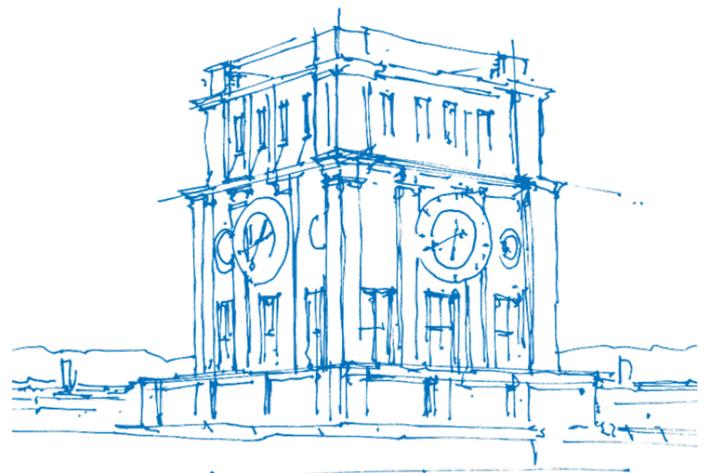


# Repräsentation von Baugrundaufschlüssen in 3D-Geoinformationssystemen am Beispiel der Deutschen Bahn

Representation of Borehole Data in 3D Geographic Information  
Systems: A Case Study of Deutsche Bahn

**Sebastian Sändig**



*TUM Uhrenturm*



# Repräsentation von Baugrundaufschlüssen in 3D-Geoinformationssystemen am Beispiel der Deutschen Bahn

Representation of Borehole Data in 3D Geographic Information  
Systems: A Case Study of Deutsche Bahn

**Sebastian Sändig**

Wissenschaftliche Arbeit zur Erlangung des Grades

**Bachelor of Science (B.Sc.)**

an der TUM School of Engineering and Design der Technischen Universität München.

**Prüfer(in):**

Prof. Dr. Thomas H. Kolbe

**Betreuer(in):**

Dr. Andreas Donaubaueer

**Eingereicht:**

München, den 18.09.2024



Ich versichere hiermit, dass ich die von mir eingereichte Abschlussarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

München, den 18.09.2024

Sebastian Sändig



# Zusammenfassung

Aktuelle Entwicklungen zur Digitalisierung von Infrastrukturprojekten erfordern eine digitale Abbildung der Datengrundlage derartiger Vorhaben („Stufenplan Digitales Planen und Bauen“ 2015). Ein zentrales Gewerk der Gesamtplanung stellt die Geotechnik dar. In diesem Bereich dienen oft eine Vielzahl an Baugrundaufschlüssen als Datengrundlage, die als Anhang geotechnischer Berichte an die entsprechenden Planer weitergegeben werden. Diese Übergabe erfolgt meist in Form nicht editierbarer PDF-Dateien, was für die Weiterverwendung ungeeignet ist und der Datenmenge nicht gerecht wird (Tegtmeier u. a. 2014). Diese Arbeit zielt in diesem Kontext darauf ab, Baugrundaufschlüsse in einem bestehenden Datenmodell zu repräsentieren und für eine dreidimensionale Visualisierung in einem Geoinformationssystem nutzbar zu machen.

Im Rahmen eines Infrastrukturprojekts der Deutschen Bahn InfraGO AG (vereinfacht Deutsche Bahn AG (DB) genannt) wurden drei Anwendungsszenarien entwickelt, die neben der Visualisierung der Daten in einer interaktiven dreidimensionalen Karte, die Integration mit Building Information Modeling (BIM)-Modellen sowie den Export der Daten umfassen. Die Wahl des Datenmodells fiel auf das explizit für Bohrdatenbanken entwickelte Datenmodell *SEP 3*, das als Grundlage für weitere Betrachtungen dienen soll. Für die Integration mit BIM-Modellen wird das Datenmodell Industry Foundation Classes (IFC) verwendet, während der Export durch das objektorientierte Datenaustauschformat *BoreholeML* erfolgen soll.

Die Umsetzung wird durch die Repräsentation der Daten in einer *PostgreSQL*-Datenbank und der anschließenden Visualisierung in *QGIS* umgesetzt. Insgesamt kann dabei gezeigt werden, dass eine dreidimensionale Zylinderdarstellung der Schicht- und Grundwasserdaten bei der Nutzung des Datenmodells *SEP 3* mithilfe von Open Source Software realisiert werden kann. Darüber hinaus kann auch ein Export dieser Daten in eine valide Geography Markup Language (GML)-Datei auf Basis des *BoreholeML*-Schemas durchgeführt werden.



# Abstract

Recent developments in the digitalization of infrastructure expansion require a digital representation of the underlying data for such projects. Geotechnical investigations represent a key discipline in planning these projects. In this field numerous borehole investigations serve as the data basis which are typically appended to geotechnical reports and passed on to the relevant planners. This data exchange is usually done in the form of non-editable PDF files, which are unsuitable for further use and do not adequately address the volume of data (Tegtmeier u. a. 2014). In this context the aim of this thesis is to represent borehole investigations in an existing data model and make them suitable for three-dimensional visualization in a Geographic Information System (GIS).

Within the framework of an infrastructure project by Deutsche Bahn InfraGO AG (referred to as DB), three application scenarios were developed. These include the visualization of data in an interactive 3D map as well as the integration with BIM models and the export of the data. The data model chosen for this purpose is *SEP 3*, specifically designed for borehole databases, and serves as the foundation for further considerations. For integration with building information models, the IFC data model is used, while the export is based on the object-oriented data exchange format *BoreholeML*.

The implementation is carried out by storing the data in a *PostgreSQL* database and visualizing it in *QGIS*. The results demonstrate that a three-dimensional representation of layer and groundwater data can be realized using the *SEP 3* data model through open-source software. In this case, the visualization was achieved by representing the boreholes as cylinders, allowing for an intuitive 3D depiction of the subsurface layers. Additionally, the export of this data into a valid GML file based on the *BoreholeML* schema can be successfully achieved.



# Abkürzungsverzeichnis

**KRB** Kleinrammbohrung

**BK** Untersuchungsbohrung

**SCH** Schurf

**SPT** Standard Penetration Test

**DPH** Schwere Rammsondierung

**CPT** Drucksondierung

**ER-Diagramm** Entity-Relationship-Diagramm

**SQL** Structured Query Language

**LBEG** Landesamt für Bergbau, Energie und Geologie in Niedersachsen

**UML** Unified Modeling Language

**XML** Extensible Markup Language

**XSD** XML Schema Definition

**OGC** Open Geospatial Consortium

**GML** Geography Markup Language

**UML** Unified Modeling Language

**WFS** Web Feature Service

**GeoSciML** Geoscience Markup Language

**CityGML** City Geography Markup Language

**ADE** Application Domain Extension

**BIM** Building Information Modeling

**IFC** Industry Foundation Classes

**STEP** Standard for the Exchange of Product model data

**DGGT** Deutsche Gesellschaft für Geotechnik

**GIM** Geographic Information Modeling

## Abkürzungsverzeichnis

**GIS** Geoinformationssystem

**DBMS** Datenbankmanagementsystem

**WKT** Well-known text

**WKB** Well-known binary

**FME** Feature Manipulation Engine

**DB** Deutsche Bahn AG

**GK 4** Gauß-Krüger Zone 4

**BMDV** Bundesministerium für Verkehr und digitale Infrastruktur

**HTML** Hyper Text Markup Language

**CSV** Comma-separated values

**OMS** Observations, Measurements and Standards

**EPSG** European Petroleum Survey Group

**CRS** Koordinatenreferenzsystem

**PL/pgSQL** Procedural Language for PostgreSQL

# Inhaltsverzeichnis

<b>Zusammenfassung</b>	<b>vii</b>
<b>Abstract</b>	<b>ix</b>
<b>Abkürzungsverzeichnis</b>	<b>xi</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Aufbau der Arbeit . . . . .	2
<b>2 Theoretische Grundlagen</b>	<b>3</b>
2.1 Inhalte geotechnischer Untersuchungen . . . . .	3
2.1.1 Direkte Aufschlüsse . . . . .	3
2.1.2 Indirekte Aufschlüsse . . . . .	4
2.1.3 Zusammenfassung und Darstellung der Ergebnisse . . . . .	5
2.2 Relationale Datenbankmodelle zur Repräsentation von Baugrunddaten . . . . .	6
2.2.1 Relationale Datenbankschemata . . . . .	6
2.2.2 Objektorientierte Datenmodelle . . . . .	11
2.3 Speicherung und Darstellung räumlicher Daten . . . . .	21
2.3.1 Geoinformationssysteme . . . . .	21
2.3.2 Geodatenbanken . . . . .	21
<b>3 Anforderungsanalyse anhand von Anwendungsszenarien</b>	<b>23</b>
3.1 Datengrundlage und bestehende Software . . . . .	23
3.1.1 Datengrundlage . . . . .	23
3.1.2 Bestehende Software . . . . .	24
3.2 Anforderungen . . . . .	24
3.2.1 Allgemeine Anforderungen . . . . .	24
3.2.2 Anforderungen des Geotechnik-Teams . . . . .	25
3.2.3 Anforderungen des BIM-Teams . . . . .	26
3.3 Ableitung von Anwendungsfällen . . . . .	26
3.3.1 Anwendungsfall 1: Abfragen von Aufschlussesdaten mithilfe einer interaktiven drei-dimensionalen Karte . . . . .	26
3.3.2 Anwendungsfall 2: 3D-Darstellung von Aufschlussesdaten und BIM-Modellen . . . . .	27
3.3.3 Anwendungsfall 3: Export einzelner Aufschlüsse für den Austausch . . . . .	27

<b>4 Wahl eines Datenmodells und Datentransformation</b>	<b>29</b>
4.1 Vergleich der Datenmodelle für die Anwendungsfälle . . . . .	29
4.1.1 Anwendungsfall 1 . . . . .	29
4.1.2 Anwendungsfall 2 . . . . .	31
4.1.3 Anwendungsfall 3 . . . . .	32
4.2 Transformation von SEP 3 zu BoreholeML . . . . .	34
4.2.1 Mapping der Bohrinformationen . . . . .	34
4.2.2 Bewertung der Transformation . . . . .	38
<b>5 Prototypische Umsetzung von zwei Anwendungsszenarien in einem DB-Projekt</b>	<b>39</b>
5.1 Verwendete Software und Datengrundlage . . . . .	39
5.1.1 Verwendete Software . . . . .	39
5.1.2 Hinweise zur Datengrundlage . . . . .	40
5.2 Anwendungsszenario 1: Abfragen von Aufschlusdaten mithilfe einer interaktiven dreidimensionalen Karte . . . . .	40
5.2.1 Konzept für Anwendungsszenario 1 . . . . .	40
5.2.2 Import der Daten in eine PostgreSQL-Datenbank mit PostGIS-Erweiterung . . . . .	42
5.2.3 Definition der Abfragen . . . . .	44
5.2.4 Visualisierung in QGIS . . . . .	50
5.3 Anwendungsszenario 3: Export einzelner Aufschlüsse für den Austausch . . . . .	53
5.3.1 Konzept für Anwendungsszenario 3 . . . . .	53
5.3.2 Voraussetzungen für den Export . . . . .	54
5.3.3 Umsetzung der Datentransformation von SEP 3 zu BoreholeML . . . . .	55
5.3.4 Überprüfung und Visualisierung der BoreholeML-Daten . . . . .	57
<b>6 Diskussion der Ergebnisse</b>	<b>59</b>
6.1 Bewertung der prototypischen Umsetzung . . . . .	59
6.1.1 Bewertung des Anwendungsszenarios 1 . . . . .	59
6.1.2 Bewertung des Anwendungsszenarios 3 . . . . .	60
6.2 Bewertung im allgemeinen Kontext . . . . .	61
<b>7 Fazit und Ausblick</b>	<b>63</b>
<b>A Digitaler Anhang</b>	<b>65</b>
<b>Literatur</b>	<b>67</b>

# Abbildungsverzeichnis

2.1	Erkundungsbohrung [1em] (eigene Aufnahme am 23.07.2024) . . . . .	4
2.2	Kleinrammbohrung und schwere Rammsondierung . . . . .	4
2.3	Darstellung eines Bohrprofils und eines Schlagzahldiagramms . . . . .	6
2.4	Zwei Tabellen in Beziehung (Krähenfußdarstellung) . . . . .	7
2.5	Krähenfußdarstellung (eigene Darstellung nach Studer 2019) . . . . .	7
2.6	SEP 3-Datenmodell inklusive der Schlüsselattribute . . . . .	8
2.7	Beispiel für Syntax nach Symbolschlüssel Geologie . . . . .	10
2.8	Darstellung von Packages und Klassen in UML . . . . .	12
2.9	Darstellung von Vererbung in UML . . . . .	12
2.10	Darstellung von Assoziation, Aggregation und Komposition in UML . . . . .	12
2.11	Packages in BoreholeML (BoreholeML o. D.) . . . . .	13
2.12	Teil des BoreholeML3 Datenmodells . . . . .	15
2.13	Fachdaten zur Bildung des Fachmodells Baugrund . . . . .	18
2.14	Vererbungshierarchie für <i>IfcBorehole</i> in IFC 4.3 . . . . .	19
4.1	Auszug aus Mapping Tabelle von <i>SEP 3</i> zu <i>BoreholeML</i> . . . . .	35
4.2	Mapping Beispiel <i>SEP 3</i> zu <i>BoreholeML</i> . . . . .	37
5.1	Workflow für Anwendungsszenario 1 . . . . .	41
5.2	SEP 3-Datenbank in GeoDin . . . . .	43
5.3	Erweiterung des Entity-Relationship-Diagramms zur Darstellung der Abfragen . . . . .	44
5.4	Abfrage der Stammdaten (ER-Diagramm) . . . . .	44
5.5	Workflow der Übersetzungsfunktionen . . . . .	46
5.6	Abfrage der Schichtdaten (ER-Diagramm) . . . . .	47
5.7	Erstellen der Zylindergeometrie . . . . .	48
5.8	Abfrage der Grundwasserdaten (ER-Diagramm) . . . . .	49
5.9	Abfrage der Sondierungsdaten (ER-Diagramm) . . . . .	49
5.10	QGIS-Visualisierung der Ergebnisse in 2D . . . . .	50
5.11	Visualisierung der 3D-Daten (Ansicht oben) . . . . .	51
5.12	Visualisierung der 3D-Daten (Ansicht unten) . . . . .	52
5.13	Workflow für Anwendungsszenario 3 . . . . .	53
5.14	Schematische Darstellung einer GML-Datei nach BoreholeML . . . . .	54
5.15	Pseudocode der Funktion <i>sep3_to_bml</i> . . . . .	55
5.16	Visualisierung einer GML-Datei in FME Data Inspector . . . . .	58
5.17	Visualisierung einer GML-Datei in QGIS . . . . .	58



# 1 Einleitung

## 1.1 Motivation

Im „Stufenplan Digitales Planen und Bauen“ stellt das deutsche Bundesministerium für Verkehr und digitale Infrastruktur ambitionierte Pläne für die Digitalisierung von Bauprojekten vor. Hierbei wird insbesondere die Bedeutung von Building Information Modeling (BIM) hervorgehoben, das den gesamten Lebenszyklus eines Bauwerks virtuell abbildet. Diese Methode soll demnach insbesondere bei öffentlichen Großprojekten wie Vorhaben zum Aus- und Neubau von Infrastruktur zum Standard werden („Stufenplan Digitales Planen und Bauen“ 2015). Dabei ist es entscheidend, dass alle Gewerke in diesen Prozess einbezogen werden. Eines dieser Gewerke, das eine wesentliche Rolle bei der Planung und Umsetzung von Bauvorhaben spielt und mit großem Risiko verbunden ist, stellt die Geotechnik dar (Kuntsche und Richter 2021). Für die digitale Planung ist allgemein eine solide strukturierte Datengrundlage notwendig, in der die Daten in editierbarer Form vorliegen (Baumgärtel u. a. 2011). Traditionell werden diese Informationen in der Geotechnik zunächst durch verschiedene Arten von Baugrundaufschlüssen im Untersuchungsgebiet gewonnen und anschließend in Form von zweidimensionalen Plänen und Berichten festgehalten (Kuntsche und Richter 2021). Diese Darstellungsform ist insbesondere bei komplexen geologischen Sachverhalten und großen Projekten fehleranfällig und nicht mehr zeitgemäß.

Geoinformationssysteme (GIS) bieten eine attraktive Möglichkeit, derartige räumliche Daten interaktiv in einer Karte darzustellen und die hinterlegten Informationen gezielt abzufragen (Bartelme 1989). Da Baugrundverhältnisse grundsätzlich ein stark dreidimensionales Thema darstellen, ist der Einsatz von 3D-Geoinformationssystemen besonders naheliegend. Diese ermöglichen die dreidimensionale Visualisierung der Baugrunddaten und schaffen so ein tieferes Verständnis der räumlichen Verteilung und Zusammensetzung des Untergrundes.

Voraussetzung für diese Darstellung ist jedoch die korrekte Erfassung und Repräsentation der Aufschlussdaten inklusive ihrer Geometrie in einem entsprechenden Datenmodell. Zudem bietet sich bei großen Datenmengen die Speicherung der formatierten Daten in einer Datenbank an. Dadurch kann das Management verbessert sowie Abfragen und die Suche nach Daten deutlich beschleunigt werden (Chang und Park 2004). Außerdem kann eine solche Datenbank die Grundlage weiterer Planungen sein, weshalb eine Exportfunktion einer Teilmenge der Daten auch eingerichtet werden sollte. Förderlich bei der Etablierung eines solchen Systems ist das Zurückgreifen auf bereits existierende Modelle und Standards. Dadurch wird die Kompatibilität mit anderen Software-Lösungen sichergestellt (Baumgärtel u. a. 2011).

Ziel dieser Arbeit ist zunächst die Auswahl eines bereits bestehenden herstellernerneutralen Datenmodells zur Repräsentation von Baugrundaufschlüssen. Die Daten sollen anschließend entsprechend des Modells in einer Datenbank gespeichert und in einem 3D-Geoinformationssystem visualisiert werden.

## 1.2 Aufbau der Arbeit

In den theoretischen Grundlagen soll zunächst ein allgemeiner Überblick über geotechnische Erkundungen im Rahmen von Bauprojekten gegeben werden, um gängige Untersuchungsmethoden und damit die Datengrundlage dieser Arbeit besser einordnen zu können. Daraufhin werden verschiedene Datenmodelle zur Repräsentation von Baugrunddaten im Hinblick auf ihre Struktur und Anwendungsbereiche vorgestellt, bevor abschließend noch auf Geoinformationssysteme und Geodatenbanken eingegangen wird.

Im darauffolgenden Kapitel werden basierend auf den Anforderungen eines Infrastrukturprojekts der Deutschen Bahn InfraGO AG (nachfolgend vereinfacht DB genannt) sowie Interviews mit Geotechnikern, drei Anwendungsszenarien entwickelt, die den praktischen Teil der Arbeit strukturieren. Diese Szenarien umfassen die Speicherung und Visualisierung von Daten in einem 3D-Geoinformationssystem, die Integration mit BIM-Modellen sowie den Export der Daten.

Darauf aufbauend wird in Kapitel 4 die Wahl eines Datenmodells für die drei Anwendungsszenarien diskutiert und anschließend die theoretische Grundlage für die Transformation der Daten in ein weiteres Datenmodell erarbeitet. Zuletzt werden Anwendungsszenario 1 und 3 auf Grundlage der Daten des DB-Projekts mittels einer *PostgreSQL*-Datenbank und dem Geoinformationssystem *QGIS* prototypisch umgesetzt und die erarbeitete Lösung im Anschluss daran diskutiert.

## 2 Theoretische Grundlagen

### 2.1 Inhalte geotechnischer Untersuchungen

Um die Abbildung geotechnischer Daten in einem Datenmodell umzusetzen, muss zunächst geklärt werden, wie diese geplant, umgesetzt und dokumentiert werden.

Gemäß der Norm DIN 4020 sind „geotechnische Untersuchungen die zur bautechnischen Beschreibung und Beurteilung von Boden und Fels notwendigen (ingenieur-) geologischen, hydrogeologischen, geophysikalischen, bodenmechanischen, felsmechanischen, umwelttechnischen und chemischen Untersuchungen“ (DIN 4020 2010). Diese Untersuchungen müssen vor Beginn einer Baumaßnahme durchgeführt werden, um diese sicher, passgenau und wirtschaftlich umzusetzen. Das zu errichtende Bauwerk wird entsprechend seiner Komplexität in eine von drei geotechnischen Kategorien eingeteilt. Kategorie eins erfordert bei geringer Komplexität lediglich einen geringen Untersuchungsaufwand, während Kategorie drei aufgrund außergewöhnlicher Tragwerke oder eines schwierigen Baugrunds einen höheren Aufwand verlangt (EN1997-1 2014). Der Begriff „Baugrund“ beschreibt in diesem Zusammenhang den Boden beziehungsweise Fels auf dem ein Bauwerk errichtet wird (Kuntsche und Richter 2021). Die Klassifikation in eine der geotechnischen Kategorien sollte in einer möglichst frühen Planungsphase erfolgen, um die darauffolgenden Untersuchungen den spezifischen Gegebenheiten anpassen zu können. (EN1997-1 2014)

Auf die Kategorisierung folgt eine Voruntersuchung, in der beispielsweise die Eignung der Lage des Bauwerks, meist anhand von Bestandsunterlagen, geprüft wird. In der anschließenden Hauptuntersuchung werden in Feld- und Laborversuchen das detaillierte Baugrundprofil, die hydrogeologischen Verhältnisse sowie weitere geologische Merkmale untersucht (ebd.). Indirekte Aufschlüsse liefern hierbei durch physikalische Messgrößen Informationen über die vorliegenden Kenngrößen, während direkte Aufschlüsse eine visuelle Erfassung und Klassifikation des angetroffenen Bodens ermöglichen (DIN 4020 2010). Einen umfassenden Überblick über gängige Aufschlusstypen gibt Kuntsche in Kapitel drei und vier seines Buchs „Geotechnik“. Die verschiedenen Typen werden im Folgenden aufgelistet und kurz beschrieben (Kuntsche und Richter 2021).

#### 2.1.1 Direkte Aufschlüsse

Eine der gebräuchlichsten Methoden direkter Aufschlüsse ist die Kleinrammbohrung (KRB). Hierbei wird ein Kernrohr zuerst in den Boden gerammt und anschließend meist hydraulisch wieder aus dem Bohrloch gezogen, um die Schichtdaten zu erfassen und Proben für nachfolgende Analysen zu entnehmen. Zusätzlich kann ein eventuell angetroffener Grundwasserspiegel mithilfe eines Lichtlots eingemessen werden. Eine qualitativ hochwertigere Methode zur direkten Erkundung des Baugrunds stellen Untersuchungsbohrungen (BK) dar. Mit schweren Bohrgeräten können hierbei auch sehr große Erkundungstiefen in hoher Qualität und inklusive Probenahme erreicht werden.

Neben Bohrungen können auch Schürfe (SCH) eingesetzt werden, um die obersten Schichten des Baugrundes zu untersuchen. Dazu wird eine kleine Grube ausgehoben, in der ein Geotechniker nach entsprechender Bearbeitung der Seitenflächen die Schichtung des Baugrunds sowie weitere Parameter direkt erfassen kann.



**Abbildung 2.1** Erkundungsbohrung  
(eigene Aufnahme am 23.07.2024)



**Abbildung 2.2** Kleinrammbohrung (links) und schwere Rammsondierung (rechts)  
(eigene Aufnahme am 23.07.2024)

Das Ergebnis direkter Aufschlüsse von Böden, auch Bodenansprache genannt, ist in einem sogenannten Schichtenverzeichnis dokumentiert. Darin werden insbesondere die Höhenlage, Bodenart und Beimengungen der einzelnen Schichten nach (DIN 14688-1 2018) erfasst. Bei der Bodenart wird zwischen Haupt- und Nebenanteilen des Bodens unterschieden, wobei der Hauptanteil die bautechnischen Eigenschaften des Bodens bestimmt und die Nebenanteile diese zusätzlich verändern. Zudem werden weitere Parameter wie Konsistenz, Plastizität, Farbe oder Bohrfortschritt beschrieben. Wasserstände und die Entnahme von Proben werden ebenfalls vermerkt. Die Proben werden anschließend im Labor untersucht, um zusätzliche Bodenparameter zu bestimmen oder um die bei der Bodenansprache erfassten Werte zu überprüfen.

### 2.1.2 Indirekte Aufschlüsse

Eine häufig verwendete Methode zur indirekten Erkundung ist die Rammsondierung. Hierbei wird eine genormte Stahlstange, mittels einem Fallgewicht, dem sogenannten Rammhären in den Boden gerammt. Das Ergebnis dieser Sondierung wird durch die Anzahl der Schläge bestimmt, die erforderlich sind, um die Stange jeweils um zehn Zentimeter in den Boden zu treiben. Abhängig vom Gewicht des Rammhärens unterscheidet man zwischen der leichten (DPL), der mittleren (DPM), der schweren (DPH) und der superschweren (DPSH-A bzw. DPSH-B) Rammsondierung.

Eine spezielle Form der Rammsondierung stellt der Standard Penetration Test (SPT) dar. Bei diesem Verfahren wird, abhängig vom Bodentyp, innerhalb eines Bohrlochs entweder eine genormte Spitze

Direkte Aufschlüsse		Indirekte Aufschlüsse	
Aufschlusstyp	Abkürzung	Aufschlusstyp	Abkürzung
Klein(-ramm)bohrung	KRB	Schwere Rammsondierung	DPH
Erkundungsbohrung	BK	Standard Penetration Test	SPT
Schurf	SCH	Drucksondierung	CPT

**Tabelle 2.1** Häufig verwendete direkte und indirekte Aufschlussverfahren

oder ein Probenentnahmeggerät in den Boden gerammt, wobei die Anzahl der Schläge gezählt wird, die benötigt wird, um eine Eindringtiefe von 30 Zentimetern zu erreichen.

Neben Rammsondierungen werden häufig auch Drucksondierungen (CPT) durchgeführt. Bei dieser Methode wird eine Sonde mit konstanter Geschwindigkeit in den Boden gedrückt, wobei in der Regel ein Eindringweg von einem Meter pro Messung genutzt wird. Sensoren an der Spitze der Sonde erfassen den Spitzendruck sowie die Mantelreibung während des Eindringens. Zusätzlich können die Abweichung der Spitze von der Lotrechten sowie der Porenwasserdruck gemessen werden. Die aus diesen Messwerten gewonnenen Daten ermöglichen Rückschlüsse auf die Lagerungsdichte beziehungsweise Konsistenz des Bodens und liefern Hinweise auf die Schichtart.

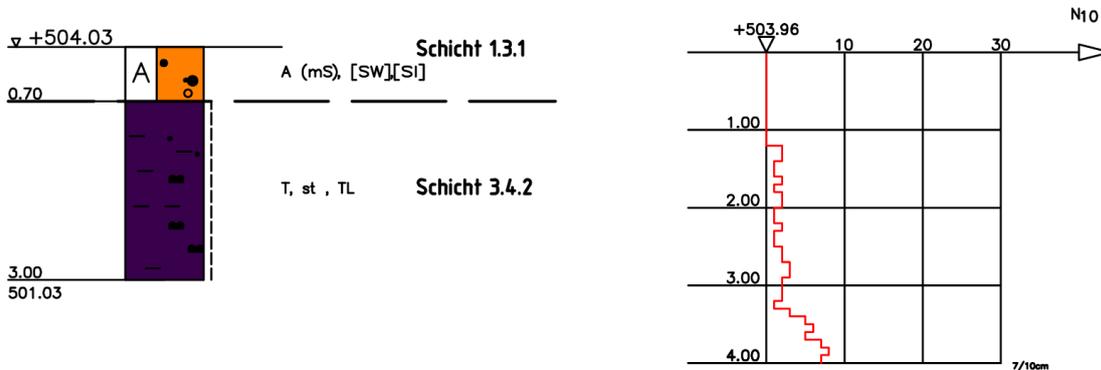
### 2.1.3 Zusammenfassung und Darstellung der Ergebnisse

In Kapitel sechs seines Werkes „Geotechnik“ beschreibt Kuntsche die Zusammenfassung und Darstellung der Ergebnisse geotechnischer Untersuchungen in Form eines geotechnischen Berichts. Diese Methodik wird im Folgenden zusammengefasst (Kuntsche und Richter 2021).

Bei direkten Aufschlüssen werden die erhobenen Daten gemäß dem Formblatt nach DIN EN ISO 22475-1:2019-09 zunächst händisch erfasst. Anschließend erfolgt die digitale Verarbeitung der Daten, aus denen dann Bohrprofile erstellt werden, die alle relevanten Informationen der Bodenansprache enthalten (siehe 2.3 links). Die Darstellung der Profile erfolgt in der Regel nach den Richtlinien der (DIN4023 2023) unter Verwendung entsprechender Software. Indirekte Aufschlüsse werden meist durch Diagramme visualisiert. Bei Rammsondierungen wird die Schlagzahl in Abhängigkeit von der Tiefe in einem Stufendiagramm dargestellt, welches es ermöglicht, indirekte Rückschlüsse auf die Konsistenz oder Lagerungsdichte der Schichten zu ziehen (siehe 2.3 rechts). Die Ergebnisse der Drucksondierungen werden hingegen in einem Liniendiagramm über die Tiefe aufgezeichnet. Die gesammelten Daten werden abschließend in einem geotechnischen Bericht zusammengefasst und aufbereitet. Dieser Bericht enthält Bohrprofile, Sondierungen und Probenergebnisse als Anlagen. Zudem wird ein Baugrundmodell entwickelt, bei dem zwischen den Aufschlüssen interpoliert wird, um einen kontinuierlichen Schichtenverlauf zu erhalten. Schichten mit gleichwertigen geotechnischen Parametern werden anschließend zu sogenannten Homogenbereichen zusammengefasst. Abschließend werden häufig Empfehlungen zur Umsetzung des Bauvorhabens aus geotechnischer Sicht formuliert.

KRB km 22,100 L  
 21.03.2017  
 AP 6,75 m von GA bl

DPH km 22,100 L  
 23.02.2017  
 AP 6,55 m von GA bl



**Abbildung 2.3** Darstellung eines Bohrprofils (links) und eines Schlagzahldiagramms (rechts)  
 (DB InfraGO AG 2020)

## 2.2 Relationale Datenbankmodelle zur Repräsentation von Baugrunddaten

Die eben beschriebenen Ergebnisse geotechnischer Untersuchungen sollen nun in einem Datenmodell repräsentiert werden. Standardisierte Datenmodelle für Aufschlussdaten sind in Deutschland insbesondere schon in den Datenbanken der Landesämter implementiert. Denn laut § 9 des deutschen Geologiedatengesetzes müssen die erfassten Fachdaten einer geologischen Untersuchung der zuständigen Landesbehörde übermittelt werden (Geologiedatengesetz 2020). Je nach Bundesland werden die Daten entweder schon in einem digitalen Bohrdatenformat oder in schriftlicher Form übergeben. Außerdem unterscheidet sich auch das Datenmodell der Bohrdatenbanken der einzelnen Landesämter. (Schirren 2020)

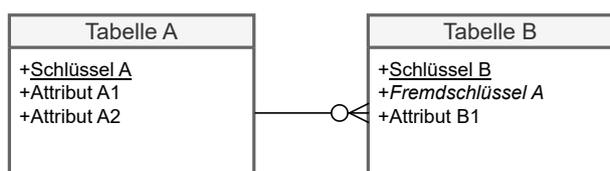
Im Folgenden wird nun eine Auswahl bereits existierender Datenmodelle zur digitalen Repräsentation von Baugrundaufschlüssen beschrieben. Sie können grundsätzlich in zwei verschiedene Typen eingeteilt werden: Relationale und objektorientierte Datenmodelle. Zunächst soll immer das Konzept und die Darstellungsform des jeweiligen Typs erklärt werden, bevor auf die einzelnen Datenmodelle eingegangen wird.

### 2.2.1 Relationale Datenbankschemata

Die Repräsentation von Bohrdaten kann beispielsweise durch ein relationales Datenbankschema erfolgen. (Studer 2019) definiert und erklärt dieses Konzept in seinem Buch „Relationale Datenbanken“. Demnach ist ein relationales Datenbankschema eine strukturierte Art der Organisation und Verwaltung von Daten, bei der Informationen in Tabellen (Relationen) gespeichert werden. Jede Tabelle besteht aus Zeilen (Datensätzen oder n-Tupeln) und Spalten (Attributen), wobei die Domäne eines Attributs der Menge aller möglichen Werte desselben Attributs entspricht. Jedes n-Tupel kann durch ein spezielles Attribut, den Primärschlüssel, eindeutig identifiziert werden. Dieser dient als sogenannter Fremdschlüssel auch zur Referenzierung des Datensatzes in einer anderen Relation. Bei diesen Beziehungen wird

zwischen drei unterschiedlichen Typen unterschieden, die nun anhand zwei beliebiger separater Tabellen A und B betrachtet werden sollen. Eine 1:1-Beziehung bedeutet, dass ein Datensatz in Tabelle A mit genau einem Datensatz in Tabelle B verbunden ist und umgekehrt. In einer 1:n-Beziehung kann ein Datensatz in Tabelle A mit mehreren Datensätzen in Tabelle B verknüpft sein, wobei jeder Datensatz in Tabelle B nur einem Datensatz in Tabelle A zugeordnet ist. Eine n:m-Beziehung ermöglicht, dass mehrere Datensätze in Tabelle A mit mehreren Datensätzen in Tabelle B verbunden sind und umgekehrt und wird im Schema durch eine zusätzliche Tabelle realisiert.

