Humboldt-Universität zu Berlin
Philosophische Fakultät
Institut für Bibliotheks- und Informationswissenschaft

Modelling and Automated Retrieval of Provenance Relationships

Masterarbeit im Rahmen des Weiterbildenden Masterstudiengangs
Bibliotheks- und Informationswissenschaft im Fernstudium

vorgelegt von
Thomas Schneider

Gutachter:
Prof. Dr. Robert Jäschke
Christian Rüter

Erfurt, den 14. Juni 2023

Contents

List of Figures

1 Introduction

1.1 Background

Provenance research is concerned with the origins and ownership history of cultural objects. Its main objective is the reconstruction of "object biographies" in the historical context. Application areas include the study of private and public collections, the detection of forgery, and the identification and restitution of loot. Since the release of the *Washington Conference Principles on Nazi-Confiscated Art* [Web1] in 1998, provenance research has received increased attention.[1]

Regarding the holdings of university and research libraries, particularly interesting provenances are those related to the change of owners when an item such as a copy of a book is passed on or distributed. Provenances can be reconstructed on the basis of *provenance marks* that are present in the item and which indicate ownership, such as stamps, bookplates (ex libris), or handwritten signatures. Provenance marks are essential for retracing the "history" of a (book) item or the extent of library holdings that have been scattered in the meantime.[2]

In order to enable provenance research, libraries record the provenances of their historical holdings in their electronic catalogues. A *provenance entry* for an item usually consists of a reference to an entity such as a person, corporate body, or collection, together with a provenance mark. The identification and disambiguation of entities and frequently occurring provenance marks is achieved via links to records in authority files. Provenance entries make it possible, for example, to query and reconstruct the items owned or held by a single person, to query the whereabouts of relevant items, or to retrace the distribution of all indexed exemplars derived from a given work.

Nowadays, provenance entries are an established part of many electronic catalogues, and are taken into account by the underlying data formats. For example, the union catalogue K10plus, which is the central database maintained and used by the libraries in the German library networks GBV and SWB, provides a dedicated data field for recording provenance entries in a structured way, plus several multi-purpose data fields which can also take provenance entries. Despite the uniform data format, there are several possibilities to record provenance entries. As Hakelberg [2016, §4] explains, libraries even within the same network often use diverse representations for the same type of provenance entry, and the differences are considerable: for example, some GBV libraries record their provenance entries in data fields at the bibliographic level ("Titelebene", describing a manifestation), while others use data fields at the exemplar level ("Exemplarebene", describing held exemplars of the manifestation). These deviations lead to large differences in the presentation of the holdings in the online catalogue, hindering the retrieval of relevant items and historical collections.

Data about persons and corporate bodies are recorded by librarians in national and international authority files, and by the general public in open, cross-domain knowledge bases such as Wikidata [Web2]. Records on persons typically contain, among other things, a unique identifier, the preferred

[1]↑ This paragraph is a brief summary of the introductory chapter in Zuschlag's Introduction to Provenance Research [Zuschlag 2022, §1].

[2]↑ This paragraph is a brief summary of the introductory section in Hakelberg's master's thesis on the status and the perspectives of provenance indexing with authority data [Hakelberg 2016, §1.1].

name, alternative name forms, places and times of birth and death, as well as relationships to corporate bodies and other persons (e.g., places of work, and family members or students). The extent of a dataset of the same person can differ between data sources; compare, for example, the records on the great mathematician and astronomer Nicolaus Copernicus (1473–1543) in various national and international authority files, which are linked at the very end of the Wikipedia page on Copernicus in the fold-out table "Authority Control" [Web3]. Hence, the state of data on persons and corporate bodies is heterogeneous as well, and it may be necessary to consult several data sources for a given entity and combine the obtained data.

Given this diversity and heterogeneity of the existing data sources, it is currently very difficult to retrieve provenance relationships which require data that is distributed over several sources. This requirement applies to a number of research questions that can arise in cultural, historical, or provenance research. The following list contains some examples for such questions, which were obtained in personal communication with researchers.[3]

Q1 Who read work X, in which manifestation and in which year?

Q2 Which exemplars[4] of work X were passed from one of its owners to a student?

Q3 What are the relationships between the recipients of manifestation Y of work X?

Q4 Which items from a collection X were passed on by its owner to a family member?

Q5 Which items from the holdings of library X were acquired from bookseller Y between 1933 and 1945?

Q6 Who participated in the sale of collection X?

Q7 Via which paths did items from collection X enter library Y?

Q8 Which libraries own the items once owned by person X?

Q9 Where did person X acquire items and did they know the previous owners?

In these examples, variables X, Y are used as abstract placeholders for specific objects such as works, manifestations, collections, etc. Therefore, **Q1**–**Q9** are actually query *patterns*, each of which represents a huge set of possible queries obtained by instantiating each variable with any such object. For the introductory purposes of this chapter, we continue to use the abstract placeholders but refer to **Q1**–**Q9** simply as *queries*.

Queries **Q1**–**Q9** are examples for a multitude of queries that arise in the context of disciplines such as history, the social sciences, cultural studies, or law, or in an interdisciplinary context. For example, answers to Query **Q1** can help trace the reception of the same work over several eras. Queries **Q2**–**Q4** aim at exploring the network that spans between the recipients of a work or collection. Queries **Q5**–**Q8** are relevant for research on Nazi loot and restitution.

Queries **Q1**–**Q9** have in common that relations of bibliographic and inter-personal (or inter-institutional) nature play an important role. These relations are typically distributed over several data sources. For example, in order to answer Query **Q2**, a researcher would have to find exemplars of work X in the catalogues of various libraries or library networks, inspect the provenance entries of each exemplar for owners, find records for those owners in one or several authority files, and inspect those records for professional relationships. This process is highly laborious and involves interaction with a multitude of heterogeneous data sources at an expert level.

[3] Dietrich Hakelberg (head of Dept. "Collection Development and Cataloguing", Research Library Gotha of the University of Erfurt) [Web4], Michaela Scheibe (deputy head of Dept. "Manuscripts and Early Printed Books"), [Web5], and Joëlle Weis (head of Research Area "Digital Literary and Cultural Studies", University of Trier), [Web6].

[4] Conforming to the FRBR model [IFLA SG FRBR 2009], the precise wording should be "examplars of manifestations of expressions of X", but we omit the intermediate entities for brevity whenever there is no chance of misunderstanding.

It is obvious that the described process would benefit greatly from automation, i.e., from a procedure that receives a query as an input, consults the various data sources autonomously, collects answers, and presents them to the user. In order to be as generic as possible, such a procedure requires an abstract model of data sources and possible queries, and it needs to be based on a careful analysis of the available data sources and their data models, and on data integration techniques. These requirements are closely linked to the question raised by Hakelberg [2016, p. 46, translated from German]: "How can historical provenance relationships be formulated and represented in a machine-readable way?"

1.2 Goal and Research Question

In this thesis, we pursue the goal of facilitating the automated retrieval of provenance relationships. More precisely, we want to develop a method for answering provenance queries that refer to bibliographic entities, people, and corporate bodies, as well as bibliographic and inter-personal or inter-institutional relationships. This method should, on input of a query, autonomously consult relevant data sources, retrieve the required information, and return the set of all answers to the user. The method should furthermore provide a high-level specification for the implementation of a retrieval system that supports the user in formulating their queries, answering them, and exploring the data that supports the query answers. It is our long-term vision that such a retrieval system will support provenance research by prospectively retrieving potentially interesting constellations.

The described goal leads to the following central research question for this thesis.

RQ	**How can provenance relationships be modelled and automatically retrieved?**

This question implies several subordinate questions:

RQ1	*What is the state of research on infrastructures for the automated retrieval of provenance relationships? Which approach(es) is/are most closely related?*
RQ2	*What are the general challenges for answering queries such as* **Q1**–**Q9***, and what are the specific challenges for an automated approach?*
RQ3	*Which data sources, standards, data formats, and further tools are available for answering provenance queries using multiple, heterogeneous data sources?*
RQ4	*Based on the structure of the identified data sources, how can data sources, queries, and query answers be modelled in an abstract framework?*
RQ5	*What is a suitable method for retrieving provenance relationships in that framework?*

1.3 Methods and Outline

In order to answer our research questions, we will proceed as follows.

In Chapter 2, we address **RQ1** by reviewing existing work and relating it to the goal of this thesis. The literature review covers works on research infrastructures for (social/historical) network analysis, data integration techniques in general and in the library domain, and provenance indexing.

In Chapter 3, we address **RQ2** via a case study based on the exemplary queries from Section 1.1. We demonstrate a manual attempt at answering them, discuss the expected quality of query answers, and compare the challenges of this manual process and of an automated approach.

In order to address Questions **RQ3**, we review data sources, techniques, and tools in Chapter 4. This review will cover standards for the description of bibliographic resources, data communication protocols, data description and exchange formats, and a range of data sources.

Based on the insights from the previous two reviews and the case study, we address Question **RQ4** in Chapter 5 by developing an abstract model of data sources, queries, and query answers. This mathematical model subsumes the previous examples while being vastly more general: it gives a formal description of how to build admissible queries, without restricting their contents (i.e., the specific names, attributes, concepts, and relations used) or their complexity.

Based on our model and all previous insights, we address Question **RQ5** in Chapter 6 by developing a method for formulating queries, retrieving query answers, and presenting them to the user. This model should serve as the basis for a future implementation of a retrieval system as indicated in the previous section.

Finally, we draw conclusions in Chapter 7.

Our central research question originates from library and information science (LIS). With the development of our model and method, we hope to provide a means for LIS to benefit from established methods from mathematics and computer science.

1.4 Acknowledgements

The author expresses his sincere thanks to Prof. Dr. Robert Jäschke for supervising and agreeing to review this thesis and for valuable suggestions; Christian Rüter for agreeing to review this thesis and for valuable feedback; Dr. Dietrich Hakelberg for an inspiring introduction to provenance research and stimulating discussions from which the main idea for this thesis originates; Prof. Vivien Petras, PhD, for valuable suggestions and the pointer to the SoNAR project; Dr. Joëlle Weis and Michaela Scheibe for inspiring conversations and valuable suggestions; Dr. Nadine Neute for inspiring discussions and valuable feedback; and, last but not least, Dr. Renate Stein for proofreading, valuable feedback, and ongoing support, especially during the challenging four months allotted for preparing this thesis.

2 Context

This thesis aims at providing support for provenance research, which is part of historical research. As we have seen in Chapter 1, networks that represent people and their relationships play a central role. Therefore, we begin our exploration of the scientific context of our work with the topic of historical (social) network analysis. In Section 2.1, we collect insights from a recent research project aimed at providing a research infrastructure for historical network analysis and apply them to our research questions.

It has also become clear in Chapter 1 that the relevant data is distributed over a multitude of heterogeneous data sources. Therefore we review the literature on linked data, data integration, and data provenance in Section 2.2.

Finally, we give a brief overview of the state of provenance indexing with authority data in Section 2.3.

2.1 Network Analysis and the SoNAR Project

2.1.1 Network Analysis

According to Jansen [2003], the notion of a *network* is a central tool for the analysis of modern societies in sociology, political science, and economics. In these disciplines, networks of political actors, companies, or researchers, among many others, are the subject of study. Additionally, networks play an important role in organisational psychology, biology, and web science [Web7; Web8]. The following paragraph briefly summarises the main constituents of network analysis as described by Jansen [2003].

Network analysis defines networks and provides a statistical toolset for describing and analysing them. A network is a graph, which consists of nodes and edges. Nodes represent actors, events or objects (e.g., people, companies, institutions, countries), and edges represent relations between those (in the case of social networks, e.g., friendship, collaboration, or family relationships). Networks are defined via certain modelling decisions such as the commitment to a set of actors and relations that are of relevance, or the decision whether relations are directed or undirected and whether they are dichotomous (an edge is present or not) or weighted by values representing frequencies or extent. The tools provided by network analysis include various metrics that apply to single nodes, pairs of nodes, paths in the network, or the whole network; those often come in several variants. Examples are connectivity, in-/outdegree, density, size, cohesion, multiplexity, reachability, and path distance. Since a graph can conveniently be represented by its adjacency matrix, methods from matrix algebra are part of the toolset [cf. Jansen 2003, §§1.1, 3.3, 5.3].

In applications where social structures are the object of study, the term *social network analysis (SNA)* is used, and it predominantly refers to the *method* of investigating social interactions [Otte and Rousseau 2002].

In the context of historical research, the notion of *historical network analysis (HNA)* has emerged recently; it focuses on the reconstruction of historical networks [Menzel, M.-J. Bludau, et al. 2020]. A distinguishing feature of HNA is its retrospective view, i.e., it is used to analyse historical data

extracted from available sources [Fangerau et al. 2022]. Menzel et al. [2020] name “a lack of awareness with regard to the availability of suitable research data” as a limiting factor of HNA; in particular, a large amount of data is distributed over heterogeneous data sources.

2.1.2 SoNAR (IDH)

The research project “SoNAR (IDH), Interfaces to Data for Historical Social Network Analysis and Research” (SoNAR) [M. Bludau et al. 2020; Menzel, M.-J. Bludau, et al. 2020; Web9], which was funded by the German Research Foundation (DFG) from 2019 to 2021, developed approaches to building “an advanced research technology environment supporting Historical Network Analysis and related research” [Web9]. In the long-term vision, that environment is expected to integrate data from a variety of existing repositories, thus providing researchers with an extensive, standardised, and interregional infrastructure for answering research questions using methods from HNA. According to the project proposal and the final reports [Web10], the project participants undertook a systematic analysis of processing and managing the source data for the purposes of HNA, designed a model of a structured data analysis for HNA based on SNA methods, developed visualization approaches and interfaces, evaluated all components for a scientific usage, and developed a concept for implementation and operation.

One of the project’s four work packages (WPs) addresses the development of the research technology environment and its evaluation against real research questions. In the remainder of this subsection, we summarise the insights described in the final report on this WP [Fangerau et al. 2022] that are relevant for the research goals of this thesis.

The report emphasises relationships and their evolution over time as central aspects of historical research and acknowledges the increased importance of methods of HNA. In this context, the authors note that visualisation plays an important role as a means for restructuring information and thus contributes to the progression of knowledge. They also address the *hermeneutic circle* [Malpas and Gander 2015] as a general approach to answering historical research questions: this term refers to an iterative, circular work process where the initial question guides the work with the source material and is, in turn, readjusted based on the answers obtained.

In order to evaluate the research technology, the project participants developed two research scenarios with several research questions. The report names four specific questions that were considered central. The nature of these questions is very general and “global” in the sense that they refer to a large part of a network: for example, they ask for the point in time when a scientific area became a separate discipline, or for the role of academic and familial links in the course of a given scientist’s career.

In this context, the report distinguishes two approaches to developing research questions in HNA. One of these is the explorative approach, where suitable questions are developed based on the available data. According to the authors, this approach includes serendipity, i.e., the hope that an inspection of the data helps identify unexpected phenomena that contribute to the shaping of the research question.

The report uses the term *data source* for original documents, media, and artefacts from which “network-compatible” (i.e., essentially relational) data can be obtained; data sources can be identified via *repositories* of various kinds, including library catalogues, archive portals, databases, and more. The report highlights the advantages of using data from authority files such as the “Gemeinsame Normdatei”, the Integrated Authority File of the German National Library (GND) [Web11], which are standardised, subject to quality control, and essential for disambiguating personal names. Moreover, authority files are freely accessible and provide information on the provenance of their data.

Concerning the repositories used, the authors report the general problem of missing or erroneous data, which leads to distorted answers to the questions, and which was not solved systematically in the project. In particular, the GND is focused on German library holdings and thus contains a disproportionately high amount of German-speaking persons. The data is biased towards people who have published at all, towards elite research, towards men (in particular among authors before the 1950s), and against certain disciplines, such as economy. The authors conclude that these restrictions significantly affect the answers to historical research questions. As possible remedies, they include further repositories, such as the German Union Catalogue of Serials (ZDB) [Web12], the German National Library (DNB) [Web13], the Kalliope Union Catalogue (KPE) [Web14], and, tentatively, other authority files (in particular, Wikidata and VIAF); we will get back to these in Section 4.3. However, the evaluation showed that even the sum of these repositories does not provide enough data for a differentiated temporal analysis; in particular, biographical data of the authors dominate, but those cover long time spans and do not provide sufficient evidence for or against dynamic relationships such as teacher-student relationships.

Concerning the selection of data, the report distinguishes between direct and indirect information on relationships. For example, if one is looking for support for the hypothesis of a social relationship between two persons A and B, then a family relationship between A and B explicit in the data supports the hypothesis directly, while matching biographical dates for A and B are an indirect indicator. In the latter case, the relationship explicit in the data (matching biographical dates) acts as *substitute information* (originally, in German: “Stellvertreterinformation”) for the relationship under consideration. The authors also address a special kind of indirect relationship, called intellectual, which mostly involves a third person or event, such as the co-citation of A’s and B’s texts by someone else.

A main constituent of the report is a catalogue of requirements to HNA and the research technology. This catalogue consists of 61 requirements that reflect the specific needs of researchers. The following of these requirements are relevant for our purposes (descriptions translated from the original German text and slightly rephrased and/or shortened):

R003 Researcher wants all information on relationships contained in the metadata of the resources to be shown as derived social relationships in the data model.

R004 Researcher wants all information on non-social relationships (see above) to be distinguished from direct social relationships.

R005 Researcher wants to connect non-social and non-explicit social relationships with further conditions (e.g., overlapping lifespans).

R006 Researcher wants to group people with comparable attributes (e.g., overlapping lifespans).

R016 Researcher wants a list of people connected to a given person.

R017 Researcher wants a list of corporate bodies connected to a given person.

R018 Researcher wants a list of people connected to a given person via some corporate body.

R029 Researcher wants, given a node, a description of the “potentially available attributes” and the provenance of the data.

R030 [dito, but with edges instead of nodes]

R043 [Researcher wants] a filter on attributes, e.g. biographical data, in order to restrict relationships to adulthood.

R061 Researcher wants to know which kinds of relationships are contained in the dataset.

2.1.3 Graph Technologies for the Analysis of Historical Social Networks Using Heterogeneous Data Sources

In the paper of the same name, Menzel, M.-J. Bludau, et al. [2020] report on work in the SoNAR project concerning the "creation and operation of research infrastructure for [HNA] based on heterogeneous data sources from cultural heritage institutions". In particular, the authors present insights on modelling and transformation of heterogeneous data sources and on the design of visualisation for historical networks. In the remainder of this subsection, we summarise their insights on data selection and processing.

In the described approach, the authors integrate data from six heterogeneous repositories into a large uniform graph that is stored in a graph database and managed by the highly performant graph database management system Neo4j [Web15]. The original repositories comprise an authority file GND, federated library catalogues (DNB, ZDB, KPE), and portals offering electronic full texts of historical newspapers (ZEFYS [Web16], Exile Press [Web17]); altogether they contain some 30 million records. The data model is restricted to 9 entity types (e.g., *Resource* or *PersonName*) and 9 very generic relation types (e.g., *RelationToPersonName*, *RelationToTopicTerm*). There is also a distinction between explicit and implicit relations: the former are present in the data, and the latter have to be inferred automatically via certain "guidelines" and marked as such.

Since some of the above data sources include full texts, it is necessary to link names of people, corporate bodies, or geographical entities to entities in authority files. The use of the underlying technique, *named entity linking*, is discussed further by Menzel, Schnaitter, et al. [2021] in the context of SoNAR, and by Meiners [2022] independently of SoNAR.

Among the challenges associated specifically with processing data, Menzel, M.-J. Bludau, et al. [2020] name the sheer size of the combined graph, which causes performance issues, and the fact that the normalisation led to errors and inconsistencies. More generic challenges include the integration of domain knowledge and the adaptation of a more performant graph database engine (GraphDB [Web18]), which requires extensive remodelling.

2.1.4 Insights Relevant for This Thesis

We now put the insights from the SoNAR project described in the previous two subsections into relation with the research goals pursued in this thesis. First, the analysis of historical networks is a broad concept, and SoNAR supports a setting that is much more general than just provenance research. Thus, the setting that we want to support in this thesis is a specific section of HNA and subsumed by the SoNAR setting.

The central role of relationships and of temporal aspects in historical research have to be reflected in the model and method that we are going to develop. Furthermore, the hermeneutic approach to historical research and the explorative approach to answering general research questions need to be supported by our method. In fact, the repeated exploration of data is exactly what we hope to support by enabling researchers to ask specific queries over a combination of data sources and use the answers to obtain "local" views of the whole network and thus inform the next steps in their research process.

Locality is an important feature that distinguishes our approach from the SoNAR setting: The exemplary research questions from the SoNAR report appear to be rather global in the sense that they focus on wider areas in the network and that their analysis requires "global" techniques such as graph metrics and statistical methods. In contrast, we aim at answering more local queries that are part of a larger research question and which focus on a single item/actor and its neighbourhood in the network (e.g., owners of a book or students of a scholar, see the exemplary questions in

Section 1.1). Answering such queries requires the retrieval of certain "candidates" from that neighbourhood which provide the necessary evidence and which then can be used to inform another iteration within the explorative approach. In the light of these considerations, we should not adopt the a-priori restriction to a fixed set of entity types or (generic) relation types; instead, it should be to use arbitrary concepts and relations from the data sources in queries and answers. Hence, we will focus on the described local perspective and leave the accommodation of global techniques for future work. In a similar spirit, we will not study visualisation; however, our envisaged retrieval system can be used *along with* visualisation tools and provide entry points to a large network.

