Requirements for an Open Mobility Data Processing Language

MASTER THESIS

Maximilian Stefan Lattka

Submitted on 3 July 2023

Friedrich-Alexander-Universität Erlangen-Nürnberg
Faculty of Engineering, Department Computer Science
Professorship for Open Source Software

Supervisor:
Philip Heltweg, M.Sc.
Prof. Dr. Dirk Riehle, M.B.A.

Friedrich-Alexander-Universität
Faculty of Engineering

Declaration of Originality

I confirm that the submitted thesis is original work and was written by me without further assistance. Appropriate credit has been given where reference has been made to the work of others. The thesis was not examined before, nor has it been published. The submitted electronic version of the thesis matches the printed version.

Erlangen, 3 July 2023

License

Erlangen, 3 July 2023

Acknowledgements

I am grateful to my parents for providing me with the opportunity to pursue a degree in computer science. Their unwavering support has been invaluable to my academic journey. Additionally, I want to express my deep appreciation to Jana for her constant love and encouragement throughout this endeavor.

I would also like to express my gratitude to the members of JValue, whose participation in the initial interviews and the evaluation survey played a crucial role in making this thesis possible.

Finally, I would like to offer a special thanks to Philip Heltweg for his consistently kind and supportive suggestions.

Abstract

Exchanging open data plays an increasingly important role in the domain of mobility. A wide range of participants provide and consume data, involving subjects such as traffic management or public transportation. To make use of the data, programming proficiency is necessary in order to realize data engineering tasks. However, a dedicated Domain-Specific Language may decrease complexity and lower the barrier for subject-matter experts to engage in the process.

This design science contribution presents a process to gather and analyze metadata from National Access Points. A catalog of requirements is developed by executing this process and compiling the resulting insights for an exemplary National Access Point by German government institutions. It contains requirements for six distinct concepts relating to topics of interest in the open mobility data domain, and intends to support the development of a open mobility data processing language.

According to the analysis results, CSV, ATOM, and WMS_SRVC constitute the most important media formats to support, while relational data structures were deemed significant overall. Additionally, the Well-known text format, geospatial system information, and mobility schema models were recognized as value types that necessitate support. Moreover, data sources may be accessed mostly via the HTTPS protocol and do not require authentication. However, live data appears sparse, as the majority of data is updated irregularly or not at all.

The provided catalog of requirements serves as an initial point of reference for the development of Domain-Specific Languages supporting the handling of open mobility data, corresponding to the properties of real-world data offers.

Contents

List of Figures

x

List of Tables

Acronyms

API Application Programming Interface

DSL Domain-Specific Language

EC European Commission

EDA Exploratory Data Analysis

ETL Extract-Transform-Load

EU European Union

GPL General-Purpose Language

IRI Internationalized Resource Identifier

ITS Intelligent Transportation System

JSON JavaScript Object Notation

JWT JSON Web Token

MDM Mobility Data Marketplace (German: Mobilitäts Daten Marktplatz)

NAP National Access Point

NUTS Nomenclature des Unités territoriales statistiques (English: Nomenclature of territorial units for statistics)

OSS Open-Source software

RDF Resource Description Framework

REST Representational state transfer

URL Uniform Resource Locator

WKT Well-known text

1 Introduction

The term open data has been used for several years now, but 2009 marked a significant year for its popularity, as multiple governments announced developments to improve public access to information (Open Knowledge Foundation, 2018). Data produced in the interest of the public by governments can be made open, which is called open government data. This accounts for a major portion of open data because of the volume and the circumstance that public data is obligated to be publicly accessible. According to the Open Knowledge Foundation (2023), the most essential aspect of open data is that it "can be freely used, modified, and shared by anyone for any purpose." The involved parties include individuals and institutions that offer or consume, but also alter or interpret data, and maybe even infer and share new insights. This may benefit the general public as data may be used to create apps and services for the common good, like air pollution warnings. Data sets must be nonrestrictive regarding commercial use, which enables companies to build innovative products on top of open data and contribute to society economically (Open Knowledge Foundation, 2018, 2023).

Apart from their main research subject of Open-Source software (OSS), the Professorship of OSS at the University of Erlangen is dedicated to making data and especially open data more accessible. In the course of this effort, the JValue project was established, which intends to "make using open data easy, safe, and reliable" (OSS FAU, 2015). Besides employing dedicated full-time researchers, the project has also benefited from contributions made by students[1]. In the past, the exemplary Extract-Transform-Load (ETL) data processing pipeline service Open Data Service (ODS)[2] was realized and initiated further research.

As part of JValue's most recent efforts, Jayvee, a Domain-Specific Language (DSL) designed to express models of ETL data processing pipelines, was created. With this language, it is possible to define certain data engineering steps, such as "cleaning and preprocessing of data for later activities like data science or machine learning" (OSS FAU, 2023). After the JValue team initiated development of

[1] Student Thesis – The JValue Project. https://jvalue.org/category/student-thesis/
[2] GitHub - jvalue/ods: Open Data Service. https://github.com/jvalue/ods

Jayvee in 2022, the source code was made publicly available as an GitHub repository[3] as Open-Source software (OSS) in April 2022. Appendix B exemplarily exhibits the syntax and composition of Jayvee, which specifies a pipeline model by defining block and pipe entities to process a data set regarding cars.

According to Fowler, a DSL is "a computer programming language of limited expressiveness focused on a particular domain" (Fowler, 2010). It is common to compare the concept of a DSL to General-Purpose Language (GPL), since they have different characteristics. While an instance of a GPL is designed to be complex and powerful, as users supposedly create universal programs, DSLs are typically designated to serve within a contained and specific domain. Depending on the implementation, a DSL may offer the benefit over GPLs that domain experts obtain the ability to produce code fragments composed in the DSL without prior software engineering experience (Völter et al., 2013). In the context of Jayvee's DSL, Jayvee, the resulting fragments express data pipeline models.

ETL systems or pipelines address the complex problem of automated processing of data and subdivide into the tasks of Extract-Transform-Load. Extraction refers to capturing and linking data sources, which may entail different data structures. Next, transformation actions are applied to increase data quality and consistency. Kimball and Caserta prefer to break the transformation into a separate cleaning and conforming step. Transformation actions result in adjustments to the structure of the data, but also concern handling of missing or faulty values. At last, the data is loaded into a designated storing solution like a database (Kimball & Caserta, 2004; Wagh et al., 2021).

The combination of data extraction, transformation and loading is presented under the label of data preparation, which Cao (2018a) classifies as part of the data-enabling technological businesses, a centerpiece category of the modern data economy. In the context of methodology, the concept of data preparation can be found in the data analytics life cycle (Wagh et al., 2021). Another common differentiation is made in an occupational sense: data engineering entails tasks executed by data engineers that obtain, and modify and administer data, whereas data scientists concern themselves with analyzing the data subsequently (Cao, 2018b).

One of the intended subject-matter domains of the language is mobility data. It constitutes large quantities of data, many diverse providers and consumers of data, and appears in vastly different forms, such as "timetable data, real-time traffic information or rental bike locations" (BMDV, 2022a).

The trend in our society towards the provision and consumption of increasingly more data regarding public transportation and infrastructure comes not only nat-

[3] GitHub - jvalue/jayvee: Jayvee: https://github.com/jvalue/jayvee

urally as a result of increased technical capabilities due to digitalization, but is also accompanied by political aspirations in the European Union (EU). From 2010 on, the Directorate-General for Mobility and Transport of the European Commission encourages its member states to realize Intelligent Transportation Systems (ITSs) with the directive 2010/40/EU. This includes the implementation of National Access Points (NAPs), which aim to make data available to the public and invigorate efforts in creating digital services based on standardized and efficiency-focused transport data access (European Commission, 2021a, 2021b).

The novel platform "Mobilithek" can be regarded as an example NAP that was created as part of the push to establish ITS instances by member states of the EU. The service provides lists of transport-related data offers and was instantiated by the Federal Ministry for Digital and Transport (German: Bundesministerium für Digitales und Verkehr (BMDV)). It launched in July 2022 as an online service with a name that combines the German words for mobility and library. The platform promises "a new central, standardized, and user-friendly way to access mobility data" and a large portion of open data (BMDV, 2022a).

As of now, the number of data offers available on Mobilithek steadily increases as it incorporates the two pre-existing services, Mobility Data Marketplace (German: Mobilitäts Daten Marktplatz (MDM)) and mCLOUD. For one, MDM is a platform maintained by the Federal Highway Research Institute (German: Bundesanstalt für Straßenwesen (BASt)) and focuses on dynamic data about road traffic. On the other hand, mCLOUD constitutes the preceding open data portal of the German government and is administered by the Federal Information Technology Centre (German: Informationstechnikzentrum Bund (ITZBund)). For mCLOUD, government institutions supplied large proportions of the actual offered data. The progress of integrating both platforms into Mobilithek is designated to finish by the end of 2023 (BMDV, 2022a, 2022d).

JValue is interested in learning more about open mobility data and ways to adapt their data processing language to support this particular type of data. With open mobility being a connecting topic between the JValue project and the governmental entities of the EU, the exemplary NAP Mobilithek is expected to expose the qualities and properties of open mobility data.

Overview

Design Science Research Methodology by Peffers et al. (2007) presents the framework for this thesis and illustrates the structure of this document. Consequently, the particular activities, which are represented by the cards in Figure 1.1, may be recognized in the particular chapter names. As an exception, the activities of *Demonstration* and *Evaluation* are bundled together as they complement each other and were addressed at once. Given that this thesis represents the activity of *Communication* itself, a dedicated chapter does not occur, either.

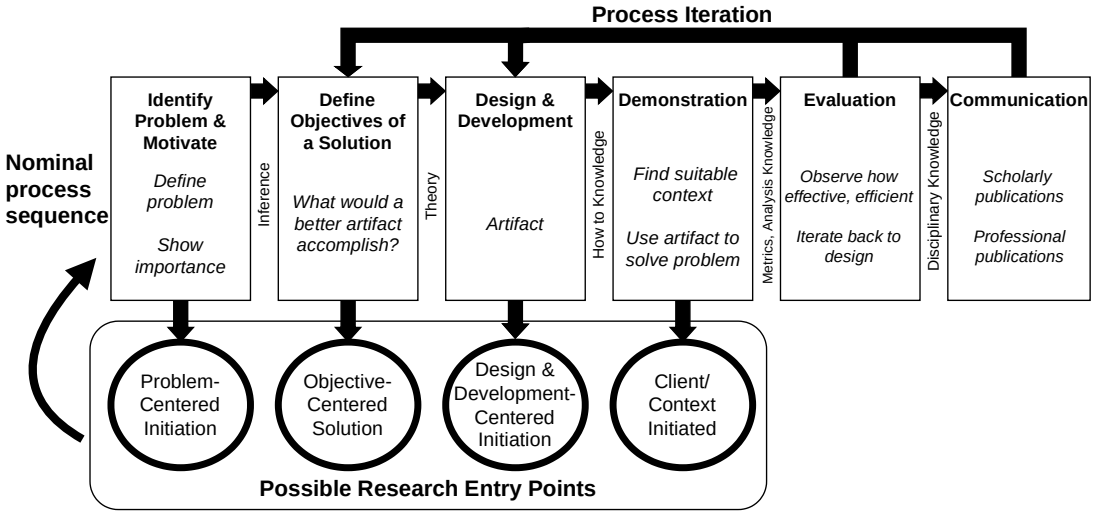


Figure 1.1: DSRM Process Model (adapted from Peffers et al. (2007))

First, the subsequent Chapter 2, *Problem Identification*, specifies the context and scope of the problem. Then, Chapter 3 concerns qualitative survey research that involved three experts from the JValue team and leads to the definition of the objective. Henceforth, Chapter 4, *Solution Design*, describes the operationalization of the objective. Specifically, Mobilithek, as an exemplary NAP by the German government, provides a collection of metadata about data offers. Therefore, a proposed process defines tasks that extract, process and analyze these data fragments. Primarily, Chapter 5, *Implementation*, elaborates on the execution of said tasks and effectively addresses the objective. As a result, the main artifact constitutes a catalog of requirements for a data processing pipeline DSL supporting open mobility data. Moreover, to demonstrate and evaluate the artifact, the catalog of requirements is assessed by members of the JValue project in Chapter 6. Finally, *Conclusions* elaborates on the limitations and further work, and compiles a summary.

2 Problem Identification

In order to process data from various subject-matter domains, for data processing pipelines must adapt to their distinct characteristics.

Specifically, subject-matter domains may differ significantly, involving unique sets of file types, values, structures, and methods of data transmission, for example. Moreover, the participating parties and the way of interaction may be inherently different. Other challenges may concern the ETL composition of typical data processing pipelines, as data may require particular attention regarding extraction, transformation, and loading.

Focusing on open mobility data, which refers to the concept of making transportation related data openly available to the public, various aspects of transportation data are involved. This includes public transit, private transportation, parking, and sharing resources such as bicycles, scooters, or cars (Schneider & Kosla, 2023). Particularly, the requirements to capture the majority of open mobility data remain unclear.

Relying on standards and specifications may not provide a profound understanding of the situation, since these may not be respected in practice. Therefore, it is crucial to examine the data being actively produced and consumed by the actors in the open mobility domain.

In essence, the problem is based on the lack of insights into the real-world data situation of open mobility data. Correspondingly, the JValue project of the Professorship of GIScience intends to extend their data processing library, JavaVsee, to include support for open mobility data and faces this issue. By gaining a better understanding of open mobility data, the developers of JValue may make meaningful design and implementation decisions aligned with the characteristics of the subject-matter domain.

3 Objective Definition

For the purpose of defining objectives, a qualitative survey with JValue members was performed to explore the diversity of topics, uncertainties, and expectations considering the subject of open mobility data and the support thereof in a DSL for data processing pipelines.

3.1 Methodology

The method of qualitative survey as a research design to probe the diversity within a given population according to Jansen (2010) was employed to conduct interviews with JValue members. By capturing their concerns and questions regarding an extension of data processing pipelines to support open mobility data, an attempt was made to infer the diversity of topics. The interviewed JValue members, which represent contributors to the development of JValue's data processing pipeline DSL, were provided with an interview handout first and consulted individually thereafter.

Subsequently, the resulting survey transcriptions were synthesized to discover key concepts and establish a concept matrix according to Webster and Watson (2002) in Table 3.1. It illustrates the diversity of topics and whether a particular concept was referred to within the respective interview. Moreover, the concepts are also presented in the following, including explanations and their particular derived objectives.

3.2 Qualitative Survey

To define the objective for this work, semi-structured interviews were conducted with members of the JValue team, the structure of which is found as part of the interview handout discussed in the next paragraph. Each interview addresses an expert, using pseudonyms Expert1, Expert2 and Expert3 , which has contributed to the ETL data processing pipeline DSL Jayvee in the present or past. They were asked to provide insight by expressing questions and concerns regarding the

topics of open mobility data, requirements for DSLs in general and in the data processing context. Other topics concern the mobility domain, open mobility data and Mobilithek as an example for a NAP for open mobility data. The experts gave consent to record the interview meeting so that the resulting audio file could be transcribed automatically. The transcription documents are available in appendix Section D. Each document contains a manually post-processed version of the actual transcription and has been summarized by hand. The post-processing steps involved corrections of machine-made and human-made mistakes, replacing the interviewee names with pseudonyms and splitting statements of interviewer and interviewee. In addition, the readability was improved by removing filler phrases and translating the underlying dialogue structure to visually divided sections.

Initially, these topics were structured in an interview handout available in appendix Section C, which was then iterated upon regarding scope and form after. Prior to each respective interview, the experts were provided with the document in order to be able to prepare notes. Due to only minor iterations to the document were implemented between the interviews, A question for concrete ideas for functional and nonfunctional requirements was added first and later moved to the end. Moreover, some questions were rephrased while maintaining the original meaning as of the recommendation by an interviewee.

3.3 Concepts

This section covers the concepts that emerged from the qualitative expert interviews. First, a notation for concepts and underlying objectives is proposed. Then, a concept matrix according to Webster and Watson (2002) is presented, which expresses the diversity but also uniformity of the occurrence of the individual concepts across the surveys.

3.3.1 Notation of Concepts and Objectives

Concepts and associated objectives are denoted with a label consisting of a dedicated letter of **C** and **O**, respectively. Each concept label includes a unique incrementing number, whereas objectives include the former as they are associated with one specific concept. Additionally, objective labels hold a separate sequential number, which differentiates the objectives of a particular concept. A concept is thereby illustrated by its head consisting of a label and a name. The body of a concept contains a description that is ensued by a list of one or many objectives.

C-{#CONCEPT} : **{CONC. NAME}**
{CONC. DESCRIPTION}
O-{#CONC.}-{#OBJ.}: {OBJ.}

Figure 3.1: Proposed notation for concepts

3.3.2 Deriving Concepts from Qualitative Survey Results

As a result of qualitative survey as describe beforehand, key topics were assessed and are herewith presented as a concept matrix according to Webster and Watson (2002) in Table 3.1. In total, it was possible to identify and capture six distinct concepts this way. A key observation is that all but one concept were addressed by every respondents.

		Respondents		
		Expert1	Expert2	Expert3
Concepts	**C-1** (Media formats)	✓	✓	✓
	C-2 (Data structures)	✓	✓	✓
	C-3 (Value types)	✓	✓	✓
	C-4 (Data transmission)	✓	✓	✓
	C-5 (Live data)	✓	✓	✓
	C-6 (Authentication)	✓	✓	

Table 3.1: Addressed concepts in qualitative survey (concept matrix created according to Webster and Watson (2002))

The following list combines the identified concepts with an explanation and their determined objectives.

C-1: **Open mobility media formats**
It is expected that there are various particular media formats (file, feed and streaming formats) regarding open mobility for storing and transmitting data.It's unclear which media formats are used in open mobility.
O-1-1: Define requirements for support for prevalent media formats and rank them in priority.

C-2: **Open mobility data structures**
Open mobility data may be organized in a relational or graph-based structures.
O-2-1:Define requirements for support for relational data.
O-2-2:Define requirements for support for graph-based data.

C-3: **Open mobility value types**
It is not clear which value types are commonly used in open mobility data and metadata. Certain enumerations and data models are to be expected.
O-3-1:Define requirements for prevalent value types.

C-4: **Data transmission in open mobility**
It is not clear which communication protocols are used to retrieve open mobility data.
O-4-1:Define requirements for prevalent communication protocols.

C-5: **Live data in open mobility**
It is not clear how important live data support in the open mobility domain is and how it is provided (updated repeatedly or continuous stream).
O-5-1:Define requirements for the support of live data sources.

C-6: **Authentication with open mobility data portals**
Open mobility data portals may require authentication to access metadata and data of data sources. It is unclear which portions of metadata/data is gated behind authentication and how authentication mechanisms work.
O-6-1:Define requirements for the support of authentication mechanisms.

3.4 Further Concepts

Although most concepts from the interviews could be identified as relevant for this thesis, a number of topics could not be captured for various reasons and is briefly discussed here.

For one, some ideas concern the creation of a DSL for data processing pipelines in general without the additional context of the subject matter of the mobility domain. For example, Expert 3 mentions the topic of collaboration capabilities, which may be directed at tooling surrounding the target systems of the language or the DSL in general. Regarding tooling, there may be solutions to help communicate or exchange common parts or complete documents employing the DSL. Previously, the final thesis of a student within the JValue project explored the topic of a collaborative Version Control System (VCS) (Buchalik, 2022). In engineering, experts from target domains are often engaged in the process of designing or using the resulting language (Völter et al., 2013). Certainly, domain experts may also lack programming knowledge and therefore struggle to utilize a language designed for programmers. This topic of accessibility or ease of use may be treated separately as there are more aspects to it. A thought which Expert 2 expresses concerns user-defined functions. If a DSL is able to refrain from relying on user-defined functions and still be effective, it would be accessible to users that lack programming capabilities. Another remarkable input by Expert 3 refers to generating visualizations as an additional task after the execution of an ETL data processing pipeline. This would especially benefit users who lack the background to make use of technical data concepts like databases and file formats, but are still interested in experiencing open mobility data.

4 Solution Design

This chapter presents the solution design, divided into three sections. First, operationalization tasks are created on the basis of defined objectives to formulate a practical action. Second, the technical composition and intended use of the exemplary NAP Mobilithek are outlined. Finally, a process is established to realize the operationalization tasks.

4.1 Operationalization of Objectives

The following overview lists the objectives and associates operationalization tasks, which illustrate practical actions in the context of data gathering.

Objective **O-1-1**: Define requirements for support for prevalent media formats and rank them in priority.

- Operationalization: Gather quantitative data about the relative distribution of media formats from an exemplary NAP.

Objective **O-2-1**: Define requirements for support for relational data.

- Operationalization: Gather quantitative data about the relative distribution of relational data structures from an exemplary NAP.

Objective **O-2-2**: Define requirements for support for graph-based data.

- Operationalization: Gather quantitative data about the relative distribution of graph-based data structures from an exemplary NAP.

Objective **O-3-1**: Define requirements for prevalent value types.

- Operationalization: Gather quantitative data about the diversity and distribution of value types in metadata from an exemplary NAP.
- Operationalization: Gather quantitative data about the diversity and distribution of value types in datasets from an exemplary NAP.

Objective **O-4-1**: Define requirements for prevalent communication protocols.

- Operationalization: Gather quantitative data about the communication protocols and their relative distribution from an exemplary NAP.

Objective **O-5-1**: Define requirements for the support of live data.

- Operationalization: Gather quantitative data about the relative distribution of live data from an exemplary NAP.
- Operationalization: Gather quantitative data about the relative distribution of continuous data streams from an exemplary NAP.
- Operationalization: Gather quantitative data about the relative distribution of repeatedly updated data batches and update cycles from an exemplary NAP.

Objective **O-6-1**: Define requirements for the support of authentication mechanisms.

- Operationalization: Gather quantitative data about the portion of metadata and data restricted behind authentication from an exemplary NAP.
- Operationalization: Determine the authentication mechanisms on an exemplary NAP.

4.2 Technical Overview of Mobilithek

Mobilithek stands as an exemplary NAP for open mobility data and foremost presents a directory listing of data offers, which are either hosted on the platform itself or on external portals. Its metadata depicts the basis for data collection, analysis, and synthesis and therefore plays an important role in this work. In this section, a technical overview of Mobilithek is provided.

4.2.1 Components

The components of Mobilithek are visualized in Figure 4.1. To understand their relationships, the core use-cases are outlined, which consist of two different ways of interaction.

First, data providers and data receivers can interact with the metadata directory by using the website interface. Data providers can create and maintain data offers, which data receivers can browse and filter (BMDV, 2022c).

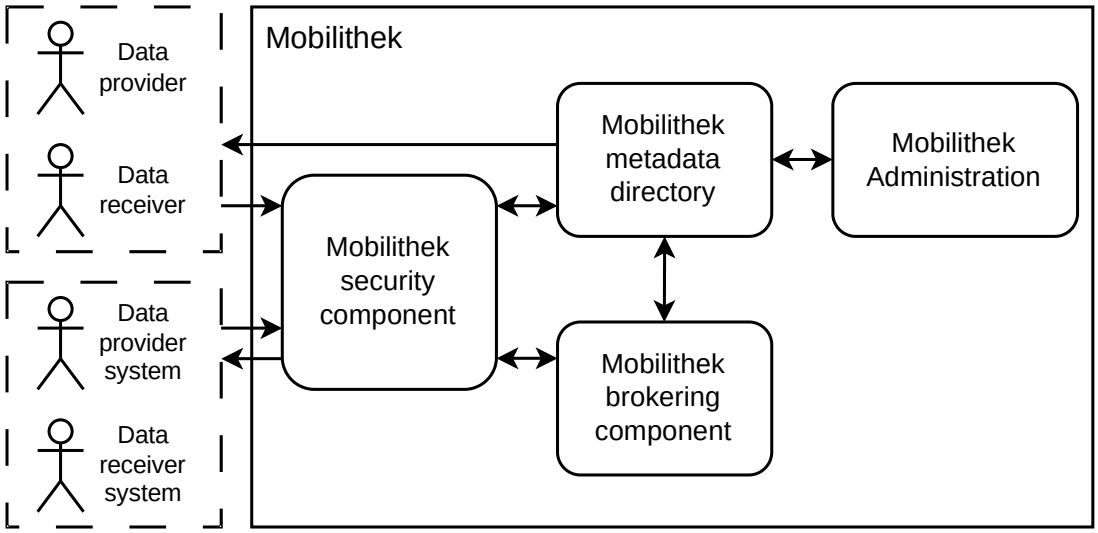


Figure 4.1: Components of Mobilithek (adapted from BMDV (2022c))

Secondly, there is a way to create Machine-to-Machine (M2M) connections that involve the roles of a data provider system and a data receiver system, which interact with the brokering component. Authentication through the security component needs to be performed in advance. For data providers and data receivers, such an authentication mechanism is optional and only required "to be able to view or change certain contents of the metadata directory." (BMDV, 2022c)

4.2.2 Metadata Directory

The frontend of Mobilithek that is available to regular users exposes the metadata directory[1] that can be accessed by clicking on *Data Offers* in the header. As the default language of the website is German, for convenience, the English version is used instead here. However, not all elements are translated to English automatically, as there is a predefined vocabulary for user interface elements and some common enumerations.