Zur vereinfachten Darstellung eines relationalen Datenbankschemas inklusive aller Schlüssel und Be-



**Abbildung 2.4** Zwei Tabellen in Beziehung (eigene Darstellung nach Studer 2019)

Bedeutung	Symbol
Keine oder eine Beziehung	—
Keine, eine oder mehrere Beziehung	—○
Genau eine Beziehung	—+
Eine oder mehrere Beziehungen	—<

**Abbildung 2.5** Krähenfußdarstellung (eigene Darstellung nach Studer 2019)

ziehungen werden oft Diagramme verwendet. Im Kapitel drei des Buches „Relationale Datenbanken“ wird die sogenannte „Krähenfußdarstellung“ vorgestellt (ebd.). Diese ist eine Abwandlung der Entity-Relationship-Diagramme (ER-Diagramme), welche allgemein eine grafische Sprache zur Abbildung relationaler Datenmodelle darstellt (Brinkhoff und Kresse 2022).

Die Tabellen werden mit ihren Attributen als Rechtecke dargestellt, während Beziehungen durch Linien zwischen den Tabellen repräsentiert werden (vergleiche Abbildung 2.4). Das Ende der Linie repräsentiert hierbei, ob eine Beziehung pflichtmäßig vorhanden (vertikaler Strich) sein muss, und ob ein oder mehrere Elemente referenziert werden (vergleiche Abbildung 2.5). Die jeweiligen Attribute der Relationen werden innerhalb des Rechtecks aufgelistet, wobei der Schlüssel unterstrichen dargestellt wird. Abbildung 2.4 stellt demnach dar, dass Tabelle A drei Attribute besitzt und mit keinem, einem oder mehreren Datensätzen von Tabelle B in Beziehung steht. Das Attribut „Schlüssel A“ dient hierbei als Fremdschlüssel in Tabelle B.

### 2.2.1.1 SEP 3

Ein Beispiel für ein relationales Datenbankschema für geologische Daten ist die Formatbeschreibung *SEP 3*, die vom Landesamt für Bergbau, Energie und Geologie in Niedersachsen (LBEG) als Schnittstelle zwischen dessen Bohrdatenbank und der Software *GeoDin* der *FUGRO Consult GmbH* entwickelt wurde. Neben den Stamm-, Schicht- und Grundwasserdaten lassen sich in dem zugrunde liegenden Datenmodell auch Probanddaten, Ausbaudaten, Bohrlochmessungen und Sondierungen erfassen, wodurch fast alle Daten einer Bohrung abgebildet werden können (Sbresny o. D.). Ein wichtiger Bestandteil zur einheitlichen Erfassung von Bohrdaten in diesem Modell ist der Symbolschlüssel Geologie, in dem Kürzel für die Codierung geologischer Sachverhalte festgehalten sind. Dadurch wird neben der Sicherstellung der korrekten Eingabe der Daten auch eine Plausibilitätsprüfung möglich.

**Aufbau des Datenmodells.** Das Datenmodell besteht insgesamt aus 15 Relationen und 64 Schlüssel-Listen, wobei das zentrale Element des Modells durch die Relation *S3STAMM* gebildet wird (ver-

gleiche Abbildung 2.6). In dieser werden die Stammdaten der einzelnen Aufschlüsse erfasst. Diese umfassen Informationen wie Koordinaten, Aufschlusstiefe, Name des Aufschlusses und weitere relevante Attribute. Jeder Aufschluss wird innerhalb eines Projekts durch das numerische Schlüsselattribut *LOCID* eindeutig identifiziert. Ein Aufschluss kann global jedoch nur durch die Kombination aus *LOCID* und *PROJ\_ID* (Projektcode) eindeutig identifiziert werden, da innerhalb einer Datenbank mehrere Projekte gespeichert werden können. Zusätzlich zur Tabelle *S3STAMM* werden, je nach Verfügbarkeit der Daten, weitere spezifische Informationen in anderen Tabellen gespeichert. Beispielsweise werden Schichtdaten in der Tabelle *S3SCHDAT* und Grundwasserdaten in der Tabelle *S3GRUNDW* abgelegt. Der Bezug zu den Aufschlüssen und damit zur Geometrie wird dabei stets durch den Fremdschlüssel *LOCID* in den entsprechenden Tabellen hergestellt, was eine klare Zuordnung und Verknüpfung der Daten zum jeweiligen Aufschluss sicherstellt (LBEG 2022). Die Kombination aus *PROJ\_ID*, *LOCID* und *RECID* in diesen Tabellen bildet zudem einen eindeutigen Schlüssel, was auch in Abbildung 2.6 dargestellt ist. Die Fremdschlüssel sind in dieser Abbildung kursiv abgebildet, während die Primärschlüssel gemäß der Krähenfußnotation unterstrichen dargestellt sind.

Grundsätzlich können einem Aufschluss in *S3STAMM* immer mehrere Datensätze in den anderen Tabellen zugeordnet werden, weshalb es sich somit um 1:n-Beziehungen handelt. Im Folgenden werden nur noch die Tabellen *S3STAMM*, *S3SCHTIT*, *S3SCHDAT*, *S3GRUNDW*, *S3SNDREG* und *S3SNDDAT* näher erklärt, da nur diese im Praxisteil der Arbeit verwendet werden (siehe grüne Relationen in Abbildung 2.6). Die restlichen grau dargestellten Klassen dienen vor allem der Repräsentation des Ausbaus und der Verfüllung des Bohrlochs. Diese Informationen sind in Bauprojekten jedoch nur selten relevant.

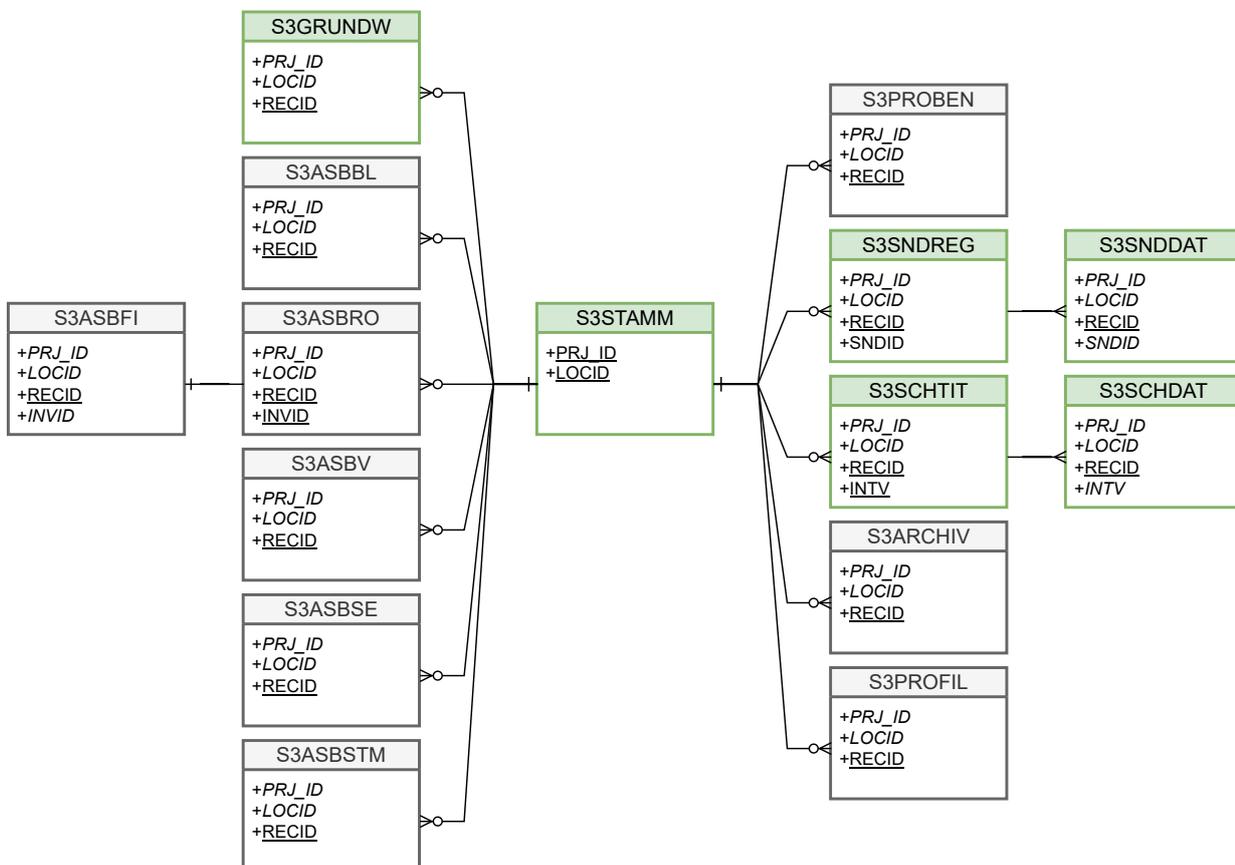


Abbildung 2.6 SEP 3-Datenmodell inklusive der Schlüsselattribute (eigene Darstellung nach LBEG 2022)

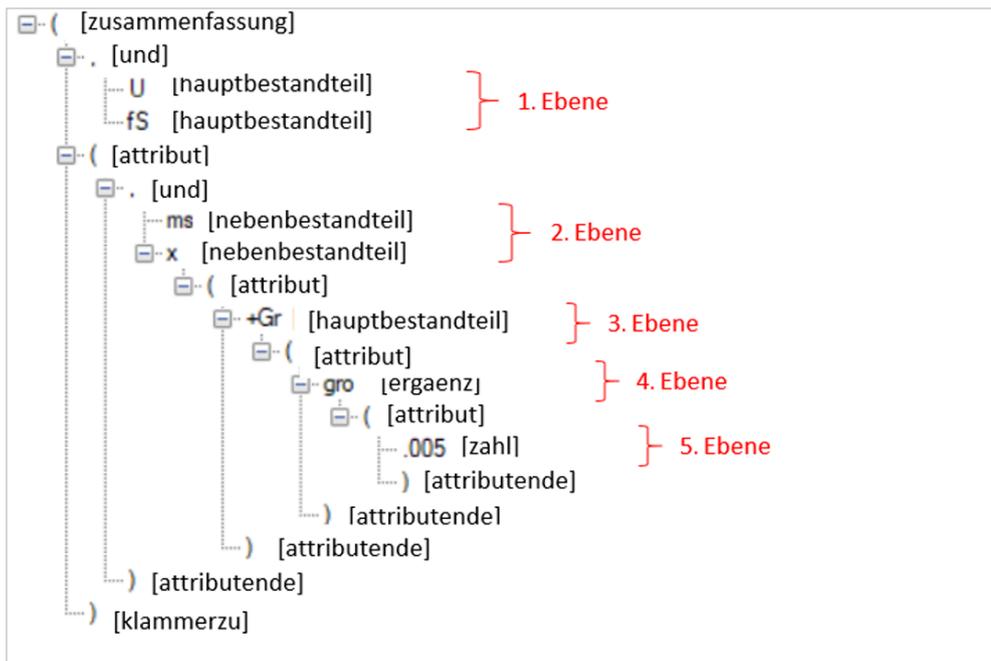
Inhalte relevanter Tabellen des <i>SEP 3</i> -Formats	
<i>S3STAMM</i> Stammdaten	Angaben zur eindeutigen Identifikation einer Bohrung (z.B. geografische Lage) und Hinweise auf das Vorhandensein weiterer Daten zu dieser Bohrung (z.B. Grundwasserdaten, Probandaten)
<i>S3SCHTIT</i> Titeldaten	Angaben zur Schichtbeschreibung (z.B. das Datum der letzten Überarbeitung des Schichtenverzeichnisses, die Versionsnummer der Interpretation, Name des Bearbeiters)
<i>S3SCHDAT</i> Schichtdaten	Angaben, die die bei einem Aufschluss angetroffenen Schichten des Untergrundes beschreiben (z.B. in ihrer Lage, ihrer Beschaffenheit, ihrer Zusammensetzung, ihrer Entstehung, ihrem Aussehen)
<i>S3SNDREG</i> Log-Messungen	Angaben zu Logmessungen bei indirekten Aufschlüssen (z.B. Schwere Rammsondierung oder Standard Penetration Test)
<i>S3SNDDAT</i> Messdaten	Messwerte mit Tiefenangaben aus Log-Messungen
<i>S3GRUNDW</i> Grundwasser	Angaben zu angetroffenem Grundwasser (z.B. Tiefe, bei der Grundwasser angetroffen wurde, Datum der Grundwassermessung)

**Tabelle 2.2** Auswahl wichtiger Relationen des Datenmodells *SEP 3*

Die Tabelle *S3STAMM* enthält die Stammdaten der Aufschlüsse. Zu diesen gehören insbesondere die Lageinformationen des Ansatzpunktes in Rechtswert (*XCOORD*), Hochwert (*YCOORD*) und Höhe (*ZCOORDB*). Folglich werden hier alle Informationen gespeichert, die für alle Typen von Aufschlüssen (Bohrungen und Sondierungen) definiert sein können.

Die anderen eben aufgeführten Tabellen existieren hingegen für die Speicherung der eigentlich ermittelten Daten. So werden beispielsweise in der Tabelle *S3SCHTIT* die Titeldaten abgelegt. Diese enthalten Angaben zur Art der Schichtbeschreibung und repräsentieren mit jedem Datensatz eine Version des Schichtenverzeichnisses eines direkten Aufschlusses (KRB, BK oder SCH). Meistens existiert nur eine Version dieses Verzeichnisses für jeden dieser Aufschlüsse, nämlich die der Ersterfassung der Schichten. Die Nummer der Version wird im Attribut *INTV* gespeichert und dient zusammen mit *PROJ\_ID* und *LOCID* zudem als Fremdschlüssel in der Tabelle *S3SCHDAT*, die die eigentlichen Informationen über die angetroffenen Schichten der Bohrungen und Schürfe enthält.

In *S3SCHDAT* entspricht jede Instanz einer dieser Schichten und wird durch die Anfangs- und Endtiefe auch hinsichtlich ihrer Lage definiert. Die in der Bodenansprache gewonnenen Daten zur Petrographie, Genese, Farbe, Konsistenz, Wassergehalt, Lagerungsdichte und Bodengruppe werden durch die im Symbolschlüssel Geologie definierten Abkürzungen in den entsprechenden Attributen gespeichert. Da die Syntaxregeln zur korrekten Beschreibung aller Datenfelder komplex und umfangreich sind, werden sie an dieser Stelle nur grob aufgezeigt. Sie spielen bei der direkten Nutzung des Formats jedoch eine entscheidende Rolle, da die gespeicherten Daten nach diesen Regeln zu menschenlesbarem Text übersetzt werden müssen. Diese Funktion wird in der Praxis meist durch entsprechende Software (z.B. *GeoDin*) übernommen. Die Abkürzungen können in einem zusammengesetzten Ausdruck einen Übergang zwischen den Werten darstellen (mit Bindestrich getrennt), kombiniert oder nebeneinander aufgezählt werden (mit Komma getrennt). Zusätzlich können durch Klammern Haupt- und Nebenebenen sowie Gruppierungen der Einzelkürzel definiert werden. Dies findet insbesondere im Attribut Petrographie (*PETRO*) Anwendung. Denn hier liegen meist Haupt- und Nebenbestandteile vor, wobei



**Abbildung 2.7** Beispiel für Syntax nach Symbolschlüssel Geologie aus LBEG 2022  
*Klartext:* (Schluff, Feinsand) (schwach mittelsandig, steinig (Granit (groß 0,005m)))  
*SEP 3-Syntax:* (U,fS)(ms2,x(+Gr(gro(0.005))))

die Nebenbestandteile immer den Hauptbestandteilen nachgestellt sind. Ein Beispiel hierzu ist in Abbildung 2.7 zu sehen, in dem die Hauptbestandteile Schluff und Feinsand vor den Nebenbestandteilen genannt werden. Ebenso können einzelne Kürzel mit Intensitäten kombiniert werden. So wird der Nebenbestandteil „mittelsandig“ in dem dargestellten Beispiel mit dem Intensitätspäfix „2“ als schwach attribuiert. Darüber hinaus gibt es ein Klartextfeld zur Ergänzung weiterer Bemerkungen *ERGBEM*, die nicht durch den Symbolschlüssel Geologie abgebildet werden können

Zusätzlich zu den Schichtdaten werden bei direkten Aufschlüssen auch lokale Grundwasserstände ermittelt, welche im Format *SEP 3* in der Tabelle *S3GRUNDW* repräsentiert werden. Da hierbei wieder mehr als ein Grundwasserstand pro Aufschluss aufgenommen werden kann, muss jeder Datensatz durch die Kombination aus *PRJ\_ID*, *LOCID* und *RECID* eindeutig identifiziert werden. Die Daten werden je nach Art des Grundwasserstands (Ruhewasserstand, angetroffener, gestiegener oder gesunkener Grundwasserstand) als Tiefenwert in den jeweiligen Spalten (*GWR*, *GW*, *GWS*, *GWF*) gespeichert. Zuletzt soll nun auf die Repräsentation indirekter Aufschlüsse am Beispiel von Rammsondierungen eingegangen werden. Da mehrere dieser Sondierungen einem direkten Aufschluss zugeordnet werden können, werden in der Tabelle *S3SNDREG* zunächst erneut alle Einzelsondierungen durch je eine Instanz repräsentiert und durch die Kombination aus *PROJ\_ID*, *LOCID* und *SNDID* eindeutig identifiziert. Die Speicherung der eigentlichen Daten erfolgt hingegen in der Tabelle *S3SNDDAT*, wobei die eben genannten Attribute hier wieder als Fremdschlüssel dienen. Die Messdaten werden pro Tiefenabschnitt *DEPTH* im Attribut *SNDVALUE* gespeichert und durch den Schlüssel *SNDID* referenziert. Im Fall einer schweren Rammsondierung würden die Tiefenabschnitte von je zehn Zentimetern im Attribut *DEPTH* und die Schlagzahlen im Feld *SNDVALUE* gespeichert werden.

**Anwendungen des Datenmodells.** Primär angewendet wird das relationale Datenbankmodell *SEP 3* in der Bohrdatenbank des Bundeslandes Niedersachsen (LBEG o. D.). Es wird auch als Industriestandard, für Bohrdaten bezeichnet und kann somit von einigen Geotechnik-Programmen gelesen werden (Schirren 2020). Beispielsweise unterstützt die Software *GeoDin* das Datenbankschema zur Speicherung und Verwaltung von Baugrunddaten und kann eine entsprechend formatierte *Microsoft Access* Datenbank exportieren (Holmok 2023). Zusätzlich existiert das kostenlose Programm *GeoDin Shuttle* zur Erstellung einer *SEP 3*-Datenbank, wodurch jedem die Erstellung einer korrekt formatierten Bohrdatenbank, mit validierten und strukturierten Daten ermöglicht wird (LBEG o. D.). Dies dient vor allem der Meldung von Bohrdaten gemäß des Geologiedatengesetzes in Niedersachsen, da die Übergabe der Bohrinformationen hier in Form einer *SEP 3*-Datenbank erfolgen muss (Schirren 2020).

### 2.2.2 Objektorientierte Datenmodelle

Neben relationalen Datenbankmodellen existieren auch objektorientierte, hierarchische Datenmodelle, deren Objekte durch das standardisierte, textbasierte Format Extensible Markup Language (XML) strukturiert gespeichert werden können (Brinkhoff und Kresse 2022). Dieses wurde ursprünglich entwickelt, um die Herausforderungen des groß angelegten elektronischen Publizierens zu bewältigen, hat sich jedoch zu einem universellen Format für den Datenaustausch in verschiedenen Bereichen entwickelt. XML ist hierarchisch organisiert, wobei Daten in einer Baumstruktur dargestellt werden, die aus Elementen, Attributen und Texten besteht (Quin 2016). Diese Struktur ermöglicht es, komplexe und verschachtelte Informationen, wie sie beispielsweise in Baugrunddaten oder geowissenschaftlichen Modellen vorkommen, verständlich zu repräsentieren. Um eine bestimmte Struktur für eine solche Datei festzulegen wird auf sogenannte XML Schema Definition (XSD)-Dateien zurückgegriffen, die zur Überprüfung einer XML-Datei herangezogen werden können. Da Baugrunddaten immer einen Raumbezug beinhalten, liegt zudem die Verwendung der XML-Syntax Geography Markup Language (GML) des Open Geospatial Consortium (OGC) nahe. Das OGC ist allgemein ein internationaler Zusammenschluss von Unternehmen, der sich der Interoperabilität und der Entwicklung von Standards für räumliche Daten widmet (Kresse, Danko und K. 2022). GML dient sowohl als Modellierungssprache für geografische Systeme als auch als offenes Austauschformat für geografische Transaktionen im Internet. Die Struktur eines GML-Dokuments wird durch ein GML-Schema beschrieben, in dem geografische Objekte durch Punkte, Linien und Flächen repräsentiert werden. Hierauf aufbauend können zudem auch spezielle Anwendungsschemata entwickelt werden, die spezifische geografische Konzepte wie Straßen oder Brücken abbilden. (GML 2012) Eine Bohrung kann in diesem Kontext beispielsweise als Punkt (Repräsentation des Ansatzpunktes der Bohrung) oder als Linie (Repräsentation des unterirdischen Bohrpfad) dargestellt werden. Zusätzlich können semantische Daten wie zum Beispiel der Aufschlusstyp oder die Schichteninformationen in dem Modell abgebildet werden.

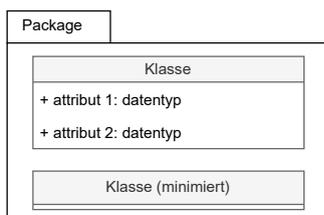
#### 2.2.2.1 Unified Modeling Language (UML)

Zur übersichtlichen grafischen Darstellung solcher Anwendungsschemata kann die grafische Sprache UML herangezogen werden, die allgemein als Werkzeug für die Analyse, das Design und die Implementierung softwarebasierter Systeme entwickelt wurde (UML 2017). Herring gibt im Kapitel „Modeling of Geographic Information“ des Buchs „Springer Handbook of Geographic Information“ einen Überblick

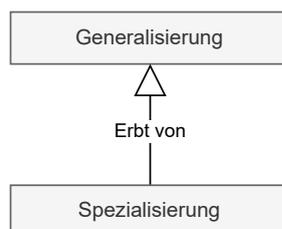
über alle wichtigen Elemente dieser Sprache (Herring, Roswell und Danko 2022). Die für den folgenden Teil dieser Arbeit relevanten UML-Elemente sollen nun aufgeführt werden.

Ein *Package* dient als Namensraum für eine Gruppe von Elementen und wird durch ein Rechteck mit einem kleineren Reiter dargestellt (vergleiche Abbildung 2.8). Klassen sind zentrale Elemente in UML, die eine Gruppe von Objekten mit gemeinsamen Eigenschaften beschreiben. Diese können durch Rechtecke mit zwei Bereichen dargestellt werden, wobei der obere Bereich den Klassennamen und der untere die Attribute enthält (vergleiche Abbildung 2.8). Ein *Interface* beschreibt das Verhalten, das von einer oder mehreren Klassen implementiert wird, ohne selbst instanziiert zu werden. *Data Types* sind ähnlich wie Klassen definiert, allerdings werden ihre Instanzen lediglich durch ihren Wert identifiziert. Eine *Enumeration* wiederum stellt einen speziellen Datentyp dar, bei dem die möglichen Werte klar definiert und auf eine Liste begrenzt sind. Real existierende Elemente, die in UML mit einer Klasse abgebildet werden, werden im Kontext räumlicher Daten auch *Feature Type* genannt. Die Verbindungen zwischen Elementen werden durch unterschiedliche Typen von Relationen verdeutlicht, die nun kurz definiert werden.

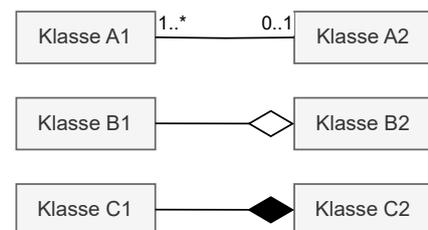
Eine Assoziation ist eine einfache Beziehung zwischen zwei Klassen, die angibt, dass Objekte einer Klasse auf Objekte einer anderen Klasse zugreifen können. Sie kann eine Leserichtung besitzen und zudem durch sogenannte Kardinalitäten definieren, wie viele Objekte einer Klasse mit Objekten einer anderen Klasse in Beziehung stehen können (vergleiche Abbildung 2.9 A). Die Vererbung stellt eine Beziehung zwischen einer Generalisierungs- und einer Spezialisierungsklasse dar, wobei die spezifische Klasse alle Eigenschaften der allgemeinen Klasse erbt, also übernimmt (vergleiche Abbildung 2.10). In diesem Kontext werden oft auch sogenannte abstrakte Klassen definiert, die zwar Attribute besitzen aber nur durch ihre Kindklassen instanziiert werden können. Ihr Klassenname wird in UML meist kursiv dargestellt. Die *Aggregation* kennzeichnet eine besondere Art der Assoziation, bei der ein Teil (in Abbildung 2.10 Klasse B2) auch unabhängig vom Ganzen (Klasse B1) existieren kann, was durch eine Linie mit einer offenen Raute symbolisiert wird (vergleiche Abbildung 2.9 B). Im Gegensatz dazu beschreibt die *Komposition* eine stärkere Bindung, bei der die Bestandteile (Klasse C2) untrennbar mit dem Ganzen (Klasse C1) verbunden sind. Dies wird durch eine Linie mit einer ausgefüllten Raute dargestellt (vergleiche Abbildung 2.9 C).



**Abbildung 2.8** Packages und Klasse in UML nach Herring, Roswell und Danko 2022



**Abbildung 2.9** Vererbung in UML nach Herring, Roswell und Danko 2022



**Abbildung 2.10** Assoziation (A), Aggregation (B) und Komposition (C) in UML nach Herring, Roswell und Danko 2022

### 2.2.2.2 BoreholeML

Das von den staatlichen geologischen Diensten Deutschland entworfene Format *BoreholeML* wurde entwickelt, um den deutschlandweiten Austausch von Bohrungsdaten unabhängig von den länderspezifischen Datenbankstrukturen zu ermöglichen (Schirren 2020; BoreholeML o. D.). Das zugrunde liegende objektorientierte Datenmodell wurde in das INSPIRE Rahmenmodell integriert und nach dessen Standards erstellt (*The Borehole Markup Language* 2010). Diesem liegt eine europäische Richtlinie zugrunde, die eine einheitliche Infrastruktur für Geodaten in der EU schaffen soll und somit zur Verbesserung der Verfügbarkeit, Qualität und Organisation räumlicher Daten beitragen soll (INSPIRE Maintenance and Implementation Group 2024). Daher wurde das Datenmodell auch gemäß bestehender Geostandards wie Metadaten (ISO 19115) oder GML (ISO 19136) formuliert und ist somit ein GML-Anwendungsschema, das als UML-Diagramm und XML-Schema abgebildet wird (BoreholeML o. D.). Metadaten beschreiben im Allgemeinen Daten über ein Datenobjekt (Kresse, Danko und K. 2022). Ein Beispiel hierfür wäre etwa das Erstellungsdatum oder der Dateiname einer Datei.

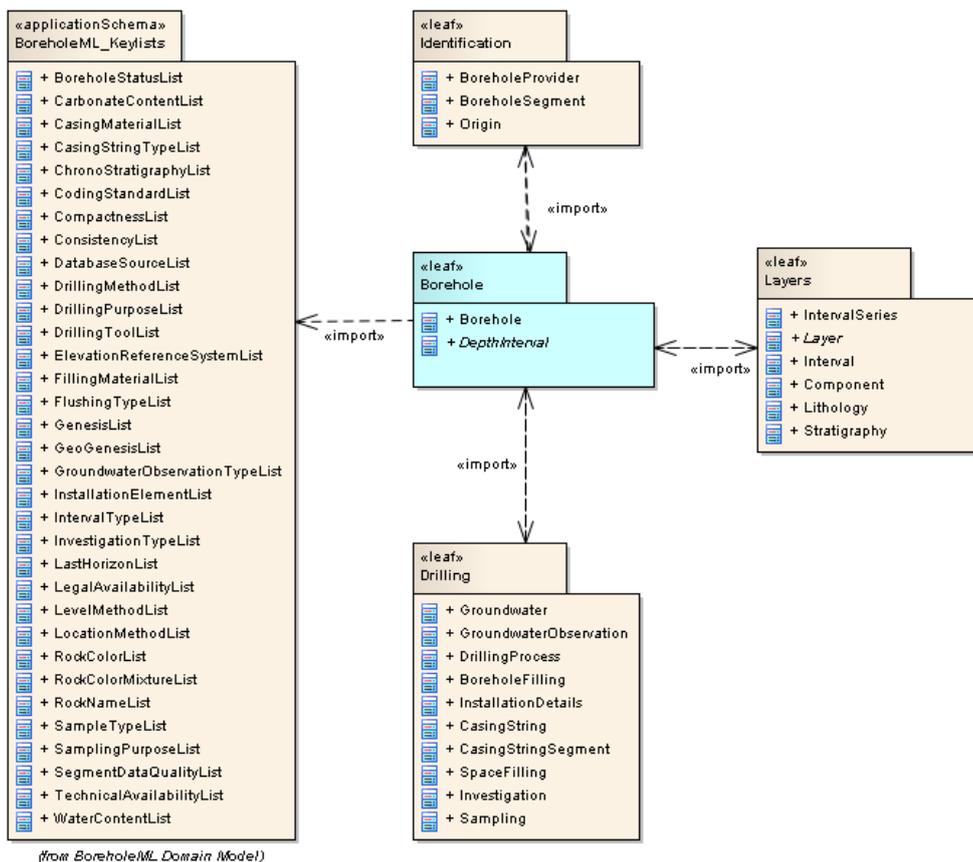


Abbildung 2.11 Packages in *BoreholeML* (entnommen aus BoreholeML o. D.)

**Aufbau des Datenmodells.** Die aktuelle Version *BoreholeML3* enthält neben Headerdaten, die beispielsweise die Koordinaten des Aufschlusspunktes enthalten, auch stratigrafische und technische Daten zur Beschreibung der Schichten und der Ausführung der Bohrung. Zudem existieren auch hier, genau wie für *SEP 3*, Codelisten zur Übersetzung der Klartext-Daten in festgelegte Codes. Insgesamt sind durch das *BoreholeML*-Datenmodell ein Großteil der Informationen abgedeckt, die das *SEP 3*-

Datenmodell beinhaltet. Somit können *SEP 3*-Daten über entsprechende Übersetzungstabellen in das andere Modell umgewandelt werden (LBEG o. D.).

Das Anwendungsschema wird in der Dokumentation in Codelisten und drei weitere Packages, sogenannte *leafs* gruppiert, die die einzelnen Elemente von *BoreholeML* enthalten (*BoreholeML* o. D.) (siehe Abbildung 2.11). Abbildung 2.12 zeigt eine Übersicht des für diese Arbeit relevanten Teils des Datenmodells. Die darin enthaltenen Elemente sollen nun kurz beschrieben werden.

Die zentrale Gruppe des Anwendungsschemas stellt das Package *Borehole* dar (in Abbildung 2.12 orange). Dieses enthält die wichtigste Komponente des Datenmodells: Den Feature Type *Borehole*, der letztlich durch zahlreiche Data Type-Elemente (nur diese sind in Abbildung 2.13 als Attribute dargestellt) auch alle Daten der Bohrung enthält. Der geometrische Verlauf der Bohrung wird hier durch einen *GML-LineString*, zusammengesetzt aus den einzelnen Bohrintervallen, gespeichert. Zusätzlich enthält dieses Package den abstrakten Datentyp *depthInterval*, der allgemein ein Intervall zwischen einer Start- und Endtiefe darstellt. Alle Elemente, die einen Tiefenabschnitt beschreiben, erben von diesem Datentyp. Dieser ist in Abbildung 2.12 jeweils separat in den Packages dargestellt, um die Vererbungsbeziehung darstellen zu können.

Zur Speicherung der Metadaten gemäß ISO 19115 dient die Untergruppe *Identification*, die neben Angaben über den Eigentümer der Bohrung (*BoreholeProvider*) auch die Koordinaten des Bohransatzpunktes in der Quelldatenbank (*Origin*) enthält. Zudem werden in der Klasse *BoreholeSegment* die dreidimensionalen Koordinaten der Bohrabschnitte gespeichert, die insbesondere für Bohrungen relevant sind, die vom Bohransatzpunkt nicht durchgehend vertikal, sondern teilweise gekrümmt verlaufen. Die Schichtdaten werden über das Package *Layers* dargestellt, wobei der Data Type *IntervalSeries* das Schichtenverzeichnis enthält, das wiederum durch eine Liste aus *Interval*-Elementen die Schichtdaten enthält. Diese können, referenziert durch das Attribut *sublayer* mehrere Unterschichten (Klasse *Component*) besitzen. *Interval* und *Component* erben von der abstrakten Klasse *Layer*, die durch ihre Attribute alle Bodeneigenschaften abbildet. Hierbei werden die Eigenschaften der Schicht hinsichtlich ihrer Lithologie (Bodenzusammensetzung) und Stratigrafie (räumliche und zeitliche Gesteinseinheit) in separierten Klassen abgebildet. Bei der Klasse *Lithology* wird dadurch die Speicherung von bis zu neun Haupt- und Nebenanteilen des Bodens in separaten Objekten ermöglicht, die inklusive ihrer Massenanteile und Farben gespeichert werden können. Von der Klasse *Stratigraphy* kann hingegen nur ein Objekt referenziert werden.