The construction of a uniform (graph) database that integrates all the available data from the heterogeneous repositories appears to be an integral part of SoNAR, according to Menzel et al.'s [2020] discussion. The obvious advantages include the interaction with a single database, the use of a single data model, autonomous hosting independently of the single repositories, and thus full control over the data. Furthermore, inconsistencies between the repositories can be resolved a-priori (at creation time), and the created database does not depend on possible changes in the data models of the repositories. However, the static nature of this database also has disadvantages: a data model has to be fixed upfront, and ultimately needs to be adapted when the models of the repositories change or new repositories are added; regular updates are necessary when the contents of the repositories change; finally, as we have learnt above, a large graph database uses a large amount of memory and requires a highly performant database system. As a result of this discussion, we do not want to commit our approach to a *static* scenario in the sense just described. Instead, we want to include the possibility of a *dynamic* scenario, where the single graph database that encompasses the distributed repositories is just an abstract notion and queries are answered *in place* over the original repositories. We will extend this discussion in Chapter 6.

A feature that we miss in the reports and publications of SoNAR is a rigorous definition of admissible queries that specifies exactly which queries are in the scope and which are not. We aim at providing such a rigorous definition for our setting which, at the same time, should be as general as possible. We consider this rigorous definition a main feature of our approach.

Data sources in the sense used in the SoNAR project report, i.e., original documents, media, and further artefacts are not part of our setting, as we do not consider the part of the research process that consults those data sources. Our work focuses on the level of what the SoNAR authors refer to as repositories, and we will relate, in Section 4.3, our choice of repositories to Menzel et al.'s [2020] selection. In this thesis, we continue to use the term "data source" for what SoNAR refers to as a repository, and we speak of *original data sources* to refer to original documents etc.

In order to allow researchers to refer to the original data sources, it becomes clear that *data provenance* plays a crucial role: the answer to a specific query in our approach needs to refer to the original data sources that provide evidence for the information contained in that answer. This reference can point to original documents (SoNAR: data sources), and/or to data sets in repositories. For a more detailed discussion of data provenance, see Section 2.2.3.

The observed relevance of indirect or substitute information in the context of SoNAR seems important to keep in mind for our work as well: It has to be noted that not all information that one might be looking for is recorded in the data. For example, there is no record of who *read* a book, and ownership has to be used as a substitute for readership although evidence of ownership is only a necessary condition for readership and not a sufficient one—although one might argue that, in the case of, say, a scientist, ownership is very likely to indicate readership. Therefore, answers to questions about readership retrieved on the grounds of recorded ownership require further interpretation and investigation by the researcher who asked the question in the first place. The example from the SoNAR report that uses "having the same biographical dates" as a

substitute for a social relationship is very similar, but in addition the fact that two persons have the same biographical dates is not a relationship that is given explicitly in the data but requires a certain amount of reasoning on several attributes of the entries for the two persons. This general observation has to be taken into account by our approach.

Finally, the insights on advantages and disadvantages of the repositories used in SoNAR will affect our discussion of data sources in Section 4.3.

2.1.5 Further Initiatives Providing Research Infrastructures

Menzel, M.-J. Bludau, et al. [2020] mention several projects that connect decentralised and heterogeneous bibliographic data sources and/or extract historical networks within the social sciences and humanities. We briefly summarise some of these projects.

DARIAH-DE [Web19] is an association that develops a digital research infrastructure for the arts and humanities following the example of large digital research infrastructures for the natural sciences. It comprises of 16 partner institutions from the academic sector and is part of the European network DARIAH-EU [Web20]. DARIAH-DE's services for research include the development and hosting of services for data analysis and visualisation [Web21].

Culturegraph [Vorndran 2018; Web22] is a service offered by the DNB which aggregates bibliographic data from library networks in German-speaking countries, from the DNB itself, and from the British Library. According to Vorndran [2018], Culturegraph makes over 160 million data sets available "for data analyses, evaluation of connections and statistical analyses". This is achieved, among other things, by enrichment via external data sources such as GND and ORCID [Web23], thus allowing for the identification and disambiguation of persons.

Menzel, M.-J. Bludau, et al. [2020] further note "an increase in joint research projects that are focused on the extraction of historical networks within the social sciences and the humanities." These projects include *Six Degrees of Francis Beacon* [Warren et al. 2016; Web24], *histoGraph* [Novak et al. 2014; Web25], *Issues with Europe* [Web26], *APIS* [Gruber and Wandl-Vogt 2017; Web27], and *Gesellschaftliche Wissensproduktion in der Aufklärung* [Purschwitz 2018]. Judging from Menzel et al.'s [2020] summary, these projects put an emphasis on statistical methods, visualisation, and/or exploration.

2.2 Linked Data and Data Integration

2.2.1 Basic Concepts

Data Integration refers to the problem of making a set of autonomous and heterogeneous data sources uniformly accessible [Doan, Halevy, and Ives 2012, p.6]. The current landscape of data sources includes highly structured data represented and accessed using classical database techniques, as well as more loosely structured data accessible via (Semantic) Web techniques. Given this vast and diverse landscape, data integration is a complex problem, and various techniques have been developed for this purpose. The textbook by Doan, Halevy, and Ives [2012] provides a comprehensive introduction to data integration.

In the remainder of this section, we restrict our focus on those aspects of data integration that are most relevant in the context of bibliographic data sources on the Web.

Linked data [Web28; Domingue, Fensel, and J. A. Hendler 2011] refers to data published on the World Wide Web that is structured and connected with other data. Based on a small and uniform

set of simple technologies the linked data paradigm provides applications with access to "a global, unbounded dataspace" [Domingue, Fensel, and J. A. Hendler 2011]. In the context of the Semantic Web [Berners-Lee, J. Hendler, and Lassila 2001; Marshall and Shipman 2003], linked data is one of several developments aimed at making data on the Web more machine-understandable.

The main technologies underlying linked data are the following.

URIs A Uniform Resource Identifier (URI) [Berners-Lee, Fielding, and Masinter 2005] is a string that is assigned to a physical or logical resource in order to identify that resource uniquely. URLs (Uniform Resource Locators) are a special case of URIs which additionally ensure that a resource can be located in a network (e.g., the internet) and that information can be retrieved from it. URIs are an integral part of RDF and the Web Ontology Language OWL (see below).

HTTP The Hypertext Transfer Protocol (HTTP) [Web29] is the fundamental protocol for data transfer and communication underlying the World Wide Web (WWW).

RDF The Resource Description Framework (RDF) [Web30] "is a framework for expressing information about resources. Resources can be anything, including documents, people, physical objects, and abstract concepts" [Web30]. Its main ingredients are *resources* and binary *properties* for linking resources. RDF statements are *triples* of the format subject–predicate–object (S–P–O) consisting of a resource, a property, and a resource (or literal). Resources and properties are described using URIs. RDF thus provides a simple and flexible data model that is particularly useful for the publication and linking of data on the Web; it has been a standard of the World Wide Web Consortium (W3C) since 2004 [Web31].

In the context of the Semantic Web, RDF and associated technologies are important tools for the definition and use of ontologies. For a definition of the term "ontology" in this setting, we cite Horrocks and Patel-Schneider [2011]: "A major feature of the Semantic Web is the ability to provide definitions for objects and types of objects that are accessible and manipulable from within the Semantic Web. In Computer Science, a collection of these sorts of definitions about a particular domain is called an ontology, although philosophers may (and probably will) have a different understanding of what constitutes an ontology." More importantly for this thesis, ontologies can be used to represent knowledge about a specific domain or about the world in general, in an unambiguous and machine-readable way, using a formal language. What is more, *reasoning* mechanisms can be employed to derive implicit knowledge, i.e., knowledge that logically follows from the explicitly represented knowledge. Last but not least, ontologies and reasoning play a role in data integration where they allow for disambiguating entities and their definitions, and for specifying and validating mappings between data sources [cf. Wache et al. 2001; Doan, Halevy, and Ives 2012, §12].

The following technologies associated with RDF and ontologies are of interest for this thesis.

RDFS RDF Schema (RDFS) [Web32] is a specific RDF vocabulary which can be used for modelling and thus serves as a very basic ontology language. It provides terms for modelling, among others, classes, properties, and domain and range restrictions.

OWL OWL is the ontology language recommended by the W3C. It is based on knowledge representation languages from the description logic (DL) family [Baader et al. 2017]. DLs have a well-defined syntax and model-theoretic semantics, which makes them suitable as a formal ontology language in the sense mentioned above. The members of the DL family vary regarding their expressive power and, closely related, regarding the computational complexity of their basic reasoning problems. OWL 2, which is the current version of OWL, is based on an expressive DL where reasoning is still decidable and reasoning algorithms have been implemented in various inference machines that perform well on a wide range of real-world ontologies. Besides these knowledge representation languages and support for reasoning, OWL includes infrastructure

for interaction with ontologies and interoperability within the Web, such as Internationalized Resource Identifiers (IRIs), XML schema datatypes, import mechanisms, and many more.

SPARQL The Simple Protocol And RDF Query Language (SPARQL) [Web33] is a W3C-recommended query language for the Semantic Web. More precisely, it is a language for querying RDF graphs via pattern matching, whose syntax is based on the Structured Query Language (SQL) [Web34], the ISO standard for managing and querying relational databases. However, SPARQL is "far more powerful than SQL, since it is designed for the *open*, *decentralized*, and *fluid* Web" [Della Valle and Ceri 2011]. SPARQL also provides a communication protocol for the interaction between clients and *endpoints*; answers are returned in RDF or XML. Additionally, SPARQL exploits inference mechanisms, i.e., answers may contain facts that are not explicitly stated in the original RDF graph but are obtained involving certain sets of inference rules, based on OWL or other standards.

Linked open data (LOD) is linked data that is also *open data*, i.e., accessible and shareable by everyone [Web35].

2.2.2 Linked Data in the Library Domain

The state of LOD in the library domain in 2013 is summarised by Pohl and Danowski [2013] in their introduction to the edited volume "(Open) Linked Data in Libraries" [Danowski and Pohl 2013] as follows (translated from German and rephrased): Libraries and related organisations began to experiment with linked data as early as 2008. This was followed by linked-data initiatives by important players such as OCLC, the Library of Congress (LoC) [Web36], and the DNB; in particular, the LoC started the initiative "Bibliographic Framework for the Digital Age" in 2011, declaring the renunciation of the library-specific standards MARC 21 and Z39.50 (see Section 4.2), and announcing the development of an infrastructure based on RDF as a basic data model. The advantages of LOD for the library domain include retrievability of, e.g., catalogue data by search engines, mechanisms for permanently linking, e.g., hit lists and hit views, the use of a much more flexible data model, interoperability ensured by the use of open web standards, and reusability via open licences. A prominent example of this development is the provision of authority data as LOD, which can be shared and reused on the Web: e.g., authority data on person names from the GND has been used by Wikipedia since 2005 in order to provide links to further reading in Wikipedia articles, see also [Hengel and Pfeifer 2005]. In addition to the GND, further providers have made authority data available as LOD, among them LoC and VIAF (see Section 4.3).

The state of L(O)D in libraries in 2013 was extensively surveyed in the special issue "Linked Data, Semantic Web and Libraries" of the *Journal of Library Metadata* [Bair 2013a]. In her preface to this issue, Bair [2013b] refers to the challenges to linked data implementations in general and in the library domain that were reported in previous work [Bizer, Heath, and Berners-Lee 2009; Byrne and Goddard 2010; Web37; Alemu et al. 2012], and she emphasises that these challenges remain. The special issue contains a survey on the perception of linked data in libraries, as well as "seven case studies of the experimental efforts of LAM institutions to use linked data to increase access to their collections and user services, plus four others that aim to increase awareness of and educate on key topics" [Bair 2013b, p.76]. The challenges highlighted in these articles fall into five areas, among them data and schema mapping and interoperability, and data quality and trust.

Further work on L(O)D in the library domain is reported in the following publications.

Burrows et al. [2021] describe an LOD model that links three large manuscript databases in the *Mapping Manuscript Migrations (MMM)* project. MMM aims at "providing large-scale analysis and visualizations of the history and provenance of medieval and Renaissance manuscripts" [Burrows et al. 2021, p.3].

Lígia Triques, Ventura Amorim Gonçalez, and Albuquerque [2022] give an overview of the data collection and integration technology in the Digital Public Library of America (DPLA), with a focus on interoperability and challenges for integrated data access.

N. Freire et al. [2019] study metadata aggregation in the context of *Europeana* [Isaac, Clayphan, and Haslhofer 2012; Petras and Stiller 2017], a web portal of the European Union that provides unified access to more than 56 million digital objects from the collections of over 4000 cultural heritage institutions in Europe [Web38]. N. Freire et al.'s case study involves two Dutch institutions as data provider and aggregator, and aims at improved discoverability of the data. The main challenge was the transition from traditional data models to flexible semantic data models. The article presents the results of a requirement analysis, the workflow that was developed, and its implementation.

Ullah et al. [2018] review the current state of L(O)D in cataloguing in the context of the trending transfer of bibliographic metadata towards L(O)D by major libraries. Their review provides an extensive survey of the recent literature. The main findings include the observations that L(O)D is becoming "the mainstream trend in library cataloging especially in the major libraries and research projects of the world" [Ullah et al. 2018, p.47] and that bibliographic metadata is becoming increasingly meaningful and reusable through the emergence of Linked Open Vocabularies.

Hauser [2014] gives an overview of the linked data service of the DNB, which started in 2010 with the publication of GND authority data as linked data. Since 2012, the DNB has been using and maintaining its own GND ontology [Web39] in order to bridge the gap from the original MARC-based data model to a flexible, open data model. In particular, the GND ontology provides a vocabulary for describing entities such as persons, corporate bodies, and places, thus allowing to disambiguate and link these entities.

2.2.3 Metadata Provenance

In the data integration scenario, the origins of (meta)data are important: When querying the combination of several data sources, the retrieved answer should contain, for every fact, information that identifies the original data source or repository from which that fact was obtained. This information is useful as a "justification" for the retrieved answer or as a pointer for further research; it is especially desirable when the consulted data sources contain conflicting information. In order to delineate the origin of data about a book (or other item) from the provenance of that book itself, we adopt the notion of *metadata provenance* [Eckert 2012].

Metadata provenance on the Web has been studied from the beginnings of linked data [see, e.g. Hartig 2009; Moreau, Groth, et al. 2008; Moreau, J. Freire, et al. 2008]. In the context of linked data from cultural heritage institutions and especially the Europeana portal, metadata provenance is studied by Eckert [2013a; 2013b; 2012]. Data provenance has also received attention in the classical database domain [see, e.g., Doan, Halevy, and Ives 2012, §14].

2.3 Provenance Indexing

Hakelberg [2016] studies the state of provenance indexing with authority data. We summarise his conclusions in the remainder of this paragraph. The GND provides authority data that is very suitable for recording provenances across institutions; this is ensured by the structured data model and the open and collaborative nature of the GND. The GND has thus become a central instrument for recording provenance information. Provenance indexing is laborious, and practices vary greatly between German libraries (see Section 1.1). Provenance records are useful only if they

are retrievable across library networks. For this purpose, indexed data needs to be homogeneous and the usability of catalogues of library networks needs to be ensured. Among other things, an overview of the normed vocabulary used for provenance indexing should be provided to users as a search entry. Hakelberg also recommends the further development of the Thesaurus of Provenance Terms (T-PRO) [Web40], which is a controlled vocabulary for the specification of provenances via single descriptors or chains thereof. In particular, the T-PRO descriptors need to be assigned URIs in order to make them reusable as linked open data.

We draw the following insights from these observations. GND as well as library catalogues should be central data sources within our approach. Obstacles may be the heterogeneous situation of indexing in library catalogues as well as the fact that T-PRO is not ready for LOD. The need for the presentation of the used vocabulary as a search entry should be taken into account.

3 Example Queries and Answers: A Case Study

As a first step towards delineating the type of queries that should be covered by our approach, we examine the queries used as motivating examples in Section 1.1 more closely. Those queries will serve as a point of reference for the analysis of the available data sources in Chapter 4, and they will be generalised by the abstract framework developed in Chapter 5. With that framework, we will provide a rigorous definition of the admissible queries that specifies their structure but does not impose any restriction on their content.

3.1 Example Queries

As already noted in Section 1.1, the queries introduced there are in fact *query patterns*, and we will make this distinction in the remainder of this section. We repeat the query patterns here in order to discuss them in more depth.

Q1 Who read work X, in which manifestation and in which year?

Q2 Which exemplars of work X were passed from one of its owners to a student?

Q3 What are the relationships between the recipients of manifestation Y of work X?

Q4 Which items from a collection X were passed on by its owner to a family member?

Q5 Which items from the holdings of library X were acquired from bookseller Y between 1933 and 1945?

Q6 Who participated in the sale of collection X?

Q7 Via which paths did items from collection X enter library Y?

Q8 Which libraries own the items once owned by person X?

Q9 Where did person X acquire items and did they know the previous owners?

These patterns can be turned into specific queries by instantiating each variable with a specific object. For example, the variable X in **Q2** can be instantiated with the seminal work *De revolutionibus orbium coelestium* (short: *De revolutionibus*; English translation: *On the Revolutions of the Heavenly Spheres*) [Copernicus 1543] by the astronomer Nicolaus Copernicus (1473–1543). Together with the additional specification that the owner of the book is a scientist, we thus obtain the following query.

Q2′ Which exemplars of *De revolutionibus* were owned by some scientist who passed them on to a student?

It is important to note that these exemplary query patterns are not meant to be representative for the range of queries that provenance researchers are interested in asking. Finding a representative choice would require a systematic analysis of queries relevant to or useful for researchers. Such an analysis would need to comprise an extensive interview study based on very generic questions of a predominantly open-ended nature, requiring a labour-intensive evaluation which could easily fill a separate thesis. As indicated above, the general framework that we will develop is informed by the available data sources and designed to cover a wide range of possible queries. Hence, it is

reasonable to assume that tools developed on its basis will be helpful for provenance researchers. In subsequent work, when our method will hopefully have been implemented in a prototype retrieval system, the extent to which researchers' needs are served can be determined by means of a more focused user study with more specific questions, which in turn can inform possible extensions of the framework.

3.2 Manual Answer Retrieval

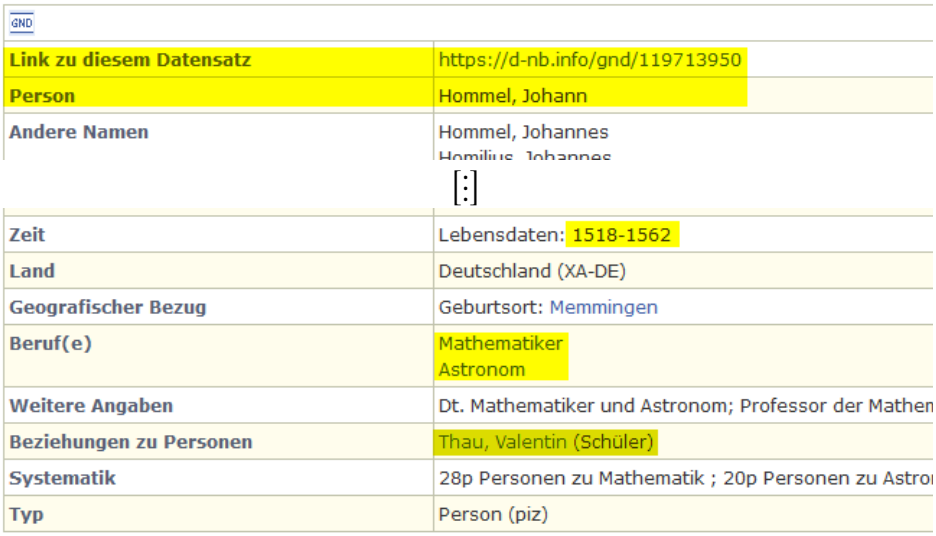
In order to demonstrate how a researcher could proceed (manually) when answering a query, we consider the query **Q2'** from above. An obvious way to proceed is the following. First, our researcher finds exemplars of *De revolutionibus* in online catalogues of libraries and library networks. For each such exemplar, they then inspect the provenance entries that name owners who were people (not corporate bodies). Finally, our researcher will have to find those names in databases such as authority files or Wikidata and, for each entry, search the listed professions and relationships to other people for the specified concept *Scientist* and relation *student*.