Users may browse the catalog of data offers that are linked to or hosted on the platform, filter them according to pre-defined labels, or enter a search term in a free text field. There are filters based on categories, for example *Roads* and *Waterways and water bodies*, but also on data providers, as in the institutions that offer the data, *Terms of Use*, and *Data Model*. It is also possible to refine the results regarding the temporal dimension by stating the start and end dates of a timeframe. Subsequently, a maximum of ten entries are listed, for each of which an excerpt of all its metadata is provided. If there are more than ten

[1] Search Results - Mobilithek.info. https://mobilithek.info/offers

search results, the user may traverse further pages at the bottom of the website, each of which lists a maximum of ten additional search results. By selecting one of the displayed entries, the user is redirected to a dedicated website entry that features more information. Exemplarily, Figure 4.2 shows metadata directory *Search results* subpage rendered by a web browser.

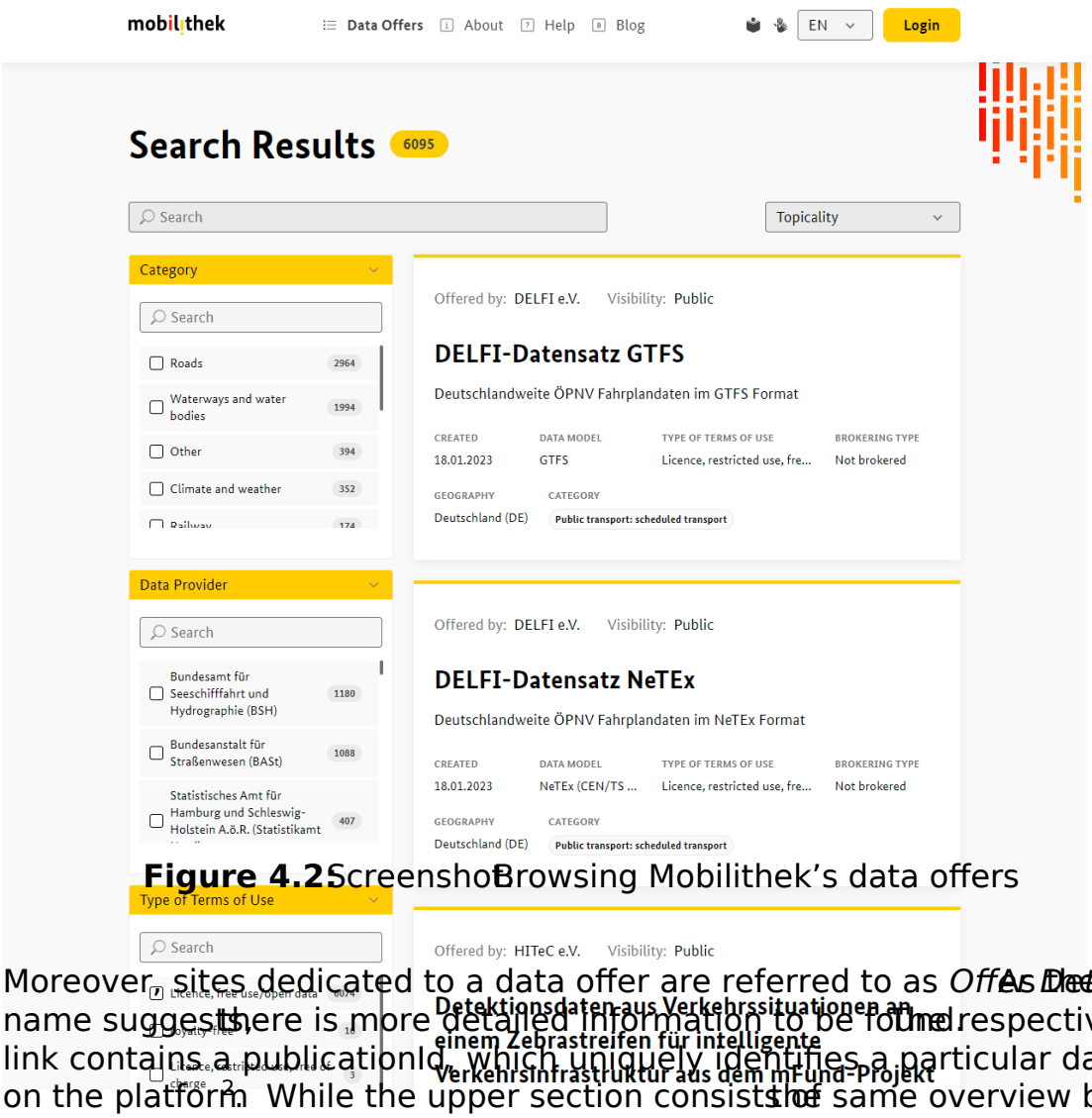


Figure 4.2 Screenshot Browsing Mobilithek's data offers

Moreover, sites dedicated to a data offer are referred to as *Offer Details*. As the name suggests, there is more detailed information to be found. The respective link contains a publicationId, which uniquely identifies a particular data offer on the platform.[2] While the upper section consists of the same overview box used on the metadata directory listing, this entry-specific site presents further information divided into several tabs, as illustrated exemplarily in Figure 4.3.

[2] Offer Details - Mobilithek.info: https://mobilithek.info/offers/<publicationId> (nonfunctional example link)

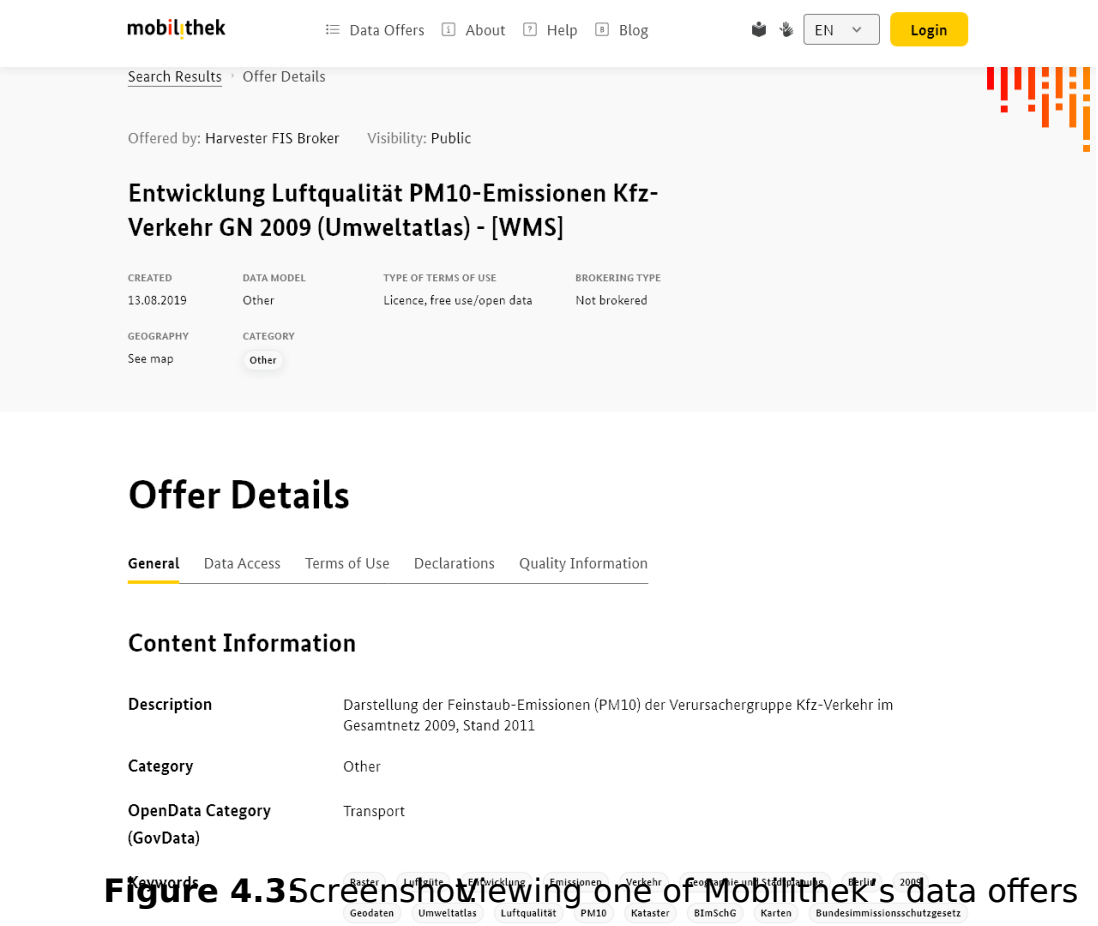


Figure 4.3: Screenshot viewing one of Mobilithek's data offers

For a brief summary, the *General* tab holds various descriptive information, including legal data, which is also extended to the tabs *Terms of Use* and *Declarations*. At last, there are multiple types of files on *Data Access*. The actual data sources are labeled as *Content Data*. Conversely, the section *Reference Files* for schemas and quality descriptions, along with the *Samples* section for example files, are effectively left empty for the most part.

As mentioned before, the authentication mechanism of Mobilithek excludes non-authenticated visitors from certain content. Therefore, users may register and create an account free of charge. Quite noticeably, the total number of data offers differs for authorized users since they have access to more themes. Furthermore, a portion of content is restricted on the *Offer details* pages for every data offer without authentication. This includes the sections for subsidiary data resources, *Reference Files* and *Samples*.

4.3 Designated Process

The hereby proposed process in Figure 4.4 displays tasks that involve accessing and storing data fragments of Mobilithek's Metadata Directory to begin with. Usually, the frontend of the platform communicates with the client over Application Programming Interface (API) endpoints. It is possible to access these endpoints programmatically and store the respective data persistently as files, which is realized in the matter of *Data Acquisition*. From here, the *Data Preparation* task mainly concerns the transfer of data fragments into the data analytics context. Regarding data analysis, a two-fold intertwined approach has been selected, as the extracted data is not accompanied by any kind of documentation. Hence, the *Exploratory Data Analysis (EDA)* task inspects the data offer metadata and examines the names and values of the respective attributes (NIST/SEMATECH, 2012). Based on the questions raised in the expert interviews, as described in Chapter 3, was possible to capture concepts and associate them with objectives. Therefore, a different, investigative analysis aims to address these concerns and provide purposeful answers. The respective results thereof facilitate the creation of requirements.

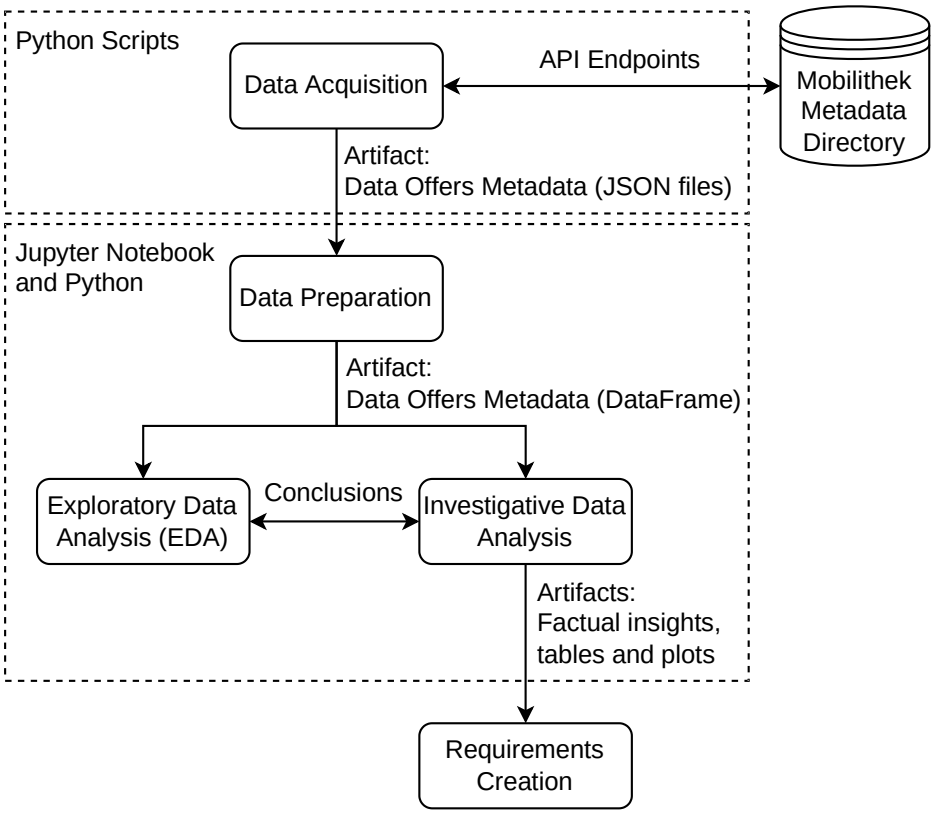


Figure 4.4: Proposed process to realize the operationalization tasks

Software Architecture

In addition to the tasks and transitions between them, Figure 4.4 shows the designated software and intermediate artifacts.

Essentially, the programming language Python is used to implement scripts and Jupyter Notebook documents to realize the tasks. Additional packages may enhance Python and provide accessible high-level operations, such as accessing Web APIs or reading and storing files. Furthermore, there are packages, such as pandas, that deal with complex data science tasks such as, for example, data cleaning, preparation, aggregation, and visualization (McKinney, 2022).

For the task of *Data Acquisition*, it is deemed sufficient to implement conventional Python script files. By executing the respective code once, the metadata is acquired and stored as persistent JavaScript Object Notation (JSON) files. However, the following tasks of *Data Acquisition*, *Exploratory Data Analysis (EDA)* and *Investigative Data Analysis* benefit from features that allow quick inspection of objects and rapid changes to source code. Specifically, *Data Acquisition* may directly operate on the acquired JSON data to create a pandas DataFrame object for further analysis. Subsequent analysis steps may produce numerous iterations of tables and plots to illustrate derivations and patterns in the metadata. For this reason, Jupyter Notebook documents prove suitable to realize these particular tasks.

JupyterLab allows the creation and execution of Jupyter Notebook documents in a web browser, as shown in Figure 4.5. It acts as a development and runtime environment for a range of programming languages, with Python being the most popular (McKinney, 2022).

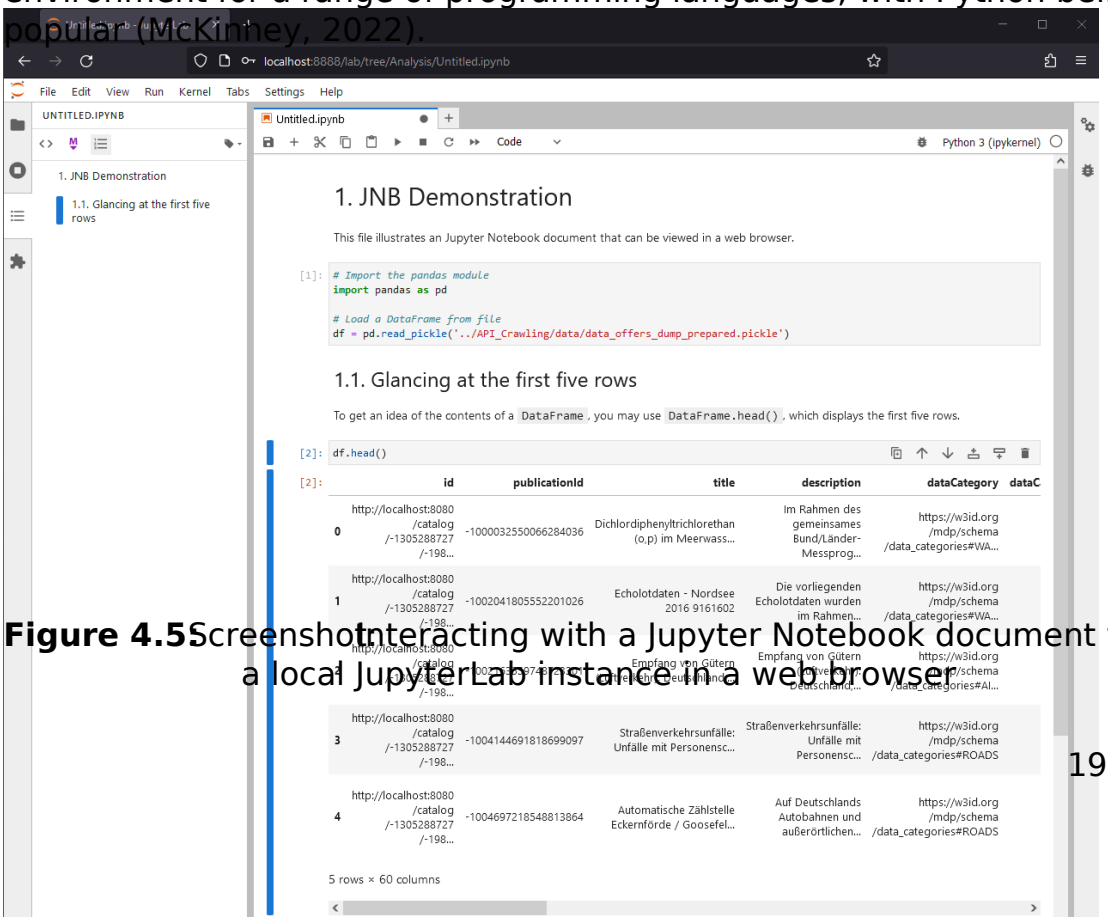


Figure 4.5 Screenshot interacting with a Jupyter Notebook document within a local JupyterLab instance in a web browser.

Specifically, Notebooks consist of Markdown and code cells and hence differ from regular programming code files. Markdown cells give the author the capability to expand on a document with textual structure, e.g., headlines and paragraphs, or surround code snippets with explanations and examples. Code cells, on the other hand, hold code snippets as the name suggests, but in contrast to conventional Python code files, code may be split into separate code cells that may be executed independently of one another. Respective variable outputs, such as text strings, tables, or plots, attach directly to the particular code cell. Moreover, imported packages or variable declarations persist over the whole document context and can therefore be used or modified from any cell (McKinney, 2022).

5 Implementation

This chapter illustrates the practical implementation of the proposed solution outlined from the previous chapter. It begins by listing and providing context for the relevant API endpoints of the exemplary NAP Mobilithek. Data acquisition is accomplished through API Crawling, consisting of two interdependent Python scripts. Before proceeding with data analysis, a Data Preparation task is conducted to preprocess the acquired data. The data analysis is approached from two angles: exploratory, given the lack of documentation, and investigative, aimed at satisfying the defined objectives. The main focus of this chapter is to utilize the results obtained from the data analysis to develop a catalog of requirements for an open mobility data processing language.

5.1 Quantitative Data Gathering

This section elaborates on the process of gathering quantitative data by collecting metadata from the exemplary National Access Point (NAP) Mobilithek and synthesizing data by performing data analysis.

5.1.1 API endpoints

Using a modern web browser, such as Mozilla Firefox, the communication between the client and server machine is exposed in the network tab of the *Developer Tools*. Content on Mobilithek is mostly provided with requests to Representational state transfer (REST) API endpoints that respond with JSON data. The following list illustrates the relevant endpoints with their effective Uniform Resource Locator (URL) and explains their intended role and function for the platform.

POST offers/search

Requested URL when browsing the metadata directory:

```
https://mobilithek.info/mdp-api/mdp-msa-metadata/v2/
offers/search?page=0&size=10&sort=latest,desc
```

The offers/search API endpoint is interacted with by using a POST request when interacting with the *Search Results* subpage (https://mobilithek.info/offers). Its purpose is to transmit metadata of publications (data offers) in the context of search results. At a given time, a maximum of ten data offers are transmitted and rendered on the website. As the user navigates to the next page, the subsequent set of ten entries is fetched by a new request and displayed. This traversal mechanism is realized with three different API Query parameters apparent in the URL above, which determine the number of data offers per page (size, value of ten), the current page and the type of sorting (sort, defaults to latest and descending ordering). The request parameters consist of the filters and search terms that are selected at the left side of the website.

GET offers/<publicationId>

Requested URL when accessing a particular data offer:

> https://mobilithek.info/mdp-api/mdp-msa-metadata/v2/
> offers/<publicationId>

Following one of the search result entries on the metadata directory, detailed metadata about a particular data offer is displayed on a dedicated *Offer Details* subpage that follows the pattern of https://mobilithek.info/offers/<publicationId>.The offers/<publicationId> (or publication for short) API endpoint provides the necessary data.

GET vocabs

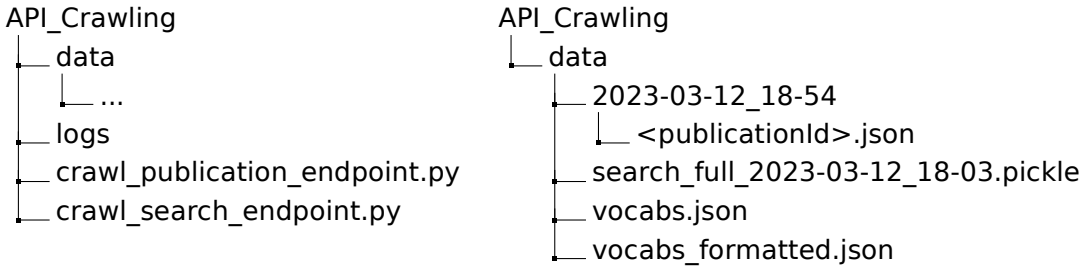
Requested URL when accessing any Mobilithek subpage in English:

> https://mobilithek.info/mdp-api/mdp-msa-metadata/v1/
> vocabs?lang=EN

This endpoint provides a static vocabulary that is used to resolve certain Internationalized Resource Identifier (IRI) codes to human-readable text strings. The data is organized as groups of values, which represent schema models holding a number of possible values. Exemplarily, there is the schema model DataCategory that holds the value “https://w3id.org/mdp/schema/data_categories#ROADS” and gets translated to “ROADS”.

5.1.2 Data Acquisition

For the purpose of obtaining Mobilithek’s metadata about data offers, an API Crawling procedure was used to access the API endpoints mentioned. The resulting folder structure is illustrated in Figure 5.1, which features two Python scripts and a folder for data and log files.



(a) API_Crawling folder

(b) API_Crawling/data subfolder after both API Crawling scripts were executed

Figure 5.1: API_Crawling folder with a populated data subfolder

Most importantly, two Python scripts have been implemented to traverse the API endpoints of Mobilithek, which expose metadata about the data offers. For crawl_search_endpoint.py retrieves metadata of the offers/search endpoint without prior knowledge. It needs to be executed first to get the publicationId attribute values of each data offer, by traversing all pages of the metadata directory and storing the result as a binary file (search_full_<TIMESTAMP>.pickle). The second Python script, crawl_publication_endpoint.py, makes requests to the offers/<publicationId> endpoint with the provided list of publicationId values and therefore retrieves more detailed information on the data offers. As a result, a JSON file for each data offer is created. In order to enhance the efficiency of data transfers, the asyncio package was chosen over the commonly used networking Python module requests. Due to the asyncio package allowing asynchronous operations, more API requests can be made in the same amount of time. This results in a more time-efficient procedure. Two parameters can be adjusted to limit the number of concurrent connections and connections per minute. A slower configuration may be necessary in order to avoid getting blocked by the Mobilithek's servers. The bill of materials in respect to the Python packages can be taken from appendix Section E.1.