Das letzte Package *Drilling* enthält alle weiteren Daten zur Bohrung wie zum Beispiel Grundwassermessungen in *Groundwater* und *GroundwaterObservation*, Informationen zu Sondierungs-Log-Daten in *Investigation* oder Informationen zu den entnommenen Proben in *Sampling*. Die eigentlichen Log-Daten sowie die Ergebnisse der Laboruntersuchungen hierzu können jedoch nicht gespeichert werden. Zusätzlich können in diesem Package noch Informationen zum Bohrvorgang sowie zum Ausbau und zur Verfüllung des Bohrlochs hinterlegt werden (*BoreholeML* o. D.; *The Borehole Markup Language* 2010). Die entsprechenden Klassen wurden in Darstellung 2.12 herausgelassen, da sie für Bauprojekte meist keine Rolle spielen.

## 2.2 Relationale Datenbankmodelle zur Repräsentation von Baugrunddaten

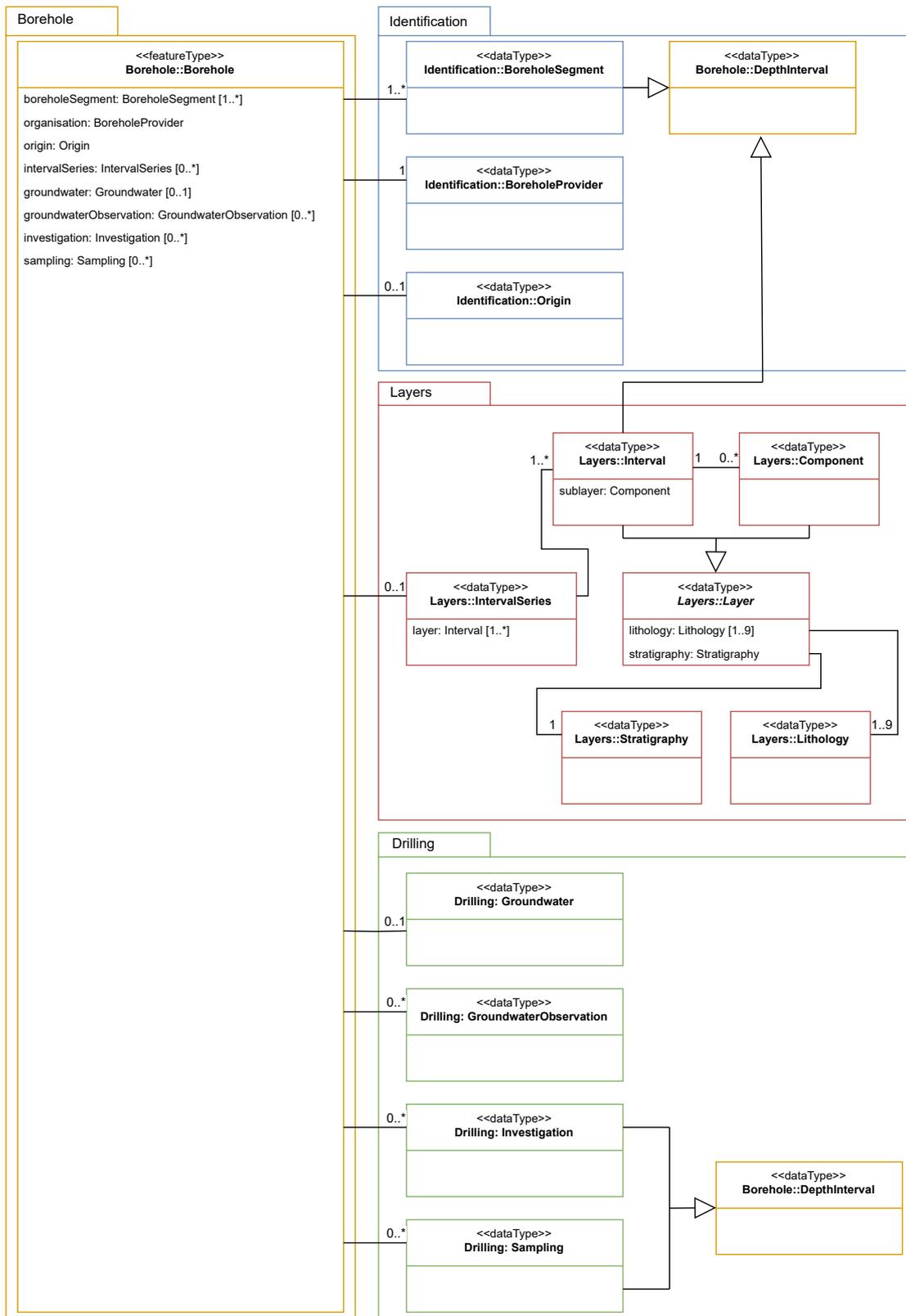


Abbildung 2.12 Teil des *BoreholeML3* Datenmodells (eigene Darstellung nach BoreholeML o. D.)

**Anwendungen des Datenmodells.** Der prominenteste Anwendungsfall des Datenmodells ist die Bohrpunktkarte Deutschland, in der die Bohrungsdaten aller geologischen Landesämter in einer Web-Oberfläche dargestellt sind. Die *BoreholeML*-Daten werden hierfür mittels eines OGC-konformen Web Feature Service (WFS) von den geologischen Landesämtern direkt zur Verfügung gestellt (Bohrpunktkarte Deutschland o. D.; Arns-Krogmann o. D.). Auf der Website der staatlichen geologischen Dienste wird das ambitionierte Ziel, „mit BoreholeML3 ein[en] Industriestandard [zu] etablieren“ formuliert (BoreholeML o. D.). Im selben Absatz wird auch die Einführung eines WFS zur direkten Abgabe von *BoreholeML*-Dateien für das Jahr 2011 angekündigt. Da dieser auf der Website nicht verlinkt ist, kann davon ausgegangen werden, dass dieses Vorhaben nicht umgesetzt wurde. Außerdem zeigt die nicht aktualisierte Ankündigung für 2011, dass die Website aktuell nicht gepflegt wird. Letztlich wird das Datenmodell in der Literatur nicht verwendet und auch Schnittstellen zu gängigen Softwareprodukten scheinen fast nicht vorhanden zu sein. Eine Anfrage an den Support der *Fugro Consult GmbH* zur Erstellung einer *BoreholeML*-Datei mit dem Programm *GeoDin* blieb unbeantwortet. Die einzige auffindbare Schnittstelle besteht zum Datenmodell *SEP 3* in Form einer Übersetzungstabelle zur Überführung des Symbolschlüssels Geologie in die Codelisten von *BoreholeML* (LBEG o. D.). Auch hier blieb eine Anfrage an die Entwickler von *SEP 3* zur konkreten Umsetzung der Umwandlung unbeantwortet. Insgesamt wird der Eindruck erweckt, dass das Datenmodell *BoreholeML* nicht weiterentwickelt wird und lediglich in der Bohrpunktkarte Deutschland Anwendung findet.

### 2.2.2.3 Geoscience Markup Language (GeoSciML)

Den Standard des OGC für die Repräsentation geologischer Daten stellt das Datenmodell GeoSciML dar (Schirren 2020). Es wurde in Zusammenarbeit der geologischen Dienste zahlreicher Länder, darunter zum Beispiel die USA, Großbritannien, Frankreich und Australien, entwickelt, um alle Teilbereiche der Geologie abzudecken. Zudem wird jedoch angemerkt, dass spezialisierte Bereiche, wie Hydrogeologie oder Geophysik, nicht im Detail durch das Modell repräsentiert werden. Stattdessen wird hier auf weitere domänenspezifische Modelle verwiesen, die gemeinsam mit GeoSciML entwickelt wurden. (Boisvert, Raymond und Sen 2017)

**Aufbau des Datenmodells.** Im Folgenden wird nur näher auf das in GeoSciML enthaltene GML-Anwendungsschema für Bohrungen eingegangen. Denn die meisten anderen Packages und Komponenten des Modells decken nur andere Bereiche der Geologie ab, die für den Praxisteil dieser Arbeit nicht relevant sind. Grundlage hierfür ist das offizielle OGC-Standard-Dokument (ebd.).

Das entsprechende Package *GeoSciML Borehole* wurde als Erweiterung des OGC-Standards *Observations, Measurements and Standards (OMS)* entwickelt und implementiert somit auch das in diesem vorgegebene konzeptionelle Modell. OMS wurde allgemein als standardisiertes Austauschformat für Beobachtungen, Messungen und Probenahmen entwickelt und wird hier für die Erfassung von Baugrunduntersuchungen verwendet (Rinne, Schleidt und Brink 2023). Neben den Schichtdaten der Bohrung können auch geologische Log-Dateien sowie weitere Details zur Bohrung repräsentiert werden. Labortechnische Untersuchungen lassen sich in einem gesonderten Package (*GeoSciML Laboratory and Analysis*) darstellen. Die eigentlichen Daten der Schichten werden dann durch eine Assoziation

des Schichtintervalls zu einem sogenannten *GFI-Feature* realisiert. Dadurch können neben der Lithologie auch weitere geologische Untersuchungsdaten repräsentiert werden. Trotzdem wird im offiziellen Standard-Dokument des OGC darauf verwiesen, dass die Klasse *GeoSciML Borehole* lediglich für Bohrungen mit geologischem Hintergrund ausgelegt ist und durch ein formelleres Modell ersetzt werden sollte, das auch in anderen Domänen (zum Beispiel Bauingenieurwesen) anwendbar ist (Boisvert, Raymond und Sen 2017).

**Anwendungen des Datenmodells.** Verwendung findet das Modell beispielsweise im Geoportal der *OneGeology*-Initiative, deren Ziel es ist, weltweit digitale geologische Karten zu erstellen und diese verfügbar zu machen (*IUGS Commission for the Management and Application of Geoscience Information (CGI)* 2024). Außerdem wurde GeoSciML als Standard für den Datentransfer geologischer Daten in der europäischen INSPIRE-Richtlinie ausgewählt (Schirren 2020).

### 2.2.2.4 Industry Foundation Classes (IFC)

Die dreidimensionale Repräsentation von Baugrunddaten ist auch im Building Information Modeling (BIM) möglich. „BIM basiert auf der durchgängigen Nutzung eines digitalen Bauwerksmodells über den gesamten Lebenszyklus eines Bauwerks – vom Entwurf, über die Planung und Ausführung bis zum Betrieb“ (Borrmann u. a. 2021). Diese Modelle integrieren nicht nur die dreidimensionale Geometrie der Bauwerke, sondern auch umfangreiche semantische Informationen. Der herstellernerneutliche Austausch dieser Modelle erfolgt über das standardisierte Datenmodell IFC, dessen Instanzen zum Beispiel durch XML (*ifcXML*), meist aber durch das Format Standard for the Exchange of Product model data (*STEP*) abgebildet werden (ebd.).

**Aufbau des Datenmodells.** Die Struktur des Datenmodells ist komplex und stark hierarchisch. Die Wurzel dieser Hierarchie stellt die abstrakte Klasse *IfcRoot* dar (vergleiche Abbildung 2.14). Dieses bildet die Grundlage für alle Entities und stellt essentielle Funktionen wie eine eindeutige Identifikation und die Verwaltung der Modifikationshistorie bereit. *IfcProduct* beschreibt Objekte wie Bauteile und räumliche Strukturen, die einem räumlichen oder geometrischen Kontext zugeordnet werden können, und ist entscheidend für die geometrische Repräsentation im Modell. Property Sets (*IfcPropertySets*) ermöglichen hingegen die Definition und Zuordnung von zusätzlichen Eigenschaften zu den Objekten, die nicht fest im IFC-Datenmodell verankert sind, und erweitern so die Flexibilität und Detailtiefe der Modellierung. (ebd.)

**Das IFC-Sub-Fachmodell zur Baugrundmodellierung.** Geotechnische Daten werden in BIM meistens in einem eigenen Fachmodell, dem Baugrundmodell, modelliert. Der Arbeitskreis 2.14 der Deutschen Gesellschaft für Geotechnik (DGGT) hat hierzu mehrere Empfehlungen herausgegeben, wovon die erste nun kurz zusammengefasst werden soll (Molzahn u. a. 2021a). Die Datengrundlage dieses Fachmodells setzt sich demnach aus zahlreichen Komponenten (siehe Abbildung 2.13) zusammen, besteht aber größtenteils aus den Informationen des geotechnischen Berichts. Wenn dieser abgeschlossen ist, sollte das Modell zumindest aus folgenden Sub-Fachmodellen bestehen, die miteinander vernetzt sind: Das Sub-Fachmodell der Aufschlüsse, das der Baugrundschichten, das Fachmodell der Homogenbereichsschichten und das der Grundwasserschichten und -körper (vergleiche Abbildung

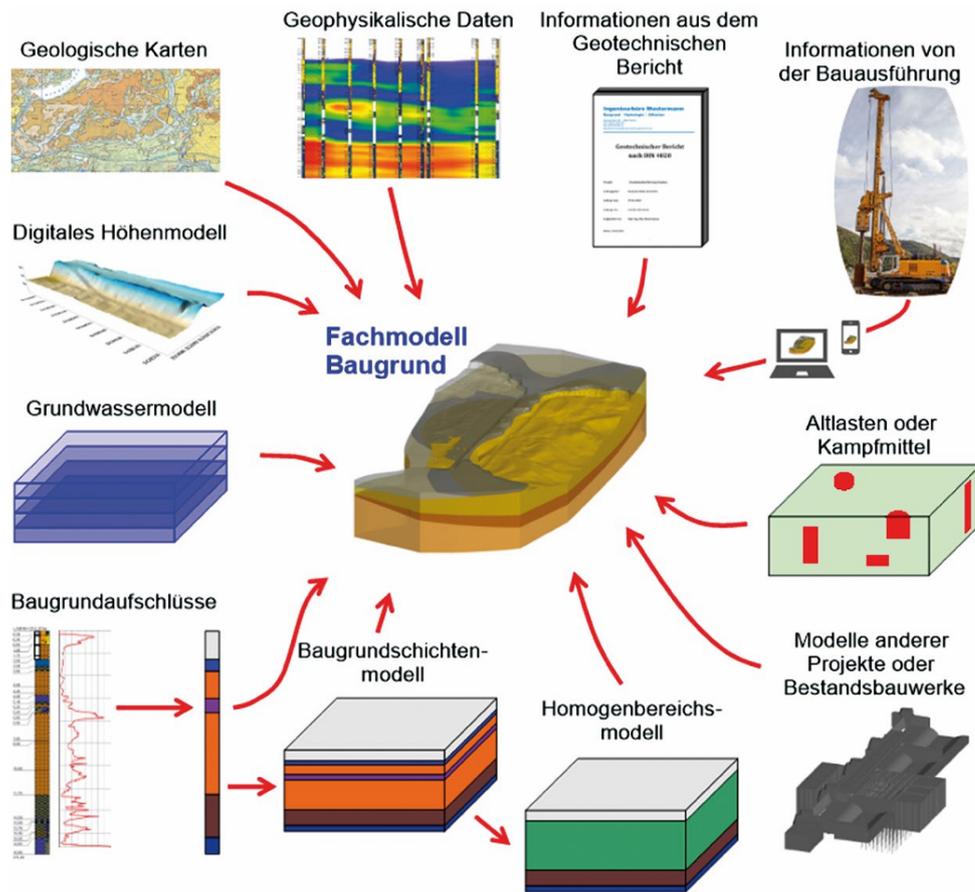
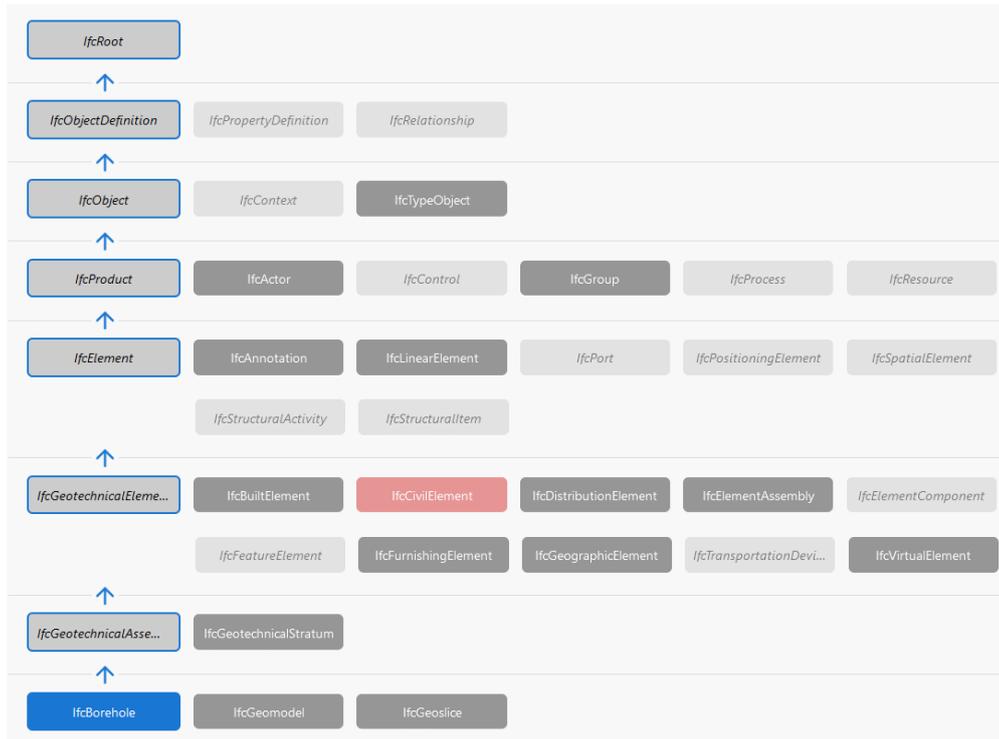


Abbildung 2.13 Fachdaten zur Bildung des Fachmodells Baugrund nach Molzahn u. a. 2021a

2.13). Die Abbildung dieser Informationen auf ein Datenmodell und später auch auf ein Datenformat soll mittels IFC erfolgen, wobei die konkreten Klassen zur Repräsentation in der Empfehlung zunächst nicht beschrieben werden.

Diese Arbeit legt den Fokus hingegen auf das Sub-Fachmodell der Baugrundaufschlüsse und strebt keine durchgehende Modellierung des Untergrunds in einem Volumenmodell an. Deshalb wird im Folgenden nur auf die Abbildung einzelner Baugrundaufschlüsse auf das IFC-Modell gemäß der Empfehlung des DGGT eingegangen. In der aktuellen Version der IFC (IFC 4x3) existieren Klassen zur Repräsentation geotechnischer Körper. Dies wird durch Unterklassen der Klasse *IfcGeotechnicalElement* bewerkstelligt. Von dieser erben zum Beispiel die Klassen *IfcGeotechnicalStratum* zur Beschreibung von Schichten oder *IfcBorehole* zur Beschreibung von Einzelbohrungen. Geotechnische Parameter der Schicht finden sich zum Beispiel in den Property Sets *PsetSolidStratumCapacity* und *PsetSolidStratumComposition* wieder, die insbesondere die Ergebnisse der Probedaten der Schichten enthalten und weniger die im Feld ermittelten Parameter der Bodenansprache. Insgesamt wird in der Dokumentation dieser Klassen auch darauf verwiesen, dass ein Attribut *PredefinedType* noch fehlt und in späteren Versionen des Modells noch hinzugefügt wird (IFC 2023). Das lässt darauf schließen, dass sich dieser Teil des Datenschemas vermutlich noch in der Entwicklung befindet.

**Anwendungen des IFC-Baugrundmodells.** Konkret angewendet wurden die Kindklassen von *IfcGeotechnicalElement* zum Beispiel von (El Sibaii u. a. 2022). Hierbei wurde ein Workflow entwickelt mit



**Abbildung 2.14** Vererbungshierarchie für *IfcBorehole* in IFC 4.3 (IFC 2023)

dem, ausgehend von einer Vorlage in Tabellenform, Daten aus traditionell erstellten geotechnischen Berichten zunächst in *IfcBorehole*-Elemente und anschließend in ein dreidimensionales Schichtenmodell transformiert werden. Aufgrund der damals fehlenden Möglichkeit des Exports der IFC-Version 4.3 in der verwendeten Software wurden nutzerspezifische Property Sets für den Export definiert, die jedoch an die offiziellen Property Sets angelehnt sind.

Aufgrund der steigendem Bedeutung der IFC und der zunehmenden Beauftragung dieser Modelle im Kontext öffentlicher Projekte ist davon auszugehen, dass sich auch die Repräsentation des Baugrunds mittels IFC zunehmend etablieren wird (Borrmann u. a. 2021).

### 2.2.2.5 City Geography Markup Language (CityGML)

Eine weitere Möglichkeit, Bohrungsdaten in einem Datenmodell zu repräsentieren, stellt der OGC-Standard *CityGML* dar. Das zugrunde liegende GML-Anwendungsschema wurde für die Repräsentation, Speicherung und den Austausch von virtuellen 3D-Stadt- und Landschaftsmodellen entwickelt. Es kann durch sogenannte Application Domain Extensions (ADEs) für weitere spezifische Anwendungen erweitert werden. Diese Möglichkeit wurde auch in mehreren Fällen zur dreidimensionalen Repräsentation geotechnischer Daten verwendet (Tegtmeier u. a. 2014; Khan, Kim und Seo 2023).

**Erweiterung von CityGML durch ADEs.** Nachfolgend soll die Vorgehensweise im ersten der beiden Artikel kurz erläutert werden. *Tegtmeier u. a. 2014* entwickelte eine *CityGML*-ADE mit dem Namen „3D-GEM“ (kurz für „3D-Geotechnical Extension Model“), für die dreidimensionale Modellierung des Baugrunds im Kontext von Infrastrukturprojekten. Zunächst wurde ausgehend von niederländischen und britischen Standards ein Modell entwickelt, um die rohen Ergebnisse geotechnischer Untersu-

chungen zu modellieren. Anschließend wurde das Format *GeoSciML* als Inspiration für die Repräsentation unterirdischer Körper herangezogen, da das Modell den Baugrund als eine Zusammensetzung unterirdischer Volumenkörpern darstellt, die die einzelnen Schichten abbilden. Diese werden als homogen betrachtet und geometrisch durch ihre Oberflächen repräsentiert. Die finale ADE schließt an das *CityGML*-Objekt *geology* mit dem von *GeoSciML* übernommenen Feature Type *GeologicFeature* und dessen Kindklassen an. Von diesem erbt folglich die neu definierte Klasse *Engineering Geology*, die weitere Kindklassen zur Modellierung der Schichten, Hohlräume und Hindernisse des Untergrunds besitzt. So kann der Untergrund zusammen mit zahlreichen überirdischen Elementen der Umwelt in einem gemeinsamen Datenmodell und inklusive Semantik und Messungen repräsentiert werden. Das Konzept wurde jedoch nur teilweise in der Literatur aufgegriffen und nicht speziell für eine Software-Anwendung entwickelt (Biljecki, Kumar und Nagel 2018). Somit ist eine Anwendung in der Praxis hier vermutlich nicht gegeben.

### 2.2.2.6 Integration von IFC und CityGML

In vielen Fällen ergänzen sich die Welten des BIM und Geographic Information Modeling (GIM), da räumliche Daten aus Geoinformationssystemen zum Beispiel oft in der Planung von Gebäuden und Infrastruktur benötigt werden (Blankenbach und Becker 2022). Somit ergibt sich zunehmend ein Bedarf der Integration beider Systeme, die aufgrund der Verschiedenheit der Modellierungsparadigmen durchaus herausfordernd sein kann (Borrmann u. a. 2021). Hier müssen zum Beispiel die verschiedenen Arten der 3D-Geometriedarstellung beachtet werden. Während in BIM der Einsatz von Volumen-geometrien in einem lokalen Koordinatensystem üblich ist, wird bei GIM meist die Repräsentation dreidimensionaler Oberflächen in einem globalen Koordinatensystem bevorzugt (Blankenbach und Becker 2022). Generell wird zwischen vier Möglichkeiten dieser Integration unterschieden. Die Umwandlung des BIM-Modells in ein GIM-Modell, die Umwandlung eines GIM-Modells in ein BIM-Modell, die einheitliche Repräsentation beider Modelle in einem neuen Modell und die Verlinkung beider Modelle (Hijazi und Donaubaue 2017).

Die beschriebene Integration lässt sich wiederum auf die Modellierung von Baugrunddaten übertragen, die sich, wie in den vorherigen Absätzen beschrieben, sowohl in BIM- als auch in GIS-Formaten umsetzen lässt. Die Integration von IFC und CityGML wurde in diesem Kontext von Khan, Kim und Seo 2023 umgesetzt. Der Artikel beschreibt zunächst die Darstellung des Untergrunds in Form von dreidimensionalen Pixeln, sogenannten Voxeln. Hiermit soll die Heterogenität des Untergrunds in einem Volumenmodell besser repräsentiert werden als in der bisherigen Darstellung homogener Untergrundschichten, die durch ihre Oberfläche repräsentiert werden. Folglich definiert Khan jeweils eine Erweiterung der Datenmodelle IFC und CityGML, um die vorgeschlagene Geometrie inklusive Semantik abzubilden und zeigt eine Möglichkeit zur Umwandlung von IFC nach CityGML auf. Eine anschließende Umfrage unter Bauprojektbeteiligten belegte die Vorteile dieser neuen Methode zur dreidimensionalen Repräsentation mittels etablierter Datenmodelle gegenüber der herkömmlichen Herangehensweise bei geotechnischen Untersuchungen. Da es sich jedoch auch hier zunächst um eine wissenschaftlich-theoretische Arbeit handelt, sind Schnittstellen etablierter Geotechnik-Software zu dieser Erweiterung nicht vorhanden.

## 2.3 Speicherung und Darstellung räumlicher Daten

### 2.3.1 Geoinformationssysteme

Neben der Repräsentation der Baugrunddaten in einem Datenmodell, muss es auch eine Möglichkeit geben, diese in einem System zu speichern und sie für Nutzer und Nutzerinnen zugänglich zu machen. Aufgrund des Raumbezugs der Baugrundaufschlüsse in Form von Koordinaten und der Vielfalt dieser Daten, liegt hierfür die Verwendung eines Geoinformationssystems nahe (Chang und Park 2004). Dieses „dient der Erfassung, Speicherung, Verarbeitung und Darstellung aller Daten, die einen Teil der Erdoberfläche [...] beschreiben“ (Bartelme 1989). Üblicherweise besteht ein Informationssystem im Allgemeinen aus einer Datenbank, die von einem Datenbankmanagementsystem (DBMS) verwaltet wird und aus Werkzeugen, die dem Anwender oder der Anwenderin zur Analyse und Manipulation der Daten zur Verfügung stehen (Zehnder 1998). Die zugrundeliegenden Daten eines GIS, auch räumliche Daten genannt, werden entweder als diskrete Objekte in Form von Vektordaten oder als kontinuierliche Felder in Form von Rasterdaten repräsentiert. Vektorisierte Daten bestehen dabei aus einzelnen Punkten, die zu Linien und Flächen zusammengesetzt werden können (Bartelme 2022).

### 2.3.2 Geodatenbanken

Konkret gespeichert werden große Mengen räumlicher Daten inklusive ihrer Semantik meist in Geodatenbanken. Dadurch kann eine effiziente Abfrage sowie die Abbildung von Datenmodellen ermöglicht werden (Brinkhoff 2013). Diese Abfragen werden in Datenbanken meist durch die Sprache Structured Query Language (SQL) ausgedrückt. SQL-Ausdrücke können in sogenannten Sichten (engl. Views) als virtuelle Tabellen definiert werden. Diese Sichten können in Abfragen wie reguläre Tabellen verwendet werden, enthalten jedoch keine physischen Daten, da sie bei einer Abfrage die Daten aus den zugrundeliegenden Relationen dynamisch abrufen.

Im Kontext von Datenbanken wird nun zusätzlich das Konzept der ersten Normalform eingeführt, die als Gütekriterium eines Datenbankschemas gilt. Ein Schema befindet sich in der ersten Normalform, wenn alle Attribute atomare Domänen haben, also nur nicht weiter unterteilbare Werte enthalten. (Studer 2019)

Die in Abschnitt 2.2 erklärten Konzepte von relationalen und objektorientierten Datenmodellen spiegeln sich auch in der Architektur von Datenbanken wieder. Während relationale Datenbanken Daten in Tabellen mit Zeilen und Spalten organisieren, speichern objektorientierte Datenbanken diese in Form von Objekten, die Konzepte wie Vererbung und Kapselung möglich machen. Geodatenbanken verwenden oft sogenannte objektrationale Datenbankmanagementsysteme, bei der die Funktionalität relationaler Datenbankmanagementsysteme durch komplexe Datentypen, zum Beispiel zur Repräsentation einer vektorisierten Linie, erweitert wird (Brinkhoff und Kresse 2022). Zur Repräsentation von Geometrien existieren standardisierte Datenmodelle, die den Austausch und die Interoperabilität zwischen verschiedenen Anwendungen ermöglichen. Instanzen dieser Klassen können durch eine menschenlesbare (Well-known text (WKT)) oder durch eine binäre Darstellung (Well-known binary (WKB)) repräsentiert werden (Portele 2022). Hierbei muss auch immer das zugehörige Koordinatenreferenzsystem (CRS) mitgespeichert werden, was meistens durch den European Petroleum Survey Group (EPSG)-Code erfolgt, da diese geodätische Referenzsysteme eindeutig identifizieren (O'Brien und Lott 2022).



# 3 Anforderungsanalyse anhand von Anwendungsszenarien

## 3.1 Datengrundlage und bestehende Software

In diesem Kapitel werden die Anforderungen an die angestrebte Repräsentation von Baugrunddaten in einem Infrastrukturprojekt der DB InfraGO AG erarbeitet. Dabei fließen sowohl allgemeine Anforderungen, die sich aus der Datengrundlage und der bereits eingesetzten Software ergeben, als auch die Ergebnisse aus zwei Interviews mit Geotechnik-Ingenieuren und -Ingenieurinnen ein. Daraus sollen anschließend Anwendungsszenarien abgeleitet werden, die die Grundlage der weiteren Arbeit darstellen werden.

### 3.1.1 Datengrundlage

Zunächst folgt ein Überblick über die zu repräsentierenden geotechnischen Daten sowie zu bereits vorhandener Software, um die technischen Voraussetzungen und die Basis der weiteren Untersuchungen zu definieren.

#### 3.1.1.1 Geotechnische Daten

Als Datengrundlage dienen alle direkten und indirekten Aufschlüsse eines circa sieben Kilometer langen Abschnitts des erwähnten Projekts der DB. Die Stammdaten des Datensatzes lagen zunächst nur in tabellarischer Form und alle restlichen Daten in Form von PDF-Dokumenten vor. Die Digitalisierung der Daten erfolgte kurz vor Start dieser Arbeit mithilfe des Baugrundprogramms *GeoDin* in einer lokalen *Microsoft Access* Datenbank. *GeoDin* diente hierbei auch zur Plausibilitätsprüfung der eingegebenen Baugrunddaten.

Die direkte Aufschlüsse umfassen Erkundungsbohrungen (BK), Kleinrammbohrungen (KRB) sowie Schürfe (SCH). Zusätzlich wurden ergänzend Schwere Rammsondierungen (DPH) sowie Drucksondierungen (CPT) als indirekte Aufschlüsse vorgenommen (siehe Tabelle 3.1). In der *GeoDin*-Datenbank, die als primäre Grundlage dieser Arbeit dienen soll, befinden sich alle Daten der Bodenansprache, der Rammsondierungen, der Grundwasserstände sowie die Stammdaten der Aufschlüsse. Die Position der Aufschlüsse wurde mithilfe eines GPS-Geräts im DB-eigenen Koordinatenreferenzsystem DB\_Ref ermittelt und für die Verwendung in *GeoDin* in das System Gauß-Krüger Zone 4 (GK 4) transformiert. Die Ergebnisse der Laboruntersuchungen sowie der Drucksondierungen wurden hingegen erst später hinzugefügt und dienen somit nicht als Grundlage dieser Arbeit.

### 3 Anforderungsanalyse anhand von Anwendungsszenarien

Direkte Aufschlüsse		Indirekte Aufschlüsse	
Aufschlusstyp	Anzahl	Aufschlusstyp	Anzahl
Kleinrammbohrung	146	Schwere Rammsondierung	111
Erkundungsbohrung	79	Standard Penetration Test	0
Schurf	43	Drucksondierung	0

**Tabelle 3.1** Anzahl der direkten und indirekten Aufschlüsse in der verwendeten *GeoDin*-Datenbank

#### 3.1.1.2 Weitere Projektdaten

Zusätzlich stehen weitere digitale Planungsdaten des Projekts zur Verfügung, die im Kontext geotechnischer Daten interessant sein können. Hiermit sind insbesondere BIM-Modelle aller Gewerke der Eisenbahninfrastruktur gemeint. Diese stehen projektintern im IFC-Format zur Verfügung.

#### 3.1.2 Bestehende Software

##### 3.1.2.1 GeoDin

*GeoDin* ist eine Geodatenmanagementsoftware zur Verwaltung, Erfassung und Visualisierung geotechnischer Daten (Fugro Consult GmbH o. D.). Sie wird seit 1994 von der *Fugro Consult GmbH* entwickelt und zur Erstellung von Bohrdatenbanken verwendet (Schirren 2020). Diese können entweder lokal in einer *Microsoft Access* Datenbank oder auf einem Server gespeichert werden. Die Datenerfassung in einer *Microsoft Access* Datenbank ist auch mit der kostenlosen Version *GeoDin Shuttle* möglich, die rein zur Dateneingabe entwickelt wurde. (Fugro Consult GmbH o. D.)