For example, the online catalogue (OPAC) of the Gotha Research Library of the University of Erfurt (Forschungsbibliothek Gotha) lists two printed exemplars of *De revolutionibus* [Web41]. One of those bears the signature Druck 4° 00466, and its provenance entries name the following previous owners [Web42]; see also Figure 3.1:

- Hieronymus Tilesius (1529–1566): autograph and date 1551
- NN: note, date 1553, name scraped out
- Johann Hommel (1518–1562), autograph
- Valentin Thau (1531–1575), note (greek proverb, possibly not denoting ownership)
- Ernest II, Duke of Saxe-Gotha-Altenburg (1745–1804): stamp/seal, initial
- Ducal Library, Gotha (a predecessor organisation of Gotha Research Library): stamp marking a duplicate
- Ernestine Gymnasium, Gotha: stamp
- Landesbibliothek Gotha: stamp

Figure 3.1: Excerpt of University of Erfurt OPAC search result for exemplar Druck 4° 00466 of "De Revolutionibus" with provenance entries (screenshot of 2023-06-05)

Our researcher can immediately decide that they can ignore the second entry (no name given) and the last three entries (corporate bodies). For the persons named in the remaining four entries, our researcher finds the corresponding GND records, which involves manual disambiguation in the case of Tilesius [Web43; Web44; Web45; Web46]. On inspection of these GND entries, it turns out that Ernest II was a regent and very probably not a scientist, and that the other three people—Tilesius, Hommel, and Thau—had professions such as theologian, mathematician, and astronomer, which qualifies them as scientists. Furthermore, Hommel's entry contains a reference to Thau via the relation *student*, see Figure 3.2 (and Thau's entry contains the inverse reference to Hommel). From this reference, our researcher can conclude that two scientists in the teacher-student relation have both possessed the exemplar.



GND	
Link zu diesem Datensatz	https://d-nb.info/gnd/119713950
Person	Hommel, Johann
Andere Namen	Hommel, Johannes Homilius, Johannes
[⋮]	
Zeit	Lebensdaten: 1518-1562
Land	Deutschland (XA-DE)
Geografischer Bezug	Geburtsort: Memmingen
Beruf(e)	Mathematiker Astronom
Weitere Angaben	Dt. Mathematiker und Astronom; Professor der Mathen
Beziehungen zu Personen	Thau, Valentin (Schüler)
Systematik	28p Personen zu Mathematik ; 20p Personen zu Astro
Typ	Person (piz)

Figure 3.2: Excerpt of the GND entry for Johann Hommel (screenshot of 2023-06-05)

Unfortunately, the data available does not imply that Hommel passed the exemplar (directly) to Thau; furthermore, the provenance entry for Thau raises doubts as to whether Thau was really an owner of the exemplar. Therefore, the retrieved data can only be regarded as a *candidate answer* which entails the hypothesis that the found exemplar was passed from Hommel to Thau. Our researcher can now engage in further research in order to verify that hypothesis.

3.3 The Quality of Query Answers

Our example illustrates that the quality of query answers strongly depends on the quality of the underlying data, regardless of whether answers are obtained manually or automatically. In particular, *missing or spurious data* will lead to *missing or spurious answers*. Before we begin to deal with automated query answering, we need to analyse the sources of missing or spurious data and their effects on the answers obtained.

For the sake of a principled discussion, we call the set of answers to a query obtained by some manual or automatic procedure an *answer set*, and we call the (set of) answers that the query has in reality the *true answers* and the *true answer set*. In the above example, the answer set obtained by the described manual process consists of the single answer "Druck 4° 00466", if we assume that our researcher interprets the data retrieved generously (i.e., as an indication that Thau might have been an owner and might have received the book directly from Hommel) and draws additional conclusions (i.e., that every mathematician or astronomer is a scientist). The true answer set is not known and will most probably never be known; it is a term of rather philosophical nature.

Ideally, both answer sets should coincide. Incomplete data may cause the answer set to exclude some true answers, and spurious data may cause it to contain some answers that are not true. We call the respective answers *missing* and *spurious*. On the basis of our example, we can identify four distinct causes for data being incomplete or spurious, as discussed in the following. We use the word *term* as an umbrella term for concepts (such as *Scientist*) and relations (such as *student*).

1. In the above example, GND contains no information on whether any of the persons involved is indeed a *scientist*. Instead, GND provides information on more specific professions, e.g., for Thau and Hommel (theologian, mathematician, and astronomer).[5] The information that they were scientists has to be *derived* based on *general knowledge of the world*.

   The same effect can occur with relations instead of concepts: Query Pattern **Q2** asks for family members. If the data only supports more specific relations such as *sister* or *father*, then relationships using *familyMember* would again need to be derived. Unfortunately, the cataloguing rules for GND authority datasets do not require that relationships are recorded exhaustively nor using a normed vocabulary; see Section 4.3 for more details.

   In summary, terms can be missing in the data sources because they are implicit in more specific terms. If the query uses such missing terms, then the answer set is always empty.

   There are at least two possible remedies: the person or machine determining the answers can either make derivations based on their knowledge of the world, or they can use *substitute information*, which has been addressed in the context of the SoNAR project (see Section 2.1, in the form of more specific terms. In the second case, we will still have to expect missing answers if the selection of more specific terms is not exhaustive.

2. In the above example for **Q2′**, there is no information available on whether any of the owners of the exemplar *passed it on* to another owner. The exact same problem affects Query Pattern **Q4**. Analogously, attempts to answer instances of Query Patterns **Q1**, **Q3**, **Q8**, and **Q9** will have to deal with the problem that there is no information available as to who *read* books, who was a *recipient* of a manifestation, who owned an item *once* (i.e., *before* another owner), or who *knows* some other person.

   More generally speaking, terms can be missing in the data sources because they are not recorded at all, as a consequence of either a general lack of evidence or a general design decision for the data source. The underlying reasons can be manifold: for example, relationships such as who actually *read* a book are very hard to confirm, or terms may not be part of the fixed vocabulary for a data field in a source. If the query uses such terms, then the answer set is always empty, as in Case 1.

   As a possible remedy, the use of substitute information can be helpful here, too: for example, the relation *owner* could be used as a substitute for “read”, “recipient of”, or “owned once”, or any familial, social, or professional relation could be used as a substitute for “knows”.

3. In the above example for **Q2′**, it is possible that single owners of the exemplar or single relationships between owners have not been recorded because no evidence has been found yet. More generally, concept memberships or relationships can be missing sporadically, which may lead to missing answers.

4. In the same example, if Hommel is erroneously recorded as the owner, then the answer obtained on the grounds of that record is spurious. More generally, spurious concept memberships or relationships may lead to spurious answers.

[5] A search in the GND catalogue reveals that GND does have an entry for the subject term *Wissenschaftler* (scientist) [Web47], but this term is used only sporadically: its 46 subordinate terms do not include, for example, *Mathematiker*, *Astronom*, or *Theologe* (mathematician, astronomer, theologian) [Web48].

These cases reveal a striking qualitative difference concerning the effects of data incompleteness or spuriousness on the answer set: Cases 3 and 4 have effects on single answers only, i.e., some answers are missing or spurious. However, Cases 1 and 2 generally make *all* answers missing unless further provisions are made. Such "provisions" might include *reasoning* (e.g., deriving the implicit knowledge that Hommel was a scientist from the explicitly recorded fact that he was a mathematician) and *hypothesising* (e.g., assuming that Thau was an owner and received the book directly from Hommel). Reasoning can be supported using semantic technologies such as (domain-specific or top-level) ontologies and related infrastructure. A possible way to support hypothesising is to provide for the use of *substitute information*, which has been addressed in the context of the SoNAR project; see Section 2.1. That is, users could be allowed to declare terms occurring in the data that may act as substitutes for certain terms used in their queries.

In the light of these observations, it is appropriate to consider query answers as *candidates* that necessitate (and inspire) further research. For this purpose, spurious answers (in manageable numbers) are less harmful than missing answers. Consequently, it is important to find ways to avoid missing answers without generating too many spurious answers. In other words, it is desirable to obtain an answer set that is a slight *overapproximation* of the true answer set.

3.4 From Manual to Automated Query Answering

The manual process that we have described in Section 3.2 is cumbersome, laborious, and prone to errors and omissions for several reasons: The search in library catalogues for exemplars of works and their provenances requires expert skills. Catalogues with potential matches need to be selected manually, and each catalogue needs to be queried individually, using its own search keys and syntax. For each retrieved exemplar, each potentially relevant provenance entry needs to be followed up in further data sources such as authority files, multiplying the amount of manual work necessary. Finally, it is not clear what an effective and efficient way to manually "explore" relationships would be: while it is easy to find direct relationships such as *student* in the view for a person's entry in databases such as GND or Wikidata, there are *indirect* relationships that are much harder to discover easily by hand, e.g., "persons P_1 and P_2 are students of the same scholar".

These considerations suggest that query answering will strongly benefit from automated support, which can help reduce the amount of manual work, integrate heterogeneous data sources, incorporate background knowledge (e.g., every mathematician is a scientist), and discover indirect relationships between entities or "substitute information". We envisage a retrieval system that enables researchers to formulate queries and which computes answers consulting data sources selected by the user. The vocabulary used for formulating queries should be based on the vocabulary present in the data sources, but it should also include concepts and relations such as *Scientist* or *read*, which are not recorded in the data, as discussed in Section 3.3. The retrieval system thus needs to implement ways to match those terms with the available vocabulary, as well as techniques for integrating data from heterogeneous sources. It is our general vision that the retrieval system will serve as an instrument for prospectively finding interesting candidates that inspire further research. In the remainder of this thesis, we want to lay the foundations by developing a formal model and an abstract method that can serve as a basis for a future implementation of a retrieval system.

4 Analysis of Available Data Sources and Techniques

In order to build a system for retrieving provenance relationships, data sources need to be selected. It is the aim of this chapter to provide information for this selection. Furthermore, the abstract model of data sources, provenance queries, and answers that we will develop in the following chapter needs to incorporate the characteristics of the data models used in the appropriate data sources. For this purpose, we review data sources that contain the relevant metadata (Section 4.3), as well as bibliographic standards, data formats, and communication interfaces relevant for these sources (Sections 4.1 and 4.2). Given the observation from the previous chapters that answering provenance queries is likely to involve several data sources, we pay special attention to data transfer and integration techniques, including linked data. At the end of this chapter (Section 4.4), we draw conclusions from the review that inform our model and method in the following chapters.

Before we start the review, we need to define the central term *metadata*. For this purpose, we follow Hider and Harvey's [2008] definition and reflection: The term *metadata* is used in multiple ways. The most general use is according to its literal meaning, "data about data". This definition is not restricted to the library domain but includes bibliographic records for documents (which contain the primary *data*). A more specific variant is the use of "metadata" to refer to structured data describing digital objects, again independently of the library domain. Hider and Harvey also note that these two meanings can no longer be clearly distinguished in view of the progressing digital transformation. In this thesis, we adopt their decision to use the term *metadata* in the most general sense, "applying to all information resources" [Hider and Harvey 2008, p.13].

4.1 Standards for the Description of Bibliographic Resources

The following two standards are widely adhered to in bibliographic data sources. This discussion is a brief summary of Wiesenmüller and Horny's introduction [2015, §§2, 3], unless indicated otherwise.

FRBR The Functional Requirements for Bibliographic Records (FRBR) [IFLA SG FRBR 2009] constitute a conceptual entity-relationship model developed by the International Federation of Library Associations and Institutions (IFLA) [Web49] in 1998 (and updated in 2009) in order to support users in finding, identifying, selecting, and accessing bibliographic items [Wiesenmüller and Horny 2015, p.17]. FRBR's entities represent things that need to be represented in the data. These entities fall into three groups, of which we introduce the first two. Entities of Group 1 are *Work*, *Expression*, *Manifestation*, *Item*, and their common superordinate concept *Endeavour*; entities of Group 2 are *Person*, *CorporateBody*, and their superordinate concept *ResponsibleEntity*. Furthermore, FRBR contains relations that link Group-1 entities with each other and with Group-2 entities, respectively. These entities and the basic relations are shown in Figure 4.1.

Beside FRBR, the IFLA developed a conceptual entity-relationship model for authority data, the Functional Requirements for Authority Data (FRAD) [IFLA WG FRANAR 2008], as well as a continuation of FRBR, the Functional Requirements for Subject Authority Data (FRSAD) [IFLA WG FRSAR 2010].

FRBR serves as a basic principle of RDA, which we discuss next.

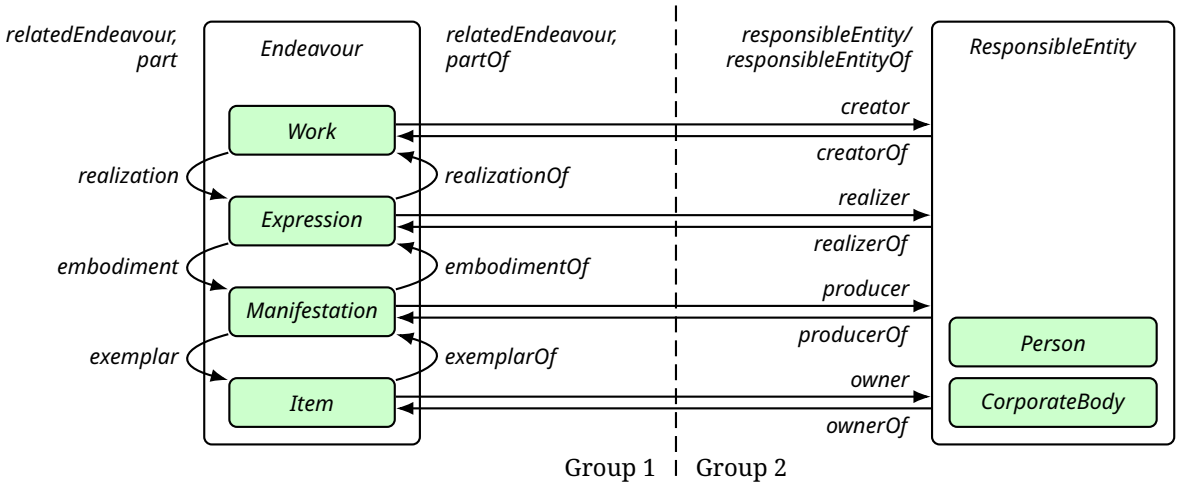


Figure 4.1: FRBR entities and basic relations of Groups 1 and 2 [following Web50; Web51]

RDA Resource Description and Access (RDA) [Web52] is a standard for the cataloguing of publications in cultural heritage institutions, and particularly in libraries. RDA is implemented and used in the library systems of several countries, including the US, the UK, Canada, Australia, and the German-speaking countries [Web53].

RDA and the application guidelines for the German-speaking area stipulate a special practice for recording FRBR Group-1 entities and relating them to each other: Bibliographic records ("Titeldatensätze") have a bibliographic level for describing a manifestation and an exemplar level for describing the related exemplars. The description of works and expressions is considered part of the description of a manifestation and thus recorded at the bibliographic level. However, it is possible to create and link an authority record for a work or expression containing the relevant description. In that case, the source of the information can be recorded as well [cf. Wiesenmüller and Horny 2015, §5.1].

For our approach, this observation means that catalogues of German libraries do not use a uniform way of linking, e.g., a work *X* with its manifestations. In particular, if there is no authority record for *X*, then *X* is represented in the data of its manifestations only implicitly (and possibly ambiguously).

4.2 Data Formats and Communication Protocols

In order to identify data sources in the following section, we first need to describe relevant standards and formats for the description and communication of (bibliographic) data. The following discussion is based on Hider and Harvey's overview [2008, §10], unless indicated otherwise.

4.2.1 Markup Languages

We briefly describe markup languages, which are repeatedly used in the technologies described in the following subsections. Markup languages are machine-readable languages used for the structuring and formatting of files.

HTML The most prominent example is the Hypertext Markup Language (HTML) [Web54], the core language of the WWW. The markup features of HTML are also used to identify and describe certain elements of digital objects. In particular, HTML provides for meta tags, i.e., labels for

metadata. One of the restrictions of HTML is its fixed set of labels, whose meanings cannot be changed. Further standards were developed to overcome this restriction, such as SGML and XML.

XML The Extensible Markup Language (XML) [Web55] is both a file format and markup language for the storage and transmission of arbitrary data; it allows for a hierarchical structuring of data in a text file format that is readable by both humans and machines [Web56]. XML is the basis for many modern web developments. Most metadata schemas have standard expressions in XML, as we will see below.

4.2.2 Data Communication Protocols

Z39.50 is a protocol for data communication between bibliographic databases. It is an ISO standard as well as an ANSI/NISO standard. Z39.50 is a set of rules developed specifically for translating search and retrieval commands between databases such as OPACs. Using Z39.50, it is possible to issue commands specific to a local database and obtain results from a remote database that might use a different command set. This protocol can be implemented over the internet and similar networks. Z39.50 is widely used in the library domain for providing access to catalogues, for example, by the LoC. Still, it is not fully implemented by all library systems, and not all libraries use the most recent version. It is also not very widely spread outside the library community.

SRU Search/Retrieval via URL (SRU) [Web57] is a successor of Z39.50 and an LoC standard. It has the same function, which is the standardisation of search and retrieval across online databases. SRU is based on HTTP, XML, and the Contextual Query Language (CQL) [Web58], a standard syntax for representing queries to information retrieval systems. Thanks to this technology, SRU is more dynamic than Z39.50 and less restricted to the library domain.

OAI-PMH The Open Archives Initiative (OAI) [Web59] is a project devoted to interoperability standards in information agencies in general, including libraries. The OAI Protocol for Metadata Harvesting (OAI-PMH) [Web60] is a standard developed by the OAI and is widely used in the library domain, albeit not as widely as Z39.50.

4.2.3 Data Description Schemata

MARC Machine-Readable Cataloguing (MARC) [Web61] "is the main data communication standard in use in libraries today" [Hider and Harvey 2008, p.198]. It is a family of formats for the *exchange* of bibliographic and further data between library systems, which is being extensively used by libraries around the world. MARC was developed by the LoC in 1969 and has been updated several times. The MARC family comprises more than 20 dialects that have been developed as an official standard in several countries. Those are very similar to each other and provide very detailed record structures for cataloguing of bibliographic data. They all adhere to an international ISO standard. The development of MARC pursued the goals of labour and cost reduction, and of standardising the cataloguing process as well as data communication and transfer. MARC "allows flexibility in library information systems" [Hider and Harvey 2008, p.201] by providing an easy way of data exchange and allowing for better resource discovery (among others, in comprehensive union catalogues).

MARC 21, was developed in 1999 and is "the major version of MARC in use internationally" [Hider and Harvey 2008, p.205]. It consists of five formats for specific kinds of data. Two of these are for bibliographic data and authorities data, respectively.

MARC has been criticised for its inflexible structure, the fragmentation into several dialects, and the restricted compatibility with current computer technologies, among other things. Hider and

Harvey [2008, p.212] give more details and further references. There have been several attempts to redeem some of these disadvantages, among them the development of new standards such as MODS (see below) and of XML schemas based on MARC 21 such as MARCXML and Turbomarc.

The problem of the proliferation of standard was not restricted to the variety of MARC dialects but also occurred within non-MARC formats. Attempts to solve this problem have been made via translation and unification. One tool for unification is Dublin Core (DC); see below.

Until recently, MARC 21 had separate data fields for the recording of provenance marks, owners, and further information belonging to a provenance. As a consequence, in data obtained via MARC 21, the important connection between provenance marks, owners, and further information is not available. For this reason, the MARC Steering Group approved, in May 2023, a proposal for introducing a new data field for "Ownership and Custodial History in Structured Form" [Web62]. This data field allows the structured recording of provenance information by providing subfields for all the constituents of the same provenance. However, it is realistic to assume that this new field will be implemented in the relevant systems with some delay.

MAB The "Maschinelles Austauschformat für Bibliotheken" (MAB) [Web63], translating as "machine data exchange format for libraries", is a legacy bibliographic data format that has been developed solely for the purpose of data exchange by the DNB. Although Hider and Harvey [2008, p.204] classify it as a dialect of MARC, it is in fact conceptually different from MARC, assigning exactly one bibliographic element to each data field and allowing a more flexible ordering of elements [Web64]. In 2013, the DNB completely abandoned the delivery of its bibliographic and authority data in MAB in favour of MARC 21 [Web63].

MODS The Metadata Object Description Schema (MODS) [Web65] is a standard developed by the LoC in order to overcome the problems with MARC mentioned above. MODS is based on XML and a subset of MARC fields; it can thus be used alongside MARC and as a "switching format between MARC and non-MARC schemas" [Hider and Harvey 2008, p.219]. The LoC also maintains an analogous standard for authority data, the Metadata Authority Description Schema (MADS) [Web66].

DC The Dublin Core Metadata Element Set, in short: Dublin Core (DC) [Web67] is an international standard consisting of 15 essential metadata terms for describing digital or physical resources. It was formulated by the Dublin Core Metadata Initiative (DCMI) [Web68], a project of the US-American non-profit organisation ASIS&T, for the purpose of locating information resources on the WWW. The 15 core elements are tailored towards the "primary metadata needs" across domains [Hider and Harvey 2008, p.215], thus making it a flexible data model. Applications of DC include resource description, the combination of metadata vocabularies from different standards, and the provision of interoperability in the linked data context. DC is extended by the *DCMI Metadata Terms*.

RDF We have already introduced RDF in Section 2.2. In the context of the following description of data sources, RDF is a very flexible data schema and the core technology for providing metadata as linked data.

BIBFRAME The Bibliographic Framework (BIBFRAME) [Web69] is a data model for bibliographic description, which was designed to replace the MARC standards and use linked data principles to improve access to bibliographic metadata within and outside the library domain [Web70]. The most recent version 2.0 was released by the LoC in 2016 [Web69]. So far, only a handful of libraries are using BIBFRAME, most of them in test mode [Web70].