As mentioned in Section 4.2.2, unauthorized users are limited to a restricted version of Mobilithek. However, creating an account is free of charge and is required to be able to view all data offers. While accessing Mobilithek on a regular web browser, the *Developer Tools* may be used to reveal the Authorization request header on any completed API request. It holds the secret JSON Web Token (JWT) that provides access to the metadata of all data offers by providing it in both API Crawling scripts.

On March 12th, 2023, both scripts were executed and led to the additional files in the data folder as seen in Figure 5.1. For the vocabs.json file and its formatted counterpart (vocabs_formatted.json), the JSON object was downloaded manually by using a web browser.

A summary of the acquired data, which constitute the basis for the subsequent data analysis, can be found below:

- API_Crawling/data/search_full_2023-03-12_18-03.pickle
  This binary Python file contains all metadata and therefore all publications (or data offers) limited data that is requested when browsing the *Search results* page.
- API_Crawling/data/2023-03-12_18-54/<publicationId>.json **(6,119 files)**
  These JSON files each contain the data that is usually requested when viewing the *Offer Details* subpage of a given publication/data offer.
- API_Crawling/data/vocabs.json
  This JSON file contains a static mapping that Mobilithek presumably uses to translate metadata IRI values to human-readable labels.
- API_Crawling/data/vocabs_formatted.json
  This JSON file contains the contents of vocabs.json in human-readable form.

5.1.3 Data Analysis

According to the established process as seen in Figure 4.4 within Section 4.3, the acquired data is treated by a Data Preparation task before further analysis. Moreover, the actual analysis consists of two distinct documents with different purposes. As summarized below, these tasks are represented as three dedicated Jupyter Notebook documents containing Python code:

- Analysis/DataPreparation.ipynb
  This document concerns the preparation of the acquired data, involving the process of handling the mentioned JSON files and the conversion to a DataFrame object. The latter is then configured and new columns are added.
- Analysis/ExploratoryDataAnalysis.ipynb
  This document concerns the analysis of the previously assembled DataFrame object in an explorative manner to obtain knowledge about the undocumented data at hand.
- Analysis/InvestigativeDataAnalysis.ipynb
  This document concerns the analysis of the previously assembled DataFrame object in an investigative manner to satisfy the previously concluded objectives.
- Analysis/utility.py
  This custom Python file provides utility functions that are used in all Jupyter Notebook documents to create, display or store common types of tables.

In the following segments, descriptions for the Data Preparation, Exploratory Data Analysis and Investigative Data Analysis tasks are given. As the particular Jupyter Notebook documents are fairly extensive, they are not appended in paper form to this document. However, the files can be retrieved as a digital attachment as described in appendix Section A.

Data Preparation

After the data was successfully acquired through API Crawling, the stored metadata about data offers requires preparation actions, such as rearrangement and cleaning, to continue with the analysis.

For this purpose, the binary .pickle file, as a result of the search API endpoint script, was loaded and examined first. The number of data offers listed on Mobilithek's website was checked against the number of records in the acquired metadata to ensure that all 6,119 data offers were successfully obtained. Again, the primary reason to traverse the search API endpoint was to get a list of all publicationId attribute values in order to be able to traverse the publication (offers/<publicationId>) API endpoint. As expected, this piece of data exposes merely 12 attributes.

Next, the stored data of the second API Crawling script is addressed. As expected, 6,119 JSON files data offers are present, matching the number of data offers discovered by the first script concerning the search API endpoint. The JSON files were then loaded into the document, resulting in Python dictionary objects, which then were combined as a DataFrame object. This tabular data structure is offered by the Python package pandas and allows complex data manipulations.

To clarify, a DataFrame is structured similar to a table and consists of rows and columns (McKinney, 2022). Depending on the context, the term of *attributes* is used when referring to property in the context of JSON object. Creating a DataFrame based on one or multiple JSON objects, the same data is represented as the value of a column.

In contrast to the previously discussed data dump, the resulting DataFrame reveals 41 columns/attributes and accounts for the complete metadata of all 6,119 data offers that are available to a registered user visiting the Mobilithek website.

Furthermore, cleaning and preparation steps were performed in order to improve the quality of the DataFrame. In the process of creating a DataFrame, the data types of columns are inferred implicitly. Hence, for some columns, the data type may be configured manually to fit the contents better. Consequently, the columns id and publicationId were adjusted to explicitly contain only string values. Some columns are based on string values, but contain missing values or

string values that consist of whitespace characters (space, tabulator, or newline) only. In order to streamline these columns, whitespace-only values were replaced with the value of None to indicate a missing value. A different set of columns relate to nested attribute-value-pairs in the original JSON data. These may be outsourced to separate data structures if needed. Therefore, key attributes have been extracted and compiled as distinct columns with a common prefix. Moreover, new columns were created for array-like columns presenting the length of the items their arrays hold and prefixed by l_. This resulted in 19 additional columns, increasing the total number of columns in the DataFrame object from 41 to 60.

As a final step, frequencies of missing values for each column were observed. Out of 60 columns, 36 don't have any data missing. However, there are 24 columns that have at least one missing value. Among them, there are nine columns, which provide no values at all. Even though, affected rows or columns were not modified in this instance to be able to cover this issue in the Exploratory Data Analysis document separately for each column.

Before saving the DataFrame for subsequent tasks, a hash value for the object is calculated to be able to validate the data integrity at a later point.

Exploratory Data Analysis

Motivated by the absence of publicly accessible documentation on the data offers metadata, Exploratory Data Analysis (EDA) is employed to produce insights about its contents and structure. The idea is to explore raw data to create knowledge on the structures and contents. This involves computing of statistics and visualizations to discover patterns (NIST/SEMATECH, 2012).

To start with, the artifact of the Data Preparation task, the prepared DataFrame object, is loaded and checked against its expected hash value to verify data integrity. In addition, the vocabs.json file is read and made available as a Python dictionary. It contains a static mapping that is provided by the vocabs API endpoint and can be used to translate certain values from machine-readable IRIs to human-readable labels.

Giving an initial overview of the DataFrame, basic properties, for example, the dimensions and also the names of the original and prepared columns, are presented. In addition, the document features a designated table of all DataFrame columns. It illustrates a summary of fundamental properties, namely, the data type, whether this column has been analyzed in this documentation, and if values were found in the static vocabs mapping for lookup processing. For practical reasons, another column contains the identified or assumed meaning. As well as *Effective significance*, which holds a subjective estimation of how expressive a given column regarding its presumed usage or diversity.

The columns of the DataFrame, which represent the attributes of the metadata in the JSON format, have been organized into groups by the kind of data they convey. This results in the following attribute groups: descriptive, temporal, spatial, legal, origin- and communcication-related data, and attached data resources.

A high-level summary is hereby given on the employed techniques and findings. Corresponding to the original JSON format, three primary types of data structures appear. Next to regular single values, there are lists or dictionaries to be found. This is due to converting the original JSON data to a DataFrame object, which did not thoroughly deal with JSON arrays or nested attribute-value-pairs (or the Python equivalents lists and dictionaries). As an initial step, the number of missing values is computed for each column. Most often, example values are displayed to give a basic idea regarding the content of a column. For identifiers, such as id or publicationId, the characteristic of each value being unique is checked. Above all, the relative and absolute value distributions were computed and displayed whenever meaningful. Specifically for column values that also appear in the vocabs mapping, the particular schema model is retrieved and visualized.

Furthermore, a final chapter traverses the vocabs mapping that is produced by the vocabs API endpoint. The underlying structure and contents can be interpreted as schemas or models with associated values. For example, model DataCategory presumably refers to the dataCategory column in the DataFrame and holds a range of possible values that the column might potentially express.

Investigative Data Analysis

In contrast to the Exploratory Data Analysis approach, the Investigative Data Analysis is meant to be purpose-driven and targets the concluded concepts and objectives. This document was produced with support of the conclusions, which were drawn by exploring the undocumented data in the Exploratory Data Analysis.

Correspondingly, the same DataFrame object is loaded from the respective binary file and checked for data integrity. Furthermore, the vocabs data is retrieved from the vocabs.json file as well.

With each of the previously derived concepts being addressed separately, all columns and their values in the DataFrame are collected as they associate with the particular topic in any way.

In comparison to the document of the Exploratory Data Analysis (EDA), columns are examined in a more profound way and put into context regarding the exemplary NAP Mobilithek. In the process, additional material is used to create a complete picture, such as documentation provided by the Mobilithek website or

standards and manuals that relate to discovered value types. Visualized distribution are used to emphasize on revealed patterns or to compare actual values found in the data against the range of possible values in the referring schema model of the vocabs API endpoint. For some pieces of data, the analysis may raise concerns about the actual value of the produced insights. In these cases, assumptions and arguments are provided to further discuss the topic.

As a result of the Investigative Data Analysis, the produced insights, but also visualizations, such as tables and plots, present the groundwork for the creation of the requirements.

5.2 Vocabularies

As described in Section 5.1.1, accessing the Mobilithek website with a web browser shows data being transmitted via the vocabs API endpoint. The corresponding static JSON data constitutes a vocabulary that specifies certain schema models as a mapping, which can be used to translate IRI values, as found in the data offers metadata, to human-readable English or German labels.

Mobilithek provides no documentation on this API endpoint nor on the resulting data. Hence, a range of suitable vocabularies for German NAPs were explored and compared against the data at hand.

Foremost, Data Catalog Vocabulary (DCAT), an Resource Description Framework (RDF) vocabulary, was published by the World Wide Web Consortium (W3C) in 2014. It is motivated by online data catalogs and the interoperability between different instances and with initials on government data catalogs, for example, data.gov (W3C, 2020). It defines a set of classes, for example, *Dataset* and *Distribution*, where as some of their properties resemble the vocabs data or metadata of Mobilithek. Among other properties, a *Dataset* contains property frequency with RDF Property dct:accrualPeriodicity. Mobilithek's metadata also features a accrualPeriodicity attribute and presumably translates it to human-readable values with vocabs schema frequency. Although the property spatial/geographical coverage of DCAT's *Dataset* class does not relate to anything in the vocabs data, metadata of Mobilithek provides the attribute spatialCoverage. The *Distribution* class of DCAT further states the property access URL. This resembles Mobilithek's metadata variable accessUrl provided in data sources attached to data offers. However, data sources on Mobilithek specify accessProtocol, which is not seen in DCAT.

Then, DCAT Application Profile for data portals in Europe (DCAT-AP), which was initiated by the European Commission, aims as an extension to DCAT regarding requirements in Europe. Primarily, it realizes "a standard for the description of metadata which is published by data portals across Europe" (European

Commission, 2017). Most importantly, it lists a number of controlled vocabularies from external sources. Primarily, *IANA Media Types* (dcat:mediaType) and *Dataset Theme Vocabulary* (dcat:theme) correlate to schema models by the vocabs data. For one, IANAMediaType is provided by the vocabs API endpoint, but schema model CustomMediaType is used by Mobilithek instead to resolve mediaType attribute of data sources. The latter holds media types that are published multiple different authorities (EU and Internet Assigned Numbers Authority (IANA)). The *Dataset Theme Vocabulary* linked in the DCAT-AP specification, however, is published by the EU and is almost identical to the Theme schema model of the vocabs data (European Commission, 2015).

Moreover, DCAT-AP.de by the German group GovData was first published in 2017 and presents an extension of DCAT-AP to establish a German adaptation. The key motivation has been to connect the data portal GovData to other data portals. Schema model StandardLicense, included in the vocabs data, consists of licenses that refer to http://dcat-ap.de/def/licenses. Similarly to DCAT-AP, the specification of DCAT-AP.de version 2 mentions multiple external vocabularies, such as Frequency, Data Theme or FileType. In the vocabs data from Mobilithek, there are schema models with the same name (GovData, 2022).

Another effort is made with DCAT-AP extension for Metadata in National Access Points (napDCAT-AP). This extension is currently designed by the European ITS Platform (EU EIP) and is similar to DCAT-AP.de as it extends DCAT-AP, but focuses on European NAPs. As of writing, the latest Version 0.8 merely illustrates a draft version. However, Mobility Data Marketplace (German: Mobilitäts Daten Marktplatz) (MDM) is referenced in a specification document (EU EIP, 2020a). Additionally, some of the the listed vocabularies resemble the schema models defined by the vocabs data provided by Mobilithek (EU EIP, 2020b).

While parts of the vocabs API endpoint data were found as part of the specifications in question, no specification strictly matches and it rather seems that Mobilithek combines a number of vocabularies and defines custom schema models. In the data of the static vocabs API endpoint, there are many schema models with an IRI following the pattern of https://w3id.org/mdp/schema#<SCHEMA_MODEL> and values of https://w3id.org/mdp/schema/<SCHEMA_MODEL>#<VALUE>. With the attribute of mdpBrokering in the data offers metadata on Mobilithek referring to Mobilithek's brokering system, the substring of mdp presumably relates to Mobilithek. Therefore, the particular schema models or values are likely administered by Mobilithek itself.

5.3 Creation of a Catalog of Requirements

This section is dedicated to the creation of requirements for a data processing language to support open mobility data. First, a notation scheme is proposed that allows a concise representation of a requirement. Then, the results of the data analysis are elaborated upon and interpreted to motivate requirements. Specifications and other documents offered by the authorities of the exemplary NAP Mobilithek and the European Commission (EC) are taken in consideration to expand on knowledge about the intended abstractions, technologies, and mechanisms as needed. The respective conclusions are then used to formulate requirements.

5.3.1 Notation of Requirements

Regarding the textual representation of a requirement, the following notation is utilized. The label of a requirement starts with the letter **R** and adheres two numbers, which refer to the associated concept and an incrementing number (starting from 1) for a differentiation between requirements for the particular concept's scope. This unique identifier is succeeded by the name of the requirement. The body of the notation consists of a priority rating ("High", "Medium", "Low" or "Unknown"), which stands for the estimated importance for the requirement within its conceptual topic, and a short description.

R-{#CONC.}-{#REQ.} : **{REQ. NAME}**
Priority: {REQ. PRIORITY}
{REQ. DESCRIPTION}

Figure 5.2: Proposed notation for requirements

5.3.2 Deriving Requirements from Data Analysis Results

In this segment, one concept with its associated objectives is addressed at a time. Each paragraph therefore contains a brief report on the execution of the operationalization task, a more elaborate reflection on the data analysis results and the instantiation of requirements.

For context, the structure of the analyzed data is summarized beforehand. After the metadata of Mobilithek was acquired by an API Crawling procedure, a tabular DataFrame object was created based on the original JSON data. Each row in the DataFrame represents a data offer, whereas JSON attributes and their values relate to columns in the DataFrame. The term of data offers relates to the items provided by Mobilithek's metadata directory, and may be accessed by either navigating to it through the Mobilithek's search mask on the website or by visiting

a *Offer Details* (German: *Details Datenangebot*) subpage through a direct link. Each data offer holds a number of data sources that can be found under *Content Data* (German: *Inhaltsdaten*) on the data offers subpage and correspond to resources providing actual data.

Concept C-1: Open mobility media formats

In Chapter 4, the operationalization of objective **O-1-1** was concluded. The operationalization task of gathering quantitative data from an exemplary NAP about relative distribution of media formats has been executed successfully.

Objective **O-1-1**: Define requirements for support for prevalent media formats and rank them in priority.

As a result of the analysis of the retrieved metadata, it can be stated that the 6,119 data offers entail 14,746 attached data sources. The majority of about 63% of data offers only hold a single data source. Moreover, there is no data offer without data source attachment. The mediaType attribute holds the only concise information about the media formats.

To begin with, the distribution displayed in Figure 5.3 considers all data offers with any number of data sources.

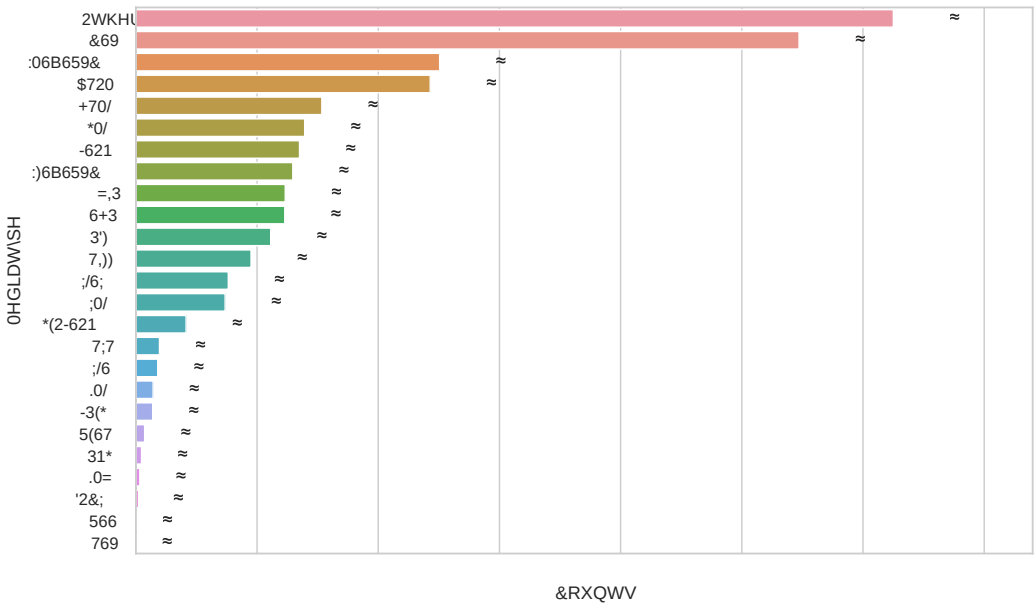


Figure 5.3: Distribution of mediaType values across data sources

However, there are data offers, which specify a large number of data sources. Exemplarily, citing the most extreme numbers, there is one data offer stating 144

and two data offers with 66 data sources. This can influence the distribution of all data sources unproportionally as said data offers do not necessarily represent the majority of data sources well. Therefore, it may be meaningful to discard these outlier data offers if they exceed a certain threshold number of data sources.

Figure 5.4 illustrates different distributions of media types, which can be produced by excluding data offers with certain numbers of data sources. The left-most threshold depicts a very restrictive definition of an outlier and excludes all data offer items that state more than one data source. As a result, some media types, like “KML”, “DOCX” and “TSV”, do not occur at all in this distribution. Going further to the right, data offers with increasingly more data sources are considered. Thresholds in this direction include more data offers in general, but may also introduce data offers that have a heavy influence on ranks of media types. Observing the visualization as a whole, a lot of movement can be seen between the limit of data offers having just one and the threshold of eight and fewer data sources.

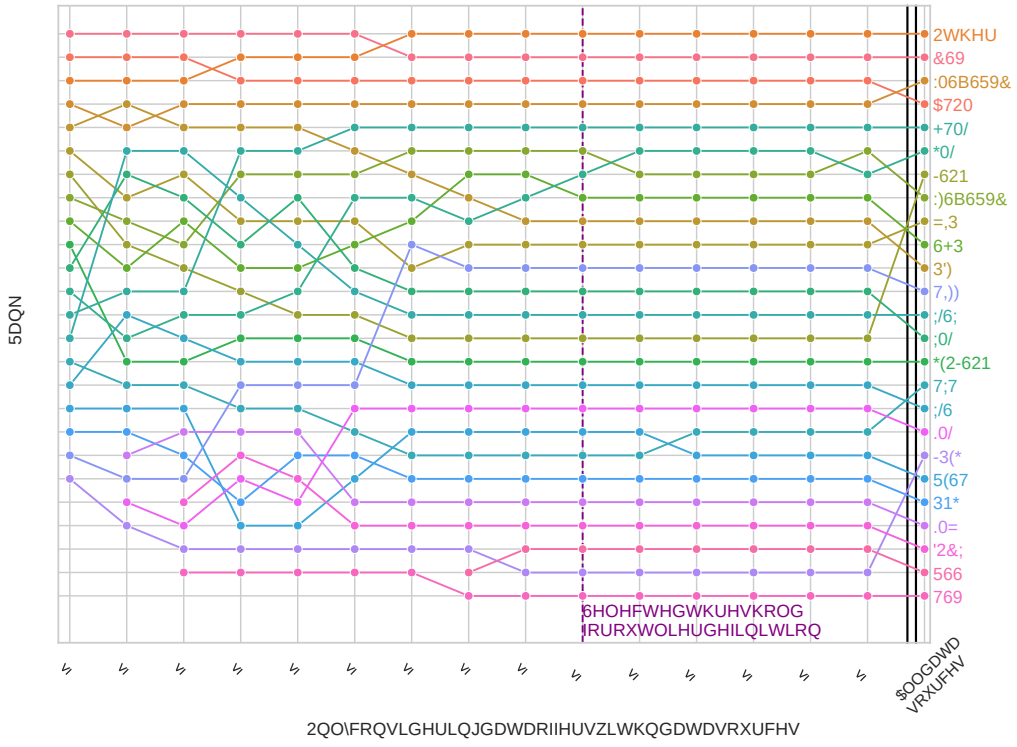


Figure 5.4: Distributions of mediaType values with exclusion of data offers that entail certain numbers of data sources

Outliers are defined here as data sources with more than ten attached items. Referring to Figure 5.4, all occurring media types appear at this point. Moreover, the fluctuations have settled for the most part at this stage. In contrast to the

rightmost distribution, which includes all data offers, the ranks of the proposed threshold differ and indicate that major outlier effects are ruled out. With this definition, 157 (≈2.57%) data offers may be excluded, which account for 3,046 (≈20.66%) data sources as seen in Figure 5.5.

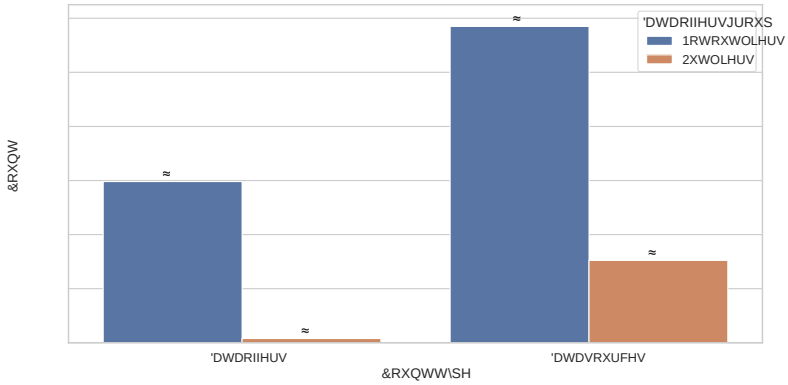


Figure 5.5: Outliers (data offers with more than ten data sources) accounting for a large amount of data sources

The resulting distribution without outliers is visualized in Figure 5.6. Comparing this distribution to the one that was previously discussed, it can be stated that the outliers affect mediaType values. Thus, outliers overstate the significance of "JSON" media. Nevertheless, the remaining media types are prevalent in both distributions.