### 3.2 Anforderungen

#### 3.2.1 Allgemeine Anforderungen

Konkrete Anforderungen an die Speicherung und Darstellung von Baugrunddaten in Infrastrukturprojekten sind in der Literatur nur selten zu finden. Deshalb wird im Folgenden ein Überblick über die entsprechenden Vorgaben zu BIM-Fachmodellen für den Baugrund gegeben. In dieser Arbeit wird lediglich die Repräsentation der Baugrundaufschlüsse und nicht des gesamten Baugrunds als Volumenmodell angestrebt. Ersteres entspricht etwa dem in Kapitel 2 genannten Sub-Fachmodell der Baugrundaufschlüsse, weshalb anschließend auch nur die Anforderungen für dieses Teilmodell behandelt werden. Dazu dient das zweite Empfehlungsschreiben des Arbeitskreises 2.14 der DGGT, welches datentechnische Anforderungen an die Modellierung von Eigenschaften in einer Datenbank für Baugrunddaten im BIM-Kontext definiert (Molzahn u. a. 2021b). Jede repräsentierte Eigenschaft soll demnach einen global eindeutigen Bezeichner erhalten, nach Typ (Parameter, Attribut, Referenz) kategorisiert und mit einem standardisierten Datentyp versehen werden. Für die Modellierung von Eigenschaften ist es wichtig, dass jeder Eigenschaftsname eindeutig und in einer passenden Kategorie organisiert ist, der Datentyp festgelegt wird, und die Einheit des Zahlenwerts nach SI-Standards definiert ist. Textwerte sind nur in Ausnahmefällen zulässig und sollten bei Mehrfachverwendung in Codelisten referenziert und nicht in der Ursprungsform gespeichert werden. Schließlich muss jede Eigenschaft eine Beschreibung und mindestens ein Beispiel enthalten, um die Anwendung in der Praxis zu erleichtern.

Ferner sollten öffentliche Auftraggeber des Infrastrukturbaus gemäß des Stufenplans für digitales Bauen des Bundesministeriums für Verkehr und digitale Infrastruktur (BMDV) bevorzugt auf herstellernerneutrale Datenformate zurückgreifen („Stufenplan Digitales Planen und Bauen“ 2015). Dies ist in diesem Fall zwar auf BIM-Anwendungen bezogen, lässt sich jedoch auch auf andere digitale Planungsdaten und somit auch auf die Wahl des Datenmodells übertragen.

Neben diesen Anforderungen aus der Literatur muss zudem noch die eingeschränkte Verfügbarkeit von Software im Projekt berücksichtigt werden. Somit sollte im Idealfall neben *GeoDin* auf Free Open Source Software zurückgegriffen werden. Leistungsstarke Extract-Transform-Load-Tools zur Transformation der Daten in komplexe Datenmodelle, wie die Software *FME*, stehen demnach nicht zur Verfügung.

Insgesamt wird eine Lösung angestrebt, die in der Praxis anwendbar ist, weshalb im Rahmen dieser Arbeit nur auf bereits verfügbare oder Open Source Software zurückgegriffen werden soll.

### 3.2.2 Anforderungen des Geotechnik-Teams

Für die Anforderungsanalyse wurden zwei Interviews Geotechnik-Ingenieuren geführt, die in dem DB-Projekt tätig sind. Die entsprechenden Transkripte dieser Gespräche können im Anhang eingesehen werden.

#### 3.2.2.1 Interview 1

Das erste Interview (siehe Anhang) wurde mit einer externen Baugrundexpertin geführt. Sie initiierte die Nutzung des Baugrundprogramms *GeoDin* im Projekt und hob hervor, dass dieses zur Eingabe der Daten, zur Verwaltung der Datenbank und für die Bearbeitung geotechnischer Aufgaben weiterhin verwendet werden soll. Somit sollte die in dieser Arbeit angestrebte Repräsentation der Daten vor allem als Ergänzung dienen und Aspekte abdecken, die von *GeoDin* nicht bedient werden. Die dreidimensionale Visualisierung der Bohrprofile in einer Karte wurde besonders dann als nützlich erachtet, wenn diese gemeinsam mit anderen Planungsdaten, insbesondere mit Modellen der BIM-Planung dargestellt werden. Die Darstellung solle insbesondere die Baugrundsichten in einer verständlichen Symbolik sowie die Daten der Bodenansprache erhalten. Die Relevanz der rohen Schlagzahlen der Rammsondierungen wurde hingegen als weniger hoch erachtet. Auf die Frage, welche Filterfunktionen einen Mehrwert bieten könnten, wurde eine räumliche Filterung auf eine Teilmaßnahme, sogenannte Teilentwurfshäfte, vorgeschlagen. Die Abgrenzung dieser Bereiche wird bei Bahn-Projekten oft anhand des Bahnkilometers vorgenommen. Als weitere Anwendung wurde eine automatische Plausibilitätsprüfung der Ergebnisse von Feld- und Laborversuchen genannt, die in einer solchen Datenbank vorgenommen werden könnte. Schließlich wurde hervorgehoben, dass die Daten primär von externen Planern benötigt werden, die auf Grundlage der Daten zum Beispiel Bauwerke planen. Hierfür wäre ein Export editierbarer Daten aus der Datenbank von Interesse.

#### 3.2.2.2 Interview 2

Das zweite Interview (siehe Anhang) wurde mit dem Geotechnik-Team eines anderen Abschnitts des Projekts, bestehend aus einem weiteren externen Baugrundexperten und einem auf Baugrund spezialisierten Projektmanagers, geführt. Beide haben bisher nicht mit *GeoDin* gearbeitet, äußerten aber

allgemein Interesse an einer Baugrund-Datenbank, die alle erfassten Daten zentral bündelt. Diese sollte neben den Schichtinformationen auch die gemessenen Grundwasserstände, sowie die Schlagzahlen der Rammsondierungen enthalten. Eine dreidimensionale Darstellung wurde vor allem im Kontext anderer dreidimensionaler Modelle der Planung als besonders gewinnbringend bezeichnet. Zusätzlich wäre die Abfrage der einzelnen Aufschlüsse in einer zweidimensionalen Karte von besonderem Interesse. Als Visualisierung wird die konventionelle Darstellung in Form von normgerechten Bohrprofilen und Schlagzahldiagrammen einer dreidimensionalen Visualisierung vorgezogen.

#### **3.2.2.3 Zusammenfassung der Ergebnisse**

Insgesamt besteht von mehreren Seiten der Wunsch nach einer zentralen Verwaltung der Baugrunddaten, die auf *GeoDin* basieren soll und möglichst alle Daten der Feld- und Laboruntersuchungen enthalten soll. Die Ergebnisse der Untersuchungen sollten nach Norm in einer Karte dargestellt werden, wobei eine dreidimensionale Darstellung vor allem im Kontext anderer Planungsdaten von Vorteil ist. Schließlich müssen die Daten in editierbarer Form aus der Datenbank exportiert werden können, um sie externen Planern als Planungsgrundlage zur Verfügung zu stellen.

#### **3.2.3 Anforderungen des BIM-Teams**

Da alle BIM-Daten, wie im Stufenplan Digitales Planen und Bauen gefordert, im Format IFC übergeben werden, steht dem BIM-Team vor allem Software zur Visualisierung von 3D-Modellen in diesem Format zur Verfügung.

### **3.3 Ableitung von Anwendungsfällen**

Ausgehend von den vorher beschriebenen Anforderungen sollen nun drei Anwendungsszenarien für eine Repräsentation der Baugrunddaten abgeleitet werden. Die Grundlage für alle drei Szenarien stellt eine Datenbank dar, die auf Grundlage eines zuvor gewählten Datenmodells aufgebaut werden soll und nun als Teil des ersten Anwendungsfalls betrachtet wird.

#### **3.3.1 Anwendungsfall 1: Abfragen von Aufschlussesdaten mithilfe einer interaktiven dreidimensionalen Karte**

Für das erste Szenario muss zunächst ein mit *GeoDin* kompatibles Datenmodell gewählt werden, da die Daten weiterhin in *GeoDin* eingegeben und verwaltet werden sollen. Dieses muss neben den Ergebnissen der Bodenansprache auch die Stammdaten, die Daten der Rammsondierungen, der Grundwasserstände und die Labordaten abbilden können. Da die zweidimensionale Visualisierung der Daten sowie die Darstellung in Bohrprofilen gemäß der einschlägigen Normen durch *GeoDin* schon gegeben ist, sollen die Informationen in diesem Szenario dreidimensional in einer interaktiven Karte dargestellt werden (Fugro Consult GmbH o. D.). Nutzerinnen und Nutzer sollen in der Lage sein, spezifische Baugrunddaten gezielt nach geografischen Abschnitten, wie Bahnkilometern, oder nach Teilprojekten, wie Teilentwurfsheften, zu filtern und anzuzeigen. Somit muss das Datenmodell zudem eine gewisse Erweiterbarkeit aufweisen, sodass die hierfür notwendigen Bahn-spezifischen Daten integriert werden können.

### **3.3.2 Anwendungsfall 2: 3D-Darstellung von Aufschlussdaten und BIM-Modellen**

Der zweite Anwendungsfall behandelt die gewünschte gemeinsame Darstellung der Aufschlussdaten mit bereits vorhandenen BIM-Modellen. Hierfür muss mindestens einer der beiden Datensätze in ein anderes Format umgewandelt werden, um beide gemeinsam in einem im Projekt vorhandenen Programm darstellen zu können. Diese Darstellung soll die Baugrundsichten in einer verständlichen Symbolik und mit den relevanten geotechnischen Attributen, wie der Bodenansprache, wiedergeben.

### **3.3.3 Anwendungsfall 3: Export einzelner Aufschlüsse für den Austausch**

Im dritten Anwendungsszenario steht die Bereitstellung einer Funktion im Vordergrund, die den Export spezifischer Baugrundaufschlüsse zur weiteren Verwendung für externe Planer ermöglicht. Zunächst muss hierfür eine Funktion zur Auswahl der gewünschten Aufschlüsse möglich sein. Diese müssen dann in ein standardisiertes, herstellernertrales Format zur Übertragung transformiert werden, das idealerweise mit gängiger Geotechnik-Software kompatibel sein sollte.



# 4 Wahl eines Datenmodells und Datentransformation

## 4.1 Vergleich der Datenmodelle für die Anwendungsfälle

In diesem Abschnitt werden die Datenmodelle für die einzelnen Anwendungsszenarien gewählt, um eine solide Grundlage für die Implementierung zu schaffen. Da die einzelnen Use-Cases unterschiedliche Anforderungen an das zugrunde liegende Datenmodell stellen, wird die Wahl des Modells für jedes Szenario separat vorgenommen. Es ist jedoch erforderlich, ein grundlegendes Datenmodell zu bestimmen, von dem aus die Transformation in die jeweiligen spezifischen Modelle erfolgen kann. Dieser Aspekt wird in Abschnitt 4.1.1 behandelt.

Im Vergleich stehen alle Datenmodelle, die in Abschnitt 2.2 beschrieben wurden (siehe Tabelle 4.1). Diese sind allesamt herstellerneutral und können entweder geologische, geotechnische oder allgemein räumliche Entitäten der Umwelt abbilden. Da das ursprüngliche Konzept von CityGML die Abbildung von Baugrunddaten nicht einschließt, wird hier zudem die von (Tegtmeier u. a. 2014) entwickelte Erweiterung *3D-GEM* betrachtet. Bei der Betrachtung von IFC werden hingegen ausschließlich die Unterklassen von *IfcGeotechnicalElement* sowie die entsprechenden Property Sets berücksichtigt, während die von (Khan, Kim und Seo 2023) vorgeschlagene Erweiterung nicht in die Analyse einbezogen wird.

### 4.1.1 Anwendungsfall 1

Zunächst ist es erforderlich, die einzelnen Datenmodelle hinsichtlich ihrer Fähigkeit zu prüfen, die relevanten Daten abzubilden. Tabelle 4.1 illustriert, dass Geometrie-, Schicht-, Grundwasser- und Probenaten prinzipiell in allen Datenmodellen abgebildet werden können. Die Speicherung roher Sondierungsdaten ist jedoch ausschließlich in *SEP 3* möglich. Dieses Modell kann allerdings weder Homogenbereiche noch dreidimensionalen Volumenmodelle abbilden. Diese beiden Teilbereiche spielen in dieser Arbeit jedoch nur eine untergeordnete Rolle, da sie nicht Teil der primären Datengrundlage sind und eine Umsetzung eines dreidimensionalen Volumenmodells nicht angestrebt wird (siehe grau markierte Zellen in Tabelle 4.1).

Neben der grundlegenden Datendarstellung ist auch die Erweiterbarkeit um Bahn-spezifische Attribute wie die Angabe des Bahnkilometers von Bedeutung. Die Datenmodelle CityGML und IFC bieten in ihrer Konzeption bereits Möglichkeiten zur Erweiterung durch ADEs (CityGML) und *Property Sets* (IFC) vor (Borrmann u. a. 2021; Gröger u. a. 2012). Ebenso können *BoreholeML* und *GeoSciML* aufgrund ihrer offen dokumentierten Struktur inklusive der Codelisten einfach ergänzt werden (*BoreholeML* o. D.; Boisvert, Raymond und Sen 2017). Im Gegensatz dazu sieht das relationale Datenmodell *SEP 3* keine Erweiterungen der bestehenden Relationen und Codelisten vor, weshalb alternative Erweiterungsmöglichkeiten erforderlich wären (Holmok 2023).

	SEP 3	BoreholeML	GeoSciML	CityGML (+ADE)	IFC 4.3
<b>Geometrie</b>	✓	✓	✓	✓	✓
<b>Schichtdaten</b>	✓	✓	✓	✓	✓
<b>Sondierungsdaten</b>	✓				
<b>Grundwasserdaten</b>	✓	✓	✓	✓	✓
<b>Probendaten</b>	✓		✓	✓	✓
<b>Homogenbereiche</b>			✓	✓	✓
<b>Baugrundmodell</b>			✓	✓	✓

**Tabelle 4.1** Vergleich der Datenmodelle im Hinblick auf die Repräsentation von Baugrunddaten

Im Folgenden werden die Datenmodelle bezüglich ihrer Eignung gemäß der in Tabelle 4.2 aufgelisteten Anforderungen jeweils einzeln bewertet.

Das Datenmodell **BoreholeML** wurde speziell für den deutschlandweiten Austausch von Bohrungsdaten entwickelt und zielt primär auf die Abbildung von Schichtinformationen im Anwendungskontext der geologischen Landesämter ab (BoreholeML o. D.). Daher ist es nicht unbedingt für die zentrale Verwaltung von Baugrunddaten eines Bauprojekts ausgelegt. Dies zeigt sich auch darin, dass die für Baumaßnahmen typischen Rammsondierungsdaten nicht als Rohdaten in dem Format integriert werden können (ebd.).

Ähnlich verhält es sich mit **GeoSciML**, dem OGC-Standard für geologische Daten. Dieses bildet primär geologische und weniger geotechnische Inhalte ab (Tegtmeier u. a. 2014). Zudem wird in der Dokumentation des Packages *Borehole* explizit darauf hingewiesen, dass dieses derzeit nur als Platzhalter für ein erweitertes Package dient, das auch die Bedürfnisse anderer Domänen berücksichtigt (Boisvert, Raymond und Sen 2017). Somit ist die Eignung dieses Datenmodells für den gegebenen Anwendungsfall ebenfalls nur eingeschränkt gegeben.

Ferner kann das Datenmodell **IFC**, das zur umfassenden Abbildung aller Bestandteile eines Bauwerks entwickelt wurde, auch zur Abbildung von Baugrunddaten verwendet werden. Obwohl entsprechende Klassen vorhanden sind, liegt der Fokus dieses Modells stärker auf den für geotechnische Zwecke ermittelten Labordaten und weniger auf den meistens als Text codierten Ergebnissen der Bodensprache (IFC 2023). Da diese jedoch den Großteil der in Kapitel 3.3.3 beschriebenen Datengrundlage ausmachen, erweist sich dieses Modell als eher ungeeignet für den spezifischen Anwendungsfall.

Anders verhält es sich mit dem Datenmodell **CityGML** inklusive der ADE *3D-GEM*. Die Erweiterung wurde gezielt für die dreidimensionale Repräsentation des Baugrunds bei Infrastrukturprojekten entwickelt und bildet somit, mit Ausnahme der rohen Sondierungsdaten, alle hierfür benötigten Parameter ab (Tegtmeier u. a. 2014). Jedoch besteht ein signifikantes Problem in der mangelnden Kompatibilität dieses Modells mit geotechnischer Software, insbesondere mit *GeoDin*. Die ADE wurde bis jetzt nur exemplarisch implementiert und wird daher von keiner Geotechnik-Software unterstützt (Biljecki, Kumar und Nagel 2018). Folglich wäre eine umfangreiche Transformation der Daten aus einem mit *GeoDin* kompatiblen Modell in das CityGML-ADE-Schema erforderlich. Dies ist insbesondere im Hinblick auf die eingeschränkte Verfügbarkeit entsprechender Software-Tools (siehe Abschnitt 3.1.2) mit erheblichem Aufwand verbunden. Dieses Problem tritt, abgesehen von *SEP 3*, auch bei allen anderen Datenmodellen auf.

Das relationale Datenmodell **SEP 3** hat den Vorteil, dass es direkt in *GeoDin* eingebunden werden

kann, wodurch keine zusätzliche Transformation der Daten notwendig ist. Dies ermöglicht eine kontinuierliche Aktualisierung der Datenbank bei der Eingabe neuer Daten (Holmok 2023). Dieser wesentliche Vorteil muss jedoch gegen die Nachteile des Formats abgewogen werden. Kritische Punkte umfassen insbesondere das grundlegende Konzept des Datenmodells. In der Kritik steht beispielsweise die Repräsentation der Petrographie einer Schicht. Haupt- und Nebenanteile werden in Textform innerhalb eines Attributs gespeichert, wodurch dieses keine atomare Domäne besitzt. Folglich erfüllt das Datenbankschema nicht die erste Datenbanknormalform (Schirren 2020). Zudem erfolgt die eindeutige Identifikation und Referenzierung einzelner Datensätze lediglich durch die Kombination mehrerer Attribute und nicht durch einen Primärschlüssel (LBEG 2022). Insgesamt müssen die Anforderungen

	BoreholeML	GeoSciML	IFC 4.3	CityGML (+ADE)	SEP 3
<b>Kompatibilität mit <i>GeoDin</i></b>	-	--	--	--	++
<b>Abb. der Datengrundlage</b>	+	+	-	+	++
<b>Konzept des Datenmodells</b>	+	+	+	+	-
<b>Erweiterbarkeit</b>	+	+	++	++	-

**Legende:**

- ++ : Erfüllt die Anforderungen vollständig
- + = Erfüllt die Anforderungen größtenteils
- = Erfüllt die Anforderungen teilweise
- = Erfüllt die Anforderungen nicht

**Tabelle 4.2** Vergleich der Datenmodelle für Anwendungsfall 1

des Geotechnik-Teams vorrangig berücksichtigt werden. Da der Schwerpunkt auf der Kompatibilität mit *GeoDin* und der vollständigen Abbildung der Datengrundlage liegt, wird daher das Modell *SEP 3* für das erste Anwendungsszenario bevorzugt (siehe grüne Spalte in Tabelle 4.2). Somit wird es trotz der Mängel im Konzept des Datenmodells und der schlechten Erweiterbarkeit als Grundlage für die weiteren Anwendungsszenarien und für die exemplarische Umsetzung in Kapitel 5 dienen. Für die Umsetzung der Abfragen und für die Speicherung der Daten soll eine Datenbank dienen. Diese stellt wiederum die Grundlage für die dreidimensionale Darstellung der Daten dar, die mittels eines Geoinformationssystems erfolgen soll. Dementsprechend werden die Daten in den folgenden Abschnitten auch als GIS-Daten bezeichnet.

#### 4.1.2 Anwendungsfall 2

Im zweiten Anwendungsszenario wird beabsichtigt, die Daten aus der *SEP 3*-Datenbank zusammen mit den BIM-Modellen des DB-Projekts darzustellen. Von den von Hijazi und Donaubaue 2017 dargestellten vier Möglichkeiten der BIM-GIS-Integration sollen nun zwei Transformationsvarianten diskutiert werden. Die erste Möglichkeit umfasst die Transformation der BIM-Modelle in ein GIS-Format, während die zweite Option die Umwandlung der *SEP 3*-Daten in ein BIM-Format vorsieht.

Angesichts der Vielzahl an BIM-Modellen unterschiedlicher Gewerke und der fehlenden geeigneten Softwarelösungen für eine BIM-GIS-Integration innerhalb des DB-Projekts wird die zweite Variante eindeutig bevorzugt. Diese Vorgehensweise ermöglicht es, die geotechnischen Daten direkt in die bereits existierenden BIM-Modelle zu integrieren, was eine konsistente und umfassende Darstellung der Baugrundinformationen gewährleistet.

Da die BIM-Modelle im IFC-Format vorliegen, liegt es nahe, auch die Baugrunddaten in dieses Format umzuwandeln, um eine optimale gemeinsame Darstellung zu gewähren. Bei der Transformation müssen die unterschiedlichen Modellierungsparadigmen der beiden Welten, insbesondere im Hinblick auf die Geometrie, berücksichtigt werden. Anders als in GIS-Systemen, bei dem Oberflächengeometrien für die Darstellung der Zylinder verwendet werden, müssen hier Volumenkörper aus den rohen Punkt-Geometriedaten erstellt werden. Zudem ist die Transformation von globalen Koordinaten in die lokalen Koordinaten der BIM-Modelle zu berücksichtigen (Blankenbach und Becker 2022). Ein weiterer wichtiger Aspekt ist die Zuordnung der semantischen Daten des Baugrunds zu den entsprechenden Klassen im IFC-Datenmodell. Hierfür ist die Erstellung eines sogenannten „Mapping Katalogs“ erforderlich, der allen Objekten des Zielmodells, einschließlich ihrer Attribute, die entsprechenden Datensätze des Ausgangsmodells zuordnet. Bevor dieser Katalog erstellt werden kann, muss jedoch entschieden werden, ob die offizielle Version 4.3 des IFC-Modells verwendet werden soll oder ob die wie von (Khan, Kim und Seo 2023) vorgeschlagene Erweiterung um zusätzliche Property Sets implementiert werden soll. Die Unterklassen von *IfcGeotechnicalElement* bilden, wie in Abschnitt 4.1.1 beschrieben, primär die Ergebnisse geotechnischer Laborversuche und weniger die der Felddaten ab. Daher wäre es sinnvoll, das Modell durch selbst definierte Property Sets zu erweitern, um die umfassende Repräsentation der in Kapitel 3 beschriebenen Datengrundlage zu ermöglichen. Allerdings würde die Erstellung eines solchen erweiterten Mapping-Katalogs sowie die konkrete Umsetzung der Transformation den Rahmen dieser Bachelorarbeit überschreiten. Daher werden hier weder der Mapping-Katalog noch die detaillierte Umsetzung der Transformation behandelt.

### 4.1.3 Anwendungsfall 3

Für das dritte Anwendungsszenario sollen verschiedene Datenmodelle hinsichtlich ihrer Eignung für den Export von Daten aus der bestehenden *SEP 3*-Datenbank in ein herstellernerutrales Modell samt geeignetem Datenformat diskutiert werden.

Die Filterfunktion zur Auswahl der zu exportierenden Aufschlüsse kann bereits durch eine Abfrage der Datenbank anhand des Bahnkilometers erfolgen, sodass ausschließlich diese Aufschlüsse in das Exportformat überführt werden. In der folgenden Analyse werden die zuvor genannten Datenmodelle im Kontext des Anwendungsszenarios 3 erneut anhand selbst gewählter Kriterien (vergleiche Tabelle 4.3) verglichen. Die Anforderungen hinsichtlich Softwarekompatibilität und Abbildung der Datengrundlage wurden bereits größtenteils im Abschnitt 4.1.1 erörtert. Der Aspekt der Softwarekompatibilität bezog sich dabei vor allem auf *GeoDin* und soll nun in einem breiteren Kontext diskutiert werden. Außerdem werden jetzt vor allem die Anforderungen „Kompatibilität zu *SEP 3*“ sowie die allgemeine Lesbarkeit für Anwender und Anwenderinnen für die einzelnen Modelle verglichen.

Eine naheliegende Option wäre der Export der Daten im Format der bestehenden **SEP 3**-Datenbank. Üblicherweise wird dies mittels einer entsprechend formatierten *Microsoft Access* Datenbank umgesetzt (Schirren 2020). Dies hätte den Vorteil, dass sämtliche vorhandene Daten der Bohrungen erhalten bleiben und, mit Ausnahme der Bahn-spezifischen Zusatzinformationen, exportiert werden können. Zudem gilt *SEP 3* als Industriestandard und wird in Form einer *Microsoft Access* Datenbank von einigen geotechnischen Software-Anwendungen unterstützt (ebd.). Allerdings besteht das Problem, dass Geotechniker ohne kompatible Software aufgrund der spezifischen Kodierung der Daten möglicherweise Schwierigkeiten mit deren Interpretation haben (siehe 2.2.1.1). Zudem ist eine direkte Integration und

Darstellung in GIS nicht ohne Weiteres möglich, da die Geometrien in *SEP 3* nicht explizit als solche gespeichert werden, sondern als rohe Koordinaten vorliegen.

Hier kommen die Stärken GML-basierter Anwendungsschemata zum Tragen, da diese unmittelbar in Geoinformationssysteme eingebunden und visualisiert werden können (Portele 2022). Allerdings muss dabei berücksichtigt werden, dass die Darstellung der Daten in einer für Anwender und Anwenderinnen leicht verständlichen Form erfolgen sollte. Bei **GeoSciML** besteht aufgrund der umfangreichen Verwendung von Codelisten die Gefahr, dass die Daten in ihrer rohen Form schwer lesbar und interpretierbar sind (Boisvert, Raymond und Sen 2017). Darüber hinaus müsste bei einer Transformation aus *SEP 3* ein Mapping der Codelisten vorgenommen werden, was aufgrund der Vielzahl von Abkürzungen mit erheblichem Aufwand verbunden ist.

Dieselbe Problematik tritt auch bei der ADE für **CityGML** auf, wobei die Inhalte der Codelisten hier zumindest in Klartext vorliegen und damit menschenlesbar sind (Tegtmeier u. a. 2014). Darüber hinaus ist die Kompatibilität des CityGML-Formats zu Geotechnik-Software, wie schon in Anwendungsszenario 1 beschrieben, erneut nicht gegeben.

Das **IFC**-Datenmodell gewinnt zunehmend an Bedeutung in der Bauplanung und wird daher auch verstärkt von Softwareanbietern unterstützt (Borrmann u. a. 2021). Für den gegebenen Anwendungsfall spricht jedoch die unzureichende Abbildung der Datengrundlage sowie der hohe Aufwand bei der Transformation aus *SEP 3* gegen den Einsatz von IFC.

**BoreholeML** stellt schließlich ein weiteres potenzielles Exportformat dar. Obwohl es nicht direkt von gängigen Geotechnik-Softwarelösungen unterstützt wird (vergleiche Abschnitt 2.2.2.2), bietet es den Vorteil, dass die Daten aufgrund der GML-Kodierung direkt in Geoinformationssystemen visualisiert werden können (Portele 2022). Hierbei spricht auch die teilweise menschenlesbare Formatierung der Bohrdaten für den Einsatz dieses Datenmodells. Der entscheidende Vorteil von *BoreholeML* liegt jedoch in der Kompatibilität zu *SEP 3*, die vor allem durch eine bereits vorhandene Mapping-Codeliste erzielt wird (LBEG o. D.). Dies reduziert den Aufwand für die Transformation erheblich.

Insgesamt ist es aufgrund des Mangels an herstellerneutralen Datenformaten zum Austausch von Bohrungsinformationen in Deutschland äußerst herausfordernd, ein Datenmodell inklusive Datenformat zu finden, das sämtliche Anforderungen erfüllt. Die Abwägung der Vor- und Nachteile der einzelnen Modelle und Formate zeigt, dass *SEP 3* und *BoreholeML* trotz ihrer jeweiligen Einschränkungen als geeignete Formate für den Datenaustausch von Aufschlussdaten mit Externen dienen können (siehe grüne Spal-

	<b>SEP 3</b>	<b>GeoSciML</b>	<b>CityGML (+ADE)</b>	<b>IFC 4.3</b>	<b>BoreholeML</b>
<b>Software-Kompatibilität</b>	+	-	-	+	-
<b>Abb. der Datengrundlage</b>	+	-	+	--	+
<b>Kompatibilität zu SEP 3</b>	++	-	--	--	+
<b>Lesbarkeit</b>	--	-	+	+	+
<b>Datenformat</b>	MS Acc. DB	GML	GML	STEP	GML

**Legende:**

- ++ : Erfüllt die Anforderungen vollständig
- + = Erfüllt die Anforderungen größtenteils
- = Erfüllt die Anforderungen teilweise
- = Erfüllt die Anforderungen nicht

**Tabelle 4.3** Vergleich der Datenmodelle für Anwendungsfall 3

ten in Tabelle 4.3). Ausschlaggebend für diese Entscheidung war vor allem der geringe Aufwand bei der Umsetzung des Exports sowie die Kompatibilität mit Geotechnik-Software und GIS-Systemen. Der Export einer *Microsoft Access* Datenbank im *SEP 3*-Format kann problemlos über *GeoDin* erfolgen, während der Export einer GML-Datei nach dem Anwendungsschema *BoreholeML* eine vorhergehende Transformation der Daten erfordert. Diese wird im folgenden Abschnitt beschrieben.

## 4.2 Transformation von SEP 3 zu BoreholeML

### 4.2.1 Mapping der Bohrinformationen

Die Transformation von *SEP 3* zu *BoreholeML*-Daten erfordert zunächst eine Analyse der Unterschiede zwischen den beiden Datenmodellen, da sie auf unterschiedlichen Konzepten basieren: *SEP 3* ist relational, während *BoreholeML* objektorientiert ist (LBEG 2022; BoreholeML o. D.). Dies ist dahingehend zu verstehen, dass die Klassen und Attribute von *BoreholeML* den Relationen und Attributen im *SEP 3*-Modell zugeordnet werden müssen. Zudem sind die Beziehungen der Klassen untereinander bei einem objektorientierten Modell nicht wie bei relationalen Modellen durch Fremdschlüssel modelliert, sondern durch Vererbungen und Assoziationen (Brinkhoff und Kresse 2022). Ein Mapping-Prozess ordnet die Elemente der beiden Datenmodelle einander zu. In diesem Zusammenhang wurde im Rahmen dieser Arbeit eine Mapping-Tabelle erstellt, die den in *SEP 3* vorhandenen Relationen und Attributen die entsprechenden Klassen und Attribute von *BoreholeML* zuweist. Diese Tabelle ist in ihrer vollständigen Form im Anhang dieser Arbeit zu finden; ein Auszug davon wird in Abbildung 4.1 dargestellt. Auf der linken Seite befinden sich alle Relationen und Attribute des *SEP 3*-Datenmodells. Diese werden ihren Pendanten im *BoreholeML*-Datenmodell auf der rechten Seite zugeordnet. Alle Attribute, deren Wertebereich durch eine Codeliste definiert ist, sind hierbei blau markiert. Außerdem bildet die Tabelle für *SEP 3* die Relationen und bei *BoreholeML* die referenzierten Objekte farblich hinterlegt ab, um den Unterschied zu Attributen mit Domänen zu symbolisieren. Die folgende Analyse basiert ausschließlich

<b>SEP 3 Relation(en)</b>	<b>BoreholeML Klasse(n)</b>	<b>Datenübertragung</b>
S3STAMM	Borehole BoreholeProvider Origin BoreholeSegment	<i>SEP 3</i> ist ausführlicher <i>BoreholeML</i> ist ausführlicher Alle Daten übertragbar In <i>SEP 3</i> nicht vorhanden
S3SCHTIT	IntervalSeries	Alle Daten übertragbar
S3SCHDAT	Layer Interval Component Lithology Stratigraphy	Alle Daten übertragbar Alle Daten übertragbar Alle Daten übertragbar <i>BoreholeML</i> ist ausführlicher <i>BoreholeML</i> ist ausführlicher
S3GRUNDW	Groundwater GroundwaterObservation	<i>SEP 3</i> ist ausführlicher In <i>SEP 3</i> nicht explizit modelliert
S3SNDREG S3SND DAT	Investigation	<i>SEP 3</i> ist ausführlicher <i>SEP 3</i> ist ausführlicher
S3PROBEN	Sampling	<i>SEP 3</i> ist ausführlicher

**Tabelle 4.4** Mapping der *SEP 3*-Relationen auf *BoreholeML*-Klassen

Farblegende					
	Ist Relation		abstrakt	Ist Klasse	
	hat Codeliste			hat Codeliste	
SEP3			BoreholeML		
Relation	Attribut	Datentyp	Klasse	Attribut (inkl. Datentyp)	Beschreibung
S3STAMM	*	Relation	Borehole	<b>Klasse</b>	Bohrung
S3STAMM	LOCID	numeric	Borehole	<b>id</b> Type: CharacterString Multiplicity:	ID
S3STAMM	LONGNAME	varchar	Borehole	<b>fullName</b> Type: LocalisedCharacterString Multiplicity:	Langname
S3STAMM	SHORTNAME	varchar	Borehole	<b>shortName</b> Type: LocalisedCharacterString Multiplicity: [0..1]	Kurzname
S3STAMM	XCOORD	numeric	Borehole	<b>location</b> Type: GM_Point Multiplicity:	Bohrlokation
S3STAMM	YCOORD	numeric	Borehole	<b>location</b> Type: GM_Point Multiplicity:	Bohrlochverlauf
S3STAMM	ZCOORDB	numeric	Borehole	<b>location</b> Type: GM_Point Multiplicity:	Bohrlokation
S3SCHTIT	*	Relation	Borehole	<b>intervalSeries</b> «voidable» Type: IntervalSeries Multiplicity: [0..*]	Schichtenverzeichnis
S3SCHTIT	SVART	varchar	IntervalSeries	<b>description</b> Type: LocalisedCharacterString Multiplicity:	Beschreibung
S3SCHTIT	INTV	numeric	IntervalSeries	<b>version</b> «voidable» Type: CharacterString Multiplicity: [0..1]	Version
S3SCHDAT	BESCHBG	varchar	Layer	<b>consistency</b> «voidable» Type: ConsistencyList Multiplicity: [0..1]	Konsistenz

Abbildung 4.1 Auszug aus Mapping Tabelle von SEP 3 zu BoreholeML

auf den Dokumentationen der beiden Datenmodelle, da eine derartige Transformation zwar im Rahmen der Bohrpunktkarte Deutschland umgesetzt wird, jedoch nicht öffentlich dokumentiert ist (LBEG 2022; BoreholeML o. D.). Die Website des LBEG informiert allerdings darüber, dass *BoreholeML* von *SEP 3* unterstützt wird und eine Mapping-Codeliste zur Verfügung steht, um die in *SEP 3* verwendeten Codierungen in die *BoreholeML*-Codes zu übersetzen (LBEG o. D.).