4.3 Data Sources

In this section, we collect and present information on data sources that provide metadata on objects, individuals, concept membership, and relationships (in short: *terms*) relevant for provenance research. Given the example queries discussed in Sections 1.1 and 3.1, relevant terms include the FRBR entities and relations (see Section 4.1), ownership, as well as professional, familial, and social relationships. We first give an overview of eligible data sources (Subsection 4.3.1) and then select a few particularly relevant ones, for which we provide a detailed analysis (Subsection 4.3.2). The choice of the latter was motivated by the insights from the SoNAR project (see Section 2.1) and Hakelberg's [2016, §4] overview of the state of provenance indexing with authority data in the German-speaking area. This analysis will have to be extended whenever the model and method that we are going to develop in the following chapters is implemented in a retrieval system in future work.

4.3.1 Collection of Data Sources

We have identified four categories of relevant data sources: library catalogues, special databases for provenance research, authority files, and knowledge bases. We next describe, for each of these categories, the relevant information that is contained in the respective data sources, and we list examples.

Library Catalogues Catalogues contain the information relevant for provenance research, such as

- bibliographic entities (works, expressions, manifestations, items);
- relations between these entities (e.g., *isManifestationOf*);
- attributes for these entities (e.g., year of publication);
- people and corporate bodies and relations such as authorship;
- provenance entries (including, e.g., owners, the ownership relation, and further attributes such as the year of ownership);
- (implicitly) the current ownership;
- (ideally) references to entries in authority files for all these types of entities.

Examples of relevant library catalogues include

- catalogues of national libraries, which often aim at collecting literature exhaustively and which are most likely to make metadata available in interoperable formats, in particular: **the catalogues of DNB, LoC, and the British Library** [Web13; Web71; Web72];
- meta-catalogues that enable a federated search in many catalogues, in particular: **WorldCat** and the **Karlsruhe Virtual Catalogue (KVK)** [Web73; Web74];
- union catalogues of library networks, which combine the holdings of the participating libraries, in particular, the German union catalogues: **K10plus, B3KAT,** the **union catalogues of hbz and hebis** [Web75; Web76; Web77; Web78];
- catalogues and local systems of single libraries, if not covered by any of the above;
- special catalogues, e.g., those used for historical research in the SoNAR project, in particular: **ZDB, KPE, ZEFYS, ExilePress** [Web12; Web14; Web16; Web17].

Clearly, the integration of catalogues of the first three groups is preferable to the integration of

catalogues of single libraries because it would allow for an interaction with as few systems as possible while giving access to a large combination of holdings. However, sometimes relevant information is recorded only in local library systems, e.g., provenance entries in some of the SWB libraries; see the description of K10plus below. In these cases, it is unavoidable to include these local systems.

Authority Files Authority files contain information relevant for provenance research, such as

- entities such as persons, corporate bodies, places, works;
- relations such as social relations, family relations, professional relations;
- attributes for the entities such as their profession.

They are usually subject to a strict quality control. Examples include the **GND** [Web11] in the German-speaking area (which has been used extensively in SoNAR), the North-American **LCNAF** [Web79], and the international projects **ISNI and VIAF** [Web80; Web81].

Knowledge Bases According to the insights from the SoNAR project, (open) knowledge bases (KBs) can be useful when trying to overcome the problem with missing or unbalanced data in authority files such as the GND (see Section 2.1). KBs are usually edited independently of the library domain by a wider community of contributors, and their quality cannot be expected to meet the standards of catalogues or authority files edited by library personnel. A prominent example of an open, cross-domain KB is **Wikidata** [Web2], which has tentatively been used in the context of the SoNAR project and which we have already identified in Section 1.1 as possible source of metadata for complementing authority data.

Cultural Heritage Databases This category contains generic federated portals of cultural heritage items as well as special databases created especially for the support of provenance research. Examples for the first kind of databases are **Europeana** [Web38] (see Section 2.2) and the **German Digital Library (DDB) [Web82]**; an example for the latter is **Proveana**—the research database of the German Lost Art Foundation [Web83].

4.3.2 Analysis of Data Sources

We now provide a review of four data sources from the previous list, namely DNB, K10plus, GND, and Wikidata. For each data source, we collect the following information.

- *Scope:* thematic focus; coverage; number of records; standards for cataloguing
- *Technical infrastructure:* data format; data model; interfaces; support for linked data
- *Data quality (from the SoNAR grant proposal [Schneider-Kempf et al. 2018, p. 19ff.]), if applicable:* use of persistent identifiers and URIs; adherence to standards for, e.g., time and date specification
- *Features useful in the context of SoNAR (see our discussion in Section 2.1.4), if applicable:* recording of temporal attributes or relations; recording of data provenance
- Further features specific to the respective data source, if applicable

DNB The following information is taken from the DNB's websites under the category "DNB Professional" [Web84; Web85; Web86].

The DNB adheres to a legal collection mandate, according to which the DNB collects "all texts, images and sound recordings published in Germany or in the German language, translated from German or relating to Germany that have been issued since 1913" [Web84]. This includes all physical publications, and, since 2006, electronic publications made available via the internet. The

mandate commits the DNB to a complete and unbiased collection that includes, among others, "printed works compiled or published between 1933 and 1945 by German-speaking emigrants" [Web84].

The DNB catalogues its entire collection both descriptively and by subject, adhering to standards such as RDA, using authority data, and including persistent identifiers such as ISBN, ISSN, URN, and DOI. The cataloguing data feeds the German National Bibliography.

According to the 2021 annual report [German National Library 2022], the DNB's holdings comprise 43.7 million physical or digitally accessible units, which are represented by almost 26 million records in the German National Bibliography.

Metadata can be obtained from the DNB freely (under the CC0 1.0 licence) via the interfaces SRU and OAI-PMH, which support XML serialisations of the data formats MARC 21, RDF, DNB Casual (an XML-based DC format), and MODS. Further data formats are available via an individual access to the DNB's *Data Shop*. The DNB's Linked Data Service provides open access to its bibliographic and authority data in *RDF* under the same license. Instructions on how to interact with these interfaces are given on the DNB's webpage on metadata services [Web86]. The DNB website also reports on a project for the conversion of its data into the BIBFRAME format [cf. Web87]: for title records, the DNB OPAC provides a download button for the BIBFRAME format. The state of this project beyond 2014 is left unclear.

According to the detailed specifications on the MARC format by the DNB [Web88; Web89], the recording of dates and times, languages, geographic area codes, and countries conforms to the respective ISO standards. Since 2015, the DNB has been using MARC field 883 for recording the metadata provenance for a selection of data fields [Web90]. Furthermore, the DNB's MARC 21 schema provides for the cataloguing of the following temporal data concerning title (i.e., manifestation) records: production, publication, distribution, and manufacturing dates (including reproductions and reissues); notes on dates and times; chronological subject term (as full text); graduation year (for dissertations etc.).

K10plus K10plus is the joint catalogue of the German library networks GBV and SWB. Together, these two networks comprise more than 1400 national, regional, academic, and public libraries [Web91; Web92], of which 838 participate in K10plus, according to the list of participating institutions in the K10plus Wiki [Web93]. Thus, K10plus comprises the data from the majority of German academic institutions [cf. Web94].

As of 31 December 2022, K10plus contains 80.8 million bibliographic records ("Titeldatensätze") with 235.4 million ownership records [Web95]. Cataloguing adheres to the same standards and guidelines as in the DNB, including similar specifications for the recording of temporal data and the use of ISO standards for dates, times, etc. In addition, the structured variants of provenance indexing used in GBV and SWB (see below) enable the recording of ownership dates if available.

K10plus provides metadata freely (under the CC0 1.0 licence), mainly via the interfaces Z39.50 and SRU. Those support the data formats MARC 21, (including its XML variants), PICA+ and PICA-XML (two provider-specific internal data formats loosely based on MARC 21), as well as DC, MODS, and the legacy format MAB2. Detailed instructions on how to interact with these interfaces are given in the K10plus Wiki [Web93]. In addition, snapshots of K10plus data are provided in MARC-XML and as linked open data (RDF-XML), as per the information in the K10plus Wiki [Web96]. However, there are currently no recent snapshots, but it is planned to provide them regularly in the near future.[6]

K10plus uses the GND as the central authority file for disambiguating persons, corporate bodies, works, subject terms, and further entities (see below for more details on the GND.). For this

[6] Personal communication with Jakob Voß, VZG (head office of the GBV).

purpose, K10plus includes copies of all GND records, which are synchronised via an OAI-PMH interface within minutes after modification [cf. BSZ and VZG 2022]. K10plus furthermore uses authority data from the Basisklassifikation (BK) [Facharbeitsgruppe Sacherschließung 1995] and the Regensburger Verbundklassifikation (RVK) [Web97], cf. the K10plus Wiki [Web98].

Concerning the state of provenance indexing in the library networks GBV and SWB, Hakelberg [2016, §4] reports the following: The catalogue systems of GBV and SWB, which are united in K10plus, support the recording and display of provenance marks at the bibliographic as well as the exemplar level of a record. However, the practice of provenance indexing differs greatly between the two networks. Since 2014, GBV has been advising the recording at the bibliographic level, in order to allow provenance research across libraries. The respective data field is structured similarly to the new MARC 21 field 361 (see Subsection 4.2.3) and thus keeps all information on the same provenance linked together. Furthermore, the names of owners are linked to the local authority record that represents, and redirects to, the respective record in GND. The physical provenance marks are documented via chains of terms from T-PRO (see Section 2.3) [plus dates if available]. As of 2014, this standard was being tested and successively introduced in the participating libraries. Before 2014, provenance marks were documented in free text in a data field for exemplar-specific comments at the exemplar level, using T-PRO descriptors and referring to local authority records that were *not* directly linked to GND records. In contrast, the SWB libraries have been, and still are, using a dedicated data field at the exemplar level to record provenance marks. This field provides subfields for the structured recording of T-PRO terms [and dates]. However, most SWB libraries have abandoned the creation of exemplar records altogether and thus needed to resort to using isolated local systems for provenance indexing [cf. Hakelberg 2016]. These systems should therefore be included in the data sources to be accessed by a comprehensive retrieval system for provenance relationships.

In summary, provenance marks are recorded within K10plus in a heterogeneous and incomplete way, and developers of a retrieval system that uses K10plus have to be aware of this problem.

GND The “Gemeinsame Normdatei”, the Integrated Authority File of the German National Library (GND) [Web11], is operated by the DNB “in cooperation with many other libraries, libra[r]y networks and other cultural and academic institutions. At present, the GND contains around 9 million authority records for persons, corporate bodies, congresses, geographic entities, specialised terms and works; these are supplemented, updated and used frequently” [Web85]. More detailed statistics can be found in the DNB’s 2021 annual report [German National Library 2022, p.49]. GND adheres to the same cataloguing standards and offers the same technical infrastructure as the DNB catalogue; in particular, GND data is accessible under the CC0 1.0 license alongside national bibliographic data via the same interfaces and data formats, including several RDF serialisations.

The DNB’s MARC 21 schema [Web88] provides for the cataloguing of the following temporal data in authority records: biographical data (persons), date of publication (works), dates of existence (conferences, corporations), year of discovery, dates of activity (persons, corporations), years of effect of relationships between entities of different kinds. Examples for temporal data of the latter kind include the time interval during which a scholar was member of a university or during which a person was another person’s student.

We have already learnt from the SoNAR project (Section 2.1) that the data in the GND is incomplete for several reasons. A further reason is certainly connected with the individualisation guidelines in the DNB’s guidelines for recording persons and families in the GND [Web99, part EH-P-16], which specify the additional information that needs to be provided in order to disambiguate a person or family. These guidelines define two groups of features that can be used to individualise (i.e., disambiguate) a person, such as biographical data or relationships to other persons, among many others. Depending on the level of the respective authority record (which indicates the state

and quality of the record), one or only a few of the many features from these two groups have to be recorded. Therefore, there is no guarantee that, for example, relationships to other people or temporal data of the kind mentioned above is recorded. In addition, relationships to other people are recorded in an unnormed way, using one of four very generic codes (e.g., acquaintance, professional relationship, family relationship) and specifying the exact kind of relationship in free text. Similarly, professions may be recorded in free text or via a link to an authority record.

Wikidata Wikidata [Web2] is "a free, collaborative, multilingual, secondary database, collecting structured data to provide support for Wikipedia, Wikimedia Commons, the other wikis of the Wikimedia movement, and to anyone in the world" [Web100]. Hence, there is no thematic restriction on the content of Wikidata.

The main unit of information in Wikidata is the *item*. Items "are used to represent all the *things* in human knowledge, including topics, concepts, and objects" [Web101]. Currently, Wikidata contains more than 100 million items [Web102]. This number is two orders of magnitudes higher than the number of articles in Wikipedia because Wikidata stores structured information on *all* Wikimedia projects, including the large media file repository Wikimedia Commons [cf. Web101]. Data on each item is stored in *statements*, which have the same S–P–O form as RDF, in Wikidata terms: *item–property–value*. Statements can have annotations [cf. Web103]. All items and properties have unique identifiers consisting of a *Q* or *P*, respectively, followed by a sequence of digits. These IDs are also part of the URL of the respective Wikidata page and of the item's or property's URI. For example, the item representing the Polish mathematician and astronomer Nicolaus Copernicus has the ID *Q619*, and the statements about this item are listed on the respective Wikidata page [cf. Web104]. The first statement in this list says that Copernicus is a human, using the property *instance_of* (*P31*) and the value *human* (item *Q5*).

Data is recorded in Wikidata collaboratively, i.e., "[d]ata is entered and maintained by Wikidata editors, who decide on the rules of content creation and management. Automated bots also enter data into Wikidata" [Web100]. Values such as times and dates are recorded conforming to the respective ISO standard.

The Wikidata page on data access [Web102] lists numerous ways to access data, all under the free CC0 license. For retrieving individual entries via URIs, there is a linked data interface. For retrieving a relatively small set of entries that are not known in advance, SPARQL queries can be posed against the Wikidata Query Service. For retrieving larger sets of entries, Wikidata offers the Linked Data Fragments endpoint, which requires computational power on the client side. Furthermore, there are two APIs mostly for editing Wikidata, as well as a recent changes stream and complete exports (dumps) [cf. Web102].

Annotations of statements (see above) seem to be suitable for recording metadata provenance and temporal attributes of relationships. It remains to investigate the use of annotations systematically in order to determine the extent to which metadata provenance and temporal attributes are recorded.

4.4 Conclusions

We have collected 20 data sources and reviewed four of them in detail. These four have in common that they provide their metadata as linked data (among other formats), and the three bibliographic data sources adhere to the FRBR entity-relationship model. From this commonality, we conclude that entities and relationships—in mathematical terms, constants and unary and binary relations—should play a central role in the model of queries, answers, and data sources that we are going to develop next. This means that a graph-based type of model, which also features in the SoNAR

project, is suitable. In case relations of higher arity turn out to be required, for example when representing the provenance of an S–P–O statement, those can always be represented by several binary relations via reification [cf. Doan, Halevy, and Ives 2012, p.339f.].

The choice of the reviewed data sources was motivated by the insights from the SoNAR project and Hakelberg's thesis [Hakelberg 2016]. For the abstract model to be developed next, the specific choice of data sources is largely unimportant because the model should be independent on the contents or specific data models of the sources. The same holds for the general method for answering queries that we are going to develop based on our model. However, future implementations of our method will have to take the specifics of the data sources into account. For this reason, it will be necessary to review additional data sources, compare them quantitatively with respect to the extent and quality of their data, and find structured ways for dealing with the problems concerning the data quality reported above, such as the ambiguous links to works and expressions in RDA-based catalogues, the heterogeneous and partly unstructured recording of provenance, and the bias and incompleteness of the GND. The latter problem might be alleviated by using Wikidata (sacrificing the strict quality control of the GND), and promising candidates for alternative databases with extensive provenance information are Proveana, the DDB, and Europeana.[7]

[7]↑ Unfortunately, Proveana does not currently offer interfaces for automated data access, as communicated by Sabrina Werner, the Proveana project manager of the German Lost Art Foundation. For possible alternatives, Ms Werner referred to the DDB and Europeana.

5 A General Model of Provenance Relationships

In this chapter, we develop a generic approach to modelling provenance relationships. More precisely, we need to model three central notions: queries that a user may want to ask, data sources that are to be consulted in order to answer a query, and answers given to a query. In order to obtain a generic approach, we aim at providing rigorous definitions for these central concepts, and we seek intensional rather than extensional definitions. In particular, those definitions should not depend on specific example queries or data sources such as the ones discussed in Chapters 3 and 4; neither should they depend on concrete objects, concepts, or relations (such as *Copernicus*, *exemplar*, or *student*). Instead, we will develop an abstract model that formalises the notions of a query, data source, and answer. This model can then be instantiated with a multitude of specific queries and data sources, and it constitutes the basis of the method for finding answers that we will develop in the next chapter.

As a basis for our abstract model, we choose standard concepts and techniques from graph theory. The concept of a graph is widely used in computer science and discrete mathematics; see standard textbooks [e.g., Diestel 2012]. Graphs and graph techniques are widely applied in various areas such as computer science, linguistics, physics and chemistry, social sciences, and biology [Web105]. They are also the foundation of social networks [Galety et al. 2022] and therefore highly relevant for historical research, e.g., in the SoNAR project, as we have seen in Section 2.1. Graphs are used to represent large knowledge bases [e.g., Ehrlinger and Wöß 2016] and are a fundamental ingredient of RDF. Furthermore, the basic definition of a graph is conceptually simple, and graph theory provides a plethora of well-understood concepts and algorithms. By utilising graph theory, our application scenario can benefit from these concepts, algorithms [Diestel 2012; Even 2012], and implementations [Web106; Web107].

The main idea of our abstract model is the following: Both queries and data sources are represented as graphs (typically a rather small graph for the query and a very large graph for the data source). Answers to the query are those parts of the graph that have the same structure as the query. In more formal words, answers are found using *pattern matching* techniques, which are well-known from querying RDF graphs via SPARQL [Della Valle and Ceri 2011] and from database and graph theory [Abiteboul, Hull, and Vianu 1995; Diestel 2012].

In the following sections, we introduce our model by defining the basic terminology (Section 5.1), the specific notion of a graph (Section 5.2), and the abstract notions of data sources, queries, and answers (Section 5.3). We also discuss decision procedures related to query answering (Section 5.4). Up to that point, the model remains conceptually simple and is based on elementary and self-contained mathematical definitions. We provide additional explanations and illustrations for the sake of readers with little or no background in mathematics. Finally, in Section 5.5, we will discuss the scope and limitations of our basic model based on the exemplary query patterns **Q1**–**Q9**, and sketch possible extensions.

5.1 Basic Notions

We start by introducing the basic terminology that we are going to use in the following, which consists of the usual terms from conceptual modelling [Brodie, Mylopoulos, and Schmidt 1984].

Objects An *object* is a specific entity, for example, the work “Graph Theory” by Reinhard Diestel, its expression in English, its manifestation as the 4th edition by Springer, a specific copy, or the person Reinhard Diestel.

Concepts A *concept* is a class of objects, such as the entities *Work*, *Expression*, etc. from the FRBR model (see Section 4.1).

We determine that names for concepts and objects start with an uppercase letter.

Relations An *n-ary relation* is a set of *n*-tuples of objects; for example, the binary relation *creator* consists of pairs of objects including (*Graph_Theory*, *Reinhard_Diestel*) if we assume for the sake of simplicity that *Graph_Theory* is the unambiguous name of said work.

Constants and concepts are in fact nullary and unary relations, but it is more intuitive to use those separate terms, As we have already argued in Section 4.4, we will hardly need to deal with relations of arity beyond 2; therefore we will mostly discuss binary relations and refer to them as *relations* when no confusion is likely to arise.

To distinguish relations from concepts and objects, we determine that their names start with a lowercase letter.

Converses of relations The *converse* of a relation *R* is the relation obtained from *R* by swapping the components in each pair, for example, the converse *creator_of* of *creator* contains the pair (*Reinhard_Diestel*, *Graph_Theory*).

Instances An *instance* of a concept or relation is an object or a pair of objects from that set; e.g., *Reinhard_Diestel* is an instance of *Person*, and (*Graph_Theory*, *Reinhard_Diestel*) is an instance of *creator*.

Relationships Instances of relations are called *relationships*.

Literals A *literal* is a fixed value, such as a year, date, or identifier.

5.2 Labelled Directed Graphs

Before we develop the technical definitions, we provide the underlying intuitions. Graphs consists of nodes and edges. Edges link nodes and can be directed or undirected. Graphs are easy to visualise: nodes are drawn as circles or rectangles, and edges as arrows (directed) or lines (undirected). Figure 5.1 shows an abstract example of a directed graph.

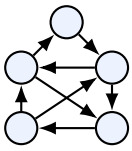


Figure 5.1: A directed graph

For our purposes, nodes represent objects or literals. Directed edges represent relations, for example, the relation *creator* is represented by a set of directed edges pointing *from* nodes representing persons or corporate bodies edges representing the relation *to* nodes representing the corresponding works; the converse relation *creatorOf* is represented by the same set of edges with their direction reversed. Since relations such as *creator* are not symmetric, direction matters and we use *directed* graphs. Symmetric relations such as *relatedEndeavour* can be represented via pairs of edges pointing in both directions.