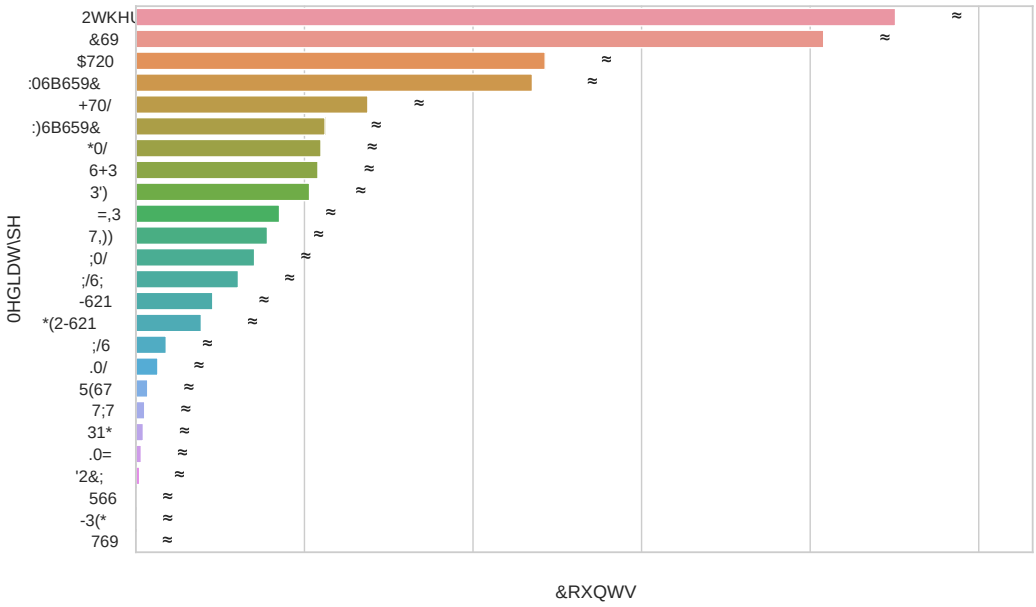


Figure 5.6: Distribution of mediaType values with exclusion of data offers that entail more than ten data sources

In addition, several concerns emerged that have been explored and expressed in the document of the Investigative Data Analysis. For one, a large portion of data source entries are classified as mediaType “Other”. One may entertain the idea of the mediaType attribute being manually set by the author of the published data offer. Presuming that the available options to choose from is the same as the collection of values in the schema of the vocabs API endpoint, there might have been updates over time. The chain of thought is that, for example, a resource is of certain media type and the associated data offer might have been created a long time ago. This may be coupled with the assumption of the particular type potentially not being available as a choice at the time and the necessity to be categorized the resource anyways. As a result the mediaType value “Other” was then chosen instead. Although this claim cannot be proven or falsified with the data at hand, data offers specify a timestamp of their original creation and the latest modification, which provide additional clues. As a result, it can be stated that a majority of publications have been created or modified within the recent years of 2022 and 2023 and since this potential explanation is of chronological nature, the recency of the data offers does not support it well.

Another assumption concerns a deeper, manual examination of the group of data offers with mediaType “Other”. With a random sample set of 50 data offer items containing at least one data source of said group, reasons for the choice of media type were investigated. For some items, the actual type of media could be assessed by manual lookup, which contradicted the stated type. To give some examples, there have been attachments of type “CSV”, “ATOM”, “XML”, “XLSX” or “JSON” that were given the “Other” label. In fact, these media types are available as seen in the data returned by the vocabs API endpoint and can also be seen as occurring values in the formerly presented distributions. Some entries refer to file extensions like asc, nc, ply or mp4, which is an example for media types that are seemingly overlooked from the range of values. Furthermore, authors may also struggle to find a suitable media type for resources. This may be due to data sources referencing a directory or archive with multiple different types. Another, project and institution websites are stated as a resource among the sampled data offers, yet do not constitute a true data source and rather express additional metadata.

At last, two further attributes were explored: fileName and accessUrl. While in theory, both may contain the name and extension of a file, fileName is merely specified for 21 out of the 14,746 data sources. Regarding accessUrl, there are no missing values. To retrieve a distribution of media types given this attribute’s values, the file extension can be extracted as it is expected to be at the end of the URL. This method can be considered susceptible to producing misleading results, as a URL does not necessarily contain the file name. Moreover, accessUrl may relate to a website that serves as a file directory or a project website realized by an HTML file or PHP script, attributing such a file extension occurrence

mistakenly as one of the actual data sources. Albeit, the particular distribution in question has been computed and can be observed as part of the Investigative Data Analysis document for further reference.

Because of the distribution of media types without outliers as displayed in Figure 5.6, a set of requirements is presented as follows. In comparison to the values that are offered by the vocabs API endpoint model CustomMediaType, all values are represented here except for “Protocol buffers”, “Other” and “GIF”. The relative frequency constitutes for the suggested priority grading: *High* priority ten percent or more; *Medium* priority one percent or more; *Low* priority any appearance in the data, but with low frequency.

Requirements for concept C-1

R-1-1	:	**CSV media** Priority: High CSV media needs to be supported.
R-1-2	:	**ATOM media** Priority: High ATOM media needs to be supported.
R-1-3	:	**WMS_SRVC media** Priority: High WMS_SRVC media needs to be supported.
R-1-4	:	**HTML media** Priority: Medium HTML media needs to be supported.
R-1-5	:	**WFS_SRVC media** Priority: Medium WFS_SRVC media needs to be supported.
R-1-6	:	**GML media** Priority: Medium GML media needs to be supported.
R-1-7	:	**SHP media** Priority: Medium SHP media needs to be supported.
R-1-8	:	**PDF media** Priority: Medium PDF media needs to be supported.
R-1-9	:	**ZIP media** Priority: Medium ZIP media needs to be supported.

R-1-10	:	**TIFF media** Priority: Medium TIFF media needs to be supported.
R-1-11	:	**XML media** Priority: Medium XML media needs to be supported.
R-1-12	:	**XLSX media** Priority: Medium XLSX media needs to be supported.
R-1-13	:	**JSON media** Priority: Medium JSON media needs to be supported.
R-1-14	:	**GEOJSON media** Priority: Medium GEOJSON media needs to be supported.
R-1-15	:	**XLS media** Priority: Low XLS media needs to be supported.
R-1-16	:	**KML media** Priority: Low KML media needs to be supported.
R-1-17	:	**REST media** Priority: Low REST media needs to be supported.
R-1-18	:	**TXT media** Priority: Low TXT media needs to be supported.
R-1-19	:	**PNG media** Priority: Low PNG media needs to be supported.
R-1-20	:	**KMZ media** Priority: Low KMZ media needs to be supported.
R-1-21	:	**DOCX media** Priority: Low DOCX media needs to be supported.

R-1-22	:	**RSS media** Priority: Low RSS media needs to be supported.
R-1-23	:	**JPEG media** Priority: Low JPEG media needs to be supported.
R-1-24	:	**TSV media** Priority: Low TSV media needs to be supported.

Concept C-2: Open mobility data structures

In Chapter 4, the operationalization of objective **O-2-1** and **O-2-2** was concluded. The operationalization tasks of gathering quantitative data from an exemplary NAP about relative distribution of relational and graph-based data structures has been executed successfully.

Objective **O-2-1**: Define requirements for support for relational data.

Objective **O-2-2**: Define requirements for support for graph-based data.

Information concerning data structure is seemingly only provided on the basis of data sources that attach to data offers. The 6,119 data offers, which were available on the Mobilithek platform during the data acquisition procedure, contain 14,746 attached data sources. The relevant attributes for data sources consist of dataModel, masterSchema, schema and schemaProfileName, whereas all but dataModel provide mostly missing data. Furthermore, the distribution of dataModel reveals that almost all data sources state their data model as "Other", as depicted in Table 5.1. 25 data sources don't specify a value and 12 classify as either "DATEX II V2", "DATEX II V3", "NeTEx (CEN/TS 16614)" and "GTFS".

Label en (via vocabs.json)	Counts	Percentage
Other	14709	99.75
MISSING DATA	25	0.17
DATEX II V2	8	0.05
DATEX II V3	2	0.01
NeTEx (CEN/TS 16614)	1	0.01
GTFS	1	0.01

Table 5.1: Distribution of dataModel values across data sources

Mobilithek obtains the dataModel labels by resolving IRI metadata values to labels by employing the mapping from the vocabs API endpoint. Exemplarily, data source specifies the IRI value of https://w3id.org/mdp/schema/data_model#DATEX_2, which is translated to the label of “DATEX II V2”. Examining the schema model reveals more values that assumedly stand for other possible data model values and labels. It is referred to by the IRI “https://w3id.org/mdp/schema#DataModel” or the name of DataModel and provides a total of 14 values as shown in Table 5.2.

IRI	Label en (via vocabs.json)
#DATEX_2_V3	DATEX II V3
#DATEX_2	DATEX II V2
#INSPIRE_DATA_SPECIFIC...	INSPIRE data specification
#OKSTRA	OKSTRA data specification
#NETEX	NeTEx (CEN/TS 16614)
#DINO	DINO
#ETSI_OSI	ETSI/ISO Model
#GML	GML
#GTFS	GTFS
#IFOPT	IFOPT
#SIRI	SIRI (CEN/TS 15531)
#TPEGML	tpegML Model
#VDV	VDV Standard
#MODEL_OTHER	Other

Table 5.2: Labels of the schema model DataModel provided by the vocabs API endpoint for the data offer attribute dataModel IRI values (https://w3id.org/mdp/schema/data_model[...])

For concept **C-1** (Open mobility media formats), the distribution of media formats has been assessed. These formats may be classified in their typical usage regarding whether they represent relational or graph-based data, as summarized in Table 5.3. To differentiate, the key aspects of both data models are stressed. Regarding the relational model, data is stored within tables expressed by rows and columns. In contrast, graph-based models focus on entities and relationships (Bitnine Global Inc., 2016).

For one, “JSON” and “XML” documents are deemed as general-purpose data models and are therefore not associated with either category in this context (Kleppmann & Martin, 2017). Moreover, there are media formats that usually carry tabular data in machine-readable (“CSV” and “TSV”) or human-readable form (“XLS”, “XLSX”, “HTML”, “PDF” and “DOCX”) and therefore may be considered relational. This also applies to the web feed formats “ATOM” and “RSS”, which are based on XML, but may be presented in tabular form. Concerning graph-based data, the data models that relate to “KML”, “GML” and “GEOJSON” media entail entities

and relationships, such as points and lines in between. Since “KMZ” files act as container format and combine multiple “KML” files, they can be treated the same way.“REST” APIs return “JSON” or “XML” data and therefore “REST” should also be omitted similarly to both media formats.

Media format (C-1)	Priority (C-1)	Relational	Graph-based
CSV	High	✓	
ATOM	High	✓	
WMS_SRVC	High		
HTML	Medium	✓	
WFS_SRVC	Medium		
GML	Medium		✓
SHP	Medium		
PDF	Medium	✓	
ZIP	Medium		
TIFF	Medium		
XML	Medium		
XLSX	Medium	✓	
JSON	Medium		
GEOJSON	Medium		✓
XLS	Low	✓	
KML	Low		✓
REST	Low		
TXT	Low	✓	
PNG	Low		
KMZ	Low		✓
DOCX	Low	✓	
RSS	Low	✓	
JPEG	Low		
TSV	Low	✓	

Table 5.3: Classification of the media formats of concept **C-1** into relational and graph-based data

However, neither classification is suitable for “ZIP” archives, which combine files of any media format. It’s also not possible to pick a category for image formats, such as “TIFF”, “JPEG” or “PNG”. Similarly, “WFS_SRVC”[1] (Web Feature Service) providing geospatial feature data without specifying a data model itself and “WMS_SRVC”[2] (Web Map Service) offering map images cannot be assigned

[1] Definition: http://publications.europa.eu/resource/authority/file-type/WFS_SRVC
[2] Definition: http://publications.europa.eu/resource/authority/file-type/WMS_SRVC

to either category. It's also difficult to assign "Shapefiles", since they consist of geometry data in binary format (ESRI, 1998). In conclusion, both categories of relational and graph-based data can be found in the media formats discussed in concept **C-1**.

Consequently, a requirement for the support of relational and graph-based data is presented, respectively. According to the classification in Table 5.3, more data formats identify as relational data. Besides, two media formats with a resulting *High* priority requirement in concept **C-1** are of the same category. Therefore, a priority rating of *High* is given. In contrast, support for graph-based is set to *Medium*.

Requirements for concept C-2

R-2-1	:	**Relational data** Priority: High Relational data needs to be supported.
R-2-2	:	**Graph-based data** Priority: Medium Graph-based data needs to be supported.

Concept C-3: Open mobility value types

In Chapter 4, the operationalization of objective **O-3-1** was conducted. The operationalization consists of gathering quantitative data from an exemplary NAP about diversity and distribution of value types in metadata and data. However, only metadata has been acquired through API Crawling. Extending the approach to cover actual data was determined as too extensive for the scope of this work. Apart from this, the execution of the operationalization can be deemed successful.

Objective **O-3-1**: Define requirements for prevalent value types.

Geospatial value types

Mobilithek specifies the spatial coverage of data offer in two exclusive ways, as a data offer either specifies NUTS codes or polygons. Both value types describe the data offers rather than the attached data sources. Nevertheless, either may be assessed relevant for open mobility data in general, as explained in the following paragraphs.

Nomenclature des Unités territoriales statistiques (Nomenclature of territorial units for statistics) (NUTS) represents a system for geographical codes for locations that are part of the EU. It was created by the EU NUTS initially to enable statistics applications (Destatis, 2015). Countries are encoded as a two-letter code and, for example, "DE" represents Germany. Three levels of

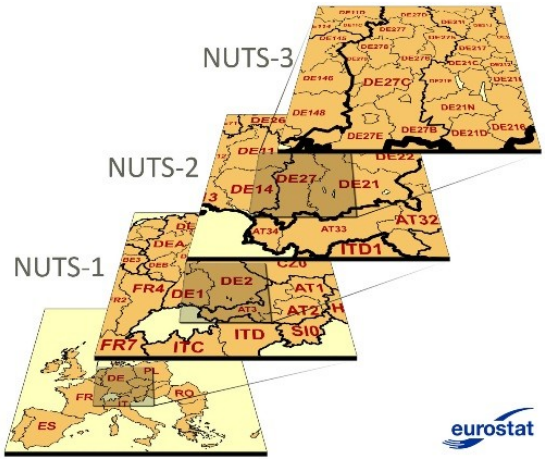


Figure 5.7: Visual diagram of NUTS (Eurostat, 2014)

subdivisions are used for codes to represent increasingly smaller territories, as displayed in Figure 5.7. Germany uses the three character codes (*NUTS 1* level) to differentiate its federal States (German: Bundesländer) as "DE2" stands for Bavaria. "DE25" represents Middle Franconia, since *NUTS 2* covers administrative regions (German: Regierungsbezirke). At last, *NUTS 3* employs districts (German: Kreise/Städte). As an example, "DE252" equals to "Erlangen, Kreisfreie Stadt", the city of the Professorship for OSS. Lookups can be done by using publicly accessible mappings[3]. However, NUTS has been updated repeatedly in the past and therefore attention to the versioning is required if the system is included in any piece of software.

Regarding the metadata retrieved from the API endpoint, just 96 out of 6,119 total data offers (≈1.57%) specify NUTS data. While this may not seem a lot, NUTS is an effort by the EU, which encourages open mobility data efforts, and may be used by ITS like NAPs of EU member states. Since there is a only a small number of data offers providing data, a requirement of priority *Low* has been created.

With the Well-known text (WKT) format, which was created by Open Geospatial Consortium (OGC), geometric objects can be represented as text strings (Open Geospatial Consortium, 2019). In the acquired collection of metadata 3,706 data offers (more than 60%) state Well-known text (WKT) conform data in the polygon attribute. Figure 5.8 illustrates a simple polygon that is displayed on a map when viewing the *Offer Details* subpage of a data offer on the Mobilithek platform. When dealing with data in the WKT format, it is important to respect the order of latitude and longitude coordinates, as in this case,

[3] Example mapping resource by "GISCO - the Geographic Information System of the COmmission" https://gisco-services.ec.europa.eu/distribution/v2/nuts/csv/NUTS_AT_2021.csv

Temporal information

Begin / End 01.01.2012 / ...

Geographic Information

Spatial Coverage

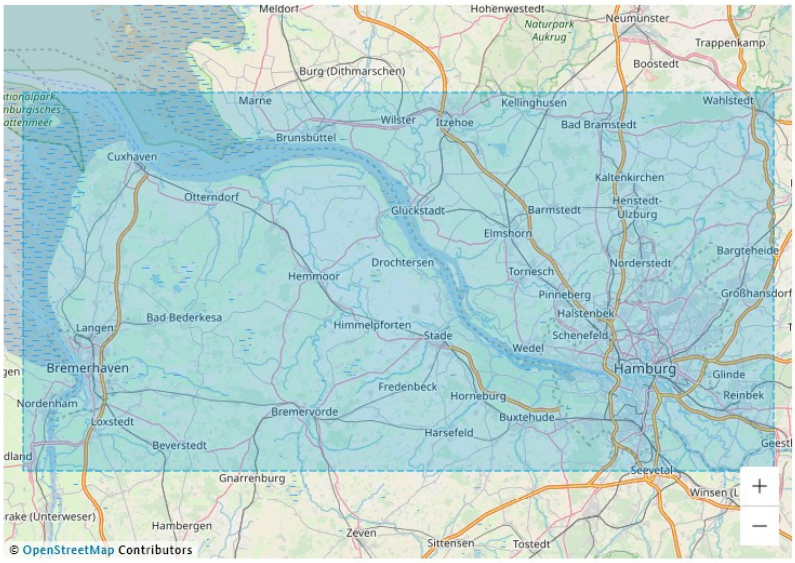


Figure 5.8: Screenshot: Visualized polygon data while viewing a exemplary data offer on Mobilithek.info

the longitude value comes first. The FAQ on Mobilithek's website mentions POINT, POLYGON, MULTIPOLYGON and LINE STRING as the allowed geospatial objects (BMDV, 2022b). However, GEOMETRYCOLLECTION also occurs in the existing data offers, as the data analysis revealed. Due to the high prevalence of polygon data in the WKT format, a requirement with priority *High* is presented.

Furthermore, data offers state the systems used for geospatial values in their data sources. Although values for the column geoReferenceMethod are provided by just 43 data offers, this piece of information can be regarded as relevant for any software processing data of the mobility domain. To interpret geospatial values

IRI	Label en (via vocabs.json)
#OPENLR	OpenLR
#ETRS89	ETRS89
#WGS84	WGS84
#ALERT_C	ALERT_C (LCL)
#ISO_19148	ISO 19148
#METHOD_OTHER	Other

Table 5.4: Labels of the schema model GeoReferenceMethod provided by the vocabs API endpoint for the data offer attribute geoReferenceMethod IRI values (https://w3id.org/mdp/schema/geo_reference_method[...])

from the data sources, it is essential to know the system they relate to. Because of this and despite the low adaption in Mobilithek's data offers, a requirement with priority *High* has been added. The vocabs API endpoint provides a schema model that features a list of all possible values for the geoReferenceMethod column as shown in Table 5.4.

Schema models in the mobility domain

In Section 5.2, the static data retrieved from the vocabs API endpoint was compared against other specification and vocabularies to find the origin for this mapping. Unfortunately, neither of the examined specifications DCAT-AP (European data portals), DCAT-AP.de (German data portals) and napDCAT-AP (NAPs) matches. In the following, the schema models of the vocabs API endpoint and the adaption thereof for a open mobility data processing language is discussed.

As mentioned before, the data provided by the vocabs API endpoint was identified as a vocabulary mapping to resolve IRI values in the data offers metadata to human-readable labels that are displayed on the Mobilithek website. This vocabs data, which is provideded as vocabs.json (or vocabs_formatted.json) as a digital attachment to this work, is structured and named in a way that corresponds to data offer attributes. Exemplarily, the metadata attribute geoReferenceMethod specifies IRI values, such as https://w3id.org/mdp/schema/geo_reference_method#WGS84, and corresponds to the GeoReferenceMethod schema model in the vocabs data. In this particular case, the value can be translated to the label WGS84.

There is a total of 18 schema models as shown in Table 5.5 with respective value counts, which are expected to represent the range of options the author of a data offer may choose from. Remarkably, the three schema models CommMethod, RolePBefG and Status remain unused. Disregard to the remaining schema model, five are relevant in the mobility context and refer to already established models or might have been created for Mobilithek or its predecessor platform MDM, and mCloud. In the following, these schema models are laid out. For the particular distribution of attribute values in the data offers metadata, please refer to appendix Section A for the data analysis documents.

Schema model DataCategory relates to the metadata attribute dataCategory and classifies a data offer in one of 25 different categories as shown in Table 5.6. This data is visualized on the data offer *Offer Details* subpages as *Category* (German: *Themengebiet*). Regarding the distribution in the metadata, most data offers specify DataCategory "Roads" (≈47.82%) or "Waterways and water bodies" (≈32.69%). Notably, there are no missing values.

Similarly, there is a DataCategoryDetail schema model in the vocabs data that corresponds to the attribute dataCategoryDetail of data offers, as multiple values may be specified for a single data offer. This information is rendered

IRI	Name	#Values
https://w3id.org/mdp/schema#AccessProtocol	AccessProtocol	13
https://w3id.org/mdp/schema#AccessRights	AccessRights	3
https://w3id.org/mdp/schema#CommMethod	CommMethod	3
https://w3id.org/idsa/core/CustomMediaType	CustomMediaType	27
https://w3id.org/mdp/schema#DataCategory	DataCategory	25
https://w3id.org/mdp/schema#DataCategoryDetail	DataCategoryDetail	95
https://w3id.org/mdp/schema#DataModel	DataModel	14
https://w3id.org/idsa/core/Frequency	Frequency	24
https://w3id.org/mdp/schema#GeoReferenceMethod	GeoReferenceMethod	6
https://w3id.org/idsa/core/IANAMediaType	IANAMediaType	9
https://w3id.org/idsa/core/Language	Language	3
https://w3id.org/mdp/schema#MdpLicense	MdpLicense	5
https://w3id.org/mdp/schema#NetworkCoverage	NetworkCoverage	4
https://w3id.org/mdp/schema#RolePBefG	RolePBefG	3
https://w3id.org/mdp/schema#StandardLicense	StandardLicense	29
https://w3id.org/mdp/schema#Status	Status	3
https://w3id.org/mdp/schema#Theme	Theme	14
https://w3id.org/mdp/schema#TransportMode	TransportMode	23

Table 5.5: All schema models and the count of their possible values provided by the vocabs API endpoint

on Mobilithek on the subpage as *Category Details* (German: *Themengebiet-Details*). Each value of DataCategoryDetail is associated with a DataCategory value as parent. In this way, this schema model subdivides DataCategory into more fine-grained subcategories. Since there is a large number of 95 values, a table is displayed for reference. However, barely any data offers (4 of 40,72%) state a dataCategoryDetail value. Hence, this schema model is not applied often.

The values of the themes attribute are resolved in the vocabs data by schema model Theme and are then displayed on the *Offer Details* subpage of a data offer similarly to the previous data offer columns and may be found as *OpenData Category (GovData)*. Notably, Theme lists values of a pre-defined model[4] by the EU illustrated in Table 5.7. However, data offers do not strictly follow these schema models. Next to some faulty data like mailto%3a/TRAN that probably refers to http://publications.europa.eu/resource/authority/data-theme/TRAN, there also are IRI values matching the *INSPIRE theme registry*[5] by the EC example, http://inspire.ec.europa.eu/theme/tn. Data offers may provide several values for themes. Yet, all data offers have at least one assigned category. The most prevalent categories are “Transport” (≈37.08%), “Environment” (≈17.95%), “Regions and cities” (≈17.50%), “Agriculture, fisheries, forestry and food” (≈12.18%) and “Science and technology” (≈9.98%).

[4] Data theme - EU Vocabularies - Publications Office of the EU: https://op.europa.eu/en/web/eu-vocabularies/concept-scheme/-/resource

[5] INSPIRE theme register: https://inspire.ec.europa.eu/theme

IRI	Label en (via vocabs.json)
#AIR_AND_SPACE_TRAVEL	Air and space travel
#CAR_UND_BIKE_SHARING	Car and Bike Sharing
#CLIMATE_AND_WEATHER	Climate and weather
#CYCLE_NETWORK_DATA	Cycle network data
#DYNAMIC_TRAFFIC_SIGNS_AND_REGULATIONS	Dynamic traffic signs and regulations
#FILLING_AND_CHARGING_STATIONS	Filling and charging stations
#FREIGHT_AND_LOGISTICS	Freight and logistics
#GENERAL_INFORMATION_FOR_TRIP_PLANNING	General information for trip planning
#INFRASTRUCTURE	Infrastructure
#TRANSPARENCY_ORGANISATION_FOR_FUEL	Market Transparency Unit for Fuel
#PARKING_AND_REST_AREA_INFORMATION	Parking and rest area
#PEDESTRIAN_NETWORK_DATA	Pedestrian network data
#PUBLIC_TRANSPORT_NONSCHEDULED_TRANSP...	Public transport: non-scheduled transport
#PUBLIC_TRANSPORT_SCHEDULED_TRANSPORT	Public transport: scheduled transport
#RAILWAY	Railway
#REALTIME_TRAFFIC_DATA	Real-time traffic data
#ROAD_WEATHER_CONDITIONS	Road weather conditions
#ROAD_WORK_INFORMATION	Road work information
#ROADS	Roads
#STATIC_ROAD_DATA	Static Road Data
#STATIC_TRAFFIC_SIGNS_AND_REGULATIONS	Static traffic signs and regulations
#TOLL_INFORMATION	Toll information
#UNEXPECTED_ROAD_EVENTS_AND_CONDITIONS	Unexpected road events and conditions
#WATERROADS_AND_WATER	Waterways and water bodies
#CAT_OTHER	Other

Table 5.6: Labels of the schema model DataCategory provided by the vocabs API endpoint for the data offer attribute dataCategory IRI values (https://w3id.org/mdp/schema/data_categories[...])

