddisfare le esigenze inerenti alla generazione di un nuovo piano schematico. Tuttavia, ulteriori test hanno dimostrato che la tecnologia SVG utilizzata per generare il rendering degli elementi raggiunge i suoi limiti in termini di prestazioni quando si tratta di simulare e rappresentare un gran numero di nodi. Attualmente si sta esplorando una nuova pista. Si tratta di una libreria che include le

funzionalità già presenti in d3-force. In questo caso la visualizzazione è basata sulla tecnologia WebGL. Quest'approccio consente di sfruttare l'accelerazione hardware del processore grafico a livello di terminale e quindi di aumentare notevolmente le capacità di calcolo durante l'esecuzione della simulazione.

Patrick Vogt  
 Tecnico in geomatica con brevetto federale  
 Responsabile documentazione delle reti  
 Energie Service Biel/Bienne  
 Rue de Gottstatt 4  
 Casella postale  
 CH-2501 Bienne  
 patrick.vogt@esb.ch

**n|w** Fachhochschule Nordwestschweiz  
 Hochschule für Architektur, Bau und Geomatik

**Startdatum: 09.02.2021**

**CAS GeoBIM**  
 Zertifikatslehrgang Geoinformation & BIM

Werden Sie BIM Profi!

[www.fhnw.ch/de/weiterbildung/architektur-bau-geomatik/cas-geo-bim](http://www.fhnw.ch/de/weiterbildung/architektur-bau-geomatik/cas-geo-bim)