Furthermore, we want to assign a unique name to each node of a graph and one or several labels to each node and each edge: The name of a node specifies the object that is represented by that node. The labels of a node specify the concepts of which that node is an instance. For example, a node representing the physicist Albert Einstein may be labelled, among others, with the concepts *Person*, *Scientist*, and *Physicist*. The labels of an edge specify the relations of which the pair of nodes represented by that edge is an instance. For example, if a person P_1 has a student P_2 and also collaborates with P_2, then this can be represented via an edge from P_1 to P_2 labelled with both *student* and *collaborator* (and/or an edge from P_2 to P_1 labelled with both *studentOf* and *collaborator*). These considerations lead us to a straightforward extension of the notion of a directed graph: a *labelled directed graph*.

In order to visualise a labelled directed graph, node names are written into the respective node, and node and edge labels are written next to the node or edge. Multiple labels of the same node or edge are delimited with commas. An example is given in Figure 5.2. The shown graph represents a part of the data described in Section 3.2 concerning an exemplar of Copernicus' *De revolutionibus* at the Gotha Research Library *(FB Gotha)*. It contains a node for the work (labelled with the FRBR entity *Work*), a node for the exemplar (labelled with the FRBR entity *Item*), and nodes for the author and two of the owners (labelled with their professions according to their GND entries). For the sake of this example, the owners are additionally labelled with the profession *Scientist*, which is implicit in the actual data. The ten edges represent relationships that are instances of FRBR relations and others between *Work*, *Item*, and *Person*. For the sake of simplicity, the graph deviates from the FRBR model [IFLA SG FRBR 2009] by omitting the FRBR entities *Expression* and *Manifestation* that should occur between the nodes labelled *Work* and *Item*.

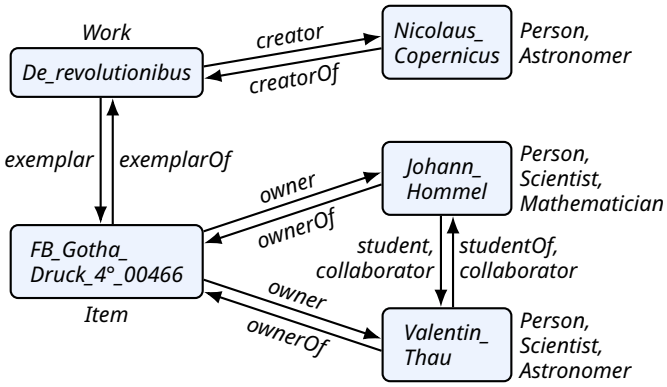


Figure 5.2: A labelled directed graph that represents data concerning an exemplar of Copernicus' *De revolutionibus* and some of its owners

As we will see in the following, labelled directed graphs can be used in our setting to represent (combinations of) data sources as well as queries. They allow us to draw on standard notions from graph theory and query answering in order to define admissible query answers and to devise methods for obtaining those.

The above explanations can be cast into a rigorous mathematical definition, which uses sets to represent nodes, a binary relation over the set of nodes to represent edges, and functions over the nodes and edges to represent names and labels. In order for the ranges[8] of those functions to be well-defined, the following definition of a labelled directed graph is relative to a *namespace*, which contains all the names of objects, concepts and relations that are relevant. This namespace is a parameter that can be freely chosen; it may consist, for example, of all the names found in the relevant data sources.

[8] The *range* or *codomain* of a function is the set of values that can occur as images of that function.

Definition 5.1. Let $\mathsf{N} = (\mathsf{N_O}, \mathsf{N_C}, \mathsf{N_R})$ be a *namespace* consisting of a set $\mathsf{N_O}$ of *object names*, a set $\mathsf{N_C}$ of *concept names*, and a set $\mathsf{N_R}$ of *relation names*. A *labelled directed graph over* N is a triple $G = (V, E, \mathcal{N}, \mathcal{L})$ where

- V is a set, whose members are called *nodes*;[a]
- $E \subseteq V \times V$ is a set of pairs of nodes, whose members are called *edges*;
- $\mathcal{N} : V \to \mathsf{N_O}$ is an injective function that assigns to each node a unique object (called the node's *name*);
- $\mathcal{L} : V \cup E \to \mathsf{N_V} \cup 2^{\mathsf{N_R}}$ is a function that assigns to each node a set of concept names (called the node's *labels*) and to each edge a non-empty set of relation names (called the edge's *labels*); we call $\mathcal{L}$ a *labelling function*.

[a] We use the standard denotation V (from "vertex") for the set of nodes.

Definition 5.1 captures the following commitments regarding names and labels: (1) Every node has a unique name, and no two nodes have the same name (the latter is ensured by injectivity). (2) A node can have an arbitrary number of labels, including no label (in case the node belongs to no concept). (3) An edge can have an arbitrary number of labels, but that number must not be zero – the effect of an edge having no labels can be achieved by simply omitting that edge.

In order to illustrate the components of Definition 5.1, we refer to the graph depicted in Figure 5.2: V consists of five nodes, and E of ten edges (each single arrow constitutes an edge since the direction matters). There are, among others, the following node names and labels:

- The node at the top left has the name *De_revolutionibus* and the single label *Work*.
- The node at the bottom right has the two labels *Person* and *Astronomer*.
- The edge from the node named *Johann_Hommel* to the node named *Valentin_Thau* has the two labels *student* and *collaborator*, and the edge pointing in the reverse direction has the labels *studentOf* and *collaborator*.

5.3 Modelling Data Sources, Queries, and Answers

We can now use our notion of a labelled directed graph to model data sources and queries, and to obtain a rigorous definition of a query answer.

5.3.1 Data Sources

Data sources correspond exactly to our notion of a graph.

Definition 5.2. A *data source over the namespace* $\mathsf{N} = (\mathsf{N_O}, \mathsf{N_C}, \mathsf{N_R})$ is a labelled directed graph over N.

In our model, we assume that there is always a *single* data source against which a query is posed and evaluated. When the model is applied to real-word queries and data sources, the abstract notion of a data source is instantiated by the union of all available actual data sources (such as catalogues, authority files, knowledge bases), including mappings between them, if applicable. We will discuss this point in more detail in Chapter 6.

5.3.2 Queries

In order to model queries based on graphs, we need to distinguish two special groups of nodes that act as placeholders (1) for the object(s) about which the query asks and (2) for further objects that are mentioned in the query without being named explicitly. For example, consider Query **Q2′** from Section 3.2:

Q2′ Which exemplars of *De revolutionibus* were owned by some scientist who passed them on to a student?

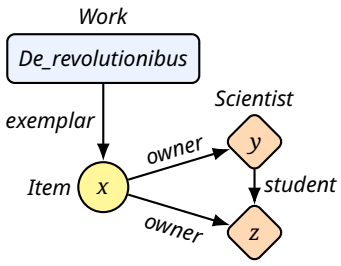


Figure 5.3: Graph representation of Query **Q2′**

To model this query, we do not only need a node representing the work *De revolutionibus*, but also a node representing an exemplar that satisfies the conditions stated in the query and whose name is asked for (Group 1), and two nodes representing the owner and their student (Group 2). Since these three individuals are not known when the query is formulated, we need to use *variables* for naming them. Query **Q2′** can now be modelled by the graph shown in Figure 5.3.

The nodes of this graph fall into three groups:

(1) The node named *De_revolutionibus* represents that work;

(2) Node x falls into Group 1 as explained above;

(3) Nodes y and z fall into Group 2 as explained above.

Node names x, y, z are the variables mentioned above, and we call x the *answer variable* and y, z the *anonymous variables* of this query. From now on, we fix two sets $\mathsf{VAR}_{\mathsf{ANS}}$ and $\mathsf{VAR}_{\mathsf{ANON}}$ of *answer variables* and *anonymous variables*, respectively, and we assume that they both contain a countably infinite number of elements (which ensures that there is an unlimited supply of variables). We furthermore require that these two sets are disjoint with each other and with any set N_O of object names. Thus, according to Definition 5.2, graphs representing data sources cannot use any variables as node names.

In order to allow variables in queries, the following definition of a query is now immediate.

Definition 5.3. A *query over the namespace* $(\mathsf{N}_\mathsf{O}, \mathsf{N}_\mathsf{C}, \mathsf{N}_\mathsf{R})$ is a labelled directed graph over $(\mathsf{N}_\mathsf{O} \uplus \mathsf{VAR}_{\mathsf{ANS}} \uplus \mathsf{VAR}_{\mathsf{ANON}}, \mathsf{N}_\mathsf{C}, \mathsf{N}_\mathsf{R})$.

The operator "$\uplus$" used in this definition stands for *disjoint union*, i.e., "$A \uplus B$" stands for the union of the *disjoint* sets A, B. The wording of the definition also ensures that we do not have to mention variables explicitly when specifying the namespace of a query.

5.3.3 Query Answers

Based on the representation of both data sources and queries as graphs, we can now define the notion of an answer to a query with respect to a data source. For this purpose, it is important to realise that, typically, a query is a small graph and a data source is a large graph, and that finding answers means finding small parts of the large graph that have the same structure as the small graph. For the example query given in Figure 5.3, this means that every subgraph of the data source consisting of four nodes with the same edges and labels should be an answer. Regarding the previous example, the subgraphs given in Figure 5.4 (a, b) constitute answers, while the subgraphs in Figure 5.4 (c, d) do not: Subgraph (a) is identical to the query graph with the only exception that it contains proper objects instead of variables in the node names; Subgraph (b) is the same graph as (a) but extended with an additional edge and additional node labels; Subgraph (c) lacks the edge labelled *student* between the two owners; Subgraph (d) resembles the structure of the query but does not contain the required node named *De_revolutionibus*.

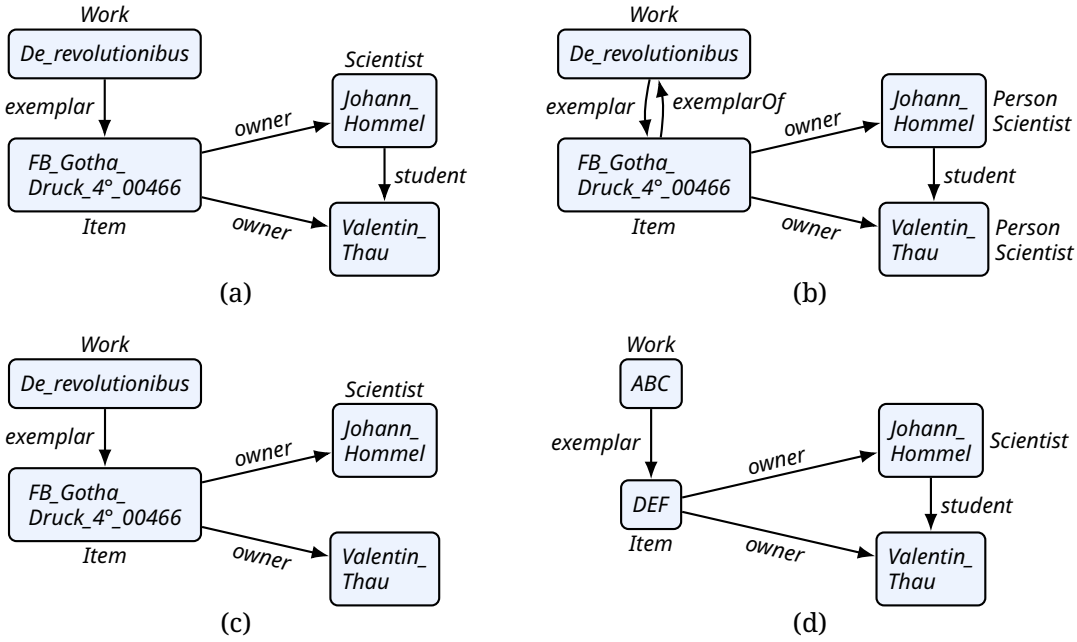


Figure 5.4: Positive (a, b) and negative (c, d) examples for query answers

In order to identify subgraphs of a given graph that have the same structure as another given graph, we use the standard notion of a homomorphism. A homomorphism is a function that maps some object to another while preserving the structure of the former. We thus need to define a variant of homomorphism that maps queries to data sources. This variant is given in the following.

Definition 5.4. Let $\mathsf{N} = (\mathsf{N_O}, \mathsf{N_C}, \mathsf{N_R})$ be a namespace, $G = (V, E, \mathcal{N}, \mathcal{L})$ a query over N, and $G' = (V', E', \mathcal{N}', \mathcal{L}')$ a data source over N. A *homomorphism from G to G'* is a map $h : V \to V'$ that satisfies the following properties.

- **H1** $\mathcal{N}(v) = \mathcal{N}'(h(v))$ for every node $v \in V$ with $\mathcal{N}(v) \in \mathsf{N_O}$.
- **H2** $\mathcal{L}(v) \subseteq \mathcal{L}'(h(v))$ for every node $v \in V$.
- **H3** $\mathcal{L}(v_1, v_2) \subseteq \mathcal{L}'(h(v_1), h(v_2))$ for every edge $(v_1, v_2) \in E$.

If h is a homomorphism from G to G', we write $h : G \to G'$. If there is some homomorphism from G to G', we write $G \lesssim G'$.

Property **H1** requires that a homomorphism maps each node in G that is named with an object to that node in G' which is named with the same object. Nodes named with variables in G can be mapped to arbitrary nodes in G'. Properties **H2** and **H3** require that homomorphisms preserve node and edge labels; more precisely, the subset relation entails that the image of a node (or edge) under h must have *at least* the same labels (and may have additional labels). Furthermore, the graph G' may have additional nodes that are not among the images of the homomorphism.

Figure 5.5 shows a homomorphism h (dashed lines) from the query depicted in Figure 5.3 to the graph from Figure 5.2.

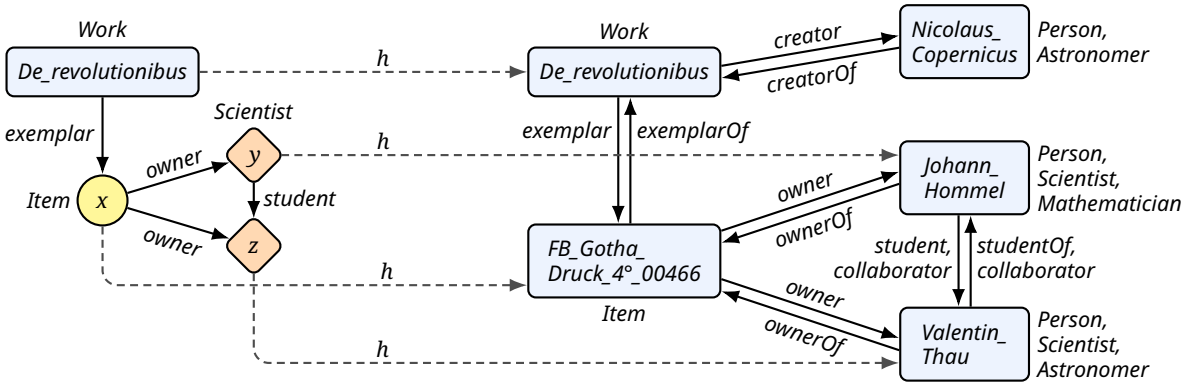


Figure 5.5: A homomorphism from **Q2′** to the graph from Figure 5.2

We can now use homomorphisms to define the notion of an answer to a query.

Definition 5.5. Let $\mathsf{N} = (\mathsf{N_O}, \mathsf{N_C}, \mathsf{N_R})$ be a namespace, $G = (V, E, \mathcal{N}, \mathcal{L})$ a query over N, and $G' = (V', E', \mathcal{N}', \mathcal{L}')$ a data source over N.

1. An *answer to G in G'* is a pair (h, G'') where h is a homomorphism $h : G \to G'$ and G'' is the subgraph of G' induced by $h(V)$.
2. The set of all answers to G in G' is called the *answer set* for G in G' and denoted $\mathsf{ans}(G, G')$.

We make the following remarks concerning Definition 5.5.

The *subgraph of G' induced by $h(V)$* in Point 1 is the graph $G'' = (V'', E'', \mathcal{N}'', \mathcal{L}'')$ where $V'' = h(V)$ and $E'', \mathcal{N}'', \mathcal{L}''$ are the restrictions of $E, \mathcal{N}, \mathcal{L}$ to V''.

The answer corresponding to the homomorphism h depicted in Figure 5.5 consists of h and the graph on the right-hand side after removal of the node *Nicolaus_Copernicus* and the two adjacent edges. This situation is shown in Figure 5.6 with the query greyed out.

Every homomorphism determines an answer. If no homomorphism $h : G \to G'$ exists, then there is no answer to G in G', i.e., $\mathsf{ans}(G, G') = \emptyset$.

From a mathematical point of view, it would suffice to equate answers with homomorphisms since G'' can be reconstructed from G and h. However, we deliberately make G'' explicit in order to take the importance of the "explorative approach" into account (see Sections 2.1.2 and 2.1.4). The subgraph G'' servers as the minimal "section" of the data that a researcher should want to inspect.

So far, we have built a model that captures data sources and query patterns such as **Q2**. In this model, answering queries amounts to finding homomorphisms between graphs. The two natural next steps consist in examining the computational properties of the problem of finding homomorphisms and in extending the model to capture further query patterns. We will realise these two steps in the following sections.

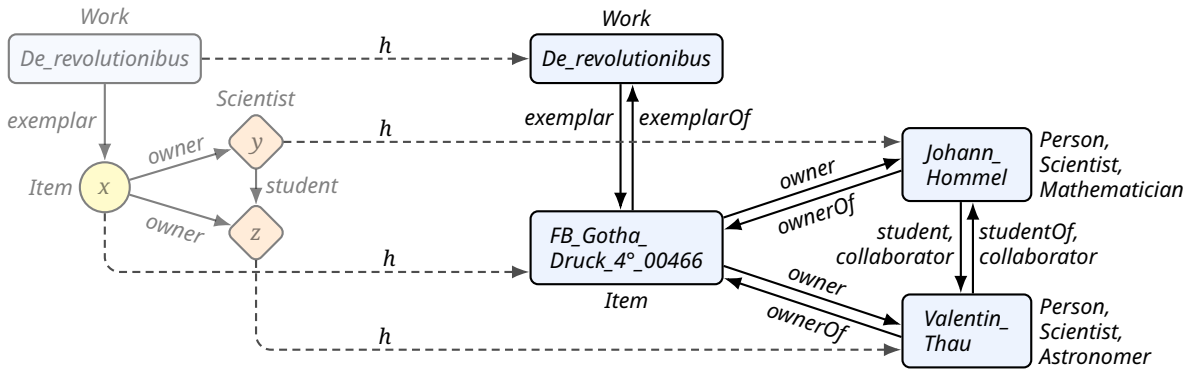


Figure 5.6: An answer consisting of the homomorphism h and the graph G'' on the right-hand side

5.4 Computational Properties

As already indicated, the problem of finding homomorphisms is well-understood in the contexts of database theory and classical logic. In particular, the favourable computational properties of this problem are exploited in successful database systems. We review these in the following, building on fundamental principles of complexity theory [Arora and Barak 2009].

The computation problem of query answering in which we are interested according to our model can be phrased as follows: Given a query and data source, compute all answers. More formally:

Input:	a query G and a data source G' over some namespace N
Output:	$\mathsf{ans}(G, G')$

For the implementation of a retrieval tool that is designed to answer questions, it is crucial to verify that this problem can be solved at all (i.e., is *computable*) and efficiently so (i.e., is *tractable*).Here, computability means that there is some algorithm which, given an arbitrary input (G, G'), returns the correct answer set $\mathsf{ans}(G, G')$ after a finite amount of time. Tractability means that there is even an algorithm whose runtime is strongly limited depending on the size of its input. Usually a *polynomial* upper bound on the amount of time is considered sufficient for tractability; that is, the runtime of the algorithm is bounded by $c \cdot n^k$, where n is the size of the input (e.g., nodes and edges in the input graphs) and c and k are some constants. Problems for which an algorithm with such guaranteed time bounds exists are called *solvable in polynomial time*. A polynomial time bound on an algorithm provides the guarantee that its runtime as a function of the input size increases moderately for all inputs, as opposed to, e.g., an exponential function. Therefore, problems for which a polynomial-time algorithm exists are generally considered tractable. However, this is still an abstract notion, given that the constants c and k can be arbitrary. It remains for practical purposes to find an optimal algorithm with c and k as small as possible, and which performs particularly well on those inputs that occur in the application at hand.

In order to check whether a computation problem is computable and tractable, is it usual to resort to a slightly more abstract level, considering the *decision problem* associated with the computation problem. Typically, *decidability* (i.e., computability) and tractability of the decision problem imply those of the computation problem; hence it suffices to consider these.

The associated decision problem additionally considers a candidate output as an input and asks whether that candidate is in fact a valid output. In the case of our computation problem, the associated decision problem is the following.

Input:	a query G over N, a data source G' over N, a function h from G to G', and a subgraph G'' of G'
Output:	YES if $(h, G'') \in \mathsf{ans}(G, G')$; NO otherwise

We denote it with QA.

QA is strongly related to the decision problem associated to answering (conjunctive) queries over relational databases. This problem has been studied from a computational point of view in seminal papers by Vardi [1982] and Chandra and Merlin [1977] with the following main result.

Theorem 5.6. [Vardi 1982] *Answers to conjunctive queries over relational databases can be decided in time polynomial with respect to the size of the database.*

This result means that the decision problem associated with answering conjunctive queries is decidable and, furthermore, tractable if the size of the query is neglected. Given that queries are typically very small compared to the size of the data, tractability under this assumption is sufficient for practical purposes; in fact, it is the basis for the success of modern (relational) database engines. As a side remark, if this assumption is dropped and the complexity with respect to the joint size of both the database and the query is considered, then it follows from Chandra and Merlin's [1977] results that the problem is intractable (under the usual reasonable complexity-theoretic assumptions) already in the presence of binary relations.