IRI	Label en (via vocabs.json)
AGRI	Agriculture, fisheries, forestry and food
ECON	Economy and finance
EDUC	Education, culture and sport
ENER	Energy
ENVI	Environment
GOVE	Government and public sector
HEAL	Health
INTR	International issues
JUST	Justice, legal system and public safety
SOCI	Population and society
OP_DATPRO	Provisional data
REGI	Regions and cities
TECH	Science and technology
TRAN	Transport

Table 5.7: Labels of the schema model Theme provided by the vocabs API endpoint for the data offer attribute themes IRI values (http://publications.europa.eu/resource/authority/data-theme/[...])

Two more models relate to the mobility domain and their values are actively referenced in the metadata of data offers, both don't provide much value as most data offers just state the value equivalent to the label "Other". For the schema

IRI	Label en (via vocabs.json)
#MOTORWAYS	Motorways
#REGIONAL_ROADS	Federal and state roads
#URBAN_LOCAL_ROADS	Urban and local roads
#NETWORK_OTHER	Other

Table 5.8: Labels of the schema model NetworkCoverage provided by the vocabs API endpoint for the data offer attribute networkCoverage IRI values (https://w3id.org/mdp/schema/network_coverage[...])

model NetworkCoverage can be associated with the attribute networkCoverage. Its values relate to types of roads as shown in Table 5.8. Secondly, TransportMode holds values used in the data offer attribute transportMode and the range of values can be seen in Table 5.9, which corresponds to types of transportation.

IRI	Label en (via vocabs.json)
#INDIVIDUAL_CAR	Car
#INDIVIDUAL_TRUCK	Truck
#SCHEDULED_BUS	Bus
#SCHEDULED_AIR	Air
#DEMAND_RESPONSIVE_BIKE_HIRE	Bike Hire
#DEMAND_RESPONSIVE_BIKE_SHARING	Bike Sharing
#DEMAND_RESPONSIVE_CAR_HIRE	Car Hire
#DEMAND_RESPONSIVE_CAR_POOLING	Car Pooling
#DEMAND_RESPONSIVE_CAR_SHARING	Car Sharing
#INDIVIDUAL_CYCLE	Cycle
#SCHEDULED_LONG_DISTANCE_COACH	Long-distance coach
#LONG_DISTANCE_RAIL	Long-distance rail
#SCHEDULED_MARITIME	Maritime (including ferry)
#SCHEDULED_METRO	Metro
#INDIVIDUAL_MOTORCYCLE	Motorcycle
#INDIVIDUAL_PEDESTRIAN	Pedestrian
#REGIONAL_AND_LOCAL_RAIL	Regional and local rail
#DEMAND_RESPONSIVE_SHUTTLE_BUS	Shuttle bus
#DEMAND_RESPONSIVE_SHUTTLE_FERRY	Shuttle ferry
#DEMAND_RESPONSIVE_TAXI	Taxi
#SCHEDULED_TRAM	Tram, Light rail
#SCHEDULED_TROLLEY_BUS	Trolley-bus
#TRANSPORT_OTHER	Other

Table 5.9: Labels of the schema model TransportMode provided by the vocabs API endpoint for the data offer attribute transportMode IRI values (https://w3id.org/mdp/schema/transport_mode[...])

These five models have an apparent meaning in the mobility domain, since they fundamentally differentiate between different types of mobility data. As a consequence, a requirement with priority *High* is presented. Admittedly, the models

DataCategoryDetail, NetworkCoverage and TransportMode have little significance in the context of the collection of data offers. This is due to the respective attributes missing values or specifying the fallback category "Other". However, the schema models DataCategory and Theme are actively used to differentiate data offers. Mobilithek does not offer documentation on these models and neither states whether the project defined these schema models or if they've been sourced from a different authority. As mentioned before, the IRI values used in the identifier for the models (https://w3id.org/mdp/schema#<SCHEMA_MODEL>) and the pattern of the IRIs for some values (https://w3id.org/mdp/schema/<SCHEMA_MODEL>#<VALUE>) may give the impression of custom schema models that have been created by in the scope of the Mobilithek project. On the other hand, schema model Theme references values of a vocabulary defined by the European Commission (EC). Further research has to show if there is consensus about vocabularies or schema models or whether the schema models provided by Mobilithek or other authorities, for example, the European Commission (EC), should be focused.

Requirements for concept C-3

R-3-1	:	**NUTS geocode standard** Priority: Low The NUTS geocode standard needs to be supported.
R-3-2	:	**WKT format** Priority: High The Well-known text (WKT) format needs to be supported.
R-3-3	:	**GeoReferenceMethod schema model** Priority: High The GeoReferenceMethod schema model needs to be supported.
R-3-4	:	**Schema models in the mobility domain** Priority: High Schema models that are used in the mobility domain need to be supported.

Concept C-4: Data transmission in open mobility

In Chapter 4, the operationalization of objective **O-4-1** was concluded. The operationalization tasks of gathering quantitative data from an exemplary NAP about communication protocols and their relative distribution has been executed successfully.

Objective **O-4-1**: Define requirements for prevalent communication protocols.

Information about the protocols used for data transmission can only be found as part of the metadata about data sources, which are attached to data offers.

For one, the column commMethod is specified, but remains completely unused. This may refer to information about a communication method with an initial intention to be presented on the data offers subpages (*Offer Details*). As the platform continues to receive software updates, this column may yield actual values in the future.

Secondly, the column accessProtocol exposes the protocol that is used in order to access the resource. Accordingly, Table 5.10 illustrates the distribution of accessProtocol values across the 14,746 data sources. Remarkably, just 23 (≈0.16%) data sources are missing values. Protocols “HTTPS” (13,606 data sources, ≈92.27%) and “HTTP” (1,107 data sources, ≈7.51%) are most frequently specified and represent almost all data sources when combined (≈99.78%). There are five data sources that make use of the “SOAP” and two of the “FTP” protocols. Besides, there is one case of Mobilithek’s proprietary formats “Mobilithek Containerformat (HTTPS)” and “Mobilithek Containerformat (SOAP)”, respectively, which may relate to Mobilithek’s brokering system. It is also worth mentioning that omitting data offers with more than ten data sources, as by the definition of an outlier in concept **C-1**, the distribution stays almost the same.

Label en (via vocabs.json)	Counts	Percentage
HTTPS	13606	92.27
HTTP	1107	7.51
MISSING DATA	23	0.16
SOAP	5	0.03
FTP	2	0.01
Mobilithek Containerformat (HTTPS)	1	0.01
Mobilithek Containerformat (SOAP)	1	0.01
Other	1	0.01

Table 5.10: Distribution of accessProtocol values across data sources

The associated vocabs schema model AccessProtocol additionally mentions the labels “OCIT”, “gRPC”, “AMQP”, “MQTT”, “RSS” and “OTS2”. In conclusion, requirements for the prevalent protocols are presented. However, the proprietary protocols of Mobilithek have been excluded as they may be potentially used on this NAP exclusively. Similarly to the priority ratings of concept **C-1**, the relative frequency of “HTTPS” justifies for a priority of *High*. “HTTP” constitutes for data sources below ten percent and above one percent and therefore receives a *Medium* priority rating. Because of the almost negligible frequencies, “SOAP” and “FTP” are of *Low* priority.

Label en (via vocabs.json)	Counts	Percentage
Public transportnon-scheduled transport	56	83.58
Parking and rest area	4	5.97
Other	2	2.99
Unexpected road events and conditions	2	2.99
Dynamic traffic signs and regulations	2	2.99
Public transportscheduled transport	1	1.49

Table 5.11: Distribution of dataCategory values across brokered data offers

A different cue is related to the brokering system of Mobilithek. The platform allows registered users that create an organization to provide or receive data through Mobilithek's brokering system. In the metadata, the column mdpBrokering is assumed to indicate a brokered data offer and identifies 67 (≈1.09%) data offers as such with a True value, while the value for the remaining data offers is set to False. As shown in Table 5.11, most of the brokered data offers (≈83.58%) refer to the dataCategory of "Public transportnon-scheduled transport". In the scope of the data analysis, it was revealed that the brokered offers relate to Taxi companies mostly. Further work may explore the brokering system and its proprietary protocols.

Requirements for concept C-4

R-4-1	:	**HTTPS protocol** Priority: High The HTTPS protocol needs to be supported.
R-4-2	:	**HTTP protocol** Priority: Medium The HTTP protocol needs to be supported.
R-4-3	:	**SOAP protocol** Priority: Low The SOAP protocol needs to be supported.
R-4-4	:	**FTP protocol** Priority: Low The FTP protocol needs to be supported.

Concept C-5: Live data in open mobility

In Chapter 4, the operationalization of objective **O-5-1** was conducted. The operationalization consists of gathering quantitative data from an exemplary NAP about relative distribution of live data, continuous data streams and repeatedly updated data batches and update cycles. While it was possible to access distributions about frequencies of updates to the data, data cover both continuous

and periodic cycles, it was not possible to make statements about the nature of data transmissions relating to streams or batch transmissions.

Objective **O-5-1**: Define requirements for the support of live data.

Cues about the frequency of updates to the data sources of data offers can be found in the accrualPeriodicity attribute. Its values are used on the *Offer Details* subpages for data offers on the *Data Access* tab. There, the particular value is shown on *Content data* as *Update Interval* if specified. Figure 5.9 illustrates the distribution of occurring accrualPeriodicity values. Most importantly, the majority of 3,512 data offers (≈57.39%) are missing values. Updating data sources “Irregularly” is the most common practice (1,315 data offers, ≈21.49%). A number of 525 data offers (≈8.58%) are “Never updated”. Regarding fixed schedules, “Up to yearly” (288 data offers, ≈4.71%) and “Up to monthly” (100 data offers, ≈1.63%) occur. Special values include “Continuously”, which is stated by 179 data offers (≈2.93%), and “On occurrence” by 5 data offers (≈0.08%).

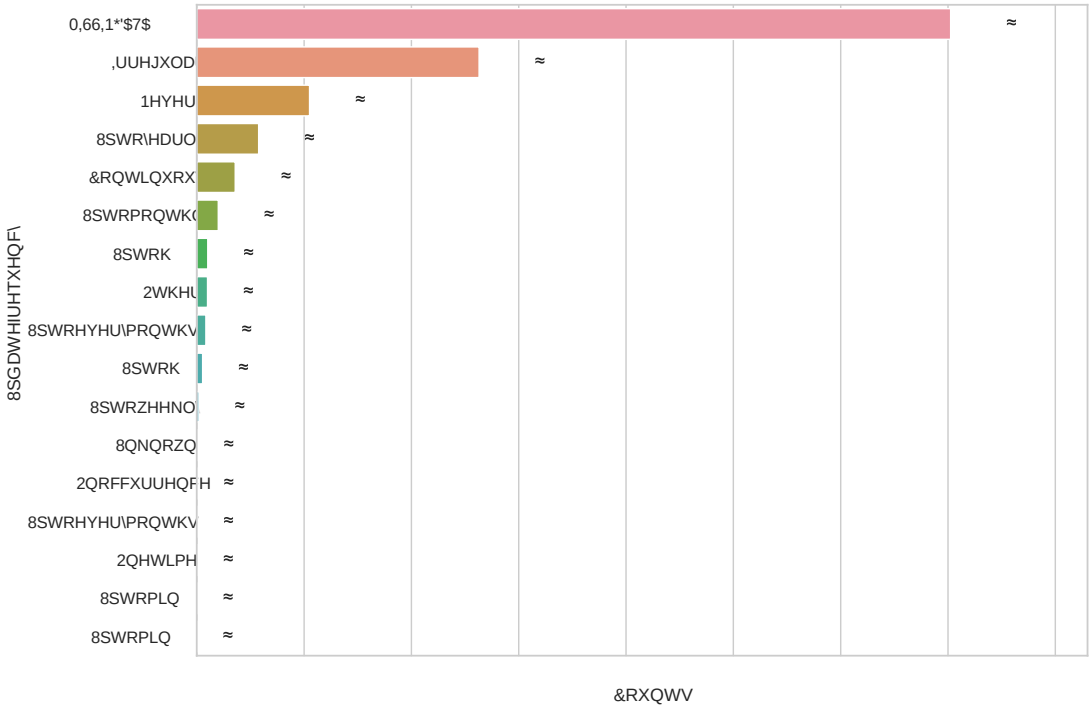


Figure 5.9: Distribution of accrualPeriodicity values across data offers

The values of accrualPeriodicity may be resolved from IRI values to English labels with the schema model provided by the vocabs API endpoint as seen in Table 5.12. In contrast to the 24 possible values given by the schema model, only 16 values do actually occur in the data offer metadata. Surprisingly, there are values of two different IRI schemas included. Besides a seemingly custom

pattern by Mobilithek (example: https://w3id.org/mdp/schema/frequency#NEVER), there are IRI values originating from the "International Data Spaces Information Model"[6] of the International Data Spaces Association (IDSA), for example, https://w3id.org/idsa/code/ANNUAL.

IRI	Label en (via vocabs.json)
mdp/schema/frequency#NEVER	Never
mdp/schema/frequency#ONCE	One-time
mdp/schema/frequency#ON_OCCURRENCE	On occurrence
idsa/code/CONTINUOUS	Continuously
idsa/code/IRREGULAR	Irregularly
idsa/code/EVERY_1_MINUTE	Up to 1 min
idsa/code/EVERY_5_MINUTES	Up to 5 min
idsa/code/EVERY_10_MINUTES	Up to 10 min
idsa/code/EVERY_15_MINUTES	Up to 15 min
idsa/code/EVERY_30_MINUTES	Up to 30 min
idsa/code/HOURLY	Up to 1h
idsa/code/BIHOURLY	Up to 2h
idsa/code/EVERY_THREE_HOURS	Up to 3h
idsa/code/TWO_TIMES_A_DAY	Up to 12h
idsa/code/DAILY	Up to 24h
idsa/code/WEEKLY	Up to weekly
idsa/code/BIWEEKLY	Up to bi-weekly
idsa/code/MONTHLY	Up to monthly
idsa/code/QUARTERLY	Up to every 3 months
idsa/code/SEMIANNUAL	Up to every 6 months
mdp/schema/frequency#LESS_THAN_YEARLY	Less frequent than yearly
idsa/code/ANNUAL	Up to yearly
mdp/schema/frequency#UNKNOWN	Unknown
mdp/schema/frequency#OTHER	Other

Table 5.12: Labels of the schema model Frequency provided by the vocabs API endpoint for the data offer attribute accrualPeriodicity IRI values (https://w3id.org/[...])

Assessing the portion of live data is not unambiguously possible, as it is a matter of the definition of live data in regard to the values in the distribution. However, the categories of "Continuously" and "On occurrence" may both be considered as an indication of live data and can be combined to about 3%. High frequent categories, such as "Up to 1 min" or "Up to 10 min" do not matter regarding the aggregation of relative frequencies, as they are barely specified by live data offers. In conclusion, a requirement is presented for the support of live data. The priority rating scheme of the other concepts involving distributions, a *Medium* priority can be justified as the portion size of 3% lies between the *High* priority

[6] International Data Spaces Information Model: https://international-data-spaces-association.github.io/InformationModel/docs/index.html

rating (ten percent or more) and the *Medium* priority rating (one percent or more). At last, no statement about the connection can be made, as there is no evidence in the metadata regarding continuous or batch data connections for transmission.

Requirements for concept C-5

R-5-1 : **Live data**
Priority: Medium
Live data needs to be supported.

Concept C-6: Authentication with open mobility data portals

In Chapter 4, the operationalization of objective **O-6-1** was concluded. The operationalization consists of two tasks. Qualitative data from an exemplary NAP about the portion of metadata and data gated behind authentication could was gathered by probing domains that represent 1% or more data sources each. The execution of the task of determining the authentication mechanism on exemplary NAP is two-fold. For one, an explanation on the regular user authentication mechanism is given. Secondly, the exemplary NAP Mobilithek also features an additional brokering system, which has not been investigated as part of this work.

Objective **O-6-1**: Define requirements for the support of authentication mechanisms.

To obtain the data portal domains of data sources, the Python package urllib and its method urlparse were used to extract the domain of the accessUrl attribute. Remarkably, all data sources specify the accessUrl attribute. By computing the frequencies of domains occurrences, it was possible to select the most prevalent cases. Domains, which constitute for 1% or more data sources (31 domains in total), were then taken into consideration for manual assessment. For each domain in the selection, the accessUrl of five data source were randomly picked and probed. Except for some instances whereas URLs were unreachable or merely referenced the index of the website, either data portals required any form of authentication to access the data. However, the 67 data offers as part of the brokering system specify the same accessUrl value of https://provider_endpoint, which obviously cannot be probed as it does not represent a valid domain. By definition, open data "can be freely used, modified, and shared by anyone for any purpose" according to Open Knowledge Foundation. Therefore, refraining from any form of authentication concerning data access is preferable and to be expected on a NAP focus on open data.

In regard to the NAP Mobilithek in particular, there are two use cases that potentially involve a form of authentication. For one, the Mobilithek metadata directory is used to browse and search data offers. Afterwards, one may register a free

account and then log in on the website. Without doing so, users are restricted to a potentially smaller number of data offers, as, parts of the *Offer Details* subpage for data offers are not displayed. Regarding the API Crawling scripts, authentication was achieved by providing the Authorization attribute-value-pair to API requests. This header attribute carries a JSON Web Token (JWT), which was extracted by manually logging in and inspecting the data traffic in the network tab of the *Developer Tools* in a web browser. Depending whether a data processing language benefits from the metadata on Mobilithek in particular, this authentication mechanism may be considered relevant and justify the implementation of a fully automated approach. During the timespan of this work, Mobilithek has expanded their services with the *Harvesting API* feature to export data offers, which may be favorable. As the second use case, Mobilithek offers a brokering system to handle data exchanges for the involved parties. Instead of regular user accounts, this requires systems to able to send or receive data to or from Mobilithek. To make use of the brokering interface either as a producing or a consuming actor, an organization needs to be registered on the platform. Then, accessing or providing data through technical interfaces may entail further authentication mechanisms. Being a representative of a company, an office or authority, a bureau, a university department or any other entity that wishes to provide or obtain data (BMDV, 2022b) is necessary to create an organization. Hence, no further actions were taken to investigate the brokering service.

The operationalization and also further considerations didn’t yield any groundwork for requirements that need to be supported by an open mobility data processing language.

5.3.3 Overview of the Catalog of Requirements

The presented table illustrates an overview of the catalog of requirements.

Concept	Requirement	Priority
C-1	R-1-1 : CSV media	High
	R-1-2 : ATOM media	High
	R-1-3 : WMS_SRVC media	High
	R-1-4 : HTML media	Medium
	R-1-5 : WFS_SRVC media	Medium
	R-1-6 : GML media	Medium
	R-1-7 : SHP media	Medium
	R-1-8 : PDF media	Medium
	R-1-9 : ZIP media	Medium
	R-1-10 : TIFF media	Medium
	R-1-11 : XML media	Medium
	R-1-12 : XLSX media	Medium
	R-1-13 : JSON media	Medium
	R-1-14 : GEOJSON media	Medium
	R-1-15 : XLS media	Low
	R-1-16 : KML media	Low
	R-1-17 : REST media	Low
	R-1-18 : TXT media	Low
	R-1-19 : PNG media	Low
	R-1-20 : KMZ media	Low
	R-1-21 : DOCX media	Low
	R-1-22 : RSS media	Low
	R-1-23 : JPEG media	Low
	R-1-24 : TSV media	Low
C-2	R-2-1 : Relational data	High
	R-2-2 : Graph-based data	Medium
C-3	R-3-1 : NUTS geocode standard	Low
	R-3-2 : WKT format	High
	R-3-3 : GeoReferenceMethod model	High
	R-3-4 : Schema models in the mobility domain	High
C-4	R-4-1 : HTTPS protocol	High
	R-4-2 : HTTP protocol	Medium
	R-4-3 : SOAP protocol	Low
	R-4-4 : FTP protocol	Low
C-5	R-5-1 : Live data	Medium
C-6	No requirements	

Table 5.13: Overview of the catalog of requirements

6 Demonstration and Evaluation

This chapter concerns the demonstration and evaluation of the established process in regard to analyzing the metadata of exemplary NAP and synthesizing insights, as well as the resulting value for the JValue project and the development of their data processing language, Jayvee.

6.1 Demonstration

For demonstration purposes, the catalog of requirements was applied to Jayvee, the concrete data processing DSL by JValue. To realize this, the JValue project team was provided with the implementation chapter and asked for feedback.

As part of the objective definition in Chapter 3, members of the JValue project were previously consulted as part of a qualitative survey. They were asked to provide questions and express their concerns in relation to working with open mobility data in the context of data processing pipelines and the creation of DSL for models thereof. On that basis, it was possible to derive concepts and objectives to cover the diversity expressed in the topics in this survey.

To demonstrate the artifact, the same group of respondents was provided with the chapter on the implementation. For one, the JValue members were able to inspect the execution of the process consisting of obtaining and analyzing the metadata about the exemplary NAP Mobilithek as described in Section 5.1 as well as the synthesis of the data analysis results in Section 5.3 to create a catalog of requirements. Attaching the actual data analysis documents (Exploratory Data Analysis and Investigative Data Analysis) may have helped the understanding of some further-reaching conclusions and justifications within the chapter. However, they were not included since these documents are excessive in length and may lead to confusion.

6.2 Evaluation Design

For evaluation purposes, a concise survey was created to capture the input of the JValue project members, which is attached as appendix Section F, including the results. Contrary to the detailed interviews for the purpose of the objective definition, a questionnaire was compiled by making use of Google Forms for two reasons: First, the procedure is more flexible because of the asynchronous nature of an online survey. Secondly, the use of single-choice or text input entails instant access to results. By applying the respective settings about anonymity, the protection of the participants' identities could be ensured.

Regarding the structure of the survey, the participant is first presented with the derived concepts from the initial qualitative survey and asked whether these cover their concerns. The respective answers describe the effectiveness of the approach for the objective definition. Then, the survey traverses through the six concepts separately and asks the participant to provide two scores for the execution of the process and the usefulness of the requirements regarding the development of Jayvee, the data processing pipeline DSL of JValue. Both questions are presented with a five-point Likert scale, illustrating a rating from "Very bad" to "Very well" [1] in regard to the execution of data analysis and data wrangling and "Very useless" to "Very useful" [2] concerning the value for the development of Jayvee. Because of the quantitative nature of this data, it is possible to aggregate the answers and emphasize patterns with visualizations. For each concept, the respondent may add a comment in the form of a free text field. Eventually, the participant may express general thoughts as part of a final free text field. While this results in qualitative data that may not be aggregated, it is meaningful to understand the respondents' reasoning or concerns leading to the assigned scores and general trends in the quantitative data.

6.3 Evaluation Results

Matching the number of initial interviewees for the qualitative survey, the evaluation survey resulted in three responses from the members of the JValue team.

6.3.1 Quantitative Results Data

The quantitative answers to questions specific to the concepts are illustrated in Figure 6.1. While the plots are divided into two subplots per row, each row represents a concept and each column relates to one of the two questions that were asked for each concept.

[1] Full scale: "Very bad" (1), "Bad" (2), "Average" (3), "Well" (4), "Very well" (5)
[2] Full scale: "Very useless" (1), "Useless" (2), "Moderate" (3), "Useful" (4), "Very useful" (5)

In regard to the execution of the data analysis and data synthesis, the scores are mixed but lean towards “Well” (4) or “Very well” (5) for the most part. Particularly, the concept **C-4**, data transmission in open mobility, performs especially well, as all respondents agree on a “Very well” (5) rating. In contrast, the concepts **C-3**, open mobility value types, and **C-5**, live data in open mobility, each exhibit a single “Average” (3) and no “Very well” (5) response.