Die Zuordnung beginnt auf der Ebene der Relationen und Klassen, wobei nur die relevanten *SEP 3*-Relationen den entsprechenden *BoreholeML*-Klassen zugeordnet werden (siehe Tabelle 4.4).

### 4.2.1.1 Mapping der Stammdaten

Die Stammdaten in der *SEP 3*-Relation *S3STAMM* können weitgehend den Klassen in den *BoreholeML*-Packages *Borehole* und *Identification* zugeordnet werden, da sie ähnliche Informationen über die Bohrungen enthalten. Dabei ist *S3STAMM* detaillierter als die *BoreholeML*-Klassen, weshalb einige Informationen nur teilweise oder gar nicht in *BoreholeML* übertragen werden können. Die Attribute der Metadaten-Klassen *BoreholeProvider* und *Origin* können größtenteils durch *SEP 3*-Attribute abgedeckt werden, wobei die Daten über den Eigentümer der Bohrung in *BoreholeProvider* etwas ausführlicher dargestellt sind und somit nicht vollständig befüllt werden können. In *SEP 3* gar nicht vorhanden sind Informationen über den exakten geometrischen Verlauf der Bohrung. Somit können entsprechende Informationen in der Klasse *BoreholeSegment* nicht abgebildet werden.

### 4.2.1.2 Mapping der Schichtdaten

Die Übertragung der Schichtdaten von *SEP 3* zu *BoreholeML* erfordert eine differenzierte Zuordnung der Daten, da beide Modelle unterschiedliche Strukturen zur Beschreibung der Schichtparameter verwenden. In beiden Modellen existiert zunächst eine Entität, die mehrere Versionen des Schichtenverzeichnisses repräsentiert (in *SEP 3* die Relation *S3SCHTIT*, in *BoreholeML* die Klasse *IntervalSeries*). Ein Datensatz in *SEP 3* entspricht dabei genau einem Objekt in *BoreholeML*.

Der Hauptunterschied liegt in der detaillierten Abbildung der Schichtdaten. Während *SEP 3* lediglich die Relation *S3SCHDAT* verwendet, verteilt *BoreholeML* diese Informationen auf fünf verschiedene Klassen. Die geotechnischen Parameter, die jede Schicht beschreiben, sind in *BoreholeML* Teil der abstrakten Klasse *Layer*. Diese entsprechen den Attributen eines Datensatzes der Relation *S3SCHDAT*, wobei die Codes entsprechend der bereitgestellten Mapping-Codeliste übersetzt werden müssen. Einige Informationen sind jedoch in beiden Modellen unterschiedlich auf Attribute aufgeteilt. So entsprechen die Daten des *BoreholeML*-Attributs *compactness* den Werten der *SEP 3*-Attribute *BESCHBV*, *ZUSATZ* und *GENESE*.

Des Weiteren entspricht die Klasse *Lithology* dem Attribut *PETRO* in *S3SCHDAT*. Dabei wurde im Abschnitt bereits 4.1.1 kritisiert, dass dieses Attribut keine atomare Domäne besitzt und somit mehrere Eigenschaften und Bestandteile der Schicht in einem codierten Textwert enthält. *BoreholeML* löst dieses Problem, indem es die Einzelinformationen zu Bodenart und Massenanteil in separaten Objekten der Klasse *Lithology* speichert. Zusätzlich werden sowohl die Ursprungsform aus der Quelldatenbank (als *rockCode*) als auch die in Klartext übersetzte Version (als *rockNameText*) in Attributen der Klasse *Layer* abgelegt. Ein Beispiel für das Mapping der Bodenanteile im *SEP 3*-Attribut *PETRO* ist in Abbil-

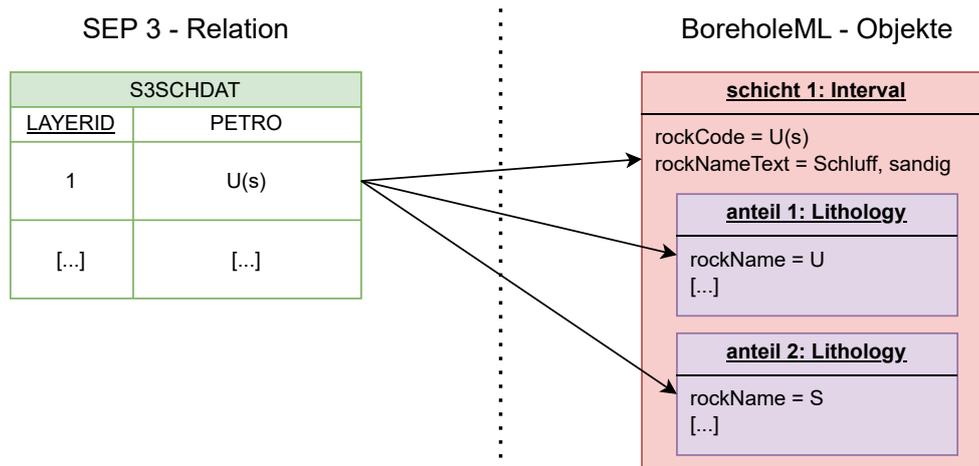


Abbildung 4.2 Mapping-Beispiel für das SEP 3-Attribut *PETRO*

dung 4.2 dargestellt. Der Attributwert („U(s)“) wird einerseits direkt in der Ursprungscodierung sowie als Klartext („Schluff, sandig“) auf das Objekt der Klasse *Interval* übertragen. Andererseits werden die einzelnen Bestandteile („U“ und „S“) zusätzlich in jeweils einem Objekt des Typs *Lithology* gespeichert. Auch die Stratigraphie wird in *BoreholeML* differenzierter dargestellt, wobei zwei Attribute zur Beschreibung verwendet werden, während *SEP 3* hierfür nur ein Attribut vorsieht.

Darüber hinaus unterscheidet sich auch die Abbildung von Unterschichten einer Hauptschicht in beiden Datenmodellen. In *SEP 3* werden diese als eigenständige Datensätze in der Relation *S3SCHDAT* gespeichert, wobei sie durch denselben *LAYERID*-Wert, aber unterschiedliche *RECID*-Werte gekennzeichnet werden. Im Gegensatz dazu löst *BoreholeML* dies durch die Klassen *Interval* und *Component*, wobei mehrere Unterschichten (*Component*) einer Hauptschicht (*Interval*) zugeordnet werden können. Insgesamt können durch Übersetzung und Aufteilung der Attributwerte in *SEP 3* alle Schichtinformationen nach *BoreholeML* transformiert werden. Die Unterschiede in der Struktur und Detaillierung der Schichtdaten müssen jedoch bei der Umsetzung der Transformation genau beachtet werden, um einen Informationsverlust zu vermeiden.

#### 4.2.1.3 Mapping der Grundwasser-, Sondierungs- und Probendaten

In Bezug auf die Transformation der Grundwasser-, Sondierungs- und Probendaten zeigt sich ebenfalls ein struktureller Unterschied zwischen den beiden Datenmodellen. Im Datenmodell *SEP 3* können für eine Bohrung mehrere angetroffene Grundwasserschichten beschrieben werden. Hierbei kann neben dem erstmalig angetroffenen Grundwasserstand zusätzlich noch der ausgepegelte Ruhewasserstand in der Relation *S3GRUNDW* eingetragen werden. Letzterer wird, je nach dem, ob der Grundwasserpegel ab dem Zeitpunkt der erstmaligen Messung angestiegen oder gesunken ist, in den entsprechenden Feldern eingetragen. Zusätzlich kann zum Beispiel noch das Datum der Messung sowie Angaben zu durchgeführten Pumpversuchen hinzugefügt werden. In *BoreholeML* kann einer Bohrung genau ein Objekt der Klasse *Groundwater* zur Repräsentation der Grundwassermessung während der Bohrung zugeordnet werden. Dies entspricht genau einem Datensatz der Relation *S3GRUNDW* in *SEP 3*. Dadurch wird das Mapping erschwert, da nur eine angetroffene Grundwasserschicht auf *BoreholeML* abgebildet werden kann. Die Objekte der Klasse *GroundwaterObservation* repräsentieren hingegen

alle Grundwassermessungen, die nach Abschluss der Bohrarbeiten im Bohrloch dokumentiert wurden. Diese Information könnte in *SEP 3* durch weitere Datensätze in *S3GRUNDW* abgebildet werden, deren Messdatum nach dem Ende der Bohrarbeiten liegt. Explizit im Datenmodell modelliert ist dieser Unterschied jedoch nicht. Zusammenfassend kann festgehalten werden, dass ein Mapping der Grundwasserstände nur dann sicher umgesetzt werden kann, wenn während der Bohrung nur eine Grundwasserschicht eingemessen wurde, was in der Datengrundlage nicht immer der Fall ist. Allgemein sollten also nur Grundwasserstände *einer* Grundwasserschicht übertragen werden.

Für die letzten drei zu übertragenden *SEP 3*-Relationen (*S3SNDREG* für Sondierungsdaten und *S3-PROBEN* für Laborergebnisse) gilt, dass in *SEP 3* die eigentlichen Daten der Messungen gespeichert werden, während *BoreholeML* nur allgemein die Art der Messung sowie deren Lage unter Geländeoberkante beschreibt. Die entsprechende *BoreholeML*-Attribute können nach Übersetzung der Codes vollständig aus den *SEP 3*-Datenfeldern übernommen werden. Der Mehrwert dieser Übertragung ist durch die fehlenden Messwerte im Kontext des dritten Anwendungsszenarios jedoch nur bedingt gegeben.

#### 4.2.2 Bewertung der Transformation

Zur quantitativen Analyse des Mappings wurde eine Mapping Matrix aller Attribute, Relationen und Objekte in *SEP 3* und *BoreholeML* erstellt, die die Abbildung der Informationen aufgrund einiger Vereinfachungen jedoch nur grob wiedergibt (siehe Tabelle 4.5). Demnach konnten von den insgesamt 243 *SEP 3*-Attributen der betrachteten Relationen insgesamt nur 87, also circa 35 Prozent übertragen werden. Dies ist unter anderem jedoch damit zu begründen, dass *SEP 3* für bestimmte Informationen (zum Beispiel Koordinaten) mehrere Attribute vorsieht. So gibt es in den Stammdaten beispielsweise mehrere Attribute für die Koordinaten unterschiedlicher Koordinatensysteme. Außerdem sind bei der Berechnung die Primär- und Fremdschlüssel der betrachteten *SEP 3*-Relationen miteingerechnet, die in *BoreholeML* durch die Abbildung auf eine hierarchisch strukturierte GML-Datei nicht benötigt werden. Somit kann dieser große Unterschied der Attributzahl vor allem auch durch eine effizientere Abbildung der Attribute erklärt werden. Trotzdem geht aus der eben angestellten Betrachtung der einzelnen Relationen und Klassen insgesamt hervor, dass *SEP 3* eine weitaus umfangreichere Abbildung der Bohrdaten ermöglicht als *BohreholeML*. Obwohl ein Großteil der Stammdaten und Schichtinformationen ohne wesentliche Informationsverluste übertragen werden können, führt die weniger umfassende Struktur von *BohreholeML* dazu, dass bestimmte wichtige Informationen, insbesondere Sondierungs- und Laborergebnisse, verloren gehen.

Die flexible Struktur von *BohreholeML* könnte allerdings durch eine Erweiterung des Datenmodells angepasst werden, um die fehlenden Informationen zu integrieren und die Übertragung von Grundwasserständen besser abzubilden.

	In <i>BoreholeML</i>	Nicht in <i>BoreholeML</i>	Gesamt
In <i>SEP 3</i>	87	156	243
Nicht in <i>SEP 3</i>	28	0	28
Gesamt	115	156	271

Tabelle 4.5 Mapping Matrix für *SEP 3* und *BoreholeML*

# 5 Prototypische Umsetzung von zwei Anwendungsszenarien in einem DB-Projekt

In diesem Abschnitt wird die konkrete Umsetzung der Repräsentation von Baugrundaufschlüssen in 3D-Geoinformationssystemen anhand der Daten eines Infrastrukturprojekts der DB InfraGO AG (vereinfacht DB genannt) beschrieben.

Kapitel 4 hat ergeben, dass für die Repräsentation in einer Datenbank das Datenmodell *SEP 3*, für die gemeinsame Darstellung mit BIM-Daten das Datenmodell IFC und für den Austausch der Daten das Datenmodell *BoreholeML* verwendet werden soll. Ebenfalls wurden in diesem Kapitel einige Schwierigkeiten in der Umsetzung des Anwendungsszenarios 2 erwähnt, weshalb dieses Szenario nun nicht weiter betrachtet wird.

## 5.1 Verwendete Software und Datengrundlage

### 5.1.1 Verwendete Software

#### 5.1.1.1 pgAdmin in PostgreSQL mit PostGIS-Erweiterung

*PostgreSQL* ist ein objekrelationales Open Source Datenbank-System, das seine Ursprünge in einem Projekt der University of California at Berkeley aus dem Jahr 1986 hat (*What is PostgreSQL* o. D.). In dieser Arbeit wird es in Kombination mit der *PostGIS*-Erweiterung verwendet, die der Speicherung von GIS-Objekten dient und umfangreiche Methoden zur Abfrage und Manipulation räumlicher Daten enthält (*PostGIS 3.4.3 Manual* o. D.). Zusätzlich wird die prozedurale Sprache *Procedural Language for PostgreSQL (PL/pgSQL)* zur Erstellung von Funktionen verwendet, welche alle Abfragen eines Vorhabens in der Datenbank bündeln und wiederverwendbar machen (The PostgreSQL Global Development Group o. D.). Als Management Tool der Datenbank wird zudem die Open Source Software *pgAdmin 4* verwendet.

#### 5.1.1.2 GIS-Software

Als Geoinformationssystem kommt die Open Source Software *QGIS* (Version 3.34.2) zum Einsatz. Sie dient hier zur Darstellung der in *PostgreSQL* erzeugten Geometrie-Objekte (*QGIS Documentation* o. D.). Um diese dreidimensional zu visualisieren, wird zudem das Plugin *Qgis2threejs* verwendet, das eine Anzeige eines Kartenausschnitts in einem zusätzlichen Fenster ermöglicht (Minuora o. D.).

### 5.1.1.3 Python-Programmierung in Pycharm

Für Aufgaben, die in PL/pgSQL aufgrund ihrer Komplexität nur schwer gelöst werden können, werden Python-Skripte (Version 3.11) in der *Pycharm*-Entwicklungsumgebung (Version 2023.3.2) erstellt. Der Zugriff auf die lokale *PostgreSQL*-Datenbank erfolgt mittels der Bibliothek *SQLAlchemy*.

### 5.1.1.4 Hyper Text Markup Language (HTML)

Um dynamische Diagramme zu erstellen wird im Rahmen dieser Arbeit das Datenformat HTML sowie die Skriptsprache *JavaScript* inklusive der Bibliothek *Plotly* verwendet (Berners-Lee und Connolly 1995; *plotly Graphing Libraries* o. D.).

### 5.1.1.5 Feature Manipulation Engine (FME)

Das Extract-Transform-Load-Tool *FME* der Safe Software Inc. wurde verwendet, um die Syntax der erzeugten *BoreholeML* Dateien zu überprüfen. Jedoch steht diese Software projektintern nicht zur Verfügung und kann somit nicht für die Datentransformation herangezogen werden.

## 5.1.2 Hinweise zur Datengrundlage

Die Aufschlussdaten wurden alle nach einem einheitlichen Schema in die *SEP 3*-Datenbank eingegeben, welches die eingegebenen Werte standardisiert und die Umsetzung damit etwas vereinfacht. Zum Beispiel bestehen die Einträge im Attribut *PETRO* nur aus Hauptbestandteilen und Nebenbestandteilen und können nicht, wie im Symbolschlüssel Geologie beschrieben, mehrere Unterebenen besitzen (LBEG o. D.). Außerdem existiert für alle Aufschlüsse nur genau eine Version des Schichtenverzeichnisses, wodurch die Relation *S3SCHTIT* inklusive ihrer Schlüssel nicht weiter betrachtet werden muss. Ebenso enthalten die Daten der Relation *S3SCHDAT* keine Unterschichten, da diese ohnehin nicht in den geotechnischen Berichten beschrieben wurden.

Neben der *SEP 3*-Datenbank wird zudem eine Liste aller Aufschlüsse mit spezifischen Daten der DB (Bahnkilometer, Position bezüglich des Gleises) verwendet, die als „Rohdatenliste“ bezeichnet wird.

## 5.2 Anwendungsszenario 1: Abfragen von Aufschlussdaten mithilfe einer interaktiven dreidimensionalen Karte

### 5.2.1 Konzept für Anwendungsszenario 1

Im ersten Anwendungsszenario werden die *SEP 3*-Baugrunddaten aus der beschriebenen *GeoDin*-Datenbank in eine *PostgreSQL*-Datenbank überführt und anschließend in einem Geoinformationssystem visualisiert. Der gesamte Workflow von der Eingabe der Daten aus den geotechnischen Berichten in *GeoDin* bis zur Visualisierung ist in Abbildung 5.1 dargestellt und wird nun weiter erläutert.

## 5.2 Anwendungsszenario 1: Abfragen von Aufschlussdaten mithilfe einer interaktiven dreidimensionalen Karte

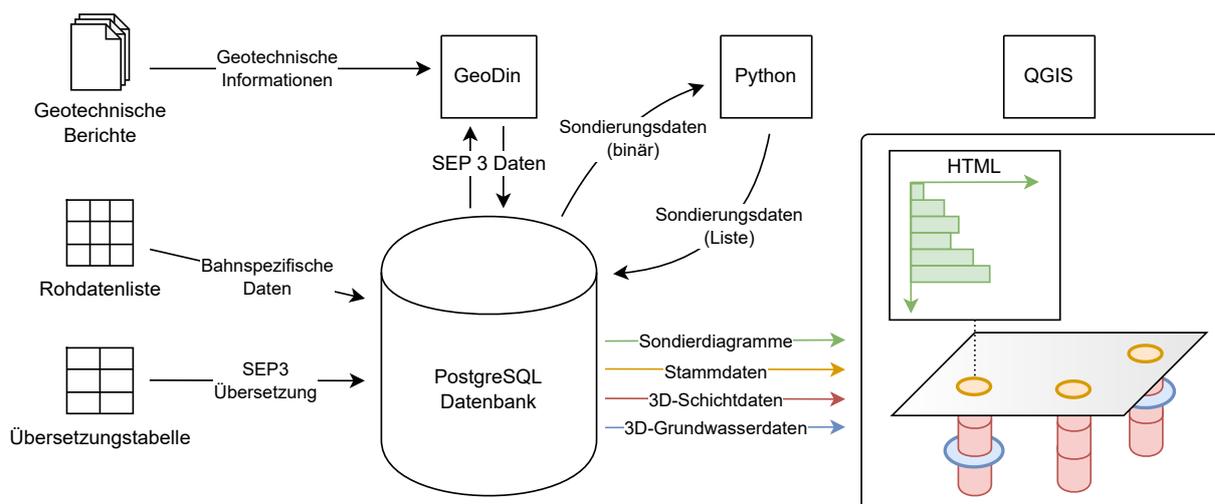


Abbildung 5.1 Workflow für Anwendungsszenario 1

### 5.2.1.1 Konzept zur Speicherung und Verarbeitung der Daten

Zunächst muss ein Datenbanksystem gewählt werden, das sowohl die *SEP 3*-Datenbank aus *GeoDin* als auch weitere eventuell benötigte Daten speichern und gleichzeitig eine direkte Anbindung an ein Geoinformationssystem ermöglichen kann. Hierfür wird das Datenbanksystem *PostgreSQL* inklusive der Erweiterung *PostGIS* zur Abfrage und Manipulation räumlicher Daten verwendet, da dieses die eben genannten Anforderungen erfüllt und zudem kostenfrei verfügbar ist (Fugro Consult GmbH o. D.; *What is PostgreSQL* o. D.; *PostGIS 3.4.3 Manual* o. D.). In dieser Datenbank sollen zunächst vor allem die Daten der *SEP 3*-Datenbank aus *GeoDin* direkt gespeichert werden, wodurch diese bei erneuter Eingabe von Daten aus geotechnischen Berichten stets automatisch aktualisiert wird. Außerdem sollen einerseits weitere Bahn-spezifische Stammdaten, die nicht auf *SEP 3* abgebildet werden können integriert werden (siehe Abbildung 5.1 links Mitte). Andererseits muss auch die in Kapitel 4 erwähnte Mapping-Codeliste (nachfolgend „Übersetzungstabelle“ genannt) zur Übersetzung der in *SEP 3* verwendeten Codes integriert werden (siehe Abbildung 5.1 links unten).

### 5.2.1.2 Konzept zur Visualisierung

Die Visualisierung soll alle in der Anforderungsanalyse erwähnten Daten in einer interaktiven Karte umfassen. Hierbei steht an erster Stelle die Darstellung der Stammdaten, welche die Position und Art der Aufschlüsse aus der *SEP 3*-Datenbank sowie Bahn-spezifische Daten wie den Bahnkilometer oder die Position bezüglich des Gleises beinhalten. Die Stammdaten sollen als Punkte in einer zweidimensionalen Karte angezeigt und die eben genannten Informationen durch einen Klick abgefragt und ausgegeben werden (siehe orange Punkte in Abbildung 5.1). Dadurch wird eine schnelle Übersicht über die geographische Verteilung der Bohrungen entlang der Bahnstrecke visualisiert.

Darüber hinaus werden die Schichtdaten jeder Bohrung, inklusive aller in der Bodensprache ermittelten Parameter, in einer dreidimensionalen Karte repräsentiert. Jede Schicht wird dabei als Zylinder dargestellt, der nach den Vorgaben der DIN 4023 farblich codiert ist, um eine intuitive Erfassung der Schichtzusammensetzung zu ermöglichen (siehe rote Zylinder in Abbildung 5.1) (DIN4023 2023).

Die Grundwasserdaten werden in Form von horizontalen Kreisen dargestellt, die den Grundwasser-

stand unter der Geländeoberkante in 3D markieren (siehe blaue Kreise in Abbildung 5.1).

Zuletzt sollen auch indirekte Aufschlüsse in Form von Rammsondierungen in die Karte integriert werden. Die Visualisierung ist in diesem Fall jedoch nicht dreidimensional, was sowohl auf geotechnische als auch auf visualisierungstechnische Überlegungen zurückzuführen ist. In der Geotechnik werden Rammsondierungen üblicherweise in Form von Schlagzahldiagrammen dargestellt, bei denen die Anzahl der Schläge über die Tiefe aufgetragen wird (Kuntsche und Richter 2021). Eine dreidimensionale Darstellung wäre somit ungewöhnlich und in der Praxis nicht unbedingt hilfreich. Technisch könnte die Visualisierung in Form von nach Schlagzahl farbcodierten Zylindern für jeden Tiefenabschnitt umgesetzt werden. Jedoch würden dadurch pro Aufschluss eine oft dreistellige Anzahl von Objekten inklusive Geometrie erzeugt, was die Performance der Visualisierung deutlich einschränken könnte. Dieses Risiko steht in keinem Verhältnis zu dem theoretischen Nutzen, weshalb eine alternative Darstellung in Form eines durch HTML generierten Schlagzahldiagramms angestrebt wird (siehe grünes Diagramm in Abbildung 5.1). Hierfür muss auf die entsprechenden Rohdaten zugegriffen werden. *GeoDin* speichert diese jedoch nicht, wie im *SEP 3*-Datenmodell vorgesehen, in Form von Datensätzen in der Relation *S3SNDDAT*, sondern in einer binärkodierten Datei im Attribut *SNDVALUE* der Relation *S3SNDREG* (LBEG 2022). Dementsprechend ist zunächst eine Dekodierung notwendig. Dies ist mit *PostgreSQL* nicht direkt möglich und muss somit mittels eines Python-Skripts erfolgen (siehe Abbildung 5.1 oben). Zur Umsetzung der Visualisierung bietet sich die Anwendung eines Geoinformationssystems (GIS) an, da dieses zur Darstellung der Daten in einer interaktiven Karte genutzt werden kann (Bartelme 1989). Konkret kommt die Open Source Software *QGIS* zum Einsatz, da diese 2D- und 3D-Geometrien aus *PostgreSQL*-Datenbanken direkt importieren kann. Außerdem ist die Einbettung von HTML-Diagrammen in sogenannten „HTML Map Tips“ möglich, wodurch auch die Darstellung der Schlagzahldiagramme ermöglicht wird (*QGIS Documentation* o. D.).

### 5.2.2 Import der Daten in eine PostgreSQL-Datenbank mit PostGIS-Erweiterung

#### 5.2.2.1 Überführen der SEP 3-Tabellen

Zunächst müssen die *SEP 3*-Baugrunddaten, die mit *GeoDin* in eine lokale *Microsoft Access*-Datenbank eingegeben wurden, in eine *PostgreSQL*-Datenbank überführt werden. Hierfür wird die Existenz eines *PostgreSQL*-Servers inklusive einer leeren Datenbank vorausgesetzt. Sobald die Verbindung von *GeoDin* zu dieser Datenbank hergestellt wurde, kann das *GeoDin*-Projekt aus der *Microsoft Access*-Datenbank kopiert und in die *PostgreSQL*-Datenbank eingefügt werden (Fugro Consult GmbH o. D.). So bleibt die vollständige Funktionalität der Datenbank in *GeoDin* erhalten (siehe Abbildung 5.2) und ein externer Zugriff auf die Daten wird ermöglicht. Sobald die Datenbank überführt wurde, sollen alle weiteren Aufschlüsse nur noch in die *PostgreSQL*-Datenbank eingetragen werden, um eine stetige Aktualisierung der Daten dieser Datenbank zu gewährleisten. Die lokale *Microsoft Access*-Datenbank wird in diesem Zuge nun nicht mehr verwendet. Anschließend wird eine zusätzliche Verbindung zur *PostgreSQL*-Datenbank im Datenbank-Management-Tool *pgAdmin 4* eingerichtet, um Abfragen in der Abfragesprache SQL auf die Datenbank ausführen zu können. Die erste Abfrage dient der Aktivierung der *PostGIS*-Erweiterung, die die Abfrage und Manipulation räumlicher Daten in der Datenbank gewährleistet (siehe Listing 5.1).

## 5.2 Anwendungsszenario 1: Abfragen von Aufschlussdaten mithilfe einer interaktiven dreidimensionalen Karte

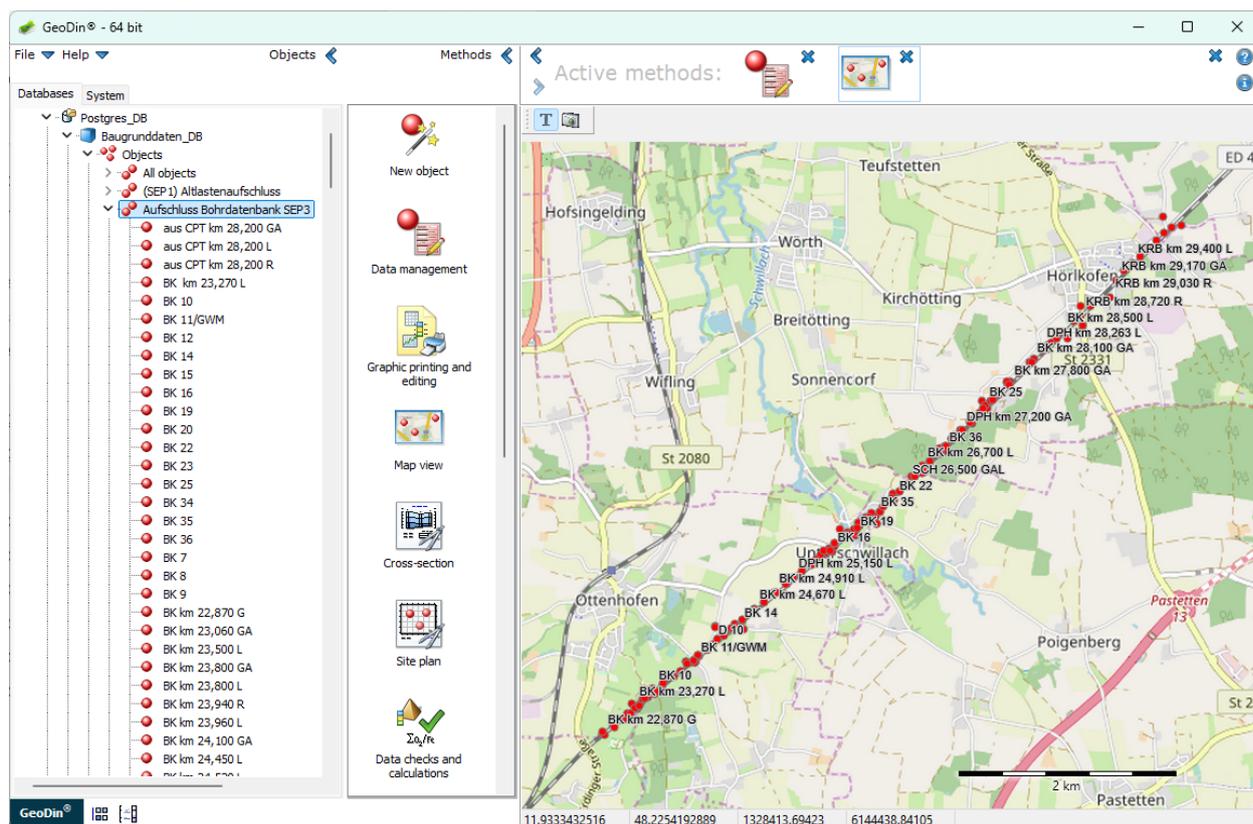


Abbildung 5.2 SEP 3-Datenbank in GeoDin - Kartenansicht

```
1 CREATE EXTENSION IF NOT EXISTS postgis;
```

Listing 5.1 Aktivierung der PostGIS Erweiterung

### 5.2.2.2 Import zusätzlicher Tabellen in die Datenbank

Um die Abfrage Bahn-spezifischer Daten zu gewähren, muss die im Konzept erwähnte Rohdatenliste in die Datenbank importiert werden. Hierfür muss eine entsprechende Tabelle im Comma-separated values (CSV)-Format vorliegen, die der in Listing 5.2 definiertem Form entspricht. Für die spätere Übertragung der Daten wird eine entsprechende Relation in der Datenbank erstellt (siehe Listing 5.2). Hierbei ist wichtig, dass die Spalte *Bezeichnung* in der Rohdatenliste eindeutig ist und dass sie in der Erstellung als Primärschlüssel gekennzeichnet wird. Außerdem müssen die Datentypen der Spalten in der Tabelle mit denen in der *PostgreSQL*-Relation übereinstimmen.

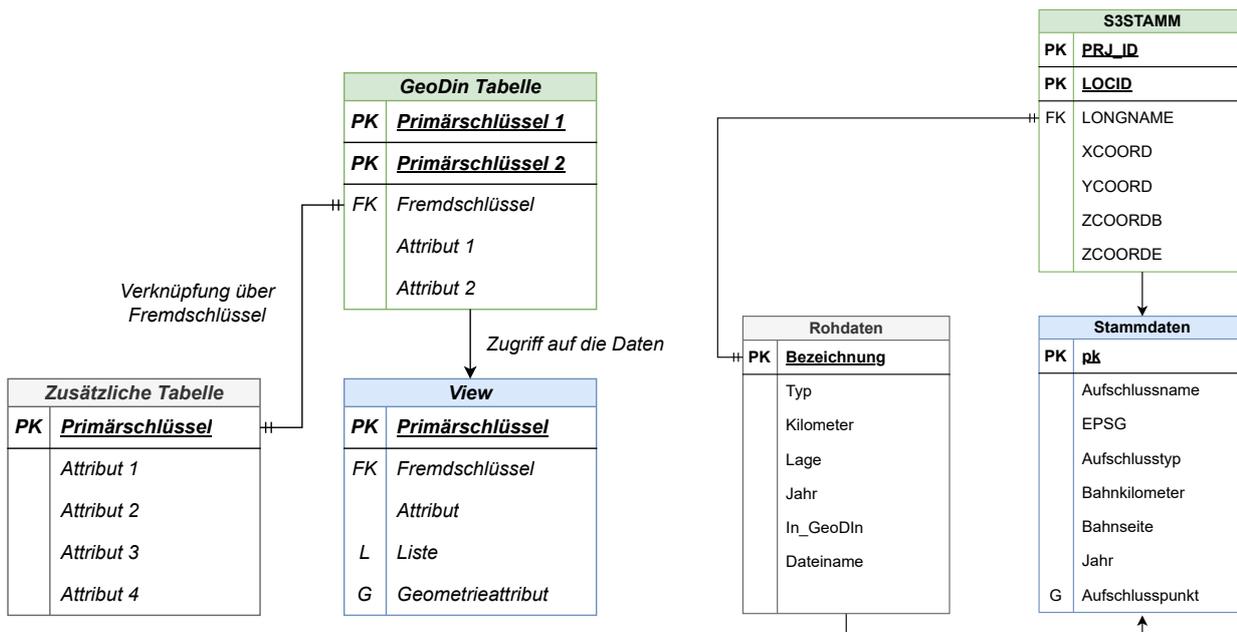
```
1 CREATE TABLE rohdatenliste
2 (
3 "km" DOUBLE PRECISION ,
4 "Bezeichnung" VARCHAR primary key ,
5 "Lage" VARCHAR (10) ,
6 "Typ" VARCHAR (10) ,
7 "Jahr" INTEGER ,
8 "In_GeoDin" BOOLEAN ,
9 "Dateiname" VARCHAR ,
10 );
```

Listing 5.2 Erstellen der Relation *rohdatenliste*

Zuletzt muss zur Übersetzung der *SEP 3*-Codierungen die entsprechende Mapping-Codeliste von der Website des LBEG heruntergeladen und ebenfalls eingepflegt werden (LBEG o. D.). Die Tabelle wurde mit *QGIS* in eine SQL-Abfrage umgewandelt und anschließend in die Datenbank importiert.