The strong relationship between our decision problem QA and the classical problem of answering conjunctive queries implies that Theorem 5.6 carries over directly to QA. In more precise terms, both problems are mutually *reducible* in polynomial time, which can be shown very easily using standard knowledge from graph theory and first-order logic. Without immersing into the technical definitions for standard notions in complexity theory, we note that polynomial-time reducibility between the two problems means that an algorithm for one of them can be turned into an algorithm for the other which uses only a polynomial (i.e., relatively small) additional amount of time.

We therefore obtain the following decidability and complexity result for our problem QA.

Corollary 5.7. QA *is decidable in time polynomial with respect to the size of the data source.*

As a consequence of Corollary 5.7, there is an algorithm for deciding QA, and thus also for computing the answer set $\mathsf{ans}(G, G')$ for a given query G and data source G' in time polynomial with respect to the size of G'. The polynomial time bound ensures that the growth of runtime as a function of the input size is guaranteed to be bounded by a function of moderate growth, independently of the specific input. However, as explained above, the existence of a polynomial-time algorithm does not immediately imply that there is an algorithm that performs well for the purposes of our envisaged application, which is the implementation of a retrieval system. Fortunately, the close relationship to conjunctive query answering over databases comes to our aid: given the existence of a (very straightforward) polynomial-time reduction of QA to conjunctive query answering, QA can be solved by either re-implementing one of the existing successful polynomial-time algorithms for conjunctive query answering or by implementing the reduction feeding its output directly into a relational database system, using the latter as a "black box". These two options correspond nicely to the distinction between the dynamic and static scenario that we have contemplated in Section 2.1.4, and which we will expand in Chapter 6.

5.5 Discussion of the Basic Model and Extensions

In Section 5.3, we have developed a basic model that is conceptually simple and easy to implement. We have shown how the exemplary query pattern **Q2** can be modelled, and we have indicated how implementations can be obtained. What is missing is a wider analysis of the scope and suitability of this model for capturing queries and data sources relevant for provenance research. It is clear that this analysis cannot be performed comprehensively within the scope of a master's thesis. However, we can estimate the extent of generality of our model by relating it to the exemplary query patterns **Q1**–**Q9**. In doing so, we will understand the limitations of the model and be able to sketch possible extensions. This is the aim of this section.

In the following discussion, we will continue to use **Q1**–**Q9** as *patterns*. That is, we will abstain from instantiating the variables X and Y because the considerations are independent of the specific instantiations. As a consequence, the drawn graphs will contain nodes labelled with uppercase letters X, Y as well as lowercase letters x, y (or similar). Only the latter type of nodes represents answer variables and anonymous variables; the former represents specific objects, for which X and Y are used as placeholders. We continue to distinguish these types of nodes by their shape and colour as in Figure 5.3, using blue rectangles for objects, yellow circles for answer variables, and orange diamonds for anonymous variables.

5.5.1 The Basic Model

The basic model that we have so far developed captures not just Query Pattern **Q2** but also **Q4**, as well as **Q1** and **Q5** if we leave out the year(s) for the moment, i.e., the following query patterns.

Q1⁻ Who read work X, and in which manifestation?

Q4 Which items from collection X were passed on by its owner to a family member?

Q5⁻ Which items from the holdings of library X were acquired from bookseller Y?

Q8 Which libraries own the items once owned by person X?

In the case of **Q1**, the relation *reads* is most certainly not recorded in provenance entries in any bibliographic data source. Therefore, we need to resort to the *substitute information* (see Section 2.1) provided by the *owner* relation. Now these two query patterns can be modelled as shown in Figure 5.7.

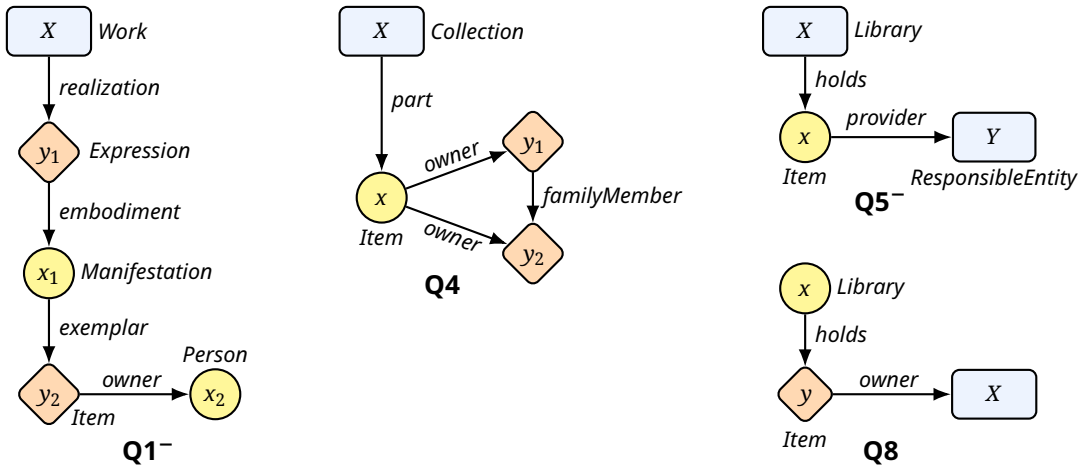


Figure 5.7: Graph representations of Query Patterns **Q1⁻**,**Q4**, **Q5⁻**, and **Q8**

5.5.2 Relations with Attributes and Relations of Higher Arity

The full wording of **Q1** and **Q5** includes references to years as attributes of relationships. In the case of **Q5**, we restrict ourselves to a variant that names a specific year rather than an interval. We will get back to intervals in Subsection 5.5.4. That is, we consider the following two query patterns (with the temporal reference highlighted).

Q1 Who read work *X*, in which manifestation and *in which year*?

Q5* Which items from the holdings of library *X* were acquired from bookseller *Y* *in 1935*?

In **Q5***, the year is specified as an attribute of the *provider* relation while, in **Q1**, it acts as an additional answer variable that represents an attribute of the *owner* relation. In order to capture **Q1** and **Q5*** fully, the model needs to be extended such that relationships can have attributes. This is straightforward in the graph representation, as the attributes can be added to the respective edge labels. This is shown in Figure 5.8 for the two respective edges of **Q1** and **Q5***.

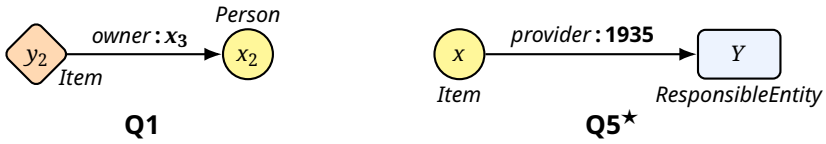


Figure 5.8: Capturing attributes on relationships

On the formal side, the notion of a graph (Definition 5.1) needs to be extended such that the labelling function $\mathcal{L}$ maps each edge to a set that consists not necessarily only of relation names (for relationships without attributes), but which may also contain pairs of relation names and literals/objects/variables (for relationships with attributes). The notion of a homomorphism (Definition 5.4) does not need to be changed, as the current formulation of Condition **H3** continues to ensure that edge labels are preserved.

Binary relations with attributes are in fact a special case of ternary relations: e.g., *owner* in **Q1** is a relation between items, persons, and literals (year numbers). As the computational properties discussed in Section 5.4 hold for relations of higher arity as well, this modest extension of the model can be implemented along with the basic model.

Ternary relations are also necessary to model **Q9** (and the supporting data):

Q9 Where did person *X* acquire items and did they know the previous owners?

In this case, the relation *acquires* needs to involve persons, items, and sellers, and this constellation can no longer be visualised as a graph in a straightforward way. A possible extension is the notion of a *hypergraph* [Voloshin 2009, §7.1], which allows edges of arbitrary arity. While the mathematical object can be defined analogously to standard graphs, it does no longer lend itself to intuitive visualisations. Instead, an SQL-like query language seems better suited but misses the advantages of an intuitive visualisation and requires expert knowledge that humanities scholars do not usually have at hand. As already mentioned, the notion of a homomorphism and the computational properties carry over directly.

Another use case for relations of higher arity is the inclusion of metadata provenance, i.e., information on the origins of a unit of data (such as a concept membership or relationship). In our scenario, it makes sense to distinguish two kinds of metadata provenance. The first is information provided by some data sources such as the GND (see Section 2.1.2), referring to original data sources on whose grounds the respective data unit was entered. The second is the reference to the data source from which the data unit entered the federated data source G' against which the query G is evaluated (see Sections 2.1.4 and 2.2.3). Both kinds of provenance information are reflected

only in the data source and not in the query, and they play a role only in the retrieved answers, enabling users to consult the original data sources for further research.

5.5.3 Answer Variables for Sets of Objects

Query Pattern **Q3** contains a reference to a *set* of objects:

Q3 What are the relationships between the recipients of manifestation Y of work X?

If we use the substitute information "ownership" for "reception", then the expected answer to **Q3** is a set of owners, together with the relationships between them—or, more precisely, the subgraph of the data source induced by the set of all owners. In order to capture this idea, our basic model needs to be extended by *set variables* representing sets of objects and *set nodes* corresponding to set variables. In the case of **Q3**, the set node acts nearly as an answer variable because the query asks for the relationships between the set of owners. In order to distinguish set variables and nodes from ordinary ones in the visualisation, we use overlined letters and octagonal shapes. Figure 5.9 depicts the graph for **Q3**.

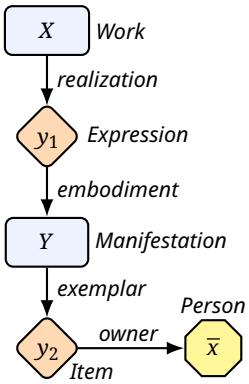


Figure 5.9: Graph representation of **Q3**

Query Pattern **Q9** (see above) also requires a set variable that represents the "previous owners". In contrast to **Q3**, the query does not ask for the relationships between the objects in that set but for the relationships between these objects and person X. These relationships are automatically included in each induced subgraph that is part of a query answer, and they need to be evaluated manually by the user in order to answer the yes/no question involving the relationship *know*, which is very likely not recorded explicitly in data sources.

In order to capture set variables in our definition of query answers, the notion of a homomorphism needs to be extended such that every set node in the query graph G is mapped to a set of nodes in G', under preservation conditions analogous to **H2** and **H3**. While all these extensions are straightforward, their effect on the complexity of query answering is not obvious. Since the number of subsets of a set is exponential in the number of its elements, it is possible that there is no longer an algorithm that requires only polynomial time (with respect to the size of the data). This question remains to be investigated.

Finally, when our basic model is extended with set variables, it can express Requirements **R016**–**R018** from the SoNAR report [Fangerau et al. 2022], which are listed in Section 2.1.2, as queries. The only additional feature required is the use of unlabelled edges in queries in order to capture the (deliberately) unspecified relation "connected to". That addition is minor: homomorphisms simply need to be required to map unlabelled edges in the query graph G' to arbitrary edges in the data source graph G. The computational properties are unaffected because query answering with unlabelled edges can be reduced to the basic query answering problem by adding a fresh relation name to all unlabelled edges in G' and all edges in G.

5.5.4 Further Types of Information

There are some types of information that should be captured in our model or by our general approach, given the specifics of the example query patterns and the insights from the SoNAR project. We briefly discuss them and their effects on modelling.

Substitute information This type of information was discussed in the SoNAR report [Fangerau et al. 2022], see Section 2.1.2. As we have already mentioned in several places in this chapter. Query Patterns **Q1**–**Q9** use several relationships that are typically not recorded in data sources explicitly, such as "read", "recipient of", "passed on to", "once owned", or "knows". Furthermore, **Q4** and **Q2′** use the relation "family member" and the concept "scientist", which are superordinate to terms that are used in data sources (such as "child" or "astronomer").

Whenever one of these terms occurs in a query, the answer set will inevitably be empty. This problem cannot be addressed at the level of the modelling, but rather at the levels of (a) the interaction between user and retrieval system or (b) the retrieval method. Option (a) refers to the way the system supports users in formulating their queries. Independently of the problem of missing relations or concepts, users should be able to select terms from the supply of terms available in the data source(s). If a term that the user wants to use is missing, then the user has no other choice than to select an alternative "substitute" term. In this case, the user should be made aware of the effect of overapproximation described in Section 3.3. Option (b) refers to using semantic technologies such as ontologies in the retrieval method (and of course in the support for formulating queries). Given that ontologies are generally useful in data integration [Doan, Halevy, and Ives 2012, §12], it would be promising to include them in our method. From the modelling perspective, it remains to proceed with caution because, depending on the ontology language used, query answering is not guaranteed to retain its favourable computational properties in the presence of ontologies [Lutz 2016; Xiao et al. 2018; Barceló et al. 2019].

Complex relationships Relationships between two objects that cannot be described using a single relation name have already been mentioned as "indirect relationships" in the context of SoNAR (Section 2.1) and in our discussion of manual versus automated query answering (Section 3.4). For example, the relationship "P_1 and P_2 are students of the same scholar" between two persons P_1 and P_2 does not correspond to a single relation. Instead, it is represented in the data graph via two *student*-edges that point from the nodes representing P_1 and P_2 to a unique node representing some other person (the scholar). Whenever a user wants to query such a complex relationship, they will have to build this small subgraph into their query. Supporting users in constructing queries is again a matter of user interaction, not of the modelling.

Temporal constraints In Subsection 5.5.2, we have already sketched one way of handling temporal information in our model of queries and data graphs. However, there are more complex kinds of temporal information, as can be seen, for example, in the full versions of Query Pattern **Q5**:

> **Q5** Which items from the holdings of library *X* were acquired from bookseller *Y* *between 1933 and 1945*?

In order to capture this query, the syntax for labels in query graphs needs to be extended such that constraints can be formulated as attributes of relationships, for example, by replacing the edge label "*provider* **: 1935**" on the right-hand side of Figure 5.8 with "*provider* **: ⩾ 1933 & ⩽ 1945**."

Temporal constraints can also be useful for modelling relations that are not explicit in the data, such as "passed on to", which occurs in **Q2** and **Q4**. Our approach so far was to consider each pair of owners as a candidate for an instance of this relation, with the risk of returning a large number of spurious answers (see also the discussion in Section 3.4). With temporal constraints, it is possible to narrow down the spurious answers: if dates of ownership are given, then a pair (y_1, y_2) of owners is a candidate only if the ownership of y_1 predates that of y_2. This can be expressed using attributes of the *owner* relation and a "<" constraint between the attributes of the *owner* labels of the edges pointing to y_1 and y_2.

In order to incorporate such constraints, the first step should be to devise a rigorous definition of the form of constraints that are allowed. The next step is to find a useful form of visualisation. Finally, the notion of a homomorphism needs to be extended to preserve constraints, which is

quite straightforward. The computational properties need to be re-examined, but it is realistic to hope that they do not deteriorate because constraints of this kind are also part of standard database query languages such as SQL.

Disjunctions in labels It is conceivable that a researcher wants to formulate a query that contains a disjunction, for example, the following variant of **Q4**.

Q4′ Which items from a collection X were passed on by its owner to *a child or a student*?

This example uses a disjunction of the relations *child* and *student*. It should be quite straightforward to extend the syntax for labels in query graphs and the notion of a homomorphism in order to represent and preserve this kind of constraint. The computational properties are not affected by adding disjunction, as Vardi's result (Theorem 5.6) applies in fact to arbitrary queries formulated in first-order logic.

5.5.5 Problematic Queries

So far, we have not addressed Query Patterns **Q6** and **Q7** in this chapter:

Q6 Who participated in the sale of collection X?

Q7 Via which paths did items from collection X enter library Y?

The reason is that they contain complex features which need to be resolved outside the scope of our model: The participation in a sale (**Q6**) is a very general notion that needs to be specified *before* formulating the query and attempting to match it to the available data. The paths of ownership featuring in **Q7** are a complex notion that cannot be captured by a simple query; instead, the query should simply ask for owners, and the paths need to be reconstructed manually *after* the retrieval system has returned the answers.

We also have not yet discussed the requirements identified by the SoNAR report and listed in Section 2.1.2, other than **R016**–**R018**. Most of these remaining requirements are of a rather exploratory nature (in particular, **R029**, **R030**, and **R061**), and they therefore need to be discussed apart from the topic of modelling. The topic of exploration should be dealt with in a separate thesis or paper.

6 A Method for Retrieving Provenance Relationships

In the previous chapter, we have developed a model for answering provenance queries. It is the vision of this thesis that, ultimately, this model is implemented in a retrieval system that answers provenance queries conforming to our model. The primary user group for that system consists of provenance researchers and historians who want to investigate, for example, the history of items and collections or the social networks of historically important figures.

In this chapter, we design a method that serves as a the general basis for a retrieval system. This method abstracts away from specific details such as programming paradigms, languages, libraries, or data structures; instead, it represents a high-level specification for future implementations.

In the discussion of the insights from the SoNAR project in Section 2.1.4, we have suggested a distinction between a dynamic and a static setting. The method that we are going to develop should be flexible and support both settings. For this purpose, we make this distinction more explicit. The *static setting* resembles the setting used in the SoNAR project: a large uniform data source *(data graph)* is constructed from the various identified data sources *(remote repositories)*, using a uniform data model that integrates the heterogeneous data models of the remote repositories in some way. This data graph is used as the sole source for computing query answers. In contrast, the *dynamic setting* foregoes the explicit construction of a uniform data graph and instead regards a collection of remote repositories as a representation of that graph. In order to answer a query, the system identifies suitable repositories, computes partial answers using the repositories' interfaces, and composes the final answer from the partial answers.

Our method will consist of two main phases, the second of which is divided into three subphases. We describe these phases in Section 6.1, addressing special requirements of the dynamic or static setting. In Section 6.2, we compare the advantages and disadvantages of the two settings.

6.1 The Main Phases

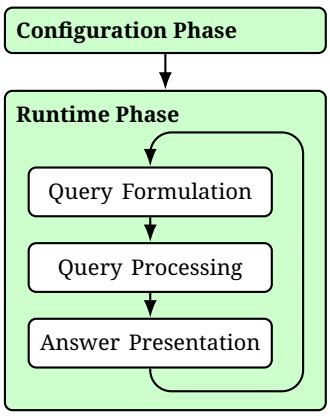


Figure 6.1: Main phases of the retrieval method

Our method consists of a *Configuration Phase*, which needs to be carried out before users can interact with the system, and a *Runtime Phase*, where the actual user interaction takes place. The latter is divided into three consecutive subphases as shown in Figure 6.1. In the *Query Formulation* phase, the system supports the user in constructing their query. In the *Query Processing* phase, the system evaluates the query and computes answers. In the *Answer Presentation* phase, the system allows the user to navigate through the computed answers. Given the significance of *exploration* in historical research (see Section 2.1.2), the answers presented by the system may prompt the user to reformulate their original query. In fact, query answers can be seen as one means of support for formulating a query in the first place. For this reason, we prefer to understand the Runtime Phase as an iterative process, which we indicate by the looping arrow in Figure 6.1. We now describe these phases in more detail.

6.1.1 The Configuration Phase

The Configuration Phase consists of the steps that need to be taken in order for the system to be able to interact with the user in the Runtime Phase. The first step is the selection of remote repositories that are available throughout the Runtime Phase. In the dynamic setting, where the data graph is implicit in this collection of repositories, it is necessary that the details of accessing each single repository have been implemented. In the static setting, the data graph needs to be built from the remote repositories and stored in memory, in a relational database management system (RDBMS), or on some suitable storage medium. As a final step, a list of terms (objects, concepts, relations) available in the data graph needs to be compiled in both settings. Ideally, each term should be annotated with the remote repository (or repositories) from which it originates, in order to trace it back for disambiguation and other purposes.

6.1.2 The Runtime Phase

The Runtime Phase is the phase in which the system receives queries from the user, computes answers, and presents them to the user. It is thus divided into three subphases.

Query Formulation In the Query Formulation Phase, the user builds the query to be answered. The system should provide support via a query editor. Since we opted for a graph-based model that facilitates intuitive visual representations, it makes perfect sense to offer a graphical interface that allows the user to create nodes, draw edges between them, and pick node names, node labels, and edge labels from the list of available terms. For disambiguation purposes, it would be helpful to have an integrated lookup functionality, i.e., the entry for a term in the original remote repository could be displayed in the system without switching to another application.

In this phase, the system should also provide support for substitute information and indirect relationships (see Section 5.5.4), i.e., for finding terms to substitute for terms that are not recorded in the data, and for constructing subgraphs of the query graphs that represent indirect relationships. For example, suggestions for the former and “templates” for the latter could be deposited in the system either through the implementation or during the Configuration Phase.

In the dynamic setting, it is possible to let the user restrict the selection of remote repositories and thus the list of available terms. This initial step is not useful in the static setting because it would require a time-consuming recomputation of the data graph (as part of the Configuration Phase).

If ternary relations or relations of higher arity play a significant role, then a visualisation is no longer feasible, as explained in Section 5.5.2. In this case, it is possible to resort to a (more technical) SQL-like syntax. The system should then support non-computer scientist users via a graphical query editor allowing to pick, arrange, and link the basic ingredients of queries as blocks. There are numerous online and standalone graphical SQL query builders [e.g., Web108; Web109], see also the reviews on dedicated webpages [e.g., Web110; Web111].