Figure 6.1: Overview of quantitative evaluation results

Moreover, the responses are also mixed overall for the usefulness of the requirements for the development of their data processing language. The participants perceive the requirements of the three concepts **C-4**, **C-1**, and **C-6** exclusively as “Very useful”. The concept **C-2**, open mobility data structures, gets a worse evaluation as two respondents assess the associated requirements as “Useful” (4) and only one gives a “Very useful” (5) score. Both concepts **C-3** and **C-5** show disagreements within the responses with a worse general rating. For the former, the ratings reside in the middle of the scale with a single score for “Useless” (2), “Moderate” (3) and “Useful” (4), respectively. The latter, concept **C-5**, is perceived better as the ratings include a single vote for “Moderate” (3), “Useful” (4) and “Very useful” (5).

6.3.2 Qualitative Results Data

For all concepts except **C-4**, the survey participants made use of the free text input and added further commentary, which is summarized here.

Concerning concept **C-1**, two comments suggest including additional data points, which may not be within the scope of this work. Moreover, one would like to see the media type in the actual data compared against the media type found in the metadata. It states that it may also be worthwhile to look further into data sources that identify as having the media type “Other”. Additionally, ZIP archives should be extracted to inspect the file types they consist of.

There are two additional comments for concept **C-2**, consisting of the suggestion to append the counts for both columns in Table 5.3 and the argument that certain formats may contain relational as well as graph-based data. One of them also denotes that the process of categorizing the media types from concept **C-1** may not be suitable to address the issue of defining requirements regarding data structures, despite a well-executed process.

For **C-3**, each of the respondents left a comment with the expectation of actual data potentially being more meaningful to determine value types, and one person even recommends addressing this in further work. Another respondent assumed mention of the International System of Units (SI). Two comments also note room for improvement concerning requirement **C-3.4**, data models in the mobility domain. Particularly, the requirement is called vague once and, in the other comment, the participant expresses that “exact data models would be more useful instead of the schema identifiers.”

The respondents provide additional comments with unique considerations for concept **C-5**, live data in open mobility. For one, an additional, more profound investigation is suggested that examines a random group of data offers that belong to the “MISSING DATA” category. Secondly, monitoring mechanisms are brought up, which may be used to observe data offers over a certain time span

and check for updates to their data. This may be based on a timestamp indicating a modification. Such endeavors would be out of the scope of this work, as stated by the author of that particular comment.

For the final concept **C-6**, authentication with open mobility data portals, one respondent is uncertain about the accuracy of the way samples were picked for manual probing of data source domains. The other suggests examining data sources automatically in greater quantity to check for particular status codes, which may indicate authentication mechanisms.

Only one participant made use of the opportunity to add further general comments at the end of the survey. The respondent approves the results and elaborates on the situation. They state that it would be possible to invest more resources in data analysis, but further results may not be of much value with respect to the efforts of the JValue project. They add that open data conceptually may be blamed for the cases of gaps in the metadata.

6.3.3 Reflecting on the Results

To start with, all participants stated that the derived concepts were indeed able to cover their concerns completely. This is of major importance, as the concepts posed a reference point throughout this work as they illustrated guidelines to associate respective objectives and operationalization tasks.

In essence, the results may be interpreted as two-sided. On the one hand, the requirements for **C-1**, **C-2**, **C-4** and **C-6** were produced by a well-executed process and also assessed as useful for the development of the data processing language according to the evaluation results. This aligns with the insights gained from the respective analysis results, as the metadata contained meaningful attribute value distribution. In that sense, the established process and the execution thereof are considered successful.

Concerning the remaining concepts **C-3** and **C-5**, the evaluation scores for the execution and usefulness indicate that further work is required. This may be due to the metadata not exposing much purposeful information, making it difficult to create requirements. Apart from obtaining and analyzing actual data to identify value types, there may not be much room for improvement. Secondly, for the topic of live data, just one weak requirements was created. Further efforts should have targeted either the data synthesis and elaborate on the data analytics results, or the objective definition activity by creating more objectives for **C-5** to guide a broader data analysis and data synthesis for this concept in particular.

While all the comments of the respondents were brought up and put in context within the previous segment, some need to be addressed as they add ideas for potential improvements.

For one, two comments for concept **C-3**, open mobility value types, relate to the topic of schema models and the respective requirement **R-3.4**, stating ambiguous conclusions and an imprecise requirement. In the implementation, the support of schema models to categorize the type of data offers was deemed to be essential in general. However, in the case of the exemplary NAP Mobilithek, schema models were discovered to be provided by the vocabs API endpoint. In Section 5.2.2, an unsuccessful attempt was made to discover the origin of these models by examination of common specifications such as DCAT-AP or DCAT-AP.de. Consequently, the connection between this particular section and the instantiation of the requirement should have been made clearer.

Moreover, an evaluation comment for concept **C-5**, live data in open mobility, mentions the idea of monitoring modification timestamps to observe updates to data offers. Specifically, data offers do indeed specify modified timestamps, but they were not mentioned in the implementation chapter presented to the survey participants. Actually, this attribute was investigated as part of the data analysis, and no meaningful conclusions were drawn. Nonetheless, the presence of the attribute and the analysis results thereof should have been addressed.

Another comment for concept **C-5** suggests bundling the occurring values into groups such as “live/streaming, on a schedule, regularly without a schedule[, or] never” to create additional requirements. The respondent assesses the differentiation into live and static live data alone as not meaningful enough for the data processing language to design scheduling operations. Certainly, the suggestion should be adapted, as their point is valid.

At last, a respondent was uncertain about the methods for selecting random accessUrl values to probe for authentication mechanisms on the most prevalent domains of actual data for concept **C-6**. In this regard, the description of methods in the artifact should have been more elaborate as the analysis documents include a particular code snippet on this issue.

6.4 Evaluation Review

Using a Likert scale to rate the execution of the process and the usefulness of the requirements for language development has proven valuable, as the requirements of different concepts may be observed at a glance and compared against each other. Furthermore, the participants made extensive use of the option to add comments, justifying their score or expressing further concerns.

Overall, the responses effectively communicate the value of the approach and motivate further work. The evaluation is regarded as meaningful, and the responses correlate well with the impressions that formed during the activity of the implementation.

7 Conclusions

To conclude, this chapter first addresses limitations and presents directions for further work. Furthermore, the second section compiles a summary of the completed activities and results of this thesis.

7.1 Limitations and Further Work

This section elaborates on the limitations of this contribution and the linked suggestions for further work. For one, the subject in the general sense is concerned. Besides, also two suggested efforts relate to the exemplary NAP Mobilithek only. Comments that were discussed in the previous Chapter 6 have been excluded from this section.

Instead of targeting actual data for analysis purposes, the process and execution thereof relied on metadata alone. Regarding the exemplary NAP Mobilithek in particular, the metadata is administered in a format the platform provides itself, resulting in a coherent structure and consistency. Because of this, data preparation and analysis were deemed feasible. However, the metadata dump features weak explanatory power in some aspects in this case. In addition, the exemplary NAP does not provide documentation about the meaning of values or categories for the displayed fields of a given data offer on their website. While there are many diverse attributes, numerous data offers miss a large portion of values or specify a placeholder value such as “Other”. Thus, actual data should be considered for examination to get concise answers, as it is possible to draw conclusions directly instead of depending on metadata alone. Nevertheless, obtaining and analyzing the actual data of a data portal such as the exemplary NAP Mobilithek is expected to be much more complex. This may be due to data offers being authored by various parties, which organize and provide data in different ways. Additionally, the quantity of data is presumed to be larger, complicating the task of obtaining the data. Nonetheless, examination of actual data instead of metadata may result in valuable insights for the creation of requirements and should be considered as another research opportunity. Further work in this regard may aim for NAPs in general, but also the particular exemplary NAP Mobilithek.

Notably, the platform uses the static data of their vocabs API endpoint to resolve IRI values in the metadata to human-readable labels. This mapping consists of schema models stating a category and associated values, giving way the diversity of a particular metadata attribute. Conversely, the combination of utilized schema models seemingly does not strictly follow the established specifications for data portals by the European Commission or German government institutions. As a result, it is difficult to assign a fixed set of schema models for an open mobility data processing language. With Mobilithek as an example, NAPs may use different selections or even custom creations. Further work may explore solutions to this problem.

As a last general direction for further work, the adaptability of implementation should be taken advantage of. The process established in light of Chapter 4 consists mostly of tasks that have been solved programmatically. In particular, only a single NAP of one member state of the European Union was considered in this work, which poses a limitation and results in a strong bias in the data analysis results. To address this, respective code segments should be adapted to fit the infrastructure and metadata format of different NAPs of the same country or others. Moreover, the implementation may also be manually executed again for the exemplary NAP at a later time to make use of a larger catalog of data offers as the basis for the data analysis. Subsequently, the procedure of obtaining and analyzing metadata may also be automated, which may be motivated by monitoring timestamps to identify data offers as live data, for example.

Because of restrictions on Mobilithek, two specific actions on the platform were not followed further. Next to the authorization mechanism, which requires the creation of a user account to view all data offers and their metadata, users need to be affiliated with an organization to access restricted features of the platform. This was elaborated in Section 5.3.2 for concept **C-6**.

For one, the values in the metadata were provided by either the author of a data offer or as a result of an automated mechanism by the platform itself. To comprehend the attributes and values better, a data offer may be created by way of trial. In particular, it may be beneficial to check whether values are entered as free text or with a predefined dropdown menu. This feature was restricted to users woh belong to an organization. Therefore, the aspect of creating a data offer was not further explored in the scope of this work and may be considered in the future.

Secondly, Mobilithek offers a brokering system, which has not been examined here as it also requires affiliation with an organization. Though, the data analysis revealed that only a small portion of the data offers utilize this feature. This may entertain the assumption that the brokering system is irrelevant. However, Mobilithek is meant to succeed the German data providers MDM and mCloud (BMDV, 2022a). As a fact, the originatedFrom attribute is set to

https://w3id.org/mdp/schema/portal#M_CLOUD for over 98% of data offers, which do not overlap with the 67 data offers that associate with Mobilithek's brokering (attribute mdp:Brokering). This indicates a feature that is exclusive to Mobilithek and that the adoption of this brokering system may improve for future data offers. With it becoming potentially more relevant over time and entailing different ways of transmitting data, utilization of this feature should be tracked in the future.

7.2 Summary

As a summary, Design Science Research Methodology was employed to create the artifact of requirements for an open mobility data processing language. Motivated by Jayvee, a DSL to specify models of a data processing pipeline, the JValue team expressed interest in extending their language to support open mobility data.

To address this, JValue project members participated in a qualitative survey to capture their concerns regarding data processing languages and open mobility data. Based on that, it was possible to determine concepts and corresponding objectives to guide the activities of solution design and implementation.

For the solution design, operationalization tasks and a process were outlined to approach the objectives. Fundamentally, the presented process employs API crawling to acquire metadata from the exemplary NAP for German transport data, Mobilithek. The platform presents itself as a suitable source for open mobility data, with its catalog of data offers being supplied by many different institutions.

Concerning the implementation, the designated process was realized and executed successfully. Consequently, metadata was gathered, prepared, and analyzed, whereas the data analysis consists of a dual approach. Exploratory Data Analysis provided essential insights about the structure and contents of the undocumented metadata. Besides, another, more objective-focused analysis was able to draw conclusions and put the respective information in context. Eventually, requirements are established in line with the insights at hand and presented as part of each particular concept.

Finally, the JValue team was approached a second time to demonstrate and evaluate the results. The three members were provided access to the implementation chapter and a subsequent evaluation survey. In the particular chapter, the responses were elaborated upon and addressed. Referring to the distribution of quantifiable evaluation results, respondents considered the requirements for three concepts as "Very useful" for the development of JValue's data processing language. For the remaining requirements, results are viewed as less useful as the ratings are mixed.

Appendices

A Additional Files

There are additional files, which may be retrieved from https://faubox.rrze.uni-erlangen.de/getlink/fi6gwAx8BdNgpzEAFynhE2/.

The folder structure is as follows:

```
thesis_files
├── Analysis
├── API_Crawling
├── Document
└── Miscellaneous
```

- Analysis
  This folder contains the Jupyter Notebook documents that constitute the data analysis as introduced in Section 5.1.3. Subfolder dist provides executed versions of these with and without code, and also in different formats. For the best experience, the HTML versions is recommended.
- API_Crawling
  This folder contains the Python scripts that were implemented to retrieve the metadata of Mobilithek as introduced in Section 6.5.1. Subfolder data provides the downloaded data as persistent files. The intermediate artifact data_offers_dump_prepared.pickle is the result of executing the *Data Preparation* Jupyter Notebook and constitutes the basis for the data analysis Jupyter Notebook documents.
- Document
  This folder contains the LaTeX source files to create this document.
- Miscellaneous
  This folder contains miscellaneous files that were used to create content for this thesis. This includes Python scripts to generate the eBM materials in appendix Section E, the survey results for the evaluation, and a Jupyter Notebook document to create Figure 6.1.

B Jayvee Example: cars.jv

Summary: A CSV file with tabular car data is downloaded and interpreted as CSV. Next, the table is interpreted as such while the column types are set up. Finally, the table data is stored in a database.

```
pipeline CarsPipeline {

  block CarsExtractor oftype HttpExtractor {
    url: "https://gist.githubusercontent.com/noamross/e5d3e859aa0c794be10b
    ↪  /raw/b999fb4425b54c63cab088c0ce2c0d6ce961a563/cars.csv";
  }

  pipe {
    from: CarsExtractor;
    to: CarsCSVInterpreter;
  }

  block CarsCSVInterpreter oftype CSVInterpreter {
  }

  pipe {
    from: CarsCSVInterpreter;
    to: NameHeaderWriter;
  }

  block NameHeaderWriter oftype CellWriter {
    at: cell A1;
    write: "name";
  }

  pipe {
      from: NameHeaderWriter;
      to: CarsTableInterpreter;
  }

  block CarsTableInterpreter oftype TableInterpreter {
    header: true;
    columns: [
      "name" oftype text,
      "mpg" oftype decimal,
      "cyl" oftype integer,
      "disp" oftype decimal,
      "hp" oftype integer,
      "drat" oftype decimal,
      "wt" oftype decimal,
      "qsec" oftype decimal,
      "vs" oftype integer,
      "am" oftype integer,
      "gear" oftype integer,
      "carb" oftype integer
```

```
    ];
  }

  pipe {
    from: CarsTableInterpreter;
    to: CarsLoader;
  }

  block CarsLoader oftype SQLiteLoader {
    table: "Cars";
    file: "./cars.sqlite";
  }
}
```

C Interview Handout

Master Thesis: Requirements for an Open Mobility Data Processing Language - Maximilian Lattka

JValue interview: Guidelines for Requirements for an Open Mobility Data Processing Language

This document creates a communication channel between the JValue team and the master thesis "Requirements for an Open Mobility Data Processing Language" for the purpose of objective definition as in requirements for the data processing language and eventually evaluation of those.

Context

As there are a couple of subjects at hand, it's reasonable to first formulate the common terminology here to start on the same page and eliminate ambiguity.

Thesis "Requirements for an Open Mobility Data Processing Language"

This thesis represents a starting point for JValue's domain specific language (DSL) Jayvee to work with open mobility data. Aligning with Design Science methodology [Peffers et al, 2007], the JValue team and I will continuously collaborate on the "Problem Identification" and "Objective Definition" over the timespan of this work. Lateron, "Solution Design" and "Implementation" will result in a catalog of requirements for Jayvee based on researching and analyzing open mobility data and Mobilithek as an implementation of a data provider system.

Jayvee

As of now, the JValue team of OSS at FAU is in development of the DSL Jayvee to realize ETL[1] pipeline models in textual form. Jayvee is supposed to describe the behavior of ODS regarding the integration of open data sources.

Mobilithek

The new national access point for mobility data Mobilithek by the German government provides a metadata listing for various data sets. Using the web GUI of Mobilithek, users are able to browse and filter data sets - some parts and some data offers require authentication (account can be created free of charge). For the majority of entries, external links point to the actual data which is then offered as a download off-site. Also, for some data offerings Mobilithek can act as a brokering service for Machine-2-Machine communication. For further context, you may refer to the "Technical interface description" document in the download section of Mobilithek.

[1] ETL stands for Extract, Transform, Load

Questionnaire

This set of questions are a starting point for the construction requirements for the DSL Jayvee working with open mobility data. In light of this thesis it is in particular important to discover the main topics that need to be covered and establish standard questions to be asked repeatedly over the course of implementing the catalog of requirements.

Open mobility data vs. open data in general

Do you expect open mobility data to have particularly different qualities from general open data when dealing with it in an ETL pipeline context? If yes, how so?

ETL data pipeline tasks and open mobility data

What questions do you have about:

- Extracting open mobility data?
- Transforming open mobility data?
- Loading open mobility data?

Mobilithek

Do you have any questions about Mobilithek as a provider for open mobility data?

Further questions

Do you have any ideas for requirements regarding DSLs?

Any other questions/concepts regarding functional or non-functional requirements?

D Expert Interviews with JValue Members

Interview with Expert1

Summary

- Open mobility data different to the general open data?
  - GTFS-RT data (real-time)
  - Up to now, only batch processing support; streaming is not supported
  - Scheduled data releases; open mobility data to be re-released
    - ∗ Contrary to static data
    - ∗ Consistent errors repeated over time
  - Dynamic aspect
    - ∗ How much of the data is actually re-released on regular schedule?
    - ∗ How much of the data is actually live?
    - ∗ How do you figure this out? (Maybe metadata?)
    - ∗ How often do you need to reload the schedule?
- Questions regarding extracting open mobility data?
  - What are the most common file formats you need to support?
  - Do you have to be concerned about encoding, e.g. ASCII for German umlauts?
  - What's the size of data? (Big data approach; performance relevant)
  - Do you need to authenticate to get the data or metadata?
  - How does authentification mechanism work?
  - What communication protocols are used?
  - What kinds of APIs are used?
    - ∗ Do you need to traverse through multiple pages whereas one API call gives you limeted results (just one page)?
    - ∗ Page-like tokens with offset?
  - Actually getting the data
    - ∗ Mobilithek: Sometimes it is necessary to follow a different external website and download the data there.
    - ∗ Most common external portals on Mobilithek?
- Questions regarding transforming open mobility data?
  - Geographic aspect of mobility data
    - ∗ Filters based on geofencing
    - ∗ longitudes and latitudes as data points
  - What kind of errors are the most common?
    - ∗ CSV-data which contains a line that is not conform with the specification
      - · Interesting if many data offers affected (~80%)
    - ∗ Fields for longitude and the latitude values mixed up
      - · Way to check for this?
      - · Potential cleanup procedures?
  - What kind of enums exist in the world of transport data?
    - ∗ E.g., well-known ids for transportation types
    - ∗ Resolve variables to human-readable format
    - ∗ Properties: numeric versus strings
    - ∗ Concept for domain modeling?
      - · E.g. postalcode; a number but also modeled differently; German Postleitzahl: Having always 5 digits; able to write specific checks
    - ∗ Type of transport might be some European-given enumeration of train, taxi and so on
      - · European-given enumeration used in how many data sets?
  - How to access metadata regarding licenses?
  - Data cleaning and getting rid of errors
    - ∗ Fixing empty values/missing values
    - ∗ What kind of values are commonly missing and how to deal with it?
      - · How to interpolate?
- Questions regarding loading open mobility data?
  - What is the structure of data?
    - ∗ Relational, graph-based (difficult to store in relational database) or like a document store or else
- Questions on Mobilithek?
  - Most questions are related to Mobilithek anyways as an example for open mobility data
  - What is the quality of data on it?
    - ∗ Potentially hard to answer
  - What are the formats of data on it?
    - ∗ Potentially easy to answer
  - What kind of metadata exists?
  - How is the compatibility (meaning in that context?)?
  - What kind of values are there for rights, like licenses?
  - Are there subdomains in Mobilithek? What are those? Overview?
    - ∗ public transport versus private (e.g. train schedule versus highways for cars)
    - ∗ "How many parking spaces are available in some region?" is very different from train schedules
- Further questions?
  - Functional/non-functional requirements of data pipeline DSLs
    - ∗ Relevant in general
    - ∗ Functional requirements: DSL should be extendable by domain experts without changing their language
      - · Modeling their own data and value types

1

Transcript (manually post-processed)

- ML: Hey Expert1, thanks for spending some time with me for my questionnaire regarding my interview for requirements for an open mobility processing language. I hope you're halfway familiar with what I'm doing and what I want to achieve with this. I have some questions prepared. Maybe you had a look [at] them already. Let's start with it.

2

Topic: Open mobility data vs. open data in general

- ML: We are dealing with open mobility data and do you expect this kind of data, the open mobility data, to be different, having particularly different qualities versus the general open data, you know and maybe have dealt with especially in the ETL pipeline context?
- Expert1: One thing I think about often is GTFS-RT data. They're like real-time data. I think currently we only support batch processing and have no plans to support streaming, but I'm asking myself if there are some requirements from real-time data and also from changing data, because I talked to some people locally. They told me that there are schedules, for example, in open mobility data that get re-released every like half a year or something and there might be consistent errors in them that we might want to fix. I think one of the things that will be a topic is that it's not only static data, but there's a regular release schedule for some of the data and some of the data might even be real-time.
- ML: So maybe in a resulting language, you might want either to define this kind of data or maybe want to specify a kind of rerun or scheduled run if you have like maybe batch data, right?
- Expert1: Yeah. I mean, especially in the context of objectives for the requirements or for the information, I think, it will be interesting to know how much of the data is actually re-released on a regular schedule and how much of it is actually live and also how you would find out, for example. If you re-release the schedule every two weeks or something, is there some kind of metadata field you need to read? How do you get this like timeframe? How often [do] you need to reload the schedule, for example.

Topic: ETL pipeline tasks and open mobility data

Extracting

- ML: Next, do you have any questions for the ETL specific tasks that you have. [As] you know, ETL stands for extraction or extracting, transformation and loading data. We can go through it one by one, maybe. Do you have any questions regarding the extraction of open mobility data?
- Expert1: I would ask myself "What are the most common file formats you need to support?". And also aside from the theoretical file formats, how are they actually available in Mobilithek in the sense that it might be CSV-data, for example, but the CSV-data is encoded in ASCII, because it contains a lot of umlauts, because it's German data for example. So then you would not only need to be able to deal with the CSV-data, but also with like these CSV-data files that have a specific encoding. Additionally, the size of data I think would be interesting. So do you move into data sizes that might require you to think about performance or big data approaches? Like if it's a GTFS data set of all of Germany, maybe it's like many gigabytes big? I don't know. Do I need to authenticate in some form to get the data? Do I need to authenticate in some form to get metadata for the data? And if I need to authenticate, how does it [work]? Because all of these things we will need to model in the language so that when we execute the pipeline model, we actually do the authentication in whatever interpreter is running. So we will need to support all of these things, for example.
- ML: You mean like communication protocols, for example, as well, right?
- Expert1: Yeah, communication protocols. Do you need to follow, for example, some APIs? If you're looking at APIs, a way you need to crawl them where you get just a limited amount of results and then you need to get to the next page. Sometimes they have like page tokens, sometimes you set a different offset and stuff like that. So how would you actually crawl the API? I think with a little stuff I've seen from Mobilithek, you might need to find the location of the data using Mobilithek, but then still follow like some external link and maybe the portals that are behind there have some again a specific way I need to get to the data. That it would be interesting what kind of portals there are and what kind of. Like if 80% of the data is on the portal of Hamburg and that one has a specific requirement, it would be interesting to know. Like that [one] is the most common one, for example.
- ML: "Accessing the data" may be a nice term for it as well.
- Expert1: Yeah, exactly.