### 5.2.3 Definition der Abfragen

Im Folgenden werden auf Grundlage der in Kapitel 3 ermittelten Anforderungen und der vorhandenen Datengrundlage eine Reihe von Abfragen entworfen, die die geforderten Daten inklusive ihrer Geometrie abbilden. Die Abfragen werden zunächst anhand der Informationen, die sie abbilden, gruppiert und durch Entity-Relationship-Diagramme in der Krähenußdarstellung visualisiert. In den erstellten Diagrammen wird im Rahmen dieser Arbeit zwischen den *SEP 3*-Relationen aus *GeoDin* (grün), zusätzlichen Tabellen (grau) und Sichten (blau) unterschieden (vergleiche Abbildung 5.3). Letztere dienen der Abfrage und Manipulation bestimmter Daten aus den abgefragten Tabellen. Diese können zwar keine Primärschlüssel besitzen, jedoch wird im Folgenden für jede Sicht ein eindeutiges Attribut (*pk*) definiert, um den Zugriff von anderen Sichten zu vereinfachen. Bei einem direkten Zugriff auf die *SEP 3*-Daten müssten sonst immer alle Schlüsselattribute verglichen werden, da nur die Kombination dieser Attribute eindeutig ist (LBEG 2022). Zusätzlich werden in der Spalte für Primär- und Fremdschlüssel komplexe Datentypen, wie Geometrien (*G*) und Listen (*L*) mit einem entsprechenden Buchstaben vermerkt, um diese hervorzuheben. Die farbliche Markierung der Relationen aus *GeoDin* soll betonen, dass diese nicht strukturell verändert werden dürfen, um die Kompatibilität mit *GeoDin* weiterhin zu gewährleisten. Anschließend werden die dargestellten Diagramme in SQL-Abfragen übersetzt, um sie in der Datenbank umzusetzen. Im Textteil werden nur relevante Teile des SQL-Codes abgebildet, während sich eine vollständige Sammlung aller Abfragen im Anhang befindet.



### 5.2.3.1 Abfrage der Stammdaten

Als Erstes soll eine Abfrage erstellt werden, die alle in der Relation *S3STAMM* vorhandenen Aufschlüsse inklusive ihrer dreidimensionalen Geometrie als Punkte darstellt. Sie soll neben dem Aufschlussnamen, den Bahnkilometer, die Lage bezüglich des Gleises, sowie das Jahr der Durchführung enthalten. Während die ersten beiden Attribute und die Geometrie aus der *SEP 3*-Datenbank übernommen werden können, müssen die restlichen Bahn-spezifischen Daten aus der zusätzlichen Tabelle (*Rohdaten*) bezogen werden. Als Verknüpfung der beiden Relationen, kann der Primärschlüssel *Bezeichnung* verwendet werden, der dem Attribut *LONGNAME* in *S3STAMM* entspricht (siehe Abbildung 5.4). Das Geometrieattribut muss aus den Koordinaten sowie dem EPSG-Code für das Koordinatensystem erstellt werden (*PostGIS 3.4.3 Manual* o. D.).

Nun soll die zuvor definierte Sicht *Stammdaten*, einschließlich des Geometrieattributs *Aufschlusspunkt*, mittels einer SQL-Abfrage erzeugt werden. Dabei wird die *PostGIS*-Funktion *st\_makepoint* verwendet, um die Geometrie basierend auf den drei Koordinatenattributen *XCOORD*, *YCOORD* und *ZCOORDB* aus *S3STAMM* zu erstellen. Zusätzlich muss mit der Funktion *st\_setsrid* das Koordinatensystem der Punkte definiert werden, um ein vollständiges Geometrieobjekt zu erzeugen. Das Koordinatensystem ist in der verwendeten *SEP 3*-Datenbank in Form eines EPSG-Codes hinterlegt und wird an der entsprechenden Stelle der Funktion abgefragt. Ausgegeben wird die Geometrie hierbei im WKB-Format, das von *QGIS* direkt gelesen werden kann. Neben der Geometrie wird ein eindeutiges Attribut erzeugt, das aus den *SEP 3*-Schlüsseln *PRJ\_ID* und *LOCID* zusammengesetzt ist und somit zur Identifikation des Aufschlusses verwendet werden kann (siehe Zeile 4 in Listing 5.3). Zuletzt wird noch die Beziehung zwischen Rohdatenliste und *S3STAMM* über eine einfache *INNER JOIN*-Operation vollzogen (siehe Zeile 15-16 in Codelistig 5.3).

```

1 CREATE VIEW Stammdaten AS
2 SELECT
3     -- Erstellen eines Text-Ersatz-Schlüssels aus PRJ\_Id und LOCID
4     stm.prj_id || '_' || locid as pk,
5     stm.longname as Aufschlussname,
6     roh."Typ" as Aufschlusstyp,
7     roh."km" as Bahnkilometer,
8     roh."Lage" as Bahnseite,
9     stm.epsg as EPSG,
10    -- Erstellen der Punktgeometrie im angegebenen Koordinatensystem
11    st_setsrid(st_makepoint(stm.xcoord, stm.ycoord, stm.zcoordb),
12              stm.epsg::INTEGER)
13 AS Aufschlusspunkt
14 FROM
15     geodin_loc_s3stamm stm INNER JOIN rohdatenliste roh
16     ON stm.longname = roh."Bezeichnung"

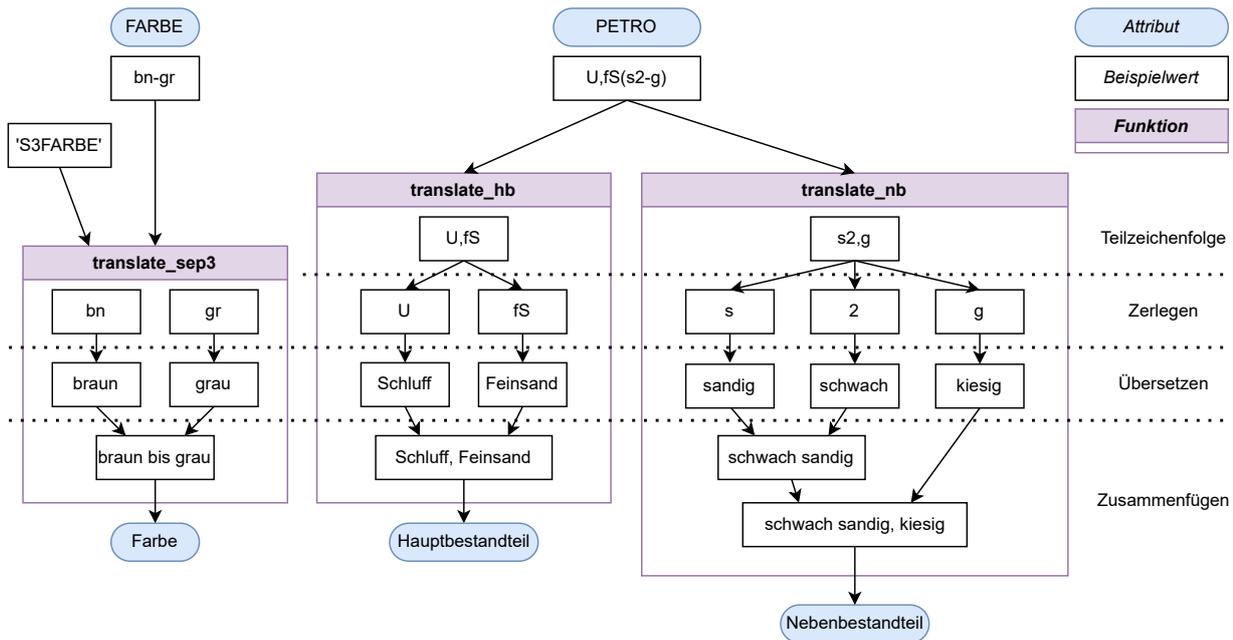
```

**Listing 5.3** Erstellen der Sicht Stammdaten

### 5.2.3.2 Abfrage und Übersetzung der Schichtdaten

Als Nächstes werden die Schichtdaten aus der Tabelle *S3SCHDAT* in der Sicht *Schichten* abgefragt. Um die Informationen in der Abfrage menschenlesbar zur Verfügung zu stellen, müssen die Codes

## 5 Prototypische Umsetzung von zwei Anwendungsszenarien in einem DB-Projekt



**Abbildung 5.5** Workflow der Übersetzungsfunktionen anhand eines Beispiels

der einzelnen Schichtattribute in *SEP 3* übersetzt werden. Hierfür dient die anfangs importierte Übersetzungstabelle des LBEG. Diese enthält alle *SEP 3*-Codelisten und ordnet immer der Kombination aus Name der Codeliste (*sep3-codelist*) und Code (*sep3-code*) einen Klartext-Wert (*sep3-term*) zu. In Abbildung 5.6 ist die Verbindung dieser Tabelle zu den Schichtdaten in *S3SCHDAT* am Beispiel des Attributs *Genese* dargestellt, kann aber auf fast alle anderen Attribute übertragen werden, die nach dem Symbolschlüssel Geologie codiert sind. In SQL wird die Übersetzung der Codes durch die Definition von Funktionen in der Sprache PL/pgSQL umgesetzt, da dieselbe Übersetzung für mehrere Attribute mit unterschiedlichen Codelisten umgesetzt werden muss. Dieser Funktion (*translate\_sep3*) wird zunächst der Code aus der Datenbank sowie der Name der Codeliste übergeben. Bevor der Code übersetzt werden kann, muss der zusammengesetzte Textwert an den Trennzeichen (Bindestrich und Komma) in seine Einzelbestandteile zerlegt werden. Die Einzelbestandteile können nun durch eine Abfrage in der Übersetzungstabelle übersetzt werden. Alle übersetzten Werte werden am Ende wieder zu einem Text zusammengesetzt. Der Workflow der Funktion ist für ein Beispiel des Attributs *Farbe* in Abbildung 5.5 links dargestellt, kann aber äquivalent auf die meisten anderen Attribute übertragen werden.

Ein Spezialfall stellt hierbei das *SEP 3*-Attribut *PETRO* dar. In diesem sind Haupt- und Nebenanteile der Schicht zusammen in einer Codierung nach dem Symbolschlüssel Geologie enthalten (vergleiche Beispiel in Abbildung 5.5 oben) (LBEG 2022). Diese sollen nun zusätzlich zur Übersetzung getrennt ausgegeben werden. Dafür muss die Textkette an der mittleren Klammer aufgeteilt werden. Der erste Teil entspricht den Haupt-, der zweite den Nebenanteilen, wobei beide wieder aufgeteilt und übersetzt werden. Der entsprechende Workflow ist in Abbildung 5.5 rechts dargestellt. Bei den Nebenanteilen müssen bei der Übersetzung zusätzlich die Intensitätszahlen (z.B. „2“ für „schwach“) am Ende eines Codebestandteils beachtet werden. Diese müssen separat durch eine weitere Codeliste übersetzt werden. Am Ende werden alle übersetzten Einzelbestandteile zu einem Textwert zusammengesetzt. Der SQL-Code der drei Funktionen *translate\_sep3*, *translate\_hb* und *translate\_nb* befindet sich im Anhang.

## 5.2 Anwendungsszenario 1: Abfragen von Aufschlussdaten mithilfe einer interaktiven dreidimensionalen Karte

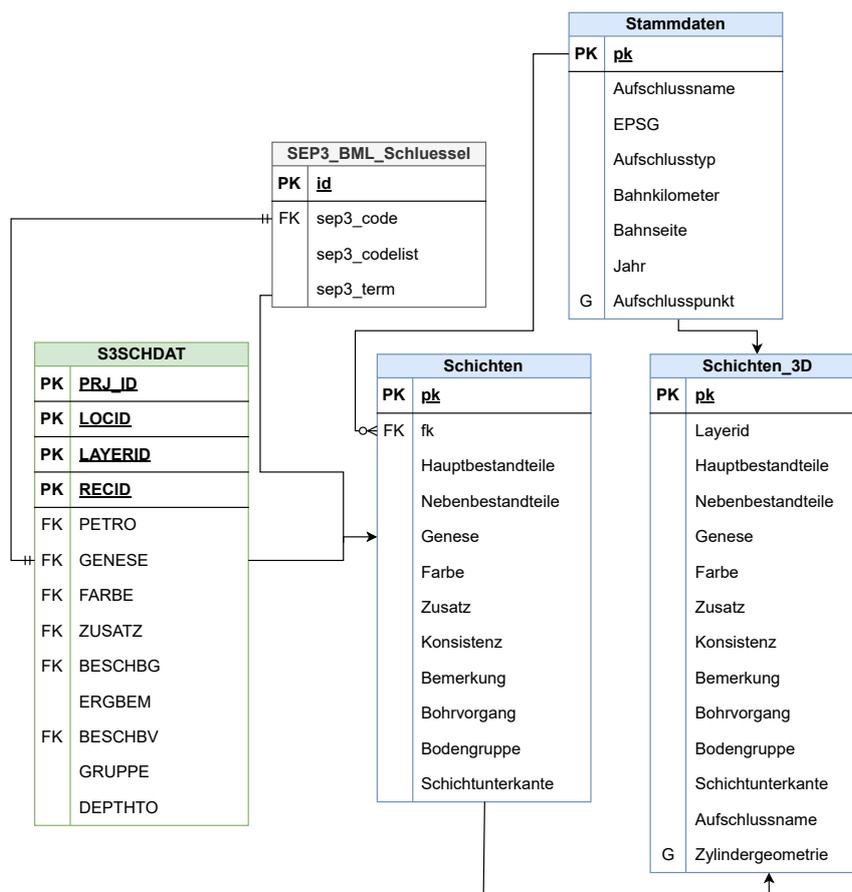


Abbildung 5.6 Abfrage der Schichtdaten

Durch die Nutzung der Funktionen können alle Schichtattribute übersichtlich innerhalb der Sicht *Schichten* abgefragt werden. In diesem wird neben der Abfrage der Schichtattribute zusätzlich ein Fremdschlüssel zur Referenzierung des zugehörigen Aufschlusses definiert. Dementsprechend entspricht dieser exakt dem Primärschlüssel der Sicht *Stammdaten* (siehe Listing 5.3).

Die Sicht *Schichten* enthält bis jetzt noch keine Geometrie, da die Koordinaten nur in den Stammdaten des zugehörigen Aufschlusses abgespeichert werden. Deshalb wird nun eine weitere Sicht (*Schichten\_3D*) erstellt, die einerseits die übersetzten Schichtdaten und andererseits die dreidimensionale Zylinderdarstellung enthält. Um eine erneute Erstellung der Geometrie des Aufschlusspunktes zu vermeiden, soll die bereits erstellte Geometrie aus der Sicht *Stammdaten* verwendet werden. Die Verbindung der Sicht *Stammdaten* und der Sicht *Schichten* erfolgt durch den äquivalent definierten Primär- beziehungsweise Fremdschlüssel in den beiden Sichten (siehe Abbildung 5.6). Nun soll aus den vorhandenen Parametern eine Zylindergeometrie erzeugt werden.

Um diese komplexe Abfolge von Abfragen und Funktionsaufrufen auszulagern, wird dies wieder durch eine PL/pgSQL Funktion umgesetzt. Der zentrale Teil dieser Funktion ist in Listing 5.4 dargestellt und wird zusätzlich durch die Abbildung 5.7 erklärt. Die grundlegende Idee besteht darin, ausgehend von der Unterkante der Schicht einen Zylinder zu extrudieren, dessen Höhe der Dicke der Schicht entspricht. Hierfür muss zunächst der dreidimensionale Punkt der Schichtunterkante bestimmt werden. Um diesen Punkt zu erstellen, kann zunächst die X- und Y-Koordinate sowie der EPSG-Code des Aufschlusspunktes übernommen werden, da von einer vertikal nach unten verlaufenden Bohrung aus-

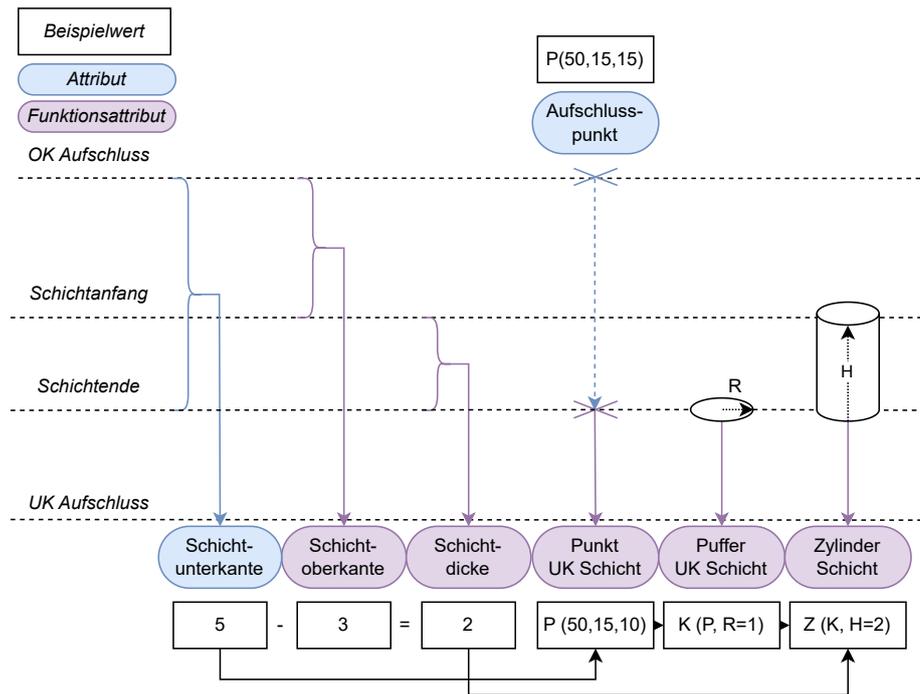


Abbildung 5.7 Visualisierung der Attribute zur Erstellung eines Schichtzylinders

gegangen wird (siehe Punkt *UK Schicht* in Abbildung 5.7 und Zeile 5-7 in Listing 5.4). Dadurch weicht nur die Höhenkoordinate des gesuchten Punkts von der des Aufschlusspunktes ab. Diese muss durch Abziehen des vertikalen Abstands (gespeichert im Attribut *schichtunterkante*) berechnet werden (Zeile 3 in Listing 5.4). Als Nächstes muss die Zylindergeometrie erzeugt werden. Hierfür wird durch die Funktion *st\_buffer* erst ein Kreis um den eben berechneten Punkt erzeugt (siehe Puffer UK Schicht in Abbildung 5.7). Der Radius wird hierbei als konstant angenommen, da in den Ausgangsdaten keine Informationen über den Bohrdurchmesser hinterlegt sind (Zeile 12). Da die Funktion *st\_buffer* nur einen Kreis mit zweidimensionalen Koordinaten ausgibt, muss dieser noch durch eine weitere Funktion *st\_force3d* auf die Höhe der Schichtunterkante gesetzt werden. Zuletzt wird die angestrebte Geometrie durch Extrusion des Kreises um die Schichtdicke erstellt. Hierfür wird die Funktion *st\_extrude* der Erweiterung *postgis\_sfcgal* verwendet (Zeile 11).

```

1  -- Vorher wird noch die Tiefe der Schichtoberkante berechnet (ok_sch)
2  -- Berechnen der Schichtdicke
3  delta_sch := schicht.schichtunterkante - ok_sch;
4  -- Erstellen des unteren Ansatzpunktes der Schicht
5  p_sch := (SELECT ST_SetSRID(ST_MakePoint(ST_X(p_stm), ST_Y(p_stm),
6           ST_Z(p_stm)-schicht.schichtunterkante),
7           ST_SRID(p_stm)));
8  -- Erzeugen eines Kreises um UK Aufschluss mit konstantem Radius
9  -- Dem entstehenden Kreis muss noch die Z-Dimension hinzugefügt werden
10 -- Extrudieren des Schichtzylinders um Schichtdicke
11 zylinder := st_extrude(st_force3D(
12   st_buffer(p_sch, 1.0, 'quad_segs=2'),ST_Z(p_sch)),0,0,delta_sch));

```

Listing 5.4 Erstellen der Geometrie in der Funktion *create\_3d\_schicht*

### 5.2.3.3 Abfrage der Grundwasserstände

Im Folgenden wird die dreidimensionale Darstellung der Grundwasserstände behandelt. Hierfür wird, äquivalent zu den Schichtdaten, zunächst eine Sicht erstellt, die wieder einen Fremdschlüssel enthält, der dem Primärschlüssel des zugehörigen Aufschlusses entspricht (siehe Abbildung 5.8). Zusätzlich werden die Attributnamen der vier theoretisch möglichen Grundwasserstände in Klartextnamen umbenannt und das Messdatum übernommen.

In SQL wird hierfür wieder eine Sicht definiert, die der Struktur von *Schichten* inklusive Primär- und Fremdschlüssel sowie der Attribute ähnelt. Darüber hinaus wird, ähnlich wie bei den Schichtdaten, eine zusätzliche Sicht erstellt, in der die dreidimensionale Geometrie des Grundwasserstandes erzeugt wird (siehe Abbildung 5.8). Für die Erstellung dieser Geometrie wird eine weitere Funktion definiert, die den Grundwasserpegel in Form eines Kreises auf Höhe des Wasserstandes darstellt. Hierfür kann dieselbe Art der Geometrierstellung wie in *create\_schicht\_3d* (siehe Listing 5.4) verwendet werden. Jedoch wird hier der letzte Schritt, die Extrusion des Zylinders, ausgelassen. Da es theoretisch drei verschiedene Grundwasserstände pro Datensatz in *S3GRUNDW* geben kann, muss vor der Geometrierstellung einer dieser Grundwasserstände ausgewählt werden. Hierfür wird in der entsprechenden Funktion eine Abfolge von Bedingungen erstellt, in der primär der Ruhewasserstand und sonst die anderen Wasserstände verwendet werden. Mit einer weiteren äquivalent aufgebauten Funktion (*get\_gw\_type*) wird zudem die Art des angezeigten Wasserstands ermittelt, um diese in GIS farblich differenziert darstellen zu können (entspricht Attribut *GW-Dargestellt*).

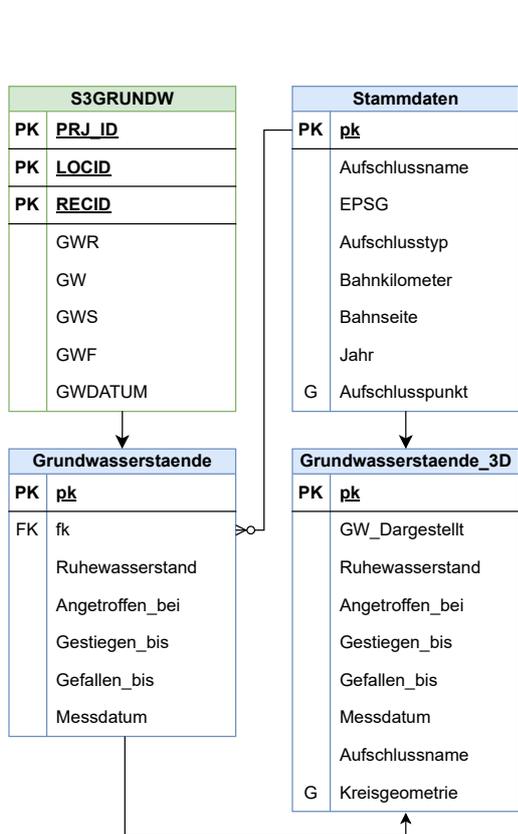


Abbildung 5.8 Abfrage der Grundwasserdaten

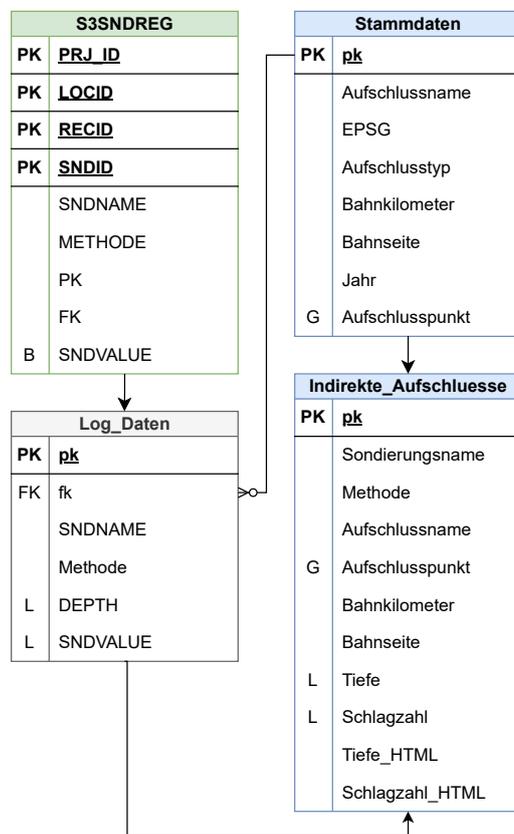


Abbildung 5.9 Abfrage der Sondierungsdaten

### 5.2.3.4 Abfrage der Sondierungen

Die letzte Abfrage soll dazu verwendet werden, die Daten der Rammsondierungen in der Karte darzustellen. Da diese, wie zuvor geschildert, von *GeoDin* in Form einer binärkodierte Datei in der Relation *S3SNDREG* gespeichert werden, müssen diese zunächst dekodiert werden. Hierfür wird in der Datenbank eine neue Relation *Log\_Daten* erstellt, die alle Informationen aus *S3SNDREG* sowie die bereits bekannten Primär- und Fremdschlüssel enthalten soll (siehe Abbildung 5.9). Die Dekodierung und Übertragung der Daten in diese Tabelle soll mithilfe eines *Python*-Skripts erfolgen, das über Funktionen der Bibliothek *sqlAlchemy* mit der Datenbank kommuniziert. Auf die Funktionsweise des Skripts soll nun nicht weiter eingegangen werden, da die verwendeten Funktionen und Bibliotheken nicht für das Verständnis des Ergebnisses benötigt werden. Der Code befindet sich jedoch in kommentierter Form im Anhang. Durch die Ausführung des Skripts werden die Daten dekodiert und pro Sondierung in jeweils ein Listenobjekt des Attributs *DEPTH* (für alle Tiefenwerte) sowie *SNDDATA* (für die Schlagzahlen der jeweiligen Tiefe) in der Relation *log-daten* geschrieben. Diese bildet nun die Grundlage für die Sicht *indirekte\_aufschlusse*, die die erstellten Sondierungsdaten und die Geometrie des Aufschlusspunkts anhand des vergebenen Fremdschlüssels *pk* verknüpft (siehe Abbildung 5.8). Zusätzlich müssen aus den Listen zwei weitere Attribute *Tiefe-HTML* und *Schlagzahl-HTML* erzeugt werden, die die Abbildung der Daten in einem HTML-Diagramm ermöglichen.

### 5.2.4 Visualisierung in QGIS

Die Abfragen können nun alle direkt in *QGIS* eingebunden werden. Hierfür muss zunächst die Verbindung zu der erstellten *PostgreSQL*-Datenbank über den *Data Source Manager* hergestellt werden. In diesem wird anschließend eine Übersicht aller Sichten und Tabellen mit Geometrieattributen ange-

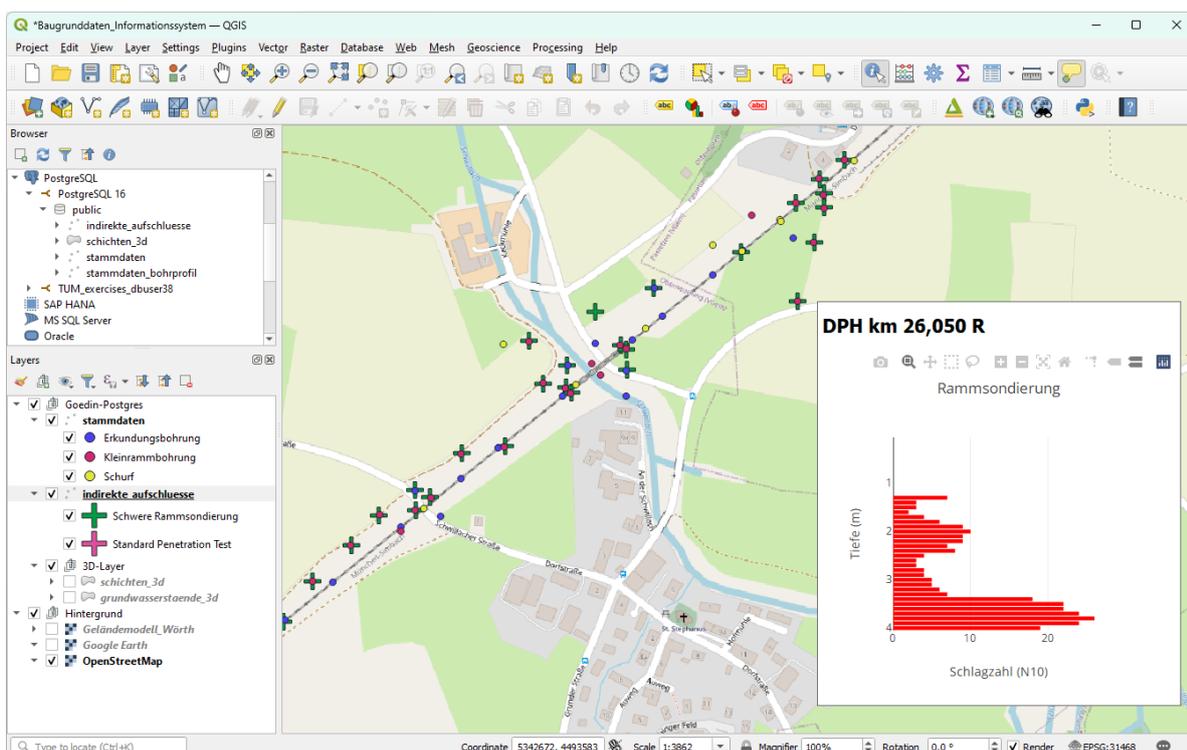


Abbildung 5.10 2D-Visualisierung direkter und indirekter Aufschlüsse in QGIS

## 5.2 Anwendungsszenario 1: Abfragen von Aufschlusdaten mithilfe einer interaktiven dreidimensionalen Karte

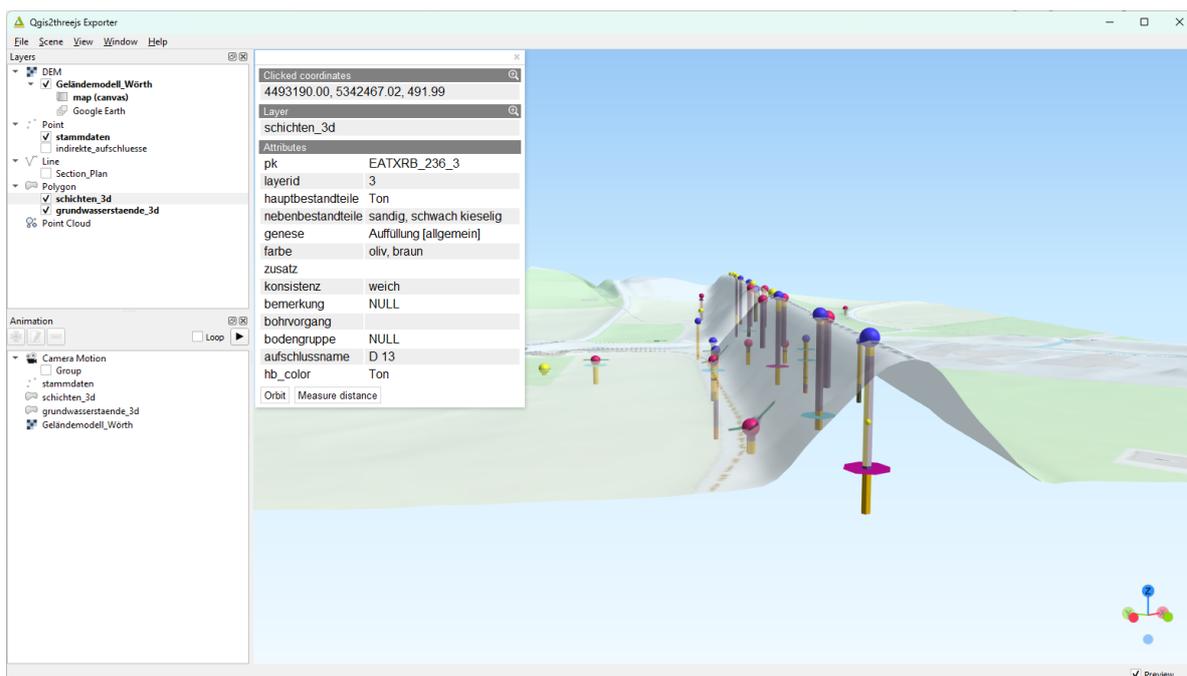


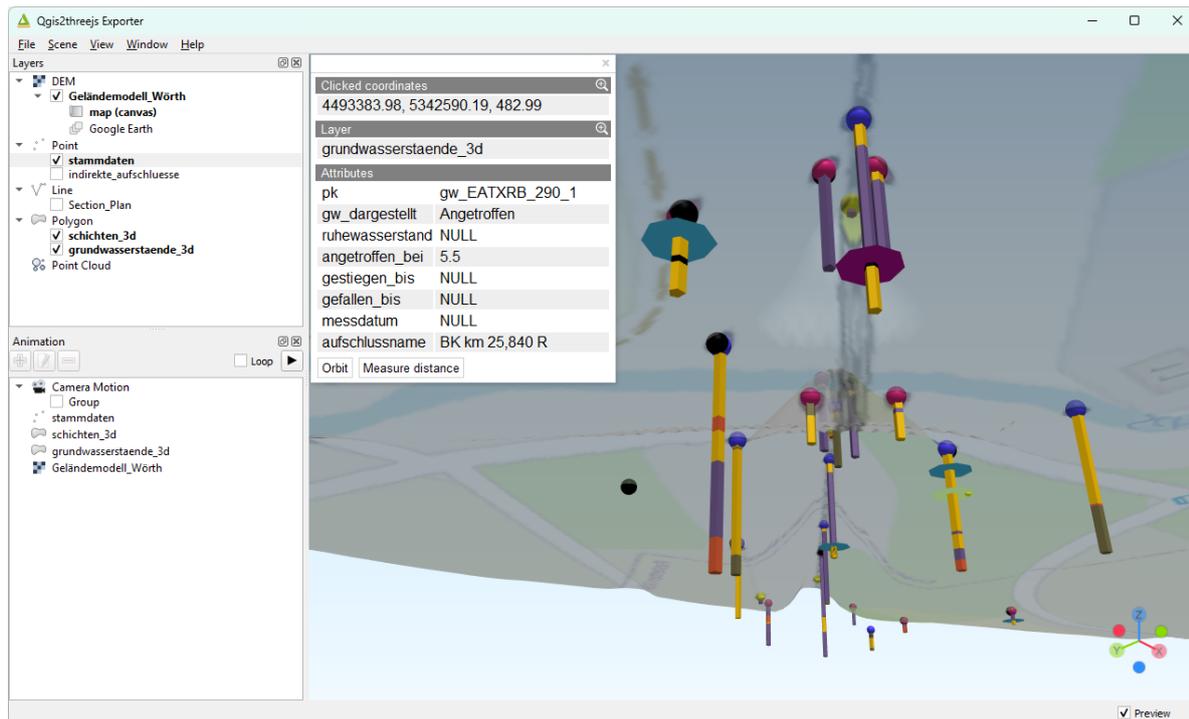
Abbildung 5.11 3D-Visualisierung der Schichtdaten in QGIS inklusive Abfrage der Daten einer Schicht

zeigt (*Stammdaten*, *Schichten\_3D*, *Grundwasserstaende\_3D* und *Indirekte\_Aufschlusse*). Diese vier Sichten können nach Durchführung aller im vorherigen Abschnitt beschriebenen Abfragen nun direkt eingefügt werden. Das QGIS-Projekt sollte eine Hintergrundkarte (z.B. Openstreetmap, Openstreetmap Contributors o. D.) und ein digitales Geländemodell (in Bayern z.B. Bayerische Vermessungsverwaltung o. D.) am Ort der Bohrungen darstellen, um diesen einen räumlichen Kontext zu verleihen. Nun muss nur noch die Darstellung der einzelnen Layer anhand der Attribute angepasst werden. Für die Stammdaten werden je nach Wert des Attributs *Aufschlusstyp* unterschiedliche Farben verwendet und zudem eine Filterfunktion angesetzt, die nur die direkten Aufschlüsse in diesem Layer darstellt (siehe Layer-Baum in Abbildung 5.10 unten links). Jeder Punkt in der Karte repräsentiert dabei einen dieser Aufschlüsse.