A possible alternative to a graphical interface that is worth considering is a text interface that allows the user to formulate their queries in natural language and translates them into SQL or SPARQL queries. For SPARQL, a framework for transforming complex natural-language queries to SPARQL queries has been developed very recently by Boumechaal and Boufaida [2023] in the context of linked data.

Query Processing In the Query Processing Phase, the system evaluates the query against the data graph. In the technical terms of our model, the system finds all homomorphisms from the query graph to the data graph and stores them along with the induced subgraphs.

In the static setting, where the data graph has been built and stored during the configuration phase,

the system has full control over the data. Concerning the implementation of query answering, the static setting leaves several options: the most ambitious one is certainly the implementation of a dedicated new algorithm, which inevitably requires extensive optimisations in order to perform well on large amounts of data. A second option is the re-implementation of existing algorithms for query answering in RDBMSs or graph databases (such as GraphDB [Web18]), exploiting the strong relationship between our query answering problem and the classical query answering problem (see Section 5.4). This "glass-box" approach to reusing existing implementations promises full control over the algorithm and thus allows optimisations tailored towards our specific query answering problem. Furthermore, numerous implementations are available [cf. Web112]. However, the downside is that a highly optimised, and thus highly complex, codebase needs to be adapted to our retrieval system, which is a highly non-trivial task. In addition, not all systems make their code freely available. A third option is the use of an existing RDBMS or graph database engine as a "black box", which requires the transfer of the data graph into that external system in the Configuration Phase, the translation of each query into an SQL or SPARQL query, the invocation of the external engine, and the retranslation of the retrieved set of answers into an answer set in our formalism, i.e., the reconstruction of homomorphisms and subgraphs. The implementation effort is restricted to realising these translations, and it is not possible to add specific optimisations.

In the dynamic setting, where the data graph is implicit in the collection of remote repositories, the system needs to interact with the repositories in real time via their interfaces. For this purpose, it is important to keep track of the repositories from which each term originates, as indicated in Subsection 6.1.1. Using this information, the query needs to be decomposed into smaller queries, and each of these needs to be answered against the respective repository. The obtained answers need to be combined to an answer set for the original query. Although the abstract description of this procedure sounds very simple, at least two parts of it require a substantial amount of work: First, it remains to find a rigorous definition of a decomposition and recomposition that provides the guarantee that the answer set obtained via this procedure coincides with the definition of an answer set with respect to the abstract notion of a data graph. Second, the evaluation of the smaller queries requires either a uniform standard (such as SPARQL, see Section 2.2.1) or a dedicated algorithm for each single remote repository and data format.

Answer Presentation In the Answer Presentation Phase, the answer set is shown to the user. Given the importance of exploration (see Sections 2.1.2 and 2.1.4) and the resulting decision that answers include the respective subgraph of the data as a "witness", a visual presentation is to be favoured. This visual presentation should allow for means to navigate through the subgraph and beyond, i.e., to explore its context in the data graph.

Furthermore, the size of an answer set can vary greatly: for "overspecified" queries, the answer set can be empty, and for very vaguely specified queries, the answer set can be even larger than the data graph. For example, if the query **Q2′** is modified to ask for a scientist who passed the item on to someone who was a student and family member at the same time, and if the owners of the item do not include two persons in those two relationships, then the answer set will be empty. The other extreme occurs, for example, if the query asks for pairs of owners that are collaborators of each other, which is a symmetric relationship. If the data contains n owners of the item and those are all in a mutual collaboration relationship, then the answer set consists of $n^2 - n$ answers because there are $n \cdot (n-1)$ pairs of distinct owners. The use of unlabelled edges (or nodes), which was mentioned as a useful extension of the model in Section 5.5.3, can have a similar effect.[9] In order to cope with the wide range of possible sizes of answer sets, ways to present answer sets in a structured way, allowing the user to group and filter answers, have to be found.

[9] In the theoretical worst case, there are pairs (G, G') of query and data graphs where each possible mapping from the nodes in G to the nodes in G' is a homomorphism. The number of such mappings is n^k, where n, k are the numbers of nodes in G and G', respectively. Given the dominating size of the data, this upper bound is prohibitively large even for very small queries.

Since we are advocating an iterative runtime phase where query answers can help users (re)formulate their queries, the presentation of query answers should include an assistant for marking unexpected or missing answers. Ideally, that assistant could suggests changes to the user's query based on these marks.

6.2 The Dynamic versus the Static Setting

We deliberately developed our method such that it can be applied in both the dynamic and the static setting. Both settings have advantages and disadvantages, and we have already learnt about some of them for the static setting from the SoNAR project (see Section 2.1.2). We now discuss the advantages, disadvantages, and common challenges of both settings, thus providing future system developers with a basis for deciding on one setting.

Advantages of the Static Setting The most important advantage of the static setting is the interaction with a single, fixed data source and data model during the Runtime Phase. The data graph is self-hosted, allowing the system full control over the data and the method of access. Thanks to the static nature, data access during the Runtime Phase is unaffected by any unforeseen unavailability of a remote repository or by changes to the external data models. Furthermore, possible inconsistencies between repositories can be resolved in the Configuration Phase, after the data graph has been built. Hiding this process from the Runtime Phase should contribute to a better user experience. Finally, as we have seen, the static setting also offers more freedom for implementations of query answering, although only the "black-box" approach seems feasible.

Disadvantages of the Static Setting In the static setting, the federated data graph requires a large amount of memory and a highly performant, scaleable implementation of query answering. The data graph also needs to be updated regularly in order to incorporate changes to the remote repositories. If it is desirable to give the user the opportunity to select repositories that are relevant for his*her query during the Runtime Phase, then additional efforts are necessary, including annotations of the data with their provenance and incorporating appropriate filters into the implementation of query answering.

Advantages of the Dynamic Setting In the dynamic setting, our method does not depend on hosting capacities such as a large amount of memory or a highly scalable query answering implementation. Furthermore, by accessing the remote repositories directly, the method will always have access to their current content. Finally, it is relatively easy to implement support for a restriction of the repositories used for answering a particular query, which can improve user experience by helping restrict the set of terms to choose from.

Disadvantages of the Dynamic Setting On the downside, the dynamic setting depends on the web services provided by the remote repositories, which means that there is no control over either the data or the time required by the interaction with the system. In the dynamic setting, the Runtime Phase is also sensitive to changes in the data models or interfaces of the remote repositories. If inconsistencies occur between the data in distinct repositories, then they need to be resolved in the Runtime Phase, too. This can be done autonomously by the system during the Query Processing Phase or through interaction with the user during the Answer Presentation Phase. Both cases are problematic because neither the system nor the user can be expected to have sufficient knowledge of the external repositories to take the necessary decisions.

An open problem of the dynamic setting is the decomposition of the query into smaller queries to be posed to the single repositories and the subsequent recomposition of the partial answers. As already mentioned above, it remains to find a formal framework with rigorous guarantees. It is conceivable that these guarantees can only be achieved by an iterative approach, which would

require a larger amount of interaction with the remote repositories. Finally, this distributed query answering would require either an implementation of several query answering routines specific to the respective repository, or a restriction to repositories offering a standardised query answering interface, such as a SPARQL endpoint.

Challenges for Both Settings In both settings, inconsistencies between the remote repositories need to be dealt with. For example, if two data sources give different years for the date of birth of the same person or for the ownership entry for the same person in the same item, then they will both induce different answers to queries that involve these specific pieces of data. The easiest way for implementers is to present all answers to the user and let them decide which answers they trust. However, this variant is not user-friendly because it tends to inflate answer sets and leaves the decision about which is the correct piece of data to the user. It would be more beneficial to user experience if the system were able to recognise (potential) inconsistencies and if domain experts could be involved in resolving those. The topic of inconsistencies should be studied in more depth; a first step should be a rigorous definition of what constitutes an inconsistency.

Another challenge is the system's reaction to changes in the data models used by the remote repositories. Every single repository and data model requires an implementation effort, and changes require a modification of the existing implementation. This fact is independent on the setting, but it affects different phases of the method (the Configuration Phase in the static setting and mostly the Query Processing Phase in the dynamic setting).

From both challenges, we conclude that the retrieval system cannot simply be shipped to the user as a standalone application but will require constant maintenance by both domain experts and implementers. This need is probably best served by the format of a web application that is hosted and maintained on a central server.

7 Conclusion

7.1 Summary of the Results

In this thesis, we have examined the problem of modelling and automated retrieval of provenance relationships that require information which is usually distributed over several data sources. We have provided a review of the related literature, a case study with example queries, and a review of available standards, data sources, and techniques for data exchange and integration. As the main contribution, we have developed an abstract and general model for queries, data sources, and query answers, and we have devised a method for implementing this model in an answer retrieval system, together with a discussion of various design choices.

In order to return to the initial research question and its subquestions, we list them here once more.

RQ	**How can provenance relationships be modelled and automatically retrieved?**

This question implies several subordinate questions:

RQ1	*What is the state of research on infrastructures for the automated retrieval of provenance relationships? Which approach(es) is/are most closely related?*
RQ2	*What are the general challenges for answering queries such as* **Q1**–**Q9***, and what are the specific challenges for an automated approach?*
RQ3	*Which data sources, standards, data formats, and further tools are available for answering provenance queries using multiple, heterogeneous data sources?*
RQ4	*Based on the structure of the identified data sources, how can data sources, queries, and query answers be modelled in an abstract framework?*
RQ5	*What is a suitable method for retrieving provenance relationships in that framework?*

We now summarise the insights obtained in this thesis relating to each of these questions.

RQ1 (Chapter 2) Our literature review has shown that social network analysis and historical network analysis are generic methods closely related to the goal of this thesis; the general context of social and historical research produces research questions similar to those underlying provenance research. The SoNAR project is a rich source of relevant insights concerning the support for historical research by research data infrastructures. While network analysis aims at answering questions of a “global” nature, our scenario covers more “local” questions.

Furthermore, our review has also shown that linked (open) data and data integration techniques have been used extensively in the library domain, and that the state of provenance indexing varies greatly between, and sometimes even within, institutions and library networks.

RQ2 (Chapter 3) The general challenges for answering queries such as **Q1**–**Q9** include several sources for missing and spurious answers. A prominent reason is the mismatch between natural-language terms for concepts or relations and the vocabulary of the data sources. A possible remedy is the use of “substitute information”, hypotheses, or reasoning. This problem carries even more

weight in an automated approach, where a machine cannot easily rely on human intuition or experience for identifying substitute information and hypothesising.

RQ3 (Chapter 4) There is a large number of bibliographic standards, bibliographic and generic data formats, communication protocols, and data sources. We have identified four groups of altogether 22 obvious data sources, and this list is by far not exhaustive. Our detailed analysis of four of the 22 data sources shows that, in particular, linked (open) data and the relevant interfaces are provided by all of them. In order to implement a prototype of a retrieval system that is supposed to provide a proof of concept, it certainly suffices to include these four data sources. However, a fully-fledged retrieval system should probably include a substantially larger selection of data sources; in this case, our analysis will need to be extended accordingly.

RQ4 (Chapter 5) We have developed a basic, conceptually simple, graph-based model for data sources, queries, and query answers, and sketched several extensions. For the basic version and some of the extensions, the query answering problem has the same good computational properties as standard query answering over relational databases. However, some desirable features in queries may require extensions of the model that no longer lend themselves to intuitive visualisations; in order to capture these features, an SQL-like syntax could be appropriate. Finally, some aspects used in queries cannot be captured in any formal model and require manual intervention.

RQ5 (Chapter 6) We have developed an abstract method consisting of two main phases and three subphases. The decision between the dynamic and the static setting is a fundamental design choice, as it affects all phases. The comparison of the two settings does not yield a clear tendency towards one or the other. The challenges that both systems have in common imply that the system needs to be constantly maintained by domain experts and implementers, which favours a web application. The developed method is a high-level specification for a future implementation of a retrieval system, and many details still remain be clarified or studied further.

RQ There is neither a single model nor a single method, but our graph-based model and two-phase method seem suitable. The abstract model captures most of the exemplary questions, and the general method has a number of parameters that need to be set or require further study.

7.2 Outlook

As immediate next steps, further data sources should be analysed, and the generic method should be elaborated in more detail. Subsequently, a prototype retrieval system should be implemented and tested. Ideally, this prototype is then turned into a fully-fledged web application.

The work reported here can be continued in a number of further directions: A systematic quantitative analysis of relevant queries and data sources in the form of an extensive user study would help underpin our ad-hoc selection of example queries as well as, hopefully, the abstract model. The testing phase for the prototype system could be integrated in this study, leading to an agile development cycle. The testing of the system should also include a pilot study on the correctness and completeness of query answers with expert users in order to validate and improve the overall approach. Furthermore, the system could be integrated in the research infrastructure in the context of the SoNAR project or in large portals such as Proveana or Europeana. Finally, the use of semantic web technologies such as ontologies and reasoning should be explored in detail.

References

Bibliography

Abiteboul, Serge, Richard Hull, and Victor Vianu (1995). *Foundations of databases*. Addison-Wesley (cit. on p. 30).

Alemu, Getaneh et al. (2012). "Linked Data for libraries: Benefits of a conceptual shift from library-specific record structures to RDF-based data models". In: *New Library World* 113.11-12, pp. 549–570. DOI: https://doi.org/10.1108/03074801211282920 (cit. on p. 12).

Arora, Sanjeev and Boaz Barak (2009). *Computational Complexity – A Modern Approach*. Cambridge University Press. URL: http://www.cambridge.org/catalogue/catalogue.asp?isbn=9780521424264 (cit. on p. 37).

Baader, Franz et al. (2017). *An Introduction to Description Logic*. Cambridge University Press. URL: http://www.cambridge.org/de/academic/subjects/computer-science/knowledge-management-databases-and-data-mining/introduction-description-logic?format=PB#17zVGeWD2TZUeu6s.97 (cit. on p. 11).

Bair, Sheila, ed. (2013a). *Journal of Library Metadata* 13.2–3. Special issue "Linked Data, Semantic Web and Libraries". URL: https://www.tandfonline.com/toc/wjlm20/13/2-3 (cit. on p. 12).

– (2013b). "Linked Data—The Right Time?" In: *Journal of Library Metadata* 13.2-3, pp. 75–79. DOI: 10.1080/19386389.2013.828527 (cit. on p. 12).

Barceló, Pablo et al. (2019). "When is Ontology-Mediated Querying Efficient?" In: *34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pp. 1–13. DOI: 10.1109/LICS.2019.8785823 (cit. on p. 42).

Berners-Lee, Tim, Roy T. Fielding, and Larry M. Masinter (2005-01). *Uniform Resource Identifier (URI): Generic Syntax*. RFC 3986. DOI: 10.17487/RFC3986 (cit. on pp. 11, 64).

Berners-Lee, Tim, James Hendler, and Ora Lassila (2001). "The Semantic Web". In: *Scientific American* 284.5, pp. 34–43. DOI: 10.1038/scientificamerican0501-34 (cit. on p. 11).

Bizer, Christian, Tom Heath, and Tim Berners-Lee (2009). "Linked Data – The Story So Far". In: *International Journal on Semantic Web and Information Systems* 5.3, pp. 1–22 (cit. on p. 12).

Bludau, Mark-Jan et al. (2020). "SoNAR (IDH): Datenschnittstellen für historische Netzwerkanalyse". In: *7. Tagung Digital Humanities im deutschsprachigen Raum (DHd 2020)*. Ed. by Tara Andrews et al. DOI: 10.5281/zenodo.4621861 (cit. on pp. 6, 64).

Boumechaal, Hasna and Zizette Boufaida (2023). "Complex Queries for Querying Linked Data". In: *Future Internet* 15.3. DOI: 10.3390/fi15030106 (cit. on p. 45).

Brodie, M. L., J. Mylopoulos, and J. W. Schmidt, eds. (1984). *On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases, and Programming Languages*. 1st. Topics in Information Systems. Springer (cit. on p. 30).

BSZ and VZG (2022-05). *Normdaten*. German. Handbook for using and creating GND authority data in K10plus, linked in K10plus Wiki. URL: https://opus.k10plus.de/frontdoor/deliver/index/docId/410/file/K10plus_Normdaten.pdf (visited on 2023-05-29) (cit. on p. 27).

Burrows, Toby et al. (2021). “A New Model for Manuscript Provenance Research: The Mapping Manuscript Migrations Project”. In: *Manuscript Studies* 6.1, pp. 131–144. URL: https://repository.upenn.edu/mss_sims/vol6/iss1/5/ (cit. on p. 12).

Byrne, Gillian and Lisa Goddard (2010). “The Strongest Link: Libraries and Linked Data”. In: *D-Lib Magazine* 16.11/12. DOI: doi:10.1045/november2010-byrne (cit. on p. 12).

Chandra, Ashok K. and Philip M. Merlin (1977). “Optimal Implementation of Conjunctive Queries in Relational Data Bases”. In: *Proceedings of the 9th Annual ACM Symposium on the Theory of Computing (SToC)*. Ed. by John E. Hopcroft, Emily P. Friedman, and Michael A. Harrison. ACM, pp. 77–90. DOI: 10.1145/800105.803397 (cit. on p. 38).

Copernicus, Nicolaus (1543). *Nicolai Copernici Torinensis De revolutionibus orbium coelestium, Libri VI*. Latin. Ed. by Andreas Osiander. Nürnberg: Petreius, Johann, [6], 196 sheets. URL: https://opac.uni-erfurt.de/LNG=EN/DB=1/PPNSET?PPN=567506266 (cit. on p. 15).

Danowski, Patrick and Adrian Pohl, eds. (2013). *(Open) Linked Data in Bibliotheken*. German. Berlin, Boston: De Gruyter Saur. DOI: doi:10.1515/9783110278736 (cit. on p. 12).

Della Valle, Emanuele and Stefano Ceri (2011). “Querying the Semantic Web: SPARQL”. In: *Handbook of Semantic Web Technologies*. Ed. by John Domingue, Dieter Fensel, and James A. Hendler. Berlin, Heidelberg: Springer. Chap. 8, pp. 299–363 (cit. on pp. 12, 30).

Diestel, Reinhard (2012). *Graph Theory*. 4th. Vol. 173. Graduate texts in mathematics. Springer (cit. on p. 30).

Doan, AnHai, Alon Y. Halevy, and Zachary G. Ives (2012). *Principles of Data Integration*. Morgan Kaufmann. DOI: https://doi.org/10.1016/C2011-0-06130-6 (cit. on pp. 10, 11, 13, 29, 42).

Domingue, John, Dieter Fensel, and James A. Hendler, eds. (2011). *Handbook of Semantic Web Technologies*. Berlin, Heidelberg: Springer. DOI: 10.1007/978-3-540-92913-0 (cit. on pp. 10, 11).

Eckert, Kai (2012). “Metadata Provenance in Europeana and the Semantic Web”. German. Master’s thesis. Humboldt-Universität zu Berlin. DOI: http://dx.doi.org/10.18452/14172 (cit. on p. 13).

– (2013a). “Die Provenienz von Linked Data”. German. In: *(Open) Linked Data in Bibliotheken*. Ed. by Patrick Danowski and Adrian Pohl. Berlin, Boston: De Gruyter Saur, pp. 97–121. DOI: doi:10.1515/9783110278736.97 (cit. on p. 13).

– (2013b-09). “Provenance and Annotations for Linked Data”. In: *International Conference on Dublin Core and Metadata Applications*, pp. 9–18. URL: https://dcpapers.dublincore.org/pubs/article/view/3669 (cit. on p. 13).

Ehrlinger, Lisa and Wolfram Wöß (2016). “Towards a Definition of Knowledge Graphs”. In: *Joint Proceedings of SEMANTiCS 2016 and SuCCESS 2016, Posters and Demos Track*. Ed. by Michael Martin, Martí Cuquet, and Erwin Folmer. Vol. 1695. CEUR Workshop Proceedings. CEUR-WS.org. URL: https://ceur-ws.org/Vol-1695/paper4.pdf (cit. on p. 30).

Even, Shimon (2012). *Graph algorithms*. Ed. by Guy Even and Richard M. Karp. 2nd. Literaturangaben. Cambridge [u.a.]: Cambridge Univ. Press. XII, 189 (cit. on p. 30).

Facharbeitsgruppe Sacherschließung, ed. (1995). *Basisklassifikation für den Bibliotheksverbund Niedersachsen/Sachsen-Anhalt/Thüringen*. German. 2nd, revised edition. Göttingen. URL: https://kxp.k10plus.de/DB=2.1/DB=2.1/PPNSET?PPN=192887122 (cit. on pp. 27, 63).

Fangerau, Heiner et al. (2022-01). *SoNAR AP 2*. German. Final report on Work Package 2 of the project. URL: https://github.com/sonar-idh/reports/blob/main/AP2-UDK-Projektdokumentation.pdf (cit. on pp. 6, 41, 42).

Freire, Nuno et al. (2019). “Aggregation of Linked Data in the Cultural Heritage Domain: A Case Study in the Europeana Network”. In: *Information* 10.8. DOI: 10.3390/info10080252 (cit. on p. 13).

Galety, Mohammad Gouse et al., eds. (2022). *Social network analysis: theory and applications*. Hoboken, NJ (cit. on p. 30).