Transforming

- ML: Alright, then maybe towards the transformation. Do you have any questions for transforming open mobility data in the context of ETL pipelines?
- Expert1: Yeah. I think new for us is how. . Because mobility data is geographic, there's a geographic aspect to it and I've often heard working with it, you kind of build filters based on geofencing. So it might be interesting to include that and how much of that is actually relevant and how that works, because it can also be part for example of CSV-data, I think, with longitudes and latitudes as data points. And then it might be interesting to know if that comes up often. How often, for example, that is in there and how often you might need to filter for that. Very interesting. In general, that is maybe a bit of a generic topic, but "What kind of errors are the most common?" Like. . . "What kind of style of error is the most common?" Let's say the CSV-data can often not be parsed automatically maybe, because it contains the wrong line that isn't like the CSV-data spec. I have seen that a lot with the Destatis exports, because they include a copyright footer, I think, often which you technically don't need to do. We already talked outside of this about it, but you can export it in perfect CSV, but I keep seeing these not correct files and if like the CSV is the most common data format and for example, 80% of them have this error line, it would be interesting for us to know that you have to remove this. Common errors I can think about or transformations might be needing to

3

4

flip longitude and latitude numbers. I've also got that from local feedback here in France. There's a lot of times they run into the issue that people use the value for longitude in the latitude field and the other way around. So everything that should be in, for example, Paris is somewhere in Africa, because it's like flipped on the globe. Detecting it might be interesting. Do we need to detect it if you deal with German GTFS data? Should there be a warning if all the data points are in Africa, for example? And a way to like clean it up would be interesting. If there are any interesting value types - also very interesting. So there might be things like ids for transportation types or something, you know, if you think of enumeration in computers science or in coding what kind of enums exist in the world of open transport data.

- ML: And maybe like resolve them to like a full human-like/human-readable string that maybe humans [find] easier to understand.
- Expert1: Exactly, also in general, what to do with it is, I think, the next level step. One important thing is to actually find just the different ones, so we know how they look, because it is a difference if they are, for example, numeric versus just strings or something. Just so we know what we need to be able to deal with. And I think, I'm not sure, I think that was for metadata in Mobilithek, but they had already some categorization depending on some catalog of values. Stuff like that would be interesting.
- ML: So an additional data source for resolving in encoded terms.
- Expert1: Yeah, I'm not sure we need like an additional data source. My point is more [that] there's a concept for domain modeling that could be the concept of different value types where you say for example, you have a value that you can can consider a number, but you might also consider it Postleitzahl like a German postal code, right? And then because you consider a postal code, it is more specifically modeled in that domain and then you're able to make more checks on it. So it has to have five numbers. You know, if it starts with a nine or something, it's probably in Berlin and so on or in Bavaria. I'm not sure. I think zeros are Berlin. Point being this might also exist in open transport data. For example, the type of transport might be some European-given enumeration of train, taxi and so on. And then it would be interesting to know "Okay, this vocabulary by the European Union, for example, is used in 10% of all data sets". Then we know that "Okay, this is something you must be able to express in the language", right? That would be a clear requirement for moving on from that - just before I forget it - for accessing data, it would also be very interesting what kind of metadata exists giving the licenses, for example, for data sets and how you would access this metadata, if you access like one block of data.
- ML: Like a data set. . .
- Expert1: Yeah, you know what kind of metadata you get added to it and what is often there.

Loading

- ML: Alright. Then we come to the last step of the ETL pipeline - loading open mobility data. Do you have any questions about that? Maybe [this] intertwines with the with the kind of data you mentioned in transformation already. . . That maybe it's different with the way how you have to load it into a database, for example, I suppose.
- Expert1: Yeah, I mean it is kind of interesting in that sense that it's not really only related to the loading step, but kind of where you can end up with with the data. Like if it's graph-based data, it might be very hard to save in a relational data schema. So just the structure of data, if it is relational, graph-based or like a document store or something. I think it would be interesting to kind of infer requirements for what kind of things you need to support based on the type of data you get from the mobility data. One last thing for transformation - actually a big point probably - is data cleaning and getting rid of errors or like fixing empty values/missing values. So for that sense, it is also very interesting to know what kind of values are commonly missing and how you might interpolate that.
- ML: Or like how to deal with it in general. Interpolating is probably one solution.
- Expert1: Yeah, exactly. Like if a value is commonly missing and it's very easy to, for example, assume something that would be interesting. But also if your value is commonly missing, it's just good to know what kind of value that is.
- ML: Okay. Yeah, I guess then that's it also for loading open mobility data or do you have something to add?
- Expert1: No.

Topic: Mobilithek

- ML: Okay. Then do you have any specific/particular questions for Mobilithek since I'm working with that? Mobilithek is a provider for open mobility data. Do you have any questions about it?
- Expert1: I mean, I would say in general, we should have most questions based on Mobilithek as an example for open mobility data. From the mFUND project we are doing - I wrote down to mention this - I would be interested in the quality of data on it, the formats of data on it, what kind of metadata exists, compatibility - which I'm not sure what that means in that context, but I'm going to put it out here - and rights, so licenses and stuff. So I think these are very interesting questions to answer and in part, very easy ones, like quality - probably hard, but formats. What kind of metadata exists and what licenses are available is probably easy to answer. So those would be very interesting for Mobilithek.
- [SHORT INTERMISSION]
- Expert1: I have some last minute things I just thought about. Because we only talked about domain experts, I think it's kind of interesting to know

if there are different large domains in mobility data as well. Like there might be public transport and infrastructure and car like highways and stuff like that and those might be different domains inside of this larger domain. And I think that it's for now.

- ML: Basically just like the domain itself. What is like particularly interesting about mobility, I guess, right?
- Expert1: No, especially like what kind of subdomains maybe exist? Like I saw stuff like "How many parking spaces are available in some region?", for example. That is very different data to schedule data for trains, for example. So one is private transport with cars, more geographical data and the other one is maybe more scheduling data. So kind of different fields even. It might be interesting to just have an overview of what kind of different fields exist even on mobility, so to speak.

Topic: Further questions

- ML: Do you think I have anything forgotten? That [there] is something to add, some kind of question I should have asked you about the whole topic?
- Expert1: Yeah, I think one of the things is functional/non-functional requirements of data pipeline DSLs. Just asking about these two things, like just straight up. Do you have any questions about what kind of requirements might be there functional or non-functional? Which I don't. So I just bring that up, because I think that's relevant in general.
- ML: About any DSL probably.
- Expert1: Yeah, exactly about any, for example, I would put out the one that it should be extendable by domain experts without changing their language. So they should be able to model their own data types and model their own value types like what kind of values they have and what kind of semantics are in there. I would say that is maybe a functional requirement for that language which, I think, is generically useful.
- ML: Alright. Well, then thank you very much for taking your time and going through these questions with me and I hope we'll have some further discussions later on. Thank you very much.
- Expert1: You're welcome.

Interview with Expert2

Summary

- Functional/non-functional requirements of data pipeline DSLs
  - Two major paradigms in pipelining: unbound and bound data or streaming data versus batch processing
    - ∗ Which is more relevant?
    - ∗ Can you depict both with one language design?
  - Support for sources/transformations?
  - Can we develop a DSL that doesn't need user-defined functions, but still is holistic enough to depict 90% of use cases? (Availability to non-coders)
  - Non-functional requirements
    - ∗ How relevant is it that execution of pipelines is fast? Performance time critical
      - · Big data that does not fit the memory?
      - · Streaming approach feels natural.
- Open mobility data different to the general open data?
  - Real-time data interesting for main users
  - Uniformity in domain-specific standards or de-facto standards?
    - ∗ Examples: repeating data formats or units; MP/H or KM/H
- Questions regarding extracting open mobility data?
  - Which data formats/protocols are used?
  - Which schemas are there?
  - Is there something very standardized? Is there something really off?
    - ∗ Example: CSV-file with multiple tables
  - Metadata inside or outside
  - Traffic limits on Mobilithek?
  - What kinds of data is available?
- Questions regarding transforming open mobility data?
  - Repeating flaws in the data - cured by repeating transformations?
  - User-defined functions needed?
  - Joining multiple files? Also data in non-machine-readable form? (parse description)
- Questions regarding loading open mobility data?
  - Amounts of data? Resulting load for cloud-based ETL service?
  - Specific tooling in open mobility data domain?
- Questions on Mobilithek?
  - How much data is hosted elsewhere and how much hosted on Mobilithek?
  - Metadata schema? Metadata useful for automation tasks?
  - Reliability of service; Downtimes? Rate limits?
  - Publicly available data versus authentication-gated data
- Further questions?
  - Question about functional/nonfunctional requirements could be phrased better.

1

Transcript (manually post-processed)

- ML: Thanks Expert2 for spending some time with me and my questionnaire for my master thesis. I want to create requirements for DSL domain specific language that's already existing more or less and we want to expand it in regards to open mobility data. So I prepared a few questions for you and maybe you can give me some concepts, some questions that are in your mind regarding the whole topic. First off: hello and again thanks for spending some time with me.

Topic: Functional and non-functional requirements

- ML: I have some sections there and the first is regarding data pipeline DSLs in general. Out of your head. What comes to your mind? Do you have any questions what kind of requirements might there be regarding functional or non-functional nature?
- Expert2: Well yeah, I have a few questions in mind. I think there are two major paradigms in pipelining: unbound and bound data or streaming data versus batch processing. So I wonder which paradigm is the most relevant or if you can depict both with one language design kind of. . . That's one of the big questions. And then obviously, I think in pipelining or ETL in general: it's all about the sources, sinks and transformations you support. So I think adoption of such a technology always comes and goes with the adapters to provide kind of and the possibilities for cleaning your data. And usually I'm aware that most ETL technology provides some kind of user-defined function mechanisms where you can write your own code to transform data, manipulate it and maybe also read or write it from sources and sinks. So big question in my mind: Can we develop a DSL that doesn't need user-defined functions, but still is holistic enough to depict 90% of use cases kind of.
- ML: Alright.
- Expert2: Another question I have in mind regarding non-functional requirements - and I think that heavily depends on the domain of data you're using - is how relevant is it that execution of pipelines is fast, so how time critical is the data and in the same way how important is parallelization which probably depends on which kinds of/loads of data you're dealing with, because you need to break it down to process it, kind of.
- ML: Sounds like you're having live data in mind, for example, where you get like a continuous stream of data and you have to process it. Or in time. maybe if you have to combine several data sources at the same time that probably is what you have in mind with the parallelism, I guess, right?
- Expert2: Yes, among others. I mean if you're facing big data chunks of one terabyte or so - I don't know how often we would deal with that to be honest, but if we do, we can't process it all at once by writing into memory. Probably depends on the machines you're running, but there will be an upper limit and probably you will need mechanisms to cope with

2

that. And I think the streaming approach feels natural to do that actually, because it also solved the problem by slicing it into smaller chunks.

Topic: Open mobility data vs. open data in general

- ML: Alright. Yeah, that's it actually for my first question. I think that kind of shoots towards the whole topic and probably we could go a bit more in detail with the next questions. Do you expect specifically open mobility data to be different in some ways, to have different kind of qualities when it comes to the comparison with general open data, maybe?
- Expert2: Yes and no. So I think there are certain aspects that might be different. For example, I could imagine that especially real-time data is interesting for the main users. Getting real time updates on, I don't know, traffics or trains or something might be data that is especially valuable for users, but that might also be just in most open data domains be the same so. . .I'm not sure if that is really differentiating. What I would expect however is that there are some domain-specific standards or de-facto standards, maybe they are not agreed on entirely, but majorly used. So probably some data formats that are most often used and also repeating data units like for example a domain agrees on using MP/H or KM/H or something like that. So maybe there's some uniformity there.

Topic: ETL pipeline tasks and open mobility data

Extracting

- ML: Alright. When we come to the ETL pipeline and the whole tasks that are involved there, we have extracting and transforming and loading data. I would go through these maybe one by one. Since we are dealing with open mobility data, do you have any questions for the extraction, maybe, extracting mobility data?
- Expert2: Yeah, obviously the obvious ones like which data formats are used, which protocols are used to pull them. Which schemas are there? Is there something very standardized? Is there something really off? Like yesterday I faced CSV-file. There was not one table, but multiples, for example. Is that the thing in that domain, how much metadata is in that file actually or is it handled outside? Or is there some weird domain specific ideas, for example, that carries some semantics? But also extraction has to do with the data source itself. So if you said like the Mobilithek has no traffic limits or so. So that would be a question if that is a general thing. And I also remember from some interesting talk about statistics with Deutsche Bahn where some developer collected data over a whole year and then some statistics about delays and all that stuff.
- ML: I think you're referring to the talk of, was it David Kriesel, I think? There he analyzed things in the CCC congress.
- Expert2: Yes, exactly. And I remember that he had to. So I think he didn't have an opportunity to get the historic data, but he rather [had] to crawl it regularly. So it's always a question which kind of data is available regarding archives for example.

Transforming

- ML: Okay. Then the next step of the ETL pipeline is transforming data. Maybe it involves some of the points you mentioned just now. Since I think strictly separating these tasks and topics and problems that come with is not always possible. But does anything come to mind to you talking about transforming mobility data or open mobility data?
- Expert2: I mean always interesting is if there are really repeating flaws in the data that might be cured by repeating transformations for example. And combined with that, if there are very unpredictable things, cleaning steps. . probably you would use user-defined functions in some way. So that's a big question in my mind. Do we need that for to cover that domain? And also - might be very related to the data sources - usually data is distributed to different sources or files and make value by joining them. So a classic one would be with this idea. Maybe there's context information in some other file or so. Then you would need to join it somehow. If we need some of these specific operations/transformations to get value out of the data and maybe there is also some semantic that cannot be found in machine-readable code? Like we would have to get it into the program itself hardcoded, maybe, I don't know.
- ML: Maybe like a description in the metadata and it says like "Hey you have this and that here", but it's like in human-readable form.
- Expert2: Yes, yes. Yeah, so that's I think what is very unclear to me from [the] transformation side.

Loading

- Expert2: From [the] loading perspective. So in the end we want to operate that language somewhere in our cloud for that's, that's how a startup might make money. So obviously a very limiting factor is with what amounts of data are we dealing? With what amounts of load do we have to calculate? Because in the end of that some kind of function, but also for example, if you want to archive the data somehow. And also interesting is if data scientists and data engineers in that domain use some fancy sync technology, I don't know. If they use SQL/SQLite or again CSV? I don't know. Maybe there are some specific tooling in that domain that they want to use that needs different input formats and that kind of things, I think.

Topic: Mobilithek

- ML: Alright. I mentioned it earlier. One of the examples of a data provider for open mobility data is Mobilithek, the website by the german government.

Since I'm also looking at that and the data that is being offered there. . . Do you have any questions about Mobilithek?

- Expert2: Yeah, also I had a brief look into it, did a few searches. And . . . it seemed to me like they're more a portal that points to other places in the web. So a question for me would be is that always the case or do they also host data on their system? And how much on the other side is just the metadata? And obviously when speaking about metadata, what is the common schema for that? Can we use that kind of to automate, for example. . . automatically generate projects on our side? I also think I saw. . . like they have a topic/a project with multiple links at the bottom where it points to multiple files. [It] wasn't entirely clear to me that . . separation, because it felt very confusing to me like having six links and they all point to different things then. And how they are intertwined in the end? So it would be interesting if that is somewhere in the metadata, I think. And apart from that it would be quite interesting how reliable their service is. Like you already said. Do they have rate limits? Probably not. Also downtimes, can they deal with load? But I'm not sure if that's actually in scope of your thesis, but it's a general question I have in mind.
- ML: Yeah, sure. It's also interesting to me, but, yeah, I don't know if it's part of my research. But [it's] certainly something that's interesting, yeah.
- Expert2: Another thing I said. . . I think they have like openly available data, but also kind of enclosed data depending on a - I'm not sure - paid plan or at least you have to register for it. So it would be interesting what differentiates the data that you need a plan for, registration for, from the ones that are publicly available.

Topic: Further questions

- ML: Yeah, definitely. Alright. Then we come almost to an end of my questionnaire. Do you have any other topics or maybe questions that are missing, you think, on the questionnaire? Just anything that you felt like that needs to be said?
- Expert2: Not really, I think the first question I had to wrap my head around first, because of the phrasing. Like, "Do you have any questions about kinds of requirements?" Yeah, but apart from that I feel like everything important to be covered.
- ML: Alright, then I'll stop the interview here. Thank you very much for your time and yeah we'll probably speak to each other in context of my master thesis, again. We'll see. Thank you very much.
- Expert2: Sure, you're welcome.

Interview with Expert3

Summary

- Open mobility data different to the general open data?
  - Expectation of more live data
  - Mode of processing of JValue pipeline? multiple iterations, repeatedly or continuously?
  - Typical file formats in mobility domain?
    - ∗ GTFS data (a zip containing multiple standardized CSV-files)
    - ∗ Maybe more file formats?
  - Typical data structures in mobility domain?
    - ∗ Graph-like, e.g. maps with places on the map
  - Typical value types in mobility domain?
    - ∗ Distances or GPS coordinates
- Questions regarding extracting open mobility data?
  - Common protocols for retieving data
    - ∗ Live data
  - Distribution of file formats
  - Structure of data: tabular, graph-like
    - ∗ Derivable from file extension or format?
  - How to treat live data?
- Questions regarding transforming open mobility data?
  - What is the goal/purpose of transforming data?
    - ∗ Data cleaning (proper format at the end)
    - ∗ Normalizing (different data sets easier to combine)
  - What transformations are there?
  - Different formats for one thing
    - ∗ Decimal separator: point or comma
  - How to combine data of different structures using transformations?
    - ∗ CSV-file from GTFS and graph-like from some other file format: Mangle together
- Questions regarding loading open mobility data?
  - What purposes of use? (Intended purposes)
  - What kind of sinks?
    - ∗ Suitable sink for a particular data structure
      - · Database suitable for tabular data, but not for graph-like data
      - · Finding these matches (or even mismatches)
      - · E.g., Journalist that wants an Excel file or diagram
- Questions on Mobilithek?
  - Distributions of file formats on Mobilithek?
    - ∗ Focus on common file formats
  - Live data as in frequently updated data or continuous data streams on Mobilithek?
    - ∗ How provided to the user?
  - Which licenses (and user rights) are used for entries in Mobilithek?
  - Availability of the platform?

1

    - ∗ Previous experience with gov data: Links broken or data not downloading
    - ∗ Caching options if files inaccessible?
- Functional/non-functional requirements of data pipeline DSLs
  - Focus on experts in the field of mobility
    - ∗ Collaboration capabilites
    - ∗ Experts being able to read/understand DSL code
  - Concept for visualizing data
    - ∗ E.g. a particular for rendering diagrams/maps
    - ∗ Useful for nontechnical users
      - · Interested in mobility data
      - · Missing the technical background to deal with databases and file formats
  - Generation of project page with visualization
    - ∗ Gives an idea what to expect from the data
- Further questions
  - None

Transcript (manually post-processed)

- ML: Hello dear Expert3. Thank you for spending some time with me and my questionnaire regarding my master thesis the requirements for an open mobility data processing language or an extension of a pre-existing one, so to speak. So I prepared some questions and you as an expert in the in the field of data pipelines and member of the JValue project could answer them. First off: hello and thank you again.
- Expert3: Hi, thanks for having me.

Topic: Open mobility data vs. open data in general

- ML: First, I wanted to ask you in general since I deal with open mobility data and it is different in in some qualities from general open data. Do you have any questions? Do you expect maybe open mobility data to be different in any way in our context with data pipelines with PTL pipelines? If yes, how do you think is it different?
- Expert3: Okay, so one thing I can imagine is that there's more kind of live data involved. Like the data is provided as it is updated or maybe sensoric data that is published or something like that. And that way we, in order to process it, we need to either run the pipeline in multiple iterations or repeatedly or continuously. So that probably is one big difference. As far as I know, there are also particular file formats that occur mostly in mobility domain. There's like GTFS data for example which is like a zip-file which contains standardized CSV-files so that maybe there may be more, I'm not sure about that, but that's also something that's special for mobility. Maybe as mobility data is like like a subset of the general open data that's out there. I think that it could be that there are particular

2

data structures that occur more often than others, like maybe a graph-like data when thinking about like maps and places on the map and maybe that's described like a graph and not like like a table or hierarchy or something like that. And also particular value types may occur more often like distances for example or GPS coordinates. So that's what I think.

Topic: ETL pipeline tasks and open mobility data

Extracting

- ML: Okay, yeah. For a different kind of section I wanted to go into more specific tasks. We have an ETL pipeline or ETL data pipeline. If you recall there is ETL as an acronym for extract, transform and load. And I thought, maybe we structure the questions like so. So you have any questions specifically regarding open mobility data about the extracting process?
- Expert3: Yes. So, it would be interesting to know what are common protocols for retrieving data. Well, that's the same as in general with general open data or maybe there are some specific protocols that are maybe also specific for live data. Then also distribution of file formats, like whether there's mostly CSV as it's in general or maybe there are some other file formats that are occurring more often. And then maybe also there are mobility-specific file formats that only occur in this particular domain. And also how data is usually structured or maybe most of the time structured. Whether they are tabular- or graph-like or have some other kind of hierarchy. And how that relates to the file format - whether we can derive the structure from the file format or the file extension. That would be interesting. And also how live data would [need to be treated] differently compared to just loading a static file and processing it. How that would interfere with each other or how it can be done in parallel together or I don't know.

Transforming

- ML: Yeah. The other letter T is for transforming - the transformation step of an ETL pipeline. Do you have any questions regarding that?
- Expert3: Yeah. So first of all, what are the goals of transforming data? When a user wants to use mobility data and needs to transform it for its use case. "What is usually the [goal/purpose of] the transformation? Maybe data cleaning, so it's in a proper format at the end or normalizing that different data sets can be combined easier, so they match together. Or even the combining itself may be a transformation that is part of the pipeline. So that different data is combined to single output so to say. And what particular transformations would be required to get there? What transformations would be necessary to clean mobility data or to normalize mobility data? What particular transformations are there?

- ML: There might be like missing values, but also like maybe faulty values, something like that?
- Expert3: Yes, exactly. Or maybe differently formatted, but meaning the same. In Germany, we have decimals with comma and in other countries they use dot and then it means the same number, but it's written differently. But maybe we need to normalize it, so we can combine. And also how we can combine data of different structures. We have like a CSV-file from GTFS and we have a graph-like structure maybe from some other file format and how we can mangle those together, how that is possible using transformations.

Loading

- ML: Very well. That leads me to the last task we have on the ETL acronym: loading open mobility data. So after transforming you probably want to transfer the data, I guess, into a database, for example, or some other data structure maybe, right? Do you have any questions for open mobility data being handled in loading?
- Expert3: Yes. So mostly what the purpose of uses are for mobility data and which kind of sinks are specifically suitable for that purpose. Imagine a journalist wanting to use mobility data and maybe he [doesn't want to] have to store in a database. He rather wants maybe to have an Excel file out of it or maybe even a diagram where you can see the data visualized that even may be a map where you have like dots on the map that visualize the data. So the question would be "What purposes of use are there among different users and which sinks are suitable for which purposes?" And so whether a particular sink is only suitable for a particular data structure. So maybe the database would be especially suitable for tabular data, but not for graph-like data or something like that. So that maybe work out those matchings and which sink suits which data so to say.
- ML: Yeah. Also more or less off the record. There's also the open question if the data structures that are available if they are good enough or suitable enough maybe for the data to be fit in all the eye. If you want to combine maybe images to like a database or whatever included, maybe there is no structure for it.
- Expert3: Yeah, that's right. Maybe mismatches between data format and the sink. Those have to be known. So that the DSL can then say "Here, that that sink doesn't match this format" or something like that.

Topic: Mobilithek

- ML: Okay, thank you very much so far. As you know, I'm dealing with Mobilithek which is a provider for open mobility data. There you have listings of data offers, things you can browse. I noted it down briefly on what it is on the handout [that] I sent you earlier. But do you have any questions about it? What are you curious about Mobilithek as a provider for open

mobility data?

- Expert3: So it would be interesting to know the distribution of file formats, so we know what to prioritize what to do first in our language. So for instance if there are most of the files are CSV then we should obviously focus on them.
- ML: Probably likely to be expected, right? CSV is very dominant.
- Expert3: Yeah. In general open data it's very dominant, but I'm not so sure about mobility data, but I also think it's quite dominant there as well. Also whether the Mobilithek provides live data like frequently updated data or maybe also continuous data streams and how those are provided to the user? About licenses, so like which different licenses are used in the Mobilithek and maybe also the different rights for the user. And also the availability of Mobilithek. When using gov data for example, it's another problem for more like general open data, I often had the problem that either links were broken or that the data wouldn't download. It just would time out. So maybe it would be interesting to know whether how good is the availability of the platform and how we could deal with it in case it's not available. Maybe caching a file when when repeatedly running a pipeline. So in case that the provider that doesn't provide the file, then we could maybe use a cache or something like that.