Die indirekten Aufschlüsse befinden sich hingegen in der Sicht *indirekte\_aufschlusse* und müssen noch mithilfe eines Schlagzahldiagramms visualisiert werden. Hierfür wird die QGIS-Funktion „HTML Map Tips“ verwendet, die beim Hovern mit der Maus über einem entsprechenden Objekt ein HTML-Fenster anzeigt (QGIS Documentation o. D.). In den Layereinstellungen wird der entsprechende HTML-Code eingefügt. Dieser umfasst eine leicht bearbeitete Vorlage für ein Balkendiagramm der *Plotly*-Bibliothek in der Sprache *JavaScript*. Diesem werden an der Definition der x- und y- Achse die entsprechend formatierten Listen *Tiefe\_HTML* und *Schlagzahl\_HTML* übergeben, womit diese für jedes Objekt entsprechend definiert werden. Als Ergebnis wird beim Hovern über einem als Kreuz dargestellten indirekten Aufschluss ein Schlagzahldiagramm angezeigt, wobei der Aufschlussname als Titel des Diagramms dargestellt wird. (siehe Abbildung 5.10 rechts unten).

Darüber hinaus muss auch die Darstellung der dreidimensionalen Layer angepasst werden. Eine dreidimensionale Ansicht des betrachteten Kartenausschnitts wird hierfür durch das QGIS-Plugin *Qgis2threejs* ermöglicht. Die Hintergrundkarte der 2D-Ansicht wird von diesem automatisch auf das digitale Geländemodell projiziert und im Rahmen dieser Arbeit leicht transparent dargestellt, um eine Ansicht

## 5 Prototypische Umsetzung von zwei Anwendungsszenarien in einem DB-Projekt



**Abbildung 5.12** Visualisierung der 3D-Grundwasserdaten inklusive Abfrage der Daten eines Grundwasserstands

der darunterliegenden Aufschlüsse zu ermöglichen (siehe Abbildung 5.11). Außerdem wird die Höhe aller Objekte zur besseren Visualisierung zweifach überhöht dargestellt.

Der nächste Schritt besteht darin, die Zylinder zur Repräsentation der Schichtdaten entsprechend ihres ersten Hauptbestandteils (Attribut *hb\_color*) einzufärben. Die Farben wurden der DIN 4023 entnommen und zusammengehörende Attributwerte zu einer Darstellung zusammengefasst (z.B. Mittelsand und Feinsand zu Sand) (DIN4023 2023). So können Bodenschichten benachbarter Aufschlüsse in der 3D-Darstellung sofort gedanklich verknüpft werden, wodurch schnell ein räumliches Verständnis der vor Ort vorliegenden Bodenverhältnisse hergestellt werden kann. So kann man anhand der violett dargestellten Schichtzylinder innerhalb des in Abbildung 5.11 rechts dargestellten Bahndamms schnell erkennen, dass dieser zum Großteil aus einer Tonschicht besteht. Ferner stellt das verwendete Plugin die Funktion bereit, die Attribute der Objekte durch einen Klick abzufragen, wodurch alle Informationen direkt in der Darstellung verfügbar werden (siehe Abfragefenster in Abbildung 5.11 links oben).

Darüber hinaus müssen noch die Grundwasserstände dargestellt werden. Die in der Abfrage erstellten dreidimensionalen Kreise werden je nach Art (z.B. Ruhewasserstand) ebenfalls eingefärbt (siehe Abbildung 5.12). Angetroffene Grundwasserstände werden hier hellblau, gestiegene Grundwasserstände violett dargestellt. Zuletzt können der dreidimensionalen Darstellung noch die Stammdaten in Form von Kugeln an den Ansatzpunkten der Bohrungen dargestellt werden. Somit können auch diese Informationen durch einen Klick in der Karte abgefragt werden.

## 5.3 Anwendungsszenario 3: Export einzelner Aufschlüsse für den Austausch

### 5.3.1 Konzept für Anwendungsszenario 3

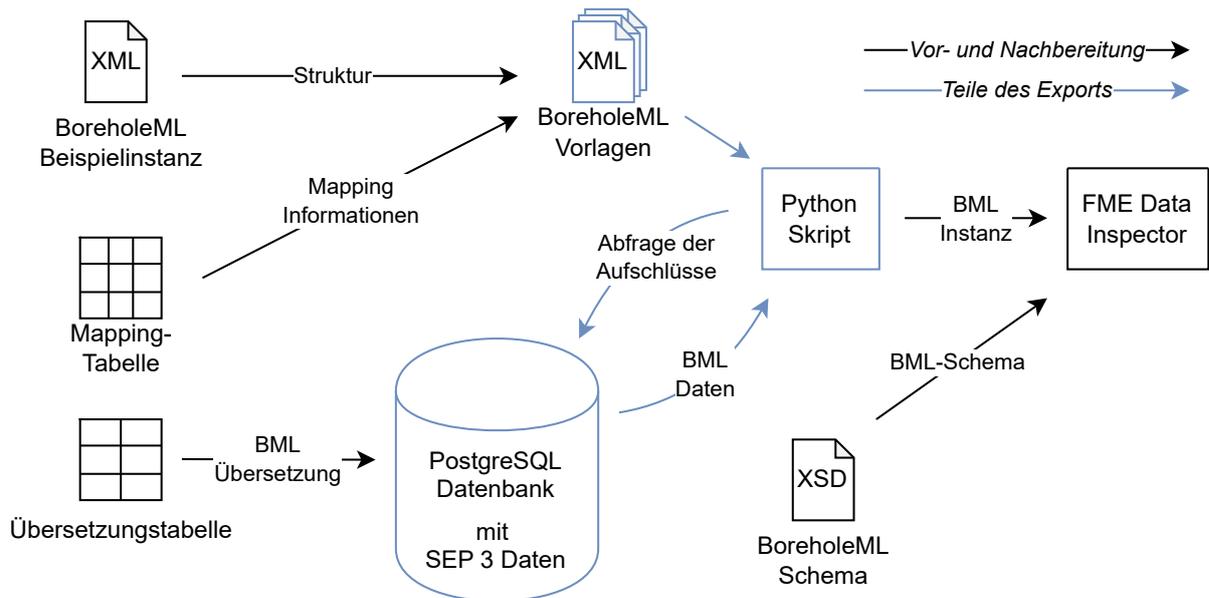


Abbildung 5.13 Workflow für Anwendungsszenario 3

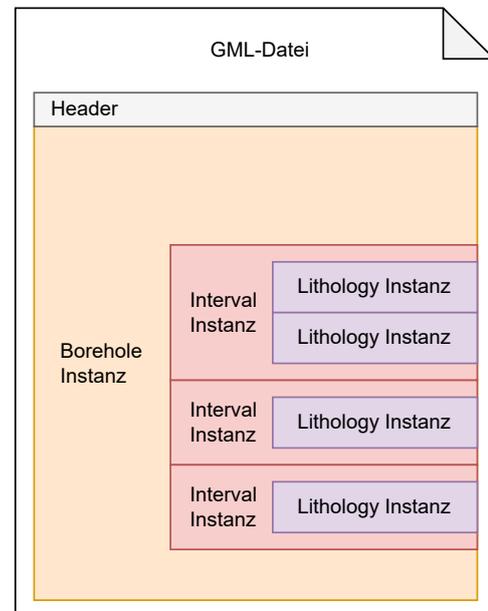
Im dritten Anwendungsszenario werden die Daten aus der *SEP 3*-Datenbank für den Austausch mit Externen in eine GML-Datei gemäß des Anwendungsschemas *BoreholeML* exportiert. Der Workflow von der *PostgreSQL*-Datenbank bis zur überprüften GML-Datei ist in Abbildung 5.13 dargestellt. Diese zeigt die Schritte der Vor- und Nachbereitung des Exports durch schwarze Pfeile, während die Teile des eigentlichen Exports durch blaue Pfeile dargestellt werden.

Als Grundlage liegt neben der *PostgreSQL*-Datenbank eine *BoreholeML*-Beispielinstantz, die in Abschnitt 4.2 erarbeitete Mappingtabelle sowie die Übersetzungstabelle zur Überführung der *SEP 3*-Codierungen in *BoreholeML* vor (siehe Abbildung 5.13 links). Aus der Beispielinstantz werden Vorlagen erstellt, die die Struktur der finalen *BoreholeML*-Instanz vorgeben. In dieser werden anhand der Mapping-Tabelle diejenigen Stellen vorgegeben, an denen die Daten aus der *SEP 3*-Datenbank eingefügt werden sollen. Die zentrale Steuerung und Durchführung des Exports wird mittels eines Python-Skripts vorgenommen. Dieses fragt zunächst die zu exportierenden Aufschlüsse in der Datenbank ab, welche die Daten der Aufschlüsse anschließend entsprechend der *BoreholeML*-Codelisten zurückgibt. Im nächsten Schritt werden die Objekte der Aufschlüsse erstellt und abschließend zu einer finalen GML-Datei zusammengefügt. Diese Instanzen werden anschließend im *FME Data Inspector* anhand des *BoreholeML*-Schemas validiert, um sicherzustellen, dass eine gültige GML-Datei erstellt wird, die dann beispielsweise in *QGIS* visualisiert werden kann.

### 5.3.2 Voraussetzungen für den Export

Als Voraussetzung für den Datenexport muss zunächst eine *PostgreSQL*-Datenbank inklusive der *SEP 3*-Daten vorliegen. Hierbei ist insbesondere auch das Vorhandensein der Übersetzungstabelle wichtig, die bereits für die Umsetzung des Anwendungsszenarios 1 in die Datenbank importiert wurde. Diese enthält neben den Übersetzungen der *SEP 3*-Codierungen in Klartext, jeweils auch eine Spalte für den entsprechenden Code in *BoreholeML* sowie den entsprechenden Namen der Codeliste. Somit können die benötigten *BoreholeML*-Codierungen in einer Abfrage direkt übersetzt werden.

Zusätzlich müssen die im Konzept erwähnten Vorlagen zur Vorgabe der finalen Struktur der Daten erstellt werden. Hierfür wird eine *BoreholeML*-Beispieldatei zunächst in eine große Vorlage und mehrere kleinere Vorlagen zerteilt. Die große Vorlage repräsentiert alle Objekte und Attribute einer Instanz des Feature Types *Borehole*, die genau ein Mal vorkommen können. In diese werden an bestimmten Stellen mehrere befüllte Vorlagen derjenigen Objekte eingefügt, die mehrfach vorkommen können (*Interval* und *Lithology*). Dieses „Baukasten-Prinzip“ ist in Abbildung 5.14 an einem Beispiel symbolisch dargestellt. Dadurch kann jede *Borehole*-Instanz flexibel anhand der gegebenen Anzahl der abzubildenden Instanzen erstellt werden. Die zu befüllenden Parameter werden durch Keywords markiert, die immer aus einem „#“, kombiniert mit dem jeweiligen *SEP 3*-Attribut in Kleinbuchstaben, das an der Stelle eingefügt werden soll, bestehen (siehe Beispiel *fullName* in Listing 5.5). Zusätzlich wird die Konvention eingeführt, die Stellen an denen Objekte (befüllte Vorlagen) eingefügt werden, wieder mit einem „#“ aber nur mit Großbuchstaben zu markieren (z.B. *#LITHOLOGY*). So wird in den Vorlagen sofort ersichtlich, an welchen Stellen dies der Fall ist. Die Vorlage für Instanzen des Typs *Interval* wird nachfolgend vereinfacht *Layer* genannt, da diese abstrakte Klasse, wie in Kapitel 4 geschildert, alle Schichtattribute enthält. Zudem enthält der Datensatz keine Unterschichten, weshalb es keine Objekte der Klasse *Component* gibt, die die zweite Kindklasse von *Layer* darstellt.



**Abbildung 5.14** Schematische Darstellung der Zusammensetzung der GML-Datei aus den Vorlagen

```

1 <bml:fullName>
2   <gmd:LocalisedCharacterString locale="http://www.infogeo.de/boreholeml
3     /3.0/codelists/DE.xml">#longname</gmd:LocalisedCharacterString>
  </bml:fullName>

```

**Listing 5.5** Auszug aus der Vorlage für Borehole (Beispiel BML-Attribut fullName)

### 5.3.3 Umsetzung der Datentransformation von SEP 3 zu BoreholeML

Für die konkrete Umsetzung der Transformation werden zuerst wieder Sichten definiert, die die Daten aus der *SEP 3*-Datenbank in die *BoreholeML*-Codierung übersetzen. Dafür wird, ähnlich wie bei der Abfrage der Schichtdaten (siehe Abbildung 5.6) eine Funktion in PL/pgSQL definiert, die jedes Element der Textketten übersetzt (*translate\_bml*). Nur wird hierbei nicht der Klartext, sondern der entsprechende Wert der *BoreholeML*-Codeliste zurückgegeben. Das Codebeispiel für die Übersetzung des Haupt- und Nebenanteils sowie einer allgemeinen Funktion für alle weiteren Attribute befindet sich im Anhang. Da *GeoDin* den Wert der Schichtoberkante nicht in das entsprechende Feld speichert, muss dieser ebenfalls über eine entsprechende Funktion aus der jeweils darüberliegenden Schicht ermittelt werden (*get\_depthfrom*).

Mit diesen Funktionen werden in zwei neu definierten Sichten *boreholeml\_stm\_data* und *boreholeml\_sch\_data* alle für die Transformation benötigten Attribute ausgegeben (siehe Anhang). Die Sicht der Stammdaten ist mit der aus Anwendungsszenario 1 bekannten Rohdatenliste verknüpft, um eine Filterung nach Bahnkilometer zu ermöglichen. Die Attributnamen innerhalb der Sichten entspricht hierbei bereits den vorher erwähnten Keywords zur Markierung der zu befüllenden Stellen in den erstellten Vorlagen.

Nun sind alle benötigten Voraussetzungen für das *Python*-Skript gegeben. Die zentrale Funktion *sep3\_to\_bml* wird im Folgenden anhand eines mit Pseudocode befüllten Schemas erläutert (siehe Abbildung 5.15). In diesem ist das Befüllen der jeweiligen Vorlage mit derselben Farbe markiert wie in Abbildung 5.14. Grundwasser- und Sondierungsdaten werden aufgrund der in Kapitel 4 geschilderten Gründe nicht transformiert.

Nachdem eine Verbindung zur *PostgreSQL*-Datenbank hergestellt wurde, erstellt die Funktion eine Abfrage, die nur diejenigen Datensätze der Sicht *boreholeml\_stm\_data* abfragt, die durch die Eingabeparameter spezifiziert werden. Dadurch werden die Datensätze anhand der Eingabeparameter gefiltert. Der Code im Anhang zeigt beispielhaft eine Filterung nach Bahnkilometer und Aufschlusstyp. Als Nächstes werden die vorher beschriebenen Vorlagen als Texttribute importiert, die nun befüllt werden müssen. Um immer die Anzahl an Instanzen der jeweiligen

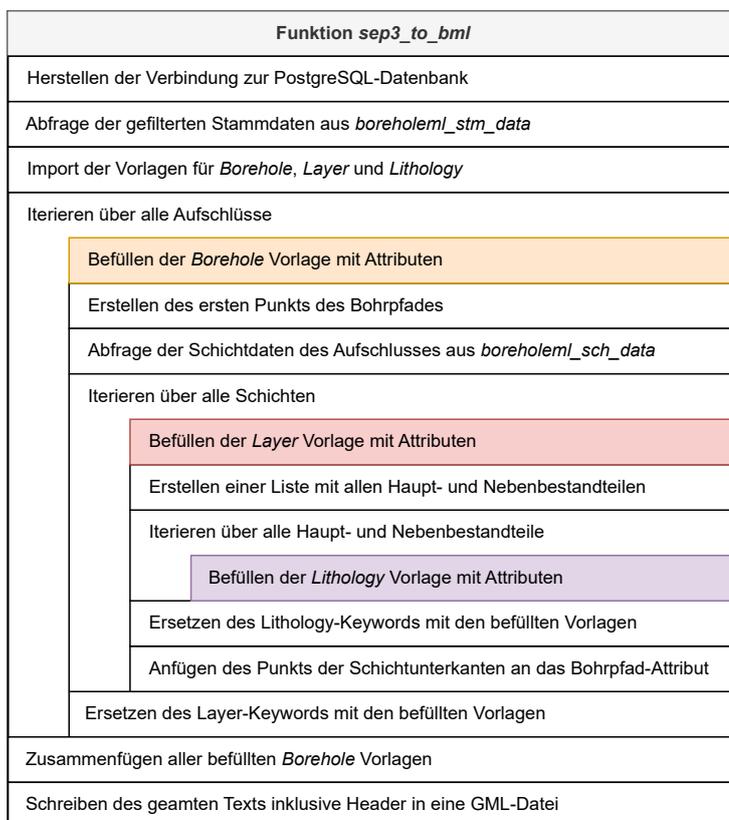


Abbildung 5.15 Pseudocode der Funktion *sep3\_to\_bml*

Vorlagen zu erzeugen, die in der Abfrage ausgegeben werden, werden Schleifen verwendet. Die erste iteriert folglich über alle Aufschlüsse (siehe erste Ebene mit eingerücktem Text in Abbildung 5.15) und befüllt die Attribute durch eine allgemein definierte Funktion *fill\_template* (siehe Listing 5.6). Diese erstellt zunächst eine Liste aller abgefragten Attributnamen und prüft anschließend die Existenz der entsprechenden Keywords in der Vorlage. Die Keywords werden daraufhin in der Vorlage durch den Wert aus der Datenbank ersetzt (siehe Listing 5.6 Zeile 14). Falls der Attributwert leer ist, wird das Keyword einfach aus der Vorlage gelöscht (siehe Zeile 12). Die Funktion ermöglicht eine Erweiterung des Exports um weitere Attribute, ohne dass der Python-Code geändert werden muss. Hierfür muss das neue Attribut nur der Vorlage sowie der entsprechenden Sicht angehängt werden.

```

1 def fill_template(data, template):
2     # Erstellen einer Liste mit allen Attributen der abgefragten Daten
3     attributes = data.columns.tolist()
4     for a in attributes:
5         # Finden der entsprechenden Stelle in der Vorlage (#attributname)
6         key = '#' + a
7         real_key = '>' + key + '<'
8         # Ersetzen der Keywords durch die Daten aus der Datenbank
9         if real_key in template:
10            # Löschen des Keywords im Fall eines leeren Attributs
11            if data[a].item() == "":
12                template = template.replace(key, "")
13            else:
14                template = template.replace(key, str(data[a].item()))
15        filled_template = template
16    return filled_template

```

**Listing 5.6** Allgemeine Funktion für das automatische Befüllen der Vorlagen

Sobald die Vorlage befüllt wurde, wird der erste Punkt des *BoreholeML*-Bohrpfads erstellt. Dieser besteht im Fall der Datengrundlage eigentlich nur aus dem Anfangs- und dem Endpunkt der Bohrung, da immer von einer rein vertikalen Bohrung ausgegangen wird. Um die Lage der Schichten innerhalb der finalen GML-Datei zu repräsentieren, werden hier jedoch auch die Punkte der Schichtunterkanten aufgenommen. Dies wird später in der Funktion umgesetzt. Laut der Mapping-Tabelle muss ein Datensatz der Relation *S3SCHDAT*, repräsentiert durch die Sicht *boreholeml\_sch\_data*, auf je eine Instanz der Klasse *Interval* gemappt werden. Hierfür werden als Nächstes die Schichtdaten des betrachteten Aufschlusses durch eine SQL-Abfrage der entsprechenden Sicht in die Funktion übernommen. Über diese wird nun in einer weiteren Schleife iteriert und die Vorlage *Layer* wieder über die vorher vorgestellte Funktion befüllt (siehe roter Block in Abbildung 5.15). In dieser Abfrage befindet sich auch das Attribut *PETRO*, das wie in Kapitel 4 geschildert, auf mehrere Instanzen der Klasse *Lithology* gemappt werden muss. Hierfür wird zunächst eine Liste mit allen Haupt- und Nebenteilen erstellt, über die wiederum eine weitere Schleife läuft. In dieser wird nun die *Lithology*-Vorlage mit den abgefragten *BoreholeML*-Codes für Petrographie befüllt (siehe violetter Block in Abbildung 5.15). Die erstellten Instanzen werden an der entsprechenden Stelle in der Vorlage *Layer* eingefügt. Zusätzlich wird der jeweilige Unterpunkt der Schicht dem vorher erstellten Bohrfad-Attribut angehängt. Nun müssen alle erstellten *Layer*-Instanzen in die *Borehole*-Vorlage eingefügt werden, womit eine vollständige *BoreholeML*-Instanz vorliegt. Zuletzt werden alle Einzelinstanzen zu einem langen Textwert zusammengehängt. Dieser kann nach Anfügen des Headers in eine finale GML-Datei nach *Boreho-*

leML-Anwendungsschema exportiert werden. Eine entsprechend erstellte Beispieldatei befindet sich im Anhang (*SEP3\_DB\_BML\_Bauwerk\_X.gml*).

#### 5.3.4 Überprüfung und Visualisierung der BoreholeML-Daten

Um die Syntax der erzeugten Datei zu überprüfen, wird diese inklusive des XSD-Schemas in die Software *FME Data Inspector* geladen (siehe Abbildung 5.16). Da die erstellte Datei dem Schema entspricht, werden die Daten hier sowohl in einer interaktiven Tabelle (rechts) als auch in einer Kartenansicht (links oben) angezeigt. Links unten sind alle fünf *Borehole*-Instanzen aufgelistet. Hier können die Stammdaten wie zum Beispiel der Name (*fullName*), die Tiefe des Aufschlusses (*totalLength*) oder Export-Attribute wie das Export-Datum (*exportDate*) angezeigt werden. Die einzelnen Instanzen, insbesondere der Schichten können rechts in einer Tabelle begutachtet werden. Für die dargestellte Schicht (*IntervalSeries.layer2*) ist, gemäß des *BoreholeML*-Anwendungsschemas, die Zusammensetzung einmal im Format der Ausgangsdatenbank (*rockCode*), in Klartext (*rockNameText*) und zusätzlich in Form mehrerer *Lithology*-Objekte (*Interval.Lithology*) gespeichert. Die erste dieser Instanzen repräsentiert den Hauptanteil (hier Grobkies) während die zweite den ersten Nebenbestandteil (hier sandig) darstellt.

Zusätzlich kann die Datei aufgrund ihres GML-Formats in *QGIS* visualisiert werden (siehe Abbildung 5.17). In der zweidimensionalen Ansicht erscheinen die als *GML LineString* gespeicherten Bohrpfade der Aufschlüsse zunächst nicht, da es sich bei diesen um vertikale Linien handelt. Hierfür muss der Linie zunächst eine *Marker Line* hinzugefügt werden, wodurch die Aufschlüsse als Punkte in der Karte sichtbar werden. Relevant ist hier jedoch vor allem die dreidimensionale Visualisierung, für die wieder das Plugin *Qgis2threejs* verwendet wird. In diesem können die Bohrpfade in Form von Zylindern dargestellt werden (siehe Abbildung 5.17). Dadurch erhält man jedoch keine einzelne Darstellung der Schichten, sondern nur des gesamten Bohrpfades. Obwohl die Schichtunterkanten in der vorher beschriebenen Python-Funktion in dem Geometrieattribut gespeichert wurden, sind diese nicht als einzelne Objekte dargestellt und außerdem nicht direkt mit den entsprechenden Schichtattributen verknüpft. Deshalb lassen diese sich auch nicht entsprechend ihres Hauptbestandteils einfärben (siehe Anwendungsszenario 1). Trotzdem ist zumindest ein dreidimensionaler Bezug hergestellt und die meisten Aufschlussdaten in strukturierter Form innerhalb der Datei gespeichert.

## 5 Prototypische Umsetzung von zwei Anwendungsszenarien in einem DB-Projekt

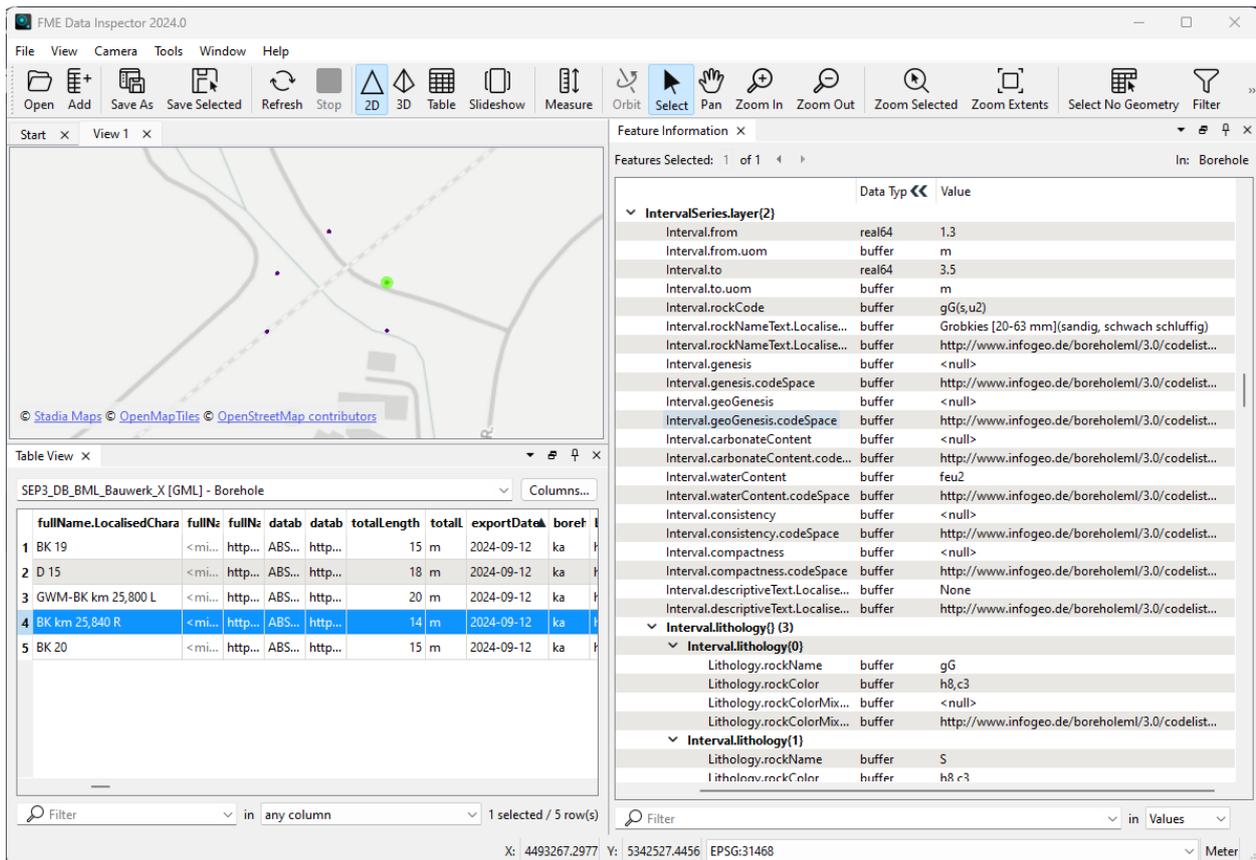


Abbildung 5.16 Prüfen einer erzeugten GML-Datei mit fünf Aufschlüssen in *FME Data Inspector*

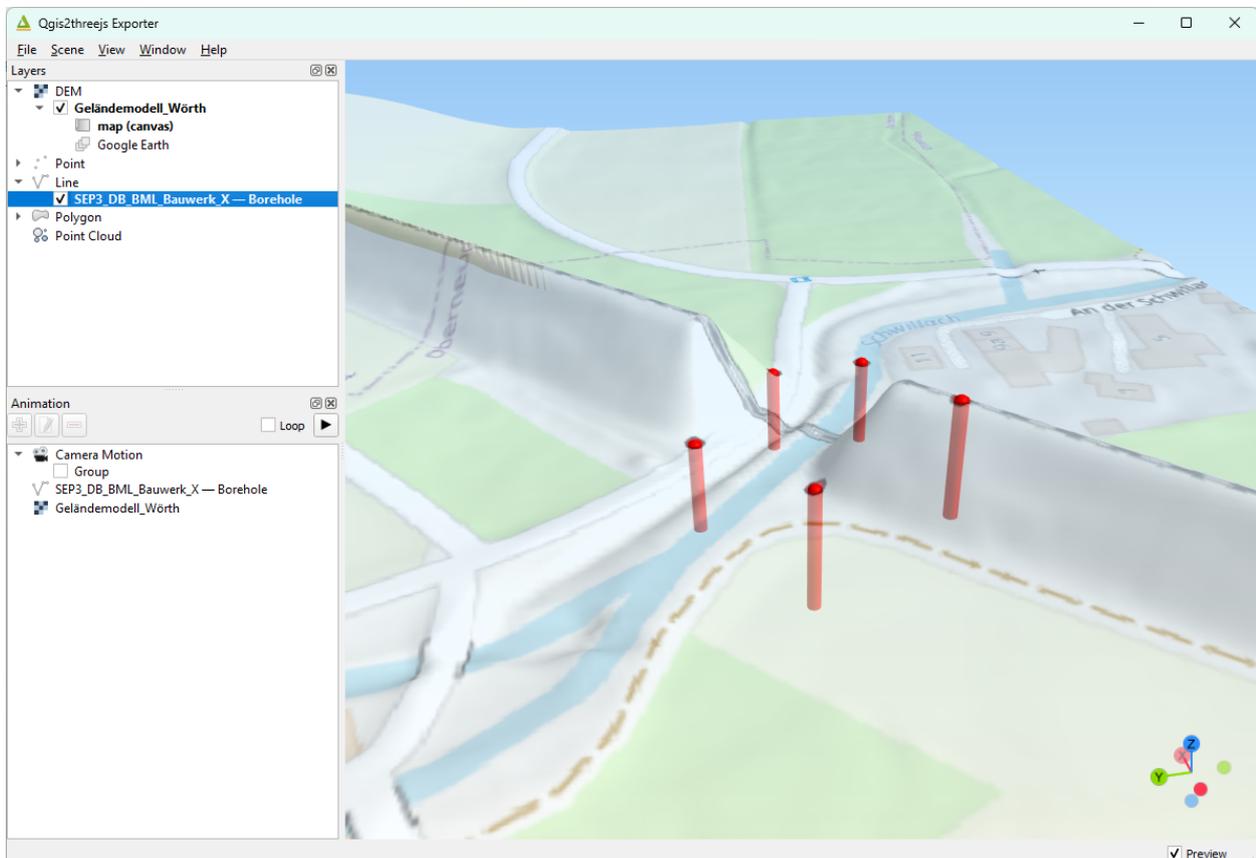


Abbildung 5.17 Dreidimensionale Visualisierung einer GML-Datei in *QGIS*

# 6 Diskussion der Ergebnisse

## 6.1 Bewertung der prototypischen Umsetzung

### 6.1.1 Bewertung des Anwendungsszenarios 1

Die Diskussion der verschiedenen Datenmodelle für den Anwendungsfall wurde bereits ausführlich in Kapitel 4 vorgenommen. Deshalb soll im Folgenden zunächst nur die in Kapitel 5 beschriebene Umsetzung bewertet werden.