Topic: Functional and nonfunctional requirements

- ML: Alright. We are almost at the end. For the rest I have some questions that target further things for the concepts/ideas that you may have. Do you have any ideas for more or less concrete requirements regarding DSLs (domain specific languages)? It's okay if if you don't have anything, but I just want to make sure that I capture all of your input.
- Expert3: Yeah, I got two. What I think is pretty important is that experts in the field of mobility are able to understand code that is written in the DSL. Especially for collaboration purposes, that different mobility experts can work together to describe a pipeline and that the source code can be [understood] by the people that are collaborating. And also, I mentioned it earlier, having a concept maybe also really embedded in the DSL for visualizing data, like having a particular sink that is rendered as a diagram or rendered as a map, which which may may not be compatible with all kinds of data, but maybe you can transform it into that structure and then have it rendered as a diagram. That would especially be useful for nontechnical users that are interested in mobility data, but don't have the technical background to really deal with databases or other file formats that may be produced by the sinks.
- ML: So you're thinking of like a pre-rendered map with map tiles and maybe the data is visualized there, for example?
- Expert3: Yeah, you could imagine a table where each row represents an address and then you could take the table and visualize it on a map. Like you have the global map and then each address in the table marks like dot on the map. Or maybe you can also draw circles where many addresses are and smaller circles where not so many addresses are. So like quantitize them. That would be maybe an interesting use case.
- ML: Yeah, in that regard probably could also extend it maybe to plotting things, maybe have a distribution graph or a line plot or something like that. . . a bar plot.
- Expert3: Yeah, absolutely. So maybe there are also types of diagrams that fit mobility data better than other domains, but I [imagine] the map, but maybe there are more. But I think that's not so important for the beginning of the language, but on the long run it would be really nice to have like project pages with like or maybe you can provide diagrams from the pipeline model and then the user knows what data he can expect from the pipeline.
- ML: Yeah, very interesting certainly.
- ML: I have a last question regarding functionality: Do you have any other questions? Maybe some that you feel were left out on my handout regarding specifically functional and nonfunctional requirements.
- Expert3: I don't have a point here, so I don't think there's anything to add.
- ML: Alright. Okay, thank you very much for again spending the time with me and working through all of my questions.
- Expert3: Sure, thanks for the invitation.
- ML: See you soon.
- Expert3: See you soon.

5 6

E Bill of Materials (BOM) for Implemented Code Documents

This bill of materials (BOM) concerns Python packages used in code-related documents that were implemented as part of this work. This relates to the two API crawling scripts and in the Jupyter Notebook documents, Data Preparation, Exploratory Data Analysis and Investigative Data Analysis. Python itself is licensed under the Python Software Foundation License (PSFL), a BSD compatible license.

E.1 API Crawling Scripts

The following output is created by Miscellaneous/generate_BOM_API_crawling.py, which utilizes the Python package pip-licenses (licensed under the MIT License).

PYTHON:
3.10.8

PACKAGES:

Name	Version	License
aiohttp	3.8.3	Apache Software License
aiolimiter	1.0.0	MIT License
pandas	1.5.3	BSD License

E.2 Data Preparation and Data Analysis Documents

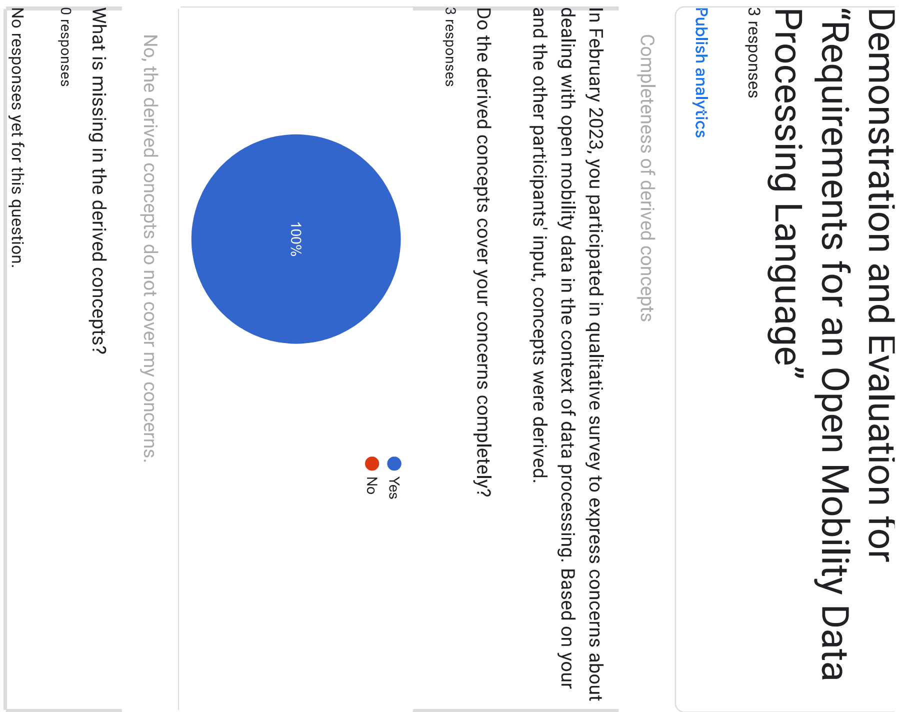
The following output is created by Miscellaneous/generate_BOM_DataPreparation_DataAnalysis.py, which utilizes the Python package pip-licenses (licensed under the MIT License).

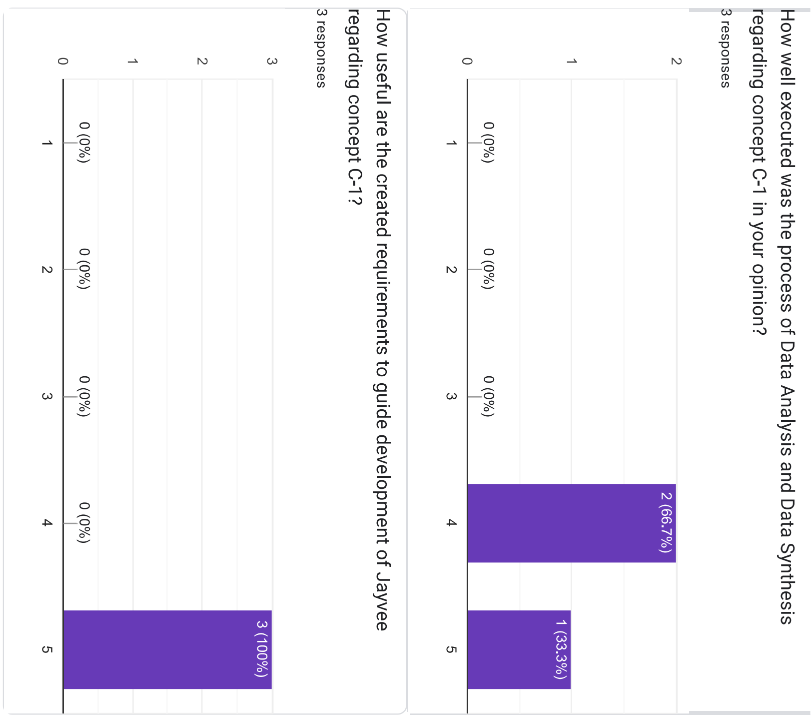
PYTHON:
3.10.8

PACKAGES:

Name	Version	License
ipython	8.11.0	BSD License
joblib	1.2.0	BSD License
jupyterlab	3.3.4	BSD License
matplotlib	3.6.2	Python Software Foundation License
numpy	1.24.2	BSD License
pandas	1.5.3	BSD License
seaborn	0.12.2	BSD License
shapely	2.0.1	BSD License

F Evaluation Survey Results





If there is anything else you want to comment on concept C-1, please do so:

3 responses

Going by mediaType is fine to classify the file type, but could be further improved by further measures (and a measure of triangulation by different factors). For the thesis absolutely sufficient but I can see some further improvements - thus only 4 out of 5 (because the question is more general and not scoped to your thesis).

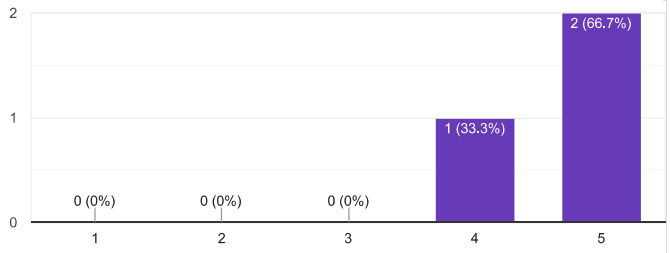
If I understood the process correctly, the media formats were mostly derived by the metadata provided by Mobilithek. A few further thoughts, probably out of scope for the thesis:
- Download actual data and compare its format with the one stated in the metadata
- Regarding the ZIP format, extract archives and see which kinds of files they contain
- Further investigate media where the "Other" format was assigned

The discussion about outliers and deeper insights into the "Other" classification are well done and make sense to me.

Concept C-2: Open mobility data structures

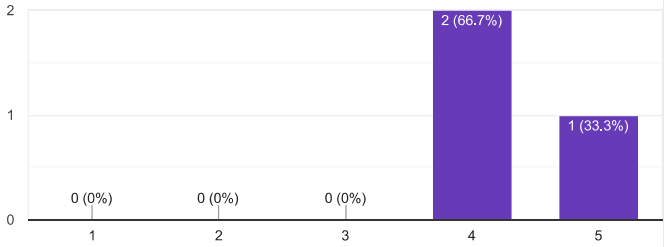
How well executed was the process of Data Analysis and Data Synthesis regarding concept C-2 in your opinion?

3 responses



How useful are the created requirements to guide development of Jayvee regarding concept C-2?

3 responses



If there is anything else you want to comment on concept C-2, please do so:

2 responses

Like you mention in your text, going by data format can roughly approximate the nature of the data structure with the assumption that the right data structure is used... I would challenge that and assume that there are CSV files that represent graph data and so on.

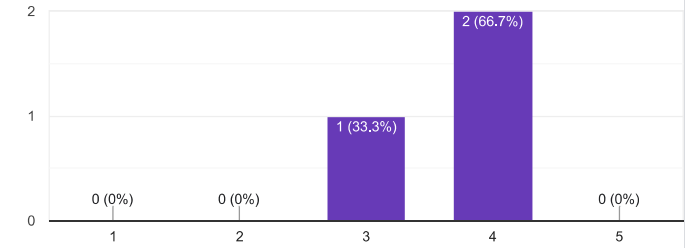
I'd say the process was well executed but I'd express some uncertainty if it can really answer the question. Again, totally fine in your thesis (I wouldn't know how to do it better), but I'd explicitly point out this limitation.

Since the data mostly derives from the file type and we know hard numbers for file types from C-1, I would have loved to see a figure or table that shows numbers for relational data vs. graph-based data.

Concept C-3: Open mobility value types

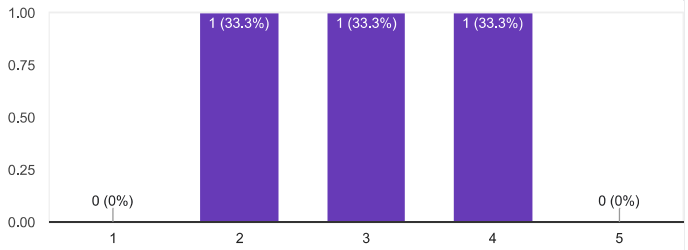
How well executed was the process of Data Analysis and Data Synthesis regarding concept C-3 in your opinion?

3 responses



How useful are the created requirements to guide development of Jayvee regarding concept C-3?

3 responses



If there is anything else you want to comment on concept C-3, please do so:

3 responses

I think your approach is a good starting point. Answering this question in the data and not only the meta-data would probably be the best approach but large enough to be its own thesis.

I subtract the usefulness by one point because an appendix with the exact data models would be more useful instead of the schema identifiers ;-)

I would have expected to see SI units here and a deeper insight to the various value types (similar to how it was done for NUTS). The required schema models (R-3-4) seem rather vague, how do such concrete values look like in the actual data?

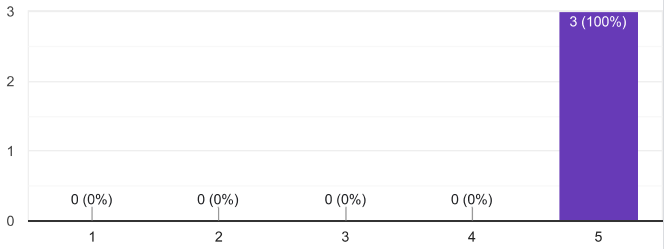
It seems like C-3 is hard to answer with the metadata based approach that was taken. The results give some insights into data types that are used with mobility data offers, but future work should explore data types as used in the actual data and not only the metadata. As it stands right now, the results can not inform language development that much.

This is fine in the context of the MA thesis of course.

Concept C-4: Data transmission in open mobility

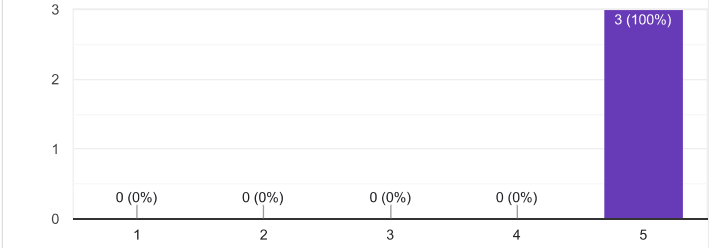
How well executed was the process of Data Analysis and Data Synthesis regarding concept C-4 in your opinion?

3 responses



How useful are the created requirements to guide development of Jayvee regarding concept C-4?

3 responses



If there is anything else you want to comment on concept C-4, please do so:

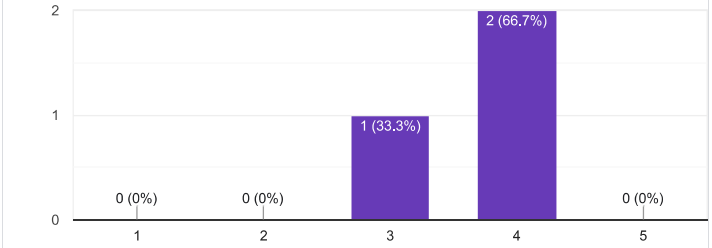
0 responses

No responses yet for this question.

Concept C-5: Live data in open mobility

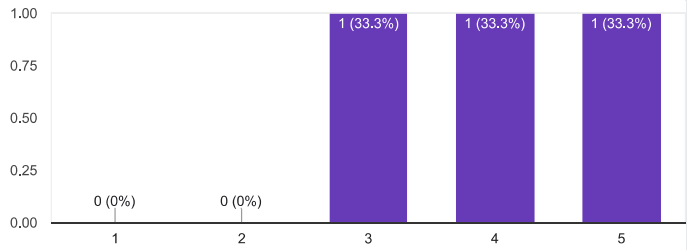
How well executed was the process of Data Analysis and Data Synthesis regarding concept C-5 in your opinion?

3 responses



How useful are the created requirements to guide development of Jayvee regarding concept C-5?

3 responses



If there is anything else you want to comment on concept C-5, please do so:

3 responses

This one is tough to answer as well... I think going the metadata way is the best thing possible for your thesis. A more elaborate way was a change detection mechanism that checks if a source got new data in a timespan X and define this as "continuous data" - or some change detection mechanism... Not scope of your thesis, though.

As a statistic, it would be interesting if the HTTP(S) sources have a last_modified or If-Modified-Since header support to check for changes.

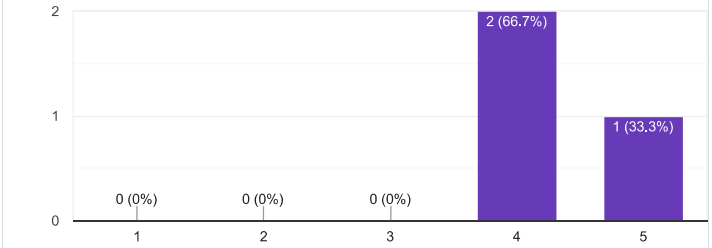
Would have been interesting to manually look into a limited number of random data offers with "MISSING DATA" to get an impression about them.

The actual requirements are surprisingly minimal (there is just one requirement), the discussion is interesting though. I would have loved to see the creation of categories of update types for data (live/streaming, on a schedule, regularly without a schedule, never...?) and then a mapping to those categories from the metadata in accrualPeriodicity. By just describing the metadata in a binary live data/no live data way, I am missing any insight on scheduling (Is polling needed? Would users define a schedule in a pipeline model?)

Concept C-6: Authentication with open mobility data portals

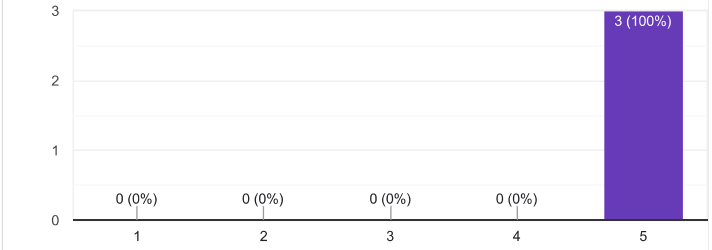
How well executed was the process of Data Analysis and Data Synthesis regarding concept C-6 in your opinion?

3 responses



How useful are the created requirements to guide development of Jayvee regarding concept C-6?

3 responses



If there is anything else you want to comment on concept C-6, please do so:

2 responses

Not sure how statistically accurate is your approach with the random sampling. But I think good enough fur us ;-)

Not surprising, but nice to have confirmation. An automated crawl of more data sets might strengthen the result (e.g., by just probing a HTTP response code and checking if it is 401/403), but might not be possible.

Final input

If you have final comments or thoughts that you want to express, please do so here:

1 response

Nice work, I love it! It gives us a good indicator what to address next.

It's obvious that you can spend countless of more hours to investigate the data (see comments), but it wouldn't make much sense for our use-case I guess. It's a pity that the collected meta-data is so incomplete - but that's open data I guess. I'd love to read and forward a "recommendation" for the mobilithek from you that worked with their API on how they can improve.

References

Bitnine Global Inc. (2016). What is the graph database? What is data model? [Retrieved May 25, 2023, from https://bitnine.net/blog-graph-database/what-is-the-graph-database/].

BMDV. (2022a). About Mobilithek – Germany's data platform that gets you moving [Retrieved January 10, 2023, from https://mobilithek.info/about].

BMDV. (2022b). FAQ general [Retrieved May 24, 2023, from https://mobilithek.info/help/FAQ].

BMDV. (2022c). mobilithek - Technical interface description [Retrieved January 4, 2023, from https://mobilithek.info/cms/assets/369959e2- 8ae0- 483f-8ce5-3d41aeb5d846?download].

BMDV. (2022d). Datenplattform: Die Mobilithek geht an den Start [Retrieved January 10, 2023, from https://bmdv.bund.de/SharedDocs/DE/Pressemitteilung 2022/043-mobilithek-geht-an-den-start.html].

Buchalik, M. (2022). *Design and implementation of a version control system for open data modelling projects* [Master's thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg]. https://oss.cs.fau.de/2022/10/14/final-thesis-design-and-implementation-of-a-version-control-system-for-open-data-modelling-projects/

Cao, L. (2018a). Data Science Thinking, 238–239. https://doi.org/10.1007/978-3-319-95092-1

Cao, L. (2018b). Data Science Thinking, 177, 323. https://doi.org/10.1007/978-3-319-95092-1

Destasis (2015). NUTS classification - German Federal Statistical Office [Retrieved May 24, 2023, from https : / / www . destatis . de / Europa / EN / Methods/Classifications/Overview_Classification_NUTS.html].

ESRI. (1998, July). *ESRI Shapefile Technical Description* [Retrieved May 28, 2023, from https : / / www . esri . com / content / dam / esrisites / sitecore - archive/Files/Pdfs/library/whitepapers/pdfs/shapefile.pdf]. ESRI.

EU EIP. (2020a, July). *napDCAT-AP – A DCAT-AP extension for Metadata in National Access Points* [Retrieved May 24, 2023, from https : / / github . com/EUEIP/napDCAT-AP/tree/master/Specification/Version0.8]. EU EIP.

EU EIP. (2020b). napDCAT-AP Vocabularies [Retrieved May 25, 2023, from https://eueip.github.io/napDCAT-AP/vocabularies/].

European Commission. (2015, October). *DCAT Application Profile for data portals in Europe Version 1.1* [Retrieved May 24, 2023, from https : / / github . com/SEMICeu/DCAT-AP/raw/master/releases/1.1/dcat-ap_1.1.pdf]. European Commission.

European Commission. (2017). About DCAT Application Profile for data portals in Europe [Retrieved May 25, 2023, from https://joinup.ec.europa.eu/collection / semantic - interoperability - community - semic / solution / dcat - application-profile-data-portals-europe/about].

European Commission. (2021a). Action Plan and Directive [Retrieved February 9, 2023, from https://transport.ec.europa.eu/transport-themes/intelligent-transport-systems/road/action-plan-and-directive_en].

European Commission. (2021b). National Access Points [Retrieved February 10, 2023, from https://transport.ec.europa.eu/transport-themes/intelligent-transport - systems / road / action - plan - and - directive / national - access - points_en].

Eurostat. (2014). Background - NUTS - Nomenclature of territorial units for statistics - Eurostat [Retrieved May 24, 2023, from https://ec.europa.eu/eurostat/web/nuts/background].

Fowler, M. (2010). Domain Specific Languages (1st), 27.

GovData. (2022, February). *DCAT-AP.de Spezifikation 2.0* [Retrieved May 24, 2023, from https://www.dcat-ap.de/def/dcatde/2.0/spec/]. GovData.

Jansen, H. (2010). The Logic of Qualitative Survey Research and its Position in the Field of Social Research Methods. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, *11* (2). https://doi.org/10.17169/fqs-11.2.1450

Kimball, R., & Caserta, J. (2004). The Data Warehouse ETL Toolkit. Wiley India Pvt. Limited.

Kleppmann, Martin. (2017). Designing data-intensive applications.

McKinney, W. (2022, August). *Python for data analysis* (3rd ed.). O'Reilly Media.

NIST/SEMATECH. (2012). Engineering Statistics Handbook, Ch. 1.1.1. https://doi.org/10.18434/M32189

Open Geospatial Consortium. (2019, August). *Well-known text representation of coordinate reference systems* [Retrieved May 26, 2023, from http://www.opengis.net/doc/is/wkt-crs/2.0.6]. Open Geospatial Consortium.

Open Knowledge Foundation. (2018). The Open Data Handbook [Retrieved April 1, 2023, from https://opendatahandbook.org/guide/en/].

Open Knowledge Foundation. (2023). Open Definition [Retrieved April 1, 2023, from http://opendefinition.org/].

OSS FAU. (2015). About – The JValue Project [Retrieved January 5, 2023, from https://jvalue.org/about/].

OSS FAU. (2023). Introduction to Jayvee [Retrieved May 2, 2023, from https://jvalue.github.io/jayvee/docs/user/intro].

Peffers, K., Tuunanen, T., Rothenberger, M., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, *24*, 45–77.

Schneider, P., & Koska, T. (2023). Mobility Data for a Just Transition: The Case for Multimodal Platforms and Data-Driven Transportation Planning [Retrieved June 28, 2023, from https://www.boell.de/sites/default/files/2023-06/e-paper-mobility-data-for-a-just-transition-endf2.pdf], 5.

Völter, M., Benz, S., Dietrich, C., Engelmann, B., Helander, M., Kats, L. C. L., Visser, E., & Wachsmuth, G. (2013). DSL engineering - designing, implementing and using domain-specific languages. dslbook.org: http://www.dslbook.org

W3C. (2020, February). *Data Catalog Vocabulary (DCAT) - Version 2* [Retrieved May 24, 2023, from https://www.w3.org/TR/vocab-dcat/]. W3C.

Wagh, S. J., Bhende, M. S., & Thakare, A. D. (2021, August). Fundamentals of data science. Chapman and Hall/CRC. https://doi.org/10.1201/9780429443237

Webster, J., & Watson, R. T. (2002). Analyzing the Past to Prepare for the Future: Writing a Literature Review. *MIS Quarterly*, *26* (2), xiii–xxiii. Retrieved March 17, 2023, from http://www.jstor.org/stable/4132319