Ein zentrales Ziel dieser Arbeit war die Bereitstellung der Aufschlusdaten eines DB-Projekts in einem herstellernerneutralen, editierbaren Datenformat, sodass diese in einer Datenbank abgerufen werden können. Dieses Ziel wurde größtenteils durch den Einsatz der Geotechnik-Software *GeoDin* bereits im Vorfeld dieser Arbeit erreicht. Durch die Entscheidung für diese Software und für das Modell *SEP 3*, wurde zudem die Diskussion der Wahl des Datenmodells in Kapitel 4 bereits maßgeblich beeinflusst. Im betrachteten Anwendungsfall war es somit am einfachsten, das bestehende Datenmodell in der Datenbank beizubehalten und die dreidimensionale Visualisierung „on-the-fly“ durch Abfragen in einer *PostgreSQL*-Datenbank vorzunehmen. Dadurch konnte, gerade im Hinblick auf die eingeschränkte Software-Verfügbarkeit im Projekt, eine praktikable Lösung erarbeitet werden, die auf den bestehenden Daten aufbaut. Als besonders positiv hervorzuheben ist dabei die automatische Aktualisierung der Datenbank bei der Eingabe neuer Daten, die somit den Aufwand für die Pflege der Datenbank reduziert. Die Wahl des Formats *SEP 3* bietet für den gegebenen Anwendungsfall neben der Kompatibilität mit *GeoDin* einige Vorteile. Durch den Einsatz dieses Modells können die Ausgangsdaten, einschließlich der Rammsondierungen, umfassend abgebildet werden, sodass alle Informationen zentral in einer einzigen Datenbank gespeichert werden. Außerdem werden die Daten bei der Eingabe direkt überprüft und entsprechen somit automatisch der in den Codelisten definierten Syntax.

Darüber hinaus wurden einige Nachteile des Datenmodells in der konkreten Umsetzung behoben. Beispielsweise können die zusammengesetzten Codes der Schichtdaten mithilfe von PL/pgSQL-Funktionen innerhalb der Abfragen so umgewandelt werden, dass sie zwar weiterhin in einem einzigen Attribut gespeichert, jedoch als Liste dargestellt werden. Dies macht sie beispielsweise besser für Filteranwendungen nutzbar. Damit ist zudem das zuvor beschriebene Problem, dass das Modell nicht der ersten Datenbanknormalform entspricht, weitestgehend behoben. Darüber hinaus kann die fehlende Erweiterbarkeit des Datenmodells durch das Hinzufügen weiterer Relationen in der *PostgreSQL*-Datenbank kompensiert werden. Hierbei entsteht durch die manuelle Aktualisierung dieser Tabelle bei der Eingabe neuer Daten jedoch ein dauerhafter Arbeitsaufwand. Außerdem müssen auch die Sondierungsdaten durch das entsprechende Python-Skript manuell aktualisiert werden, um die Binärdarstellung in eine Listendarstellung umzuwandeln.

Insgesamt kann die Datengrundlage durch die vorgestellte Lösung jedoch nahezu vollständig repräsentiert und die Nachteile des Datenmodells *SEP 3* weitestgehend kompensiert werden. Die Anforde-

rungen werden aus dieser Sicht somit größtenteils erfüllt.

Ferner muss auch die Erfüllung der Anforderungen im Hinblick auf die Visualisierung in *QGIS* bewertet werden. Diese kann beispielsweise mit der bereits vorhandenen Visualisierung der Karten- und Bohrprofilansicht in *GeoDin* verglichen werden. Die zweidimensionale Visualisierung in *QGIS* (siehe Abbildung 5.10) ist insgesamt relativ ähnlich zu der Kartendarstellung in *GeoDin* (siehe Abbildung 5.2). Die Vorteile der eigenen Umsetzung liegen zum Beispiel in den erweiterten Filtermöglichkeiten, die auch nach bahnspezifischen Attributen vorgenommen werden können. Außerdem können die Aufschlussdaten sehr einfach im Kontext anderer Daten, wie zum Beispiel der technischen Planung, dargestellt werden, da *QGIS* eine umfangreiche Zahl an Formaten unterstützt.

Eine dreidimensionale Visualisierung ist mit *GeoDin* nicht möglich. Im Gegensatz dazu visualisiert die im *QGIS*-Plugin *Qgis2threejs* umgesetzte Darstellung alle Schichten einschließlich ihrer Attribute dreidimensional. Einerseits ist die Hauptbodenart gemäß der DIN 4023-Farbcodes codiert, wodurch ein schneller Überblick über die örtlichen Bodenverhältnisse gewonnen werden kann. Andererseits fehlen in der Visualisierung weitere geotechnische Parameter wie Nebenbestandteile oder Konsistenz, die in den zweidimensionalen Bohrprofilansichten üblicherweise dargestellt werden und für geotechnische Aufgabenstellungen, wie die Erstellung von Homogenbereichen, von zentraler Bedeutung sind (Kuntsche und Richter 2021). Deswegen wurde in Interview 1 auch betont, dass eine derartige dreidimensionale Darstellung in der Praxis nur der Gewinnung eines allgemeinen Überblicks dient (vergleiche Interview 2 in Anhang). Eine weitere Einschränkung der entwickelten Lösung betrifft die Darstellung der Rammsondierungen. Diese können zwar über die *HTML Map Tips* ähnlich wie in klassischen Schlagzahl diagrammen angezeigt werden, jedoch fehlt der räumliche Kontext zu den Bohrprofilen, wodurch diese Darstellungsweise in der Praxis wahrscheinlich nur selten verwendet werden wird. Auch die dargestellten Grundwasserstände haben nur eine eingeschränkte Aussagekraft, da diese jahreszeitlich schwanken und der aufgenommene Stand somit nur eine Momentaufnahme darstellt (ebd.). Trotzdem kann dadurch ein grober Überblick über die lokal vorliegenden Grundwasserverhältnisse gegeben werden. Darüber hinaus zeigen die geführten Interviews, dass eine dreidimensionale Darstellung der Schichtdaten insbesondere im Zusammenspiel mit BIM-Modellen von großem Nutzen sein kann. Dieser Anwendungsfall, der in der vorliegenden Arbeit als Anwendungsszenario 2 bezeichnet wurde, konnte aufgrund der im Kapitel 4 dargelegten Gründe jedoch nicht realisiert werden. Dadurch wird diese Anforderung durch die erstellte Lösung nicht erfüllt.

Insgesamt stellt die umgesetzte dreidimensionale Darstellung eine attraktive Ergänzung zur Software *GeoDin* dar, sie kann jedoch bei geotechnischen Aufgabenstellungen mit hoher Sicherheit nicht deren umfassende Funktionalität ersetzen.

### 6.1.2 Bewertung des Anwendungsszenarios 3

Das Ziel des Anwendungsszenarios 3 war die Transformation einer Teilmenge der Daten in ein herstellerneutrales Datenmodell inklusive Datenformat. Hierfür wurde die Wahl des Datenmodells *BoreholeML* sowie die allgemeine Bewertung der Transformation von *SEP 3* bereits ausführlich in Kapitel 4 diskutiert. Deshalb soll erneut nur die konkrete Umsetzung bewertet werden. Die Transformation mittels einer Python-Funktion ermöglichte die Übertragung der meisten Stamm- und Schichtdaten der Ausgangsdaten in eine GML-Datei gemäß dem Anwendungsschema *BoreholeML*. Allerdings konnten zusätzliche Detaildaten wie Grundwasserstände und Sondierungen nicht in der gewünschten Form ins

Zielmodell übertragen werden. Dennoch wurde eine valide GML-Datei erstellt, die erfolgreich anhand des XSD-Schemas überprüft wurde, wodurch die Anforderungen an die Transformation weitgehend erfüllt wurden.

Zuletzt muss in diesem Kontext noch die Anwendbarkeit der transformierten GML-Datei betrachtet werden. Hierbei ist insbesondere die mangelnde Kompatibilität dieses Dateiformats mit Geotechnik-Software zu bemängeln. Zwar streben die staatlichen geologischen Dienste mit *BoreholeML* nach eigener Aussage die Etablierung eines Industriestandards, an, jedoch konnte bei einer tieferen Recherche nur *Autodesk Civil 3D* als kompatible Software gefunden werden (Scheel 2018). Obwohl GML-Dateien grundsätzlich von gängiger GIS-Software gelesen werden können, ist die Hinterlegung des XSD-Schemas notwendig, um eine korrekte Darstellung der Attributwerte zu gewährleisten (Portele 2022). Da dies in der Praxis häufig nicht möglich ist, kann in der Regel nur die Geometrie der Bohrung übertragen werden. Daher ist es fraglich, ob die entwickelte Exportfunktion innerhalb des Projekts tatsächlich Anwendung finden wird.

In diesem Kontext muss aber grundsätzlich die mangelnde Verfügbarkeit etablierter Austauschformate für Bohrinformationen nach deutschen Normen kritisiert werden. Das einzige Datenformat, das in Deutschland von verschiedenen Softwarelösungen unterstützt wird, ist eine Microsoft Access Datenbank im *SEP 3*-Format. Diese kann beispielsweise von den Programmen *GeoDin* und *WinBohr* gelesen werden (Schirren 2020).

## 6.2 Bewertung im allgemeinen Kontext

Aus einer allgemeinen Perspektive betrachtet hat die dargestellte Umsetzung vor allem den Nachteil, dass sie stark von der kostenpflichtigen Software *GeoDin* abhängt. Nur durch die Verwendung dieses Programms kann eine *SEP 3*-Datenbank erstellt werden, die die Durchführung aller in Kapitel 5 beschriebenen Abfragen ermöglicht. Eine mögliche Lösung für diese Einschränkung wäre die Nutzung der kostenlosen Dateneingabe-Software *GeoDin Shuttle*. Mit dieser kann ebenfalls eine *SEP 3*-Datenbank angelegt werden, allerdings nur als lokal gespeicherte Microsoft Access-Datenbank und nicht direkt auf einem Server (Fugro Consult GmbH o. D.). Die Übertragung der Daten müsste hier demnach manuell vorgenommen werden, wodurch insbesondere bei einer Aktualisierung der Daten zusätzlicher Arbeitsaufwand entsteht. Trotzdem könnte diese Möglichkeit die allgemeine und kostenfreie Nutzung der in Kapitel 5 vorgestellten Lösung gewährleisten.

Im Gegensatz zur aktuellen Forschung auf diesem Gebiet beschäftigt sich diese Arbeit nicht mit der Entwicklung eines dreidimensionalen Volumenmodells des Untergrunds. Hierfür muss das zugrundeliegende Datenmodell die Abbildung dreidimensionaler Volumenkörper, Oberflächen und Homogenbereiche unterstützen. Wie in Tabelle 4.1 dargestellt, bieten weder *BoreholeML* noch *SEP 3* diese Möglichkeit (Schirren 2020). Dies liegt unter anderem daran, dass diese Modelle primär für die Speicherung und Übertragung einer großen Anzahl von Baugrundaufschlüssen verwendet werden, die räumlich häufig weit voneinander entfernt liegen. Im Gegensatz dazu müssen Homogenbereiche im Rahmen des geotechnischen Berichts für ähnliche Schichten mehrerer Aufschlüsse innerhalb eines räumlich begrenzten Gebiets definiert werden (Kuntsche und Richter 2021). Deshalb ist eine allgemeine Abbildung dieser Information auf die gesamte Datenbank nicht sinnvoll, wäre aber beispielsweise bei einem Export einer Teilmenge der Aufschlüsse denkbar.

In der Literatur wird zur Abbildung dieser Daten meist auf bestehende Modelle zur allgemeinen dreidimensionalen Abbildung von Umweltobjekten zurückgegriffen, um diesen spezifische geotechnische Informationen hinzuzufügen. Tegtmeier u. a. 2014 sowie Khan, Kim und Seo 2023 erweitern jeweils das Datenmodell CityGML durch die Anwendung des ADE-Konzepts und bilden dadurch Bodenschichten durch dreidimensionale Oberflächen- oder Voxelmodelle ab. El Sibaii u. a. 2022 nutzt stattdessen die bereits existierenden Strukturen des IFC-Datenmodells zur Repräsentation dieser Informationen. Die Konzepte der genannten Papers gehen jedoch weit über den Umfang und das Ziel dieser Arbeit hinaus. Vergleichbar mit der in Kapitel 5 erstellten Lösung sind beispielsweise Arbeiten von Mc Carthy und Graniero 2006 oder Chang und Park 2004. In diesen wurden GIS-basierte Systeme für das Management und die Visualisierung geologischer Informationen erstellt. Hierbei wurden jeweils landesspezifische Standards für die Speicherung von Bohrinformationen erweitert, um diese in einer Datenbank zu speichern. Da diese Arbeiten bereits fast 20 Jahre alt sind und somit auf überholten technischen Grundlagen basieren, wurden diese in den theoretischen Grundlagen nicht näher betrachtet. Diese Arbeit hebt sich in der Hinsicht von den genannten ab, dass die Nutzung eines in Deutschland bereits etablierten Standards angestrebt wurde, anstatt ein bestehendes Modell zu erweitern. Außerdem nutzen Mc Carthy und Graniero 2006 zur Visualisierung der Daten das kostenpflichtige Geoinformationssystem *ArcGIS*, während diese Arbeit, abgesehen von der Erstellung der Daten mit *GeoDin*, auf Open Source Systeme wie *QGIS* und *PostgreSQL* zurückgreift.

Insgesamt konzentriert sich die aktuelle Forschung zu diesem Thema deutlich stärker auf die Modellierung dreidimensionaler Baugrundmodelle als auf die Abbildung von Baugrundaufschlüssen auf bestehende Datenmodelle, wobei dies eine wichtige Grundlage für die dreidimensionale Modellierung darstellt (Baumgärtel u. a. 2011).

## 7 Fazit und Ausblick

Insgesamt zeigt die Arbeit zunächst verschiedene Datenmodelle zur Repräsentation von Baugrunddaten auf und kommt aufbauend auf einer Anforderungsanalyse zu dem Schluss, dass das Datenmodell *SEP 3* aufgrund der breiten Abbildung der Datengrundlage und der Kompatibilität mit bestehender Software für den gegebenen Anwendungsfall am besten geeignet ist. Die Daten werden entsprechend dieses Datenmodells anschließend in einer *PostgreSQL*-Datenbank gespeichert und mittels Funktionen und Abfragen dekodiert. Die anschließende dreidimensionale Visualisierung erfolgt über ein Plugin des Geoinformationssystems *QGIS*. Zusätzlich wurde mittels eines Python-Skripts eine Exportfunktion programmiert, die eine ausgewählte Teilmenge der Daten in eine GML-Datei nach dem Anwendungsschema *BoreholeML* transformiert. Die vorgestellten Lösungen könnten, wie im Folgenden beschrieben, umfassend optimiert und erweitert werden.

Eine Erweiterung der in Anwendungsszenario 1 vorgestellten Datenbank und Visualisierung könnte durch die Integration von Labordaten ergänzt werden. Dadurch könnte eine in Interview 1 vorgeschlagene automatische Plausibilitätsprüfung dieser Ergebnisse mit den Felddaten umgesetzt werden. Ferner könnte die Aussagekraft der Grundwasserdarstellung erheblich verbessert werden, indem kontinuierlich aktualisierte Grundwasserstände von vor Ort installierten Messstellen importiert werden. Die hierfür benötigten Daten stehen projektintern bereits zur Verfügung, können jedoch bis jetzt nicht anschaulich visualisiert werden.

In den Kapiteln 3 und 4 wird zunächst die gemeinsame Darstellung der Baugrundaufschlüsse mit BIM-Daten des Projekts diskutiert, jedoch aufgrund des hohen Aufwands nicht umgesetzt. Die vorgeschlagene Transformation in das IFC-Datenmodell könnte aufbauend auf der durch Probanden erweiterten Datenbank aus Anwendungsszenario 1 sowie den Erkenntnissen zur Datentransformation in Anwendungsszenario 3 ebenfalls realisiert werden. Dadurch könnte wiederum eine wesentliche Anforderung erfüllt werden, die aus den projektintern geführten Interviews hervorging.

Außerdem wäre eine Optimierung der Exportfunktion in Anwendungsszenario 3 denkbar, bei der das Datenmodell *BoreholeML* für eine verbesserte Abbildung von Grundwasser-, Sondierungs- und Probanden erweitert wird, um diese bei einem Austausch ebenfalls übermitteln zu können.

Zusammenfassend stellt die vorgestellte Lösung eine solide Grundlage für den Einsatz digitaler Methoden in der Geotechnik dar und eröffnet neue Möglichkeiten zur effizienten Verarbeitung und Visualisierung geotechnischer Daten, wodurch sie den Weg für weiterführende Entwicklungen und die Integration zusätzlicher digitaler Techniken ebnet.



# A Digitaler Anhang

- Digitale Version dieser Arbeit im PDF-Format
- Transkripte der Interviews 1 und 2
- Vollständige Mapping-Tabelle (*SEP 3* zu *BoreholeML*) inklusive Kommentaren zur Umsetzung
- Programmier-Code
  - SQL-Code
  - Python-Projektordner
  - HTML-Code
- *BoreholeML*-Vorlagen
- Erstellte *BoreholeML*-Datei



# Literatur

- [1] „Stufenplan Digitales Planen und Bauen“. In: (2015). URL: <https://bmdv.bund.de/SharedDocs/DE/Publikationen/DG/stufenplan-digitales-bauen.html>.
- [2] W. Tegtmeier u. a. „3D-GEM: Geo-technical extension towards an integrated 3D information model for infrastructural development“. In: *Computers & Geosciences* 64 (2014), S. 126–135. ISSN: 0098-3004. DOI: 10.1016/j.cageo.2013.11.003. URL: <https://www.sciencedirect.com/science/article/pii/S0098300413002902>.
- [3] K. Kuntsche und S. Richter. *Geotechnik. Erkunden – Untersuchen – Berechnen – Ausführen – Messen*. Springer Vieweg, 2021. ISBN: 9783658322908. DOI: 10.1007/978-3-658-32290-8.
- [4] T. Baumgärtel u. a. „Integrierte Planung auf Basis von 3D-Modellen“. In: *Digitale Baustelle- innovativer Planen, effizienter Ausführen: Werkzeuge und Methoden für das Bauen im 21. Jahrhundert*. Hrsg. von W. Günthner und A. Borrmann. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, S. 23–115. ISBN: 9783642164866. DOI: 10.1007/978-3-642-16486-6\_2.
- [5] N. Bartelme. *GIS Technologie. Geoinformationssysteme, Landinformationssysteme und ihre Grundlagen*. Springer Berlin, Heidelberg, 1989. ISBN: 9783662074947. DOI: 10.1007/978-3-662-07494-7.
- [6] Y. Chang und H. Park. „Development of a web-based Geographic Information System for the management of borehole and geological data“. In: *Computers & Geosciences* 30.8 (2004), S. 887–897. ISSN: 0098-3004. DOI: 10.1016/j.cageo.2004.07.006. URL: <https://www.sciencedirect.com/science/article/pii/S0098300404001335>.
- [7] *DIN 4020:2010-12, Geotechnische Untersuchungen für bautechnische Zwecke - Ergänzende Regelungen zu DIN EN 1997-2*. DIN 4020:2010-12. 2010.
- [8] *DIN EN 1997-1, Entwurf, Berechnung und Bemessung in der Geotechnik Teil 1: Allgemeine Regeln*. DIN EN 1997-1:2014-03. 2014.
- [9] *Geotechnische Erkundung und Untersuchung — Benennung, Beschreibung und Klassifizierung von Boden — Teil 1: Benennung und Beschreibung (ISO 14688-1:2017)*. DIN EN ISO 14688-1:2018. 2018.
- [10] *Geotechnische Erkundung und Untersuchung— Zeichnerische Darstellung der Ergebnisse von Bohrungen und sonstigen direkten Aufschlüssen*. DIN 4023:2006-02. 2023.
- [11] *Geotechnischer Bericht des DB-Projekts*. Techn. Ber. 2020. Exakter Name und Autor nach projektinternen Vorgaben nicht angegeben.
- [12] „Geologiedatengesetz vom 19. Juni 2020. (BGBl. I S. 1387)“. In: (2020).

- [13] G. Schirren. „Aufbereitung von Bohrungsdaten für das digitale Baugrundmodell“. In: *Kolloquium Digitalisierung in der Geotechnik* (2020). URL: [https://izw.baw.de/publikationen/kolloquien/0/08\\_Schirren\\_Aufbereitung%20von%20Bohrungsdaten...\\_.pdf](https://izw.baw.de/publikationen/kolloquien/0/08_Schirren_Aufbereitung%20von%20Bohrungsdaten..._.pdf).
- [14] T. Studer. *Relationale Datenbanken. Von den theoretischen Grundlagen zu Anwendungen mit PostgreSQL*. Springer Vieweg Berlin, 2019. ISBN: 9783662589762. DOI: 10.1007/978-3-662-58976-2.
- [15] T. Brinkhoff und W. Kresse. „Modeling of Geographic Information“. In: *Springer Handbook of Geographic Information*. Cham: Springer International Publishing, 2022. ISBN: 9783030531256. DOI: 10.1007/978-3-030-53125-6\_3.
- [16] J. Sbresny. *SEP 3 - Die Schnittstelle zur Bohrdatenbank Niedersachsens*. Landesamt für Bergbau, Energie und Geologie Niedersachsen. URL: [https://www.lbeg.niedersachsen.de/karten\\_daten\\_publicationen/bohrdatenbank](https://www.lbeg.niedersachsen.de/karten_daten_publicationen/bohrdatenbank). (Zugriff: 01.08.2024).
- [17] Arbeitskreis SSG/SEP3 des LBEG. *Dokumentation SEP3*. Landesamt für Bergbau, Energie und Geologie Niedersachsen. 2022. URL: <https://www.lbeg.niedersachsen.de/download/189634>.
- [18] Arbeitskreis SSG/SEP3 des LBEG. *Software-Downloads - SEP3*. Landesamt für Bergbau, Energie und Geologie Niedersachsen. URL: [https://www.lbeg.niedersachsen.de/karten\\_daten\\_publicationen/bohrdatenbank/sep\\_3/softwaredownloads/software-downloads-875.html](https://www.lbeg.niedersachsen.de/karten_daten_publicationen/bohrdatenbank/sep_3/softwaredownloads/software-downloads-875.html). (Zugriff am 21.08.2024).
- [19] S. Holmok. *Aufschluss Bohrdatenbank SEP 3*. 2023. URL: <https://support.geodin.com/hc/de/articles/360016072840-Aufschluss-Bohrdatenbank-SEP-3>.
- [20] L. Quin. *Extensible Markup Language*. World Wide Web Consortium (W3C). 2016. URL: <https://www.w3.org/XML/>. (Zugriff: 08.08.2024).
- [21] W. Kresse, D. Danko und Fadaie K. „Standardization“. In: *Springer Handbook of Geographic Information*. Springer Nature Switzerland, 2022. ISBN: 9783030531256. DOI: 10.1007/978-3-030-53125-6\_15.
- [22] *OGC® Geography Markup Language (GML) — Extended schemas and encoding rules*. Open Geospatial Consortium. 2012. URL: [https://portal.ogc.org/files/?artifact\\_id=46568](https://portal.ogc.org/files/?artifact_id=46568).
- [23] *Unified Modeling Language*. Object Management Group (OMG). 2017. URL: <https://www.omg.org/spec/UML/>.
- [24] J. Herring, C. Roswell und D. Danko. „Modeling of Geographic Information“. In: *Springer Handbook of Geographic Information*. Cham: Springer International Publishing, 2022. ISBN: 9783030531256. DOI: 10.1007/978-3-030-53125-6\_1.
- [25] *The Borehole Markup Language - BoreholeML. Das verständliche und anwendungsbezogene Austauschformat für Bohrdaten*. Staatliche geologische Dienste Deutschlands. URL: [https://www.infogeo.de/Infogeo/DE/Home/BoreholeML/boreholeml\\_node.html](https://www.infogeo.de/Infogeo/DE/Home/BoreholeML/boreholeml_node.html). (Zugriff am 09.08.2024).

- [26] *The Borehole Markup Language. A Comprehensive and Applicable Borehole Information Exchange Format - Introduction*. State Geological Surveys of Germany, Federal Institute for Geosciences und Natural Resources. Juni 2010. URL: [https://www.infogeo.de/boreholeml/3.0/zip/BoreholeML3\\_Submitted2INSPIRE.zip](https://www.infogeo.de/boreholeml/3.0/zip/BoreholeML3_Submitted2INSPIRE.zip).
- [27] *D2.8.II.4 Data Specification on Geology – Technical Guidelines*. Techn. Ber. Version 3.3.0. INSPIRE Maintenance und Implementation Group (MIG), 2024. URL: <https://github.com/INSPIRE-MIF/technical-guidelines/releases/tag/v2024.2>.
- [28] *Bohrpunktkarte Deutschland*. Staatliche geologische Dienste Deutschlands. URL: <https://boreholemap.bgr.de>. (Zugriff am 10.08.2024).
- [29] B. Arns-Krogmann. *Bohrpunktkarte Deutschland*. Bundesanstalt für Geowissenschaften und Rohstoffe (BGR). URL: [https://www.bgr.bund.de/DE/Themen/Geodatenmanagement/Bohrpunktkarte-Deutschland/bohrpunktkarte-deutschland\\_inhalt.html](https://www.bgr.bund.de/DE/Themen/Geodatenmanagement/Bohrpunktkarte-Deutschland/bohrpunktkarte-deutschland_inhalt.html). (Zugriff am 10.08.2024).
- [30] E. Boisvert, O. Raymond und M. Sen. *OGC Geoscience Markup Language 4.1 (GeoSciML)*. IUGS Commission for the Management und Application of Geoscience Information. 2017. URL: <http://docs.opengeospatial.org/is/16-008/16-008.html>. (Zugriff am 22.08.2024).
- [31] I Rinne, K. Schleidt und Van den Brink. *OGC Abstract Specification Topic 20 - Observations, measurements and samples*. Open Geospatial Consortium. 2023. URL: <http://www.opengis.net/doc/as/om/3.0>.
- [32] *IUGS Commission for the Management and Application of Geoscience Information (CGI)*. 2024. URL: <http://docs.opengeospatial.org/is/16-008/16-008.html>. (Zugriff am 22.08.2024).
- [33] A. Borrmann u. a. *Building Information Modeling. Technologische Grundlagen und industrielle Praxis*. Springer Vieweg Wiesbaden, 2021. ISBN: 9783658333614. DOI: 10.1007/978-3-658-33361-4.
- [34] M. Molzahn u. a. *Das Fachmodell Baugrund – Empfehlungen Nr. 1 des Arbeitskreises 2.14 der DGGT „Digitalisierung in der Geotechnik“*. Techn. Ber. 1. Deutsche Gesellschaft für Geotechnik (DGGT), 2021. DOI: 10.1002/gete.202000040.
- [35] *Official documentation for IFC 4.3.2.0 (IFC 4.3)*. buildingSMART. 2023. URL: [https://standards.buildingsmart.org/IFC/RELEASE/IFC4\\_3/index.html](https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/index.html). (Zugriff am 24.08.2024).
- [36] M. El Sibai u. a. „Towards efficient BIM use of geotechnical data from geotechnical investigations“. In: *ITcon 27* (2022). DOI: 10.36680/j.itcon.2022.019.
- [37] M. S. Khan, I. S. Kim und J. Seo. „A boundary and voxel-based 3D geological data management system leveraging BIM and GIS“. In: *International Journal of Applied Earth Observation and Geoinformation* 118 (2023), S. 103277. ISSN: 1569-8432. DOI: 10.1016/j.jag.2023.103277. URL: <https://www.sciencedirect.com/science/article/pii/S1569843223000997>.
- [38] F. Biljecki, K. Kumar und C. Nagel. „CityGML Application Domain Extension (ADE): overview of developments“. In: *Open Geospatial Data, Software and Standards* 3 (2018), S. 103277. ISSN: 2363-7501. DOI: 10.1186/s40965-018-0055-6.

- [39] J. Blankenbach und R. Becker. „Building Information Modeling“. In: *Springer Handbook of Geographic Information*. Springer Nature Switzerland, 2022. ISBN: 9783030531256. DOI: 10.1007/978-3-030-53125-6\_21.
- [40] I. Hijazi und A. Donaubaue. „Integration of Building and Urban Information Modeling Opportunities and Integration Approaches“. In: *Geoinformationssysteme 2017 - Beiträge zur 4. Münchner GI-Runde* (2017). URL: <https://mediatum.ub.tum.de/doc/1350517/1350517.pdf>.
- [41] C. A. Zehnder. *Informationssysteme und Datenbanken*. Vieweg + Teubner Verlag Wiesbaden, 1998. ISBN: 9783322899354. DOI: 10.1007/978-3-322-89935-4.
- [42] N. Bartelme. „Geographic Information Systems“. In: *Springer Handbook of Geographic Information*. Springer Nature Switzerland, 2022. ISBN: 9783030531256. DOI: 10.1007/978-3-030-53125-6\_6.
- [43] T. Brinkhoff. *Geodatenbanksysteme in Theorie und Praxis. Einführung in objektrelationale Geodatenbanken unter besonderer Berücksichtigung von Oracle Spatial*. Wichmann Berlin, 2013. ISBN: 9783879075720. URL: [http://content-select.com/index.php?id=bib\\_view%5C&ean=9783879075720](http://content-select.com/index.php?id=bib_view%5C&ean=9783879075720).
- [44] Clemens Portele. „Encoding of Geographic Information“. In: *Springer Handbook of Geographic Information*. Springer Nature Switzerland, 2022. ISBN: 9783030531256. DOI: 10.1007/978-3-030-53125-6\_4.
- [45] C. O'Brien und R. Lott. „Registration of Geospatial Information Elements“. In: *Springer Handbook of Geographic Information*. Springer Nature Switzerland, 2022. ISBN: 9783030531256. DOI: 10.1007/978-3-030-53125-6\_18.
- [46] *Über GeoDin*. Fugro Consult GmbH. URL: <https://geodin.com/de/pages/uber-uns>. (Zugriff am 26.08.2024).
- [47] M. Molzahn u. a. *Das Fachmodell Baugrund – Empfehlungen Nr. 1 des Arbeitskreises 2.14 der DGGT „Digitalisierung in der Geotechnik“*. Techn. Ber. 1. 2021. DOI: 10.1002/gete.202000040.
- [48] G. Gröger u. a. „OGC city geography markup language (CityGML) encoding standard“. In: (2012). URL: <http://www.opengis.net/spec/citygml/2.0>[https://portal.ogc.org/files/?artifact\\_id=47842](https://portal.ogc.org/files/?artifact_id=47842).
- [49] *What is PostgreSQL*. The PostgreSQL Global Development Group. URL: <https://www.postgresql.org/about/>. (Zugriff am 26.08.2024).
- [50] *PostGIS 3.4.3 Manual*. The PostGIS Development Group. URL: <https://postgis.net/docs/manual-3.4/>. (Zugriff am 26.08.2024).
- [51] *PostgreSQL Documentation*. The PostgreSQL Global Development Group. URL: <https://www.postgresql.org/docs/>. (Zugriff am 02.09.2024).
- [52] *QGIS Documentation*. QGIS Development Team. URL: <https://docs.qgis.org/3.34/de/docs/index.html>. (Zugriff am 01.09.2024).
- [53] A. Minuora. *Qgis2threejs Plugin Document*. URL: <https://minorua.github.io/Qgis2threejs/docs/ObjectTypes.html>. (Zugriff am 01.09.2024).

- [54] T. Berners-Lee und D. Connolly. *Hypertext Markup Language*. World Wide Web Consortium (W3C). 1995. URL: <https://www.w3.org/TR/WD-html2/>. (Zugriff am 01.09.2024).
- [55] *Bar Charts in JavaScript*. Plotly Graphing Libraries. URL: <https://plotly.com/javascript/bar-charts/>. (Zugriff am 02.09.2024).
- [56] *Openstreetmap*. Openstreetmap Contributors. URL: <https://www.openstreetmap.org>.
- [57] *Open Data Portal Bayern - Digitales Geländemodell*. Bayerische Vermessungsverwaltung. URL: <https://geodaten.bayern.de/opengeodata/OpenDataDetail.html?pn=dgm1>. (Download am 16.05.2024).
- [58] T. Scheel. *Visualisierung Von Bodenprofilen - vom Bohrkern zum 3D-Schichtenmodell*. 2018. URL: <https://forums.autodesk.com/t5/au-germany-vortrage/visualisierung-von-bodenprofilen-vom-bohrkern-zum-3d/td-p/8332281>.
- [59] J. Mc Carthy und P. Graniero. „A GIS-based borehole data management and 3D visualization system“. In: *Computers & Geosciences* 32 (2006). DOI: doi:10.1016/j.cageo.2004.07.006.