German National Library (2022-06). *Jahresbericht 2021*. German. Annual report 2021 by the DNB. URL: https://d-nb.info/1257467816/34 (cit. on pp. 26, 27).

Gruber, Christine and Eveline Wandl-Vogt (2017). "Mapping historical networks: Building the new Austrian Prosopographical / Biographical Information System (APIS). Ein Überblick". German. In: *Europa baut auf Biographien. Aspekte, Bausteine, Normen und Standards für eine europäische Biographik*. Ed. by Ágoston Bernad, Christine Gruber, and Maximilian Kaiser. new academic press (cit. on p. 10).

Hakelberg, Dietrich (2016). "Herkunft finden und vernetzen: Stand und Perspektiven der Provenienzerschließung mit Normdaten". German. Master's thesis. Humboldt-Universität zu Berlin (cit. on pp. 1, 3, 13, 24, 27, 29).

Hartig, Olaf (2009). "Provenance Information in the Web of Data". In: *Proceedings of the Second Workshop on Linked Data on the Web (LDOW 2009)* (cit. on p. 13).

Hauser, Julia (2014). "Der Linked Data Service der Deutschen Nationalbibliothek". German. In: *Dialog mit Bibliotheken* 26.1, pp. 38–42 (cit. on p. 13).

Hengel, Christel and Barbara Pfeifer (2005). "Kooperation der Personennamendatei (PND) mit Wikipedia." German. In: *Dialog mit Bibliotheken* 17.3, pp. 18–24 (cit. on p. 12).

Hider, Philip and Ross Harvey (2008). *Organising Knowledge in a Global Society: Principles and Practice in Libraries and Information Centres*. Topics in Australasian Library and Information Studies 29. Wagga Wagga: Centre for Information Studies, Charles Stuart University. URL: http://ebookcentral.proquest.com/lib/huberlin-ebooks/detail.action?docID=1639577 (cit. on pp. 20–23).

Horrocks, Ian and Peter F. Patel-Schneider (2011). "KR and Reasoning on the Semantic Web: OWL". In: *Handbook of Semantic Web Technologies*. Berlin, Heidelberg: Springer, pp. 365–398 (cit. on p. 11).

IFLA Study Group on the Functional Requirements for Bibliographic Records (2009-02). *Functional Requirements for Bibliographic Records. Final report*. Ed. by International Federation of Library Associations and Institutions (IFLA). München. URL: https://repository.ifla.org/handle/123456789/811 (cit. on pp. 2, 20, 32, 63).

IFLA Working Group on Functional Requirements and Numbering of Authority Records (FRANAR) (2008-12). *Functional Requirements for Authority Data. A Conceptual Model*. Ed. by International Federation of Library Associations and Institutions (IFLA). URL: http://www.ifla.org/files/assets/cataloguing/frad/frad_2013.pdf (cit. on pp. 20, 63).

IFLA Working Group on Functional Requirements for Subject Authority Records (FRSAR) (2010-06). *Functional Requirements for Subject Authority Data (FRSAD). A Conceptual Model*. Ed. by International Federation of Library Associations and Institutions (IFLA). URL: https://repository.ifla.org/handle/123456789/835 (cit. on pp. 20, 63).

Isaac, Antoine, Robina Clayphan, and Bernhard Haslhofer (2012). "Europeana: Moving to Linked Open Data". In: *Information Standards Quarterly* 24.2/3, p. 34. DOI: 10.3789/isqv24n2-3.2012.06 (cit. on p. 13).

Jansen, Dorothea (2003). *Einführung in die Netzwerkanalyse. Grundlagen, Methoden, Forschungsbeispiele*. German. 2nd. UTB 2241. Opladen: Leske + Budrich. 312 pp. (cit. on p. 5).

Lígia Triques, Maria, Paula Regina Ventura Amorim Gonçalez, and Ana Cristina de Albuquerque (2022). "Integration of cultural data from digital repositories: an overview of the DPLA Hubs."

In: *Revista Digital de Biblioteconomia e Ciência da Informação* 20, pp. 1–21. DOI: 10.20396/rdbci.v20i00.8666967 (cit. on p. 13).

Lutz, Carsten (2016). "Complexity and Expressive Power of Ontology-Mediated Queries (Invited Talk)". In: *33rd Symposium on Theoretical Aspects of Computer Science (STACS)*. Ed. by Nicolas Ollinger and Heribert Vollmer. Vol. 47. Leibniz International Proceedings in Informatics (LIPIcs). Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2:1–2:11. DOI: 10.4230/LIPIcs.STACS.2016.2 (cit. on p. 42).

Malpas, Jeffrey Edward and Hans-Helmuth Gander, eds. (2015). *The Routledge companion to hermeneutics*. Routledge Philosophy Companions. London: Routledge. 778 p (cit. on p. 6).

Marshall, Catherine C. and Frank M. Shipman (2003). "Which semantic web?" In: *UK Conference on Hypertext* (cit. on p. 11).

Meiners, Ole (2022). "Evaluation von Named Entity Recognition-Modellen für historische deutschsprachige Texte am Beispiel frühneuzeitlicher Ego-Dokumente". German. Bachelor's thesis. Humboldt-Universität zu Berlin. 62 pp. (cit. on p. 8).

Menzel, Sina, Mark-Jan Bludau, et al. (2020). "Graph Technologies for the Analysis of Historical Social Networks Using Heterogeneous Data Sources". In: *Graph Technologies in the Humanities 2020 (GRAPH 2020)*. Vol. 3110. CEUR Workshop Proceedings, pp. 124–149 (cit. on pp. 5, 6, 8–10, 64).

Menzel, Sina, Hannes Schnaitter, et al. (2021). "Named Entity Linking mit Wikidata und GND – Das Potenzial handkuratierter und strukturierter Datenquellen für die semantische Anreicherung von Volltexten". In: *Qualität in der Inhaltserschließung*. Ed. by Michael Franke-Maier et al. Berlin, Boston: De Gruyter Saur, pp. 229–258. DOI: doi:10.1515/9783110691597-012 (cit. on p. 8).

Moreau, Luc, Juliana Freire, et al. (2008). "The Open Provenance Model: An Overview". In: *Provenance and Annotation of Data and Processes*. Ed. by Juliana Freire, David Koop, and Luc Moreau. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 323–326 (cit. on p. 13).

Moreau, Luc, Paul Groth, et al. (2008-04). "The Provenance of Electronic Data". In: *Commun. ACM* 51.4, pp. 52–58. DOI: 10.1145/1330311.1330323 (cit. on p. 13).

Novak, Jasminko et al. (2014). "HistoGraph – A Visualization Tool for Collaborative Analysis of Networks from Historical Social Multimedia Collections". In: *18th International Conference on Information Visualisation*, pp. 241–250. DOI: 10.1109/IV.2014.47 (cit. on p. 10).

Otte, Evelien and Ronald Rousseau (2002). "Social network analysis: a powerful strategy, also for the information sciences". In: *Journal of Information Science* 28.6, pp. 441–453. DOI: 10.1177/016555150202800601 (cit. on p. 5).

Petras, Vivien and Juliane Stiller (2017). "A Decade of Evaluating Europeana – Constructs, Contexts, Methods & Criteria". In: *Proceedings of the 21st International Conference on Theory and Practice of Digital Libraries (TPDL)*. Ed. by Jaap Kamps et al. Vol. 10450. Lecture Notes in Computer Science. Springer, pp. 233–245. DOI: 10.1007/978-3-319-67008-9_{1}{9} (cit. on p. 13).

Pohl, Adrian and Patrick Danowski (2013). "Linked Open Data in der Bibliothekswelt: Grundlagen und Überblick". German. In: *(Open) Linked Data in Bibliotheken*. Ed. by Patrick Danowski and Adrian Pohl. Berlin, Boston: De Gruyter Saur, pp. 1–44. DOI: doi:10.1515/9783110278736.1 (cit. on p. 12).

Purschwitz, Anne (2018-12). "Netzwerke des Wissens – Thematische und personelle Relationen innerhalb der halleschen Zeitungen und Zeitschriften der Aufklärungsepoche (1688–1818)". German. In: *Journal of Historical Network Research* 2.1, pp. 109–142. URL: http://jhnr.uni.lu/index.php/jhnr/article/view/47 (cit. on p. 10).

Schneider-Kempf, Barbara et al. (2018). *Beschreibung des Vorhabens – Projektanträge im Bereich "Wissenschaftliche Literaturversorgungs- und Informationssysteme" (LIS)*. Project proposal. URL: https://github.com/sonar-idh/reports/blob/main/SoNAR-Projektantrag-short.pdf (cit. on p. 25).

Ullah, Irfan et al. (2018-12). "An Overview of the Current State of Linked and Open Data in Cataloging". In: *Information Technology and Libraries (Online)* 37.4, pp. 47–80. DOI: 10.6017/ital.v37i4.10432 (cit. on p. 13).

Vardi, Moshe Y. (1982). "The Complexity of Relational Query Languages (Extended Abstract)". In: *Proceedings of the 14th Annual ACM Symposium on the Theory of Computing (SToC)*. Ed. by Harry R. Lewis et al. ACM, pp. 137–146. DOI: 10.1145/800070.802186 (cit. on p. 38).

Voloshin, Vitaly I. (2009). *Introduction to graph and hypergraph theory*. Nova Science Publishers. URL: https://ebookcentral.proquest.com/lib/huberlin-ebooks/detail.action?docID=3019796 (cit. on p. 40).

Vorndran, Angela (2018-12). "Hervorholen, was in unseren Daten steckt! Mehrwerte durch Analysen großer Bibliotheksdatenbestände". German. In: *o-bib* 5.4, pp. 166–180. DOI: 10.5282/o-bib/2018H4S166-180 (cit. on p. 10).

Wache, Holger et al. (2001). "Ontology-Based Integration of Information – A Survey of Existing Approaches". In: *Proceedings of the IJCAI'01 Workshop on Ontologies and Information Sharing*. Ed. by A . Gómez-Pérez et al. URL: https://ceur-ws.org/Vol-47/wache.pdf (cit. on p. 11).

Warren, Christopher N. et al. (2016). "Six Degrees of Francis Bacon: A Statistical Method for Reconstructing Large Historical Social Networks". In: *Digital humanities quarterly* 10.3 (cit. on p. 10).

Wiesenmüller, Heidrun and Silke Horny (2015). *Basiswissen RDA*. German. Berlin, München, Boston: De Gruyter Saur. DOI: doi:10.1515/9783110311471 (cit. on pp. 20, 21).

Xiao, Guohui et al. (2018). "Ontology-Based Data Access: A Survey". In: *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*. IJCAI Organization, pp. 5511–5519. DOI: 10.24963/ijcai.2018/777 (cit. on p. 42).

Zuschlag, Christoph (2022). *Einführung in die Provenienzforschung*. German. München: C.H.BECK (cit. on p. 1).

Web Resources

[Web1] U.S. Department of State. *Washington Conference Principles on Nazi-Confiscated Art*. URL: https://web.archive.org/web/20170426113213/https://www.state.gov/p/eur/rt/hlcst/270431.htm (visited on 2023-05-12) (cit. on p. 1).

[Web2] Wikimedia Foundation. *Wikidata*. URL: https://www.wikidata.org/wiki/Wikidata:Main_Page (visited on 2023-03-31) (cit. on pp. 1, 25, 28).

[Web3] Wikipedia. *Nicolaus Copernicus*. URL: 10.1.1.142.4390 (visited on 2023-06-12) (cit. on p. 2).

[Web4] Research Library Gotha of the University of Erfurt. *Contact persons: Dr. Dietrich Hakelberg*. URL: https://www.uni-erfurt.de/en/gotha-research-library/library/contact/contact-persons/dr-dietrich-hakelberg (visited on 2023-05-18) (cit. on p. 2).

[Web5] Stiftung Preußischer Kulturbesitz: Staatsbibliothek zu Berlin. *Manuscripts and Early Printed Books: Contact*. URL: https://staatsbibliothek-berlin.de/en/about-the

-library/departments/manuscripts-and-early-printed-books/contact (visited on 2023-05-18) (cit. on p. 2).

[Web6] Kompetenzzentrum – Trier Center for Digital Humanities, University of Trier. *Dr Joëlle Weis: Head of Research Area*. URL: https://tcdh.uni-trier.de/en/person/dr-joelle-weis (visited on 2023-05-18) (cit. on p. 2).

[Web7] Wikipedia. *Soziale Netzwerkanalyse*. German. URL: https://de.wikipedia.org/wiki/Soziale_Netzwerkanalyse (visited on 2023-04-26) (cit. on p. 5).

[Web8] — *Network Science. Network Analysis*. URL: https://en.wikipedia.org/wiki/Network_science#Network_analysis (visited on 2023-04-26) (cit. on p. 5).

[Web9] University of Applied Sciences Potsdam. *SoNAR (IDH) – Interfaces to Data for Historical Social Network Analysis and Research*. URL: https://sonar.fh-potsdam.de/ (visited on 2023-04-20) (cit. on pp. 6, 64).

[Web10] SoNAR (IDH) research collective. *Reports*. URL: https://github.com/sonar-idh/reports (visited on 2023-04-20) (cit. on p. 6).

[Web11] German National Library. *The Integrated Authority File (GND)*. URL: https://www.dnb.de/EN/Professionell/Standardisierung/GND/gnd.html (visited on 2023-03-30) (cit. on pp. 6, 25, 27, 63).

[Web12] — *Zeitschriftendatenbank (German Union Catalogue of Serials)*. URL: https://zdb-katalog.de (visited on 2023-05-05) (cit. on pp. 7, 24, 64).

[Web13] — *DNB Catalogue*. URL: https://katalog.dnb.de/EN/home.html?v=plist (visited on 2023-03-31) (cit. on pp. 7, 24, 63).

[Web14] Prussian Cultural Heritage Foundation: Berlin State Library. *Kalliope Union Catalog*. URL: https://kalliope-verbund.info/ (visited on 2023-05-05) (cit. on pp. 7, 24, 63).

[Web15] Neo4j, Inc. *neo4j*. URL: https://neo4j.com/ (visited on 2023-05-05) (cit. on p. 8).

[Web16] Prussian Cultural Heritage Foundation: Berlin State Library. *ZEFYS – the newspaper information system of Berlin State Library*. URL: https://zefys.staatsbibliothek-berlin.de/ (visited on 2023-05-05) (cit. on pp. 8, 24, 64).

[Web17] German National Library. *Digital Exile Press*. URL: https://www.dnb.de/EN/Sammlungen/DEA/Exilpresse/exilpresse_node.html (visited on 2023-05-05) (cit. on pp. 8, 24).

[Web18] Ontotext. *Ontotext GraphDB*. URL: https://www.ontotext.com/products/graphdb/ (visited on 2023-06-11) (cit. on pp. 8, 46).

[Web19] Göttingen State and University Library. *DARIAH-DE*. URL: https://de.dariah.eu/ (visited on 2023-05-06) (cit. on p. 10).

[Web20] DARIAH ERIC. *DARIAH-EU*. URL: https://www.dariah.eu/ (visited on 2023-05-06) (cit. on p. 10).

[Web21] Wikipedia. *DARIAH-DE*. German. URL: https://de.wikipedia.org/wiki/DARIAH-DE (visited on 2023-05-06) (cit. on p. 10).

[Web22] German National Library. *Culturegraph*. German. URL: Culturegraph.org (visited on 2023-05-06) (cit. on p. 10).

[Web23] ORCID, Inc. *ORCID. Connecting research and researchers*. URL: https://orcid.org/ (visited on 2023-05-06) (cit. on p. 10).

[Web24] Carnegie Mellon University Libraries. *Six Degrees of Francis Bacon*. URL: http://www.sixdegreesoffrancisbacon.com/ (visited on 2023-05-06) (cit. on p. 10).

[Web25] CVCE Digital Humanities Lab. *histograph*. URL: http://histograph.eu/ (visited on 2023-05-06) (cit. on p. 10).

[Web26] University of Innsbruck. *"Issues with Europe"*. URL: https://www.uibk.ac.at/projects/issues-with-europe/index.html.en (visited on 2023-05-06) (cit. on p. 10).

[Web27] Austrian Centre for Digital Humanities and Cultural Heritage. *APIS*. URL: https://www.oeaw.ac.at/acdh/projects/completed-projects/apis (visited on 2023-05-06) (cit. on p. 10).

[Web28] W3C. *LinkedData*. URL: https://www.w3.org/wiki/LinkedData (visited on 2023-05-07) (cit. on p. 10).

[Web29] IETF HTTP Working Group. *HTTP Documentation*. URL: https://httpwg.org/specs/ (visited on 2023-05-12) (cit. on pp. 11, 63).

[Web30] Schreiber, Guus and Yves Raimond, eds. *RDF 1.1 Primer*. W3C Working Group Note, 24 June 2014. URL: https://www.w3.org/TR/rdf11-primer/ (visited on 2023-05-13) (cit. on pp. 11, 64).

[Web31] World Wide Web Consortium. *Resource Description Framework (RDF)*. URL: https://www.w3.org/RDF/ (visited on 2023-04-08) (cit. on p. 11).

[Web32] Brickley, Dan and R.V. Guha, eds. *RDF Schema 1.1*. W3C Recommendation, 25 February 2014. URL: https://www.w3.org/TR/rdf-schema/ (visited on 2023-05-13) (cit. on pp. 11, 64).

[Web33] Harris, Steve and Andy Seaborne, eds. *SPARQL 1.1 Query Language*. W3C Recommendation, 21 March 2013. URL: https://www.w3.org/TR/sparql11-query/ (visited on 2023-05-13) (cit. on pp. 12, 64).

[Web34] International Organization for Standardization (ISO). *ISO/IEC 9075-1:2016. Information technology — Database languages — SQL — Part 1: Framework (SQL/Framework)*. URL: https://www.iso.org/standard/63555.html (visited on 2023-05-13) (cit. on pp. 12, 64).

[Web35] Wikipedia. *Linked data*. URL: https://en.wikipedia.org/wiki/Linked_data (visited on 2023-05-09) (cit. on p. 12).

[Web36] Library of Congress. *Library of Congress*. URL: https://www.loc.gov/ (visited on 2023-05-20) (cit. on pp. 12, 64).

[Web37] Gonzalez, Gloria. *Linked Open Data: A Beckoning Paradise*. Guest post in the Library of Congress Blog "The Signal – Digital Happenings at the Library of Congress". URL: https://blogs.loc.gov/thesignal/2011/06/linked-open-data-a-beckoning-paradise/ (visited on 2023-05-14) (cit. on p. 12).

[Web38] Europeana Foundation. *Discover Europe's digital cultural heritage*. URL: https://www.europeana.eu (visited on 2023-05-14) (cit. on pp. 13, 25).

[Web39] German National Library. *GND Ontology*. URL: https://d-nb.info/standards/elementset/gnd# (visited on 2023-05-14) (cit. on p. 13).

[Web40] GBV Common Library Network. *T-PRO Thesaurus der Provenienzbegriffe*. German. URL: https://provenienz.gbv.de/T-PRO_Thesaurus_der_Provenienzbegriffe (visited on 2023-05-17) (cit. on pp. 14, 64).

[Web41] University of Erfurt. *OPAC search result for "De Revolutionibus"*. URL: https://opac.uni-erfurt.de/DB=1/CMD?ACT=SRCHA&IKT=1016&SRT=YOP&TRM=tit+de+revolutionibus+and+per+kopernikus+and+jah+15**+and+bbg+a* (visited on 2023-03-31) (cit. on p. 16).

[Web42] — *OPAC entry for one exemplar of "De Revolutionibus"*. URL: https://opac.uni-erfurt.de/LNG=EN/DB=1/XMLPRS=N/PPN?PPN=567506266 (visited on 2023-03-31) (cit. on p. 16).

[Web43] German National Library. *GND entry for Tilesius, Hieronymus*. German. URL: https://d-nb.info/gnd/10419975X (visited on 2023-06-05) (cit. on p. 17).

[Web44] — *GND entry for Hommel, Johann*. German. URL: https://d-nb.info/gnd/119713950 (visited on 2023-06-05) (cit. on p. 17).

[Web45] — *GND entry for Thau, Valentin*. German. URL: https://d-nb.info/gnd/119847469 (visited on 2023-06-05) (cit. on p. 17).

[Web46] — *GND entry for Ernest II, Duke of Saxe-Gotha-Altenburg*. German. URL: https://d-nb.info/gnd/102109109 (visited on 2023-06-05) (cit. on p. 17).

[Web47] Gerrnan National Library. *Dataset for the subject term "Wissenschaftler" (scientist) in the GND catalogue*. German. URL: https://portal.dnb.de/opac/opacPresentation?cqlMode=true&reset=true&referrerPosition=5&referrerResultId=wissenschaftler+and+mat=subjects&any&query=idn=040665674 (visited on 2023-05-18) (cit. on p. 18).

[Web48] — *List of all subordinate terms for the subject term "Wissenschaftler" (scientist) in the GND catalogue*. German. URL: https://portal.dnb.de/opac/simpleSearch?reset=true&cqlMode=true&query=ubRef=040665674&selectedCategory=any (visited on 2023-05-18) (cit. on p. 18).

[Web49] International Federation of Library Associations and Institutions. *IFLA*. URL: https://www.ifla.org/ (visited on 2023-05-23) (cit. on pp. 20, 63).

[Web50] Voss, Jakob. *File:FRBR-Group-1-entities-and-basic-relations.svg*. URL: https://en.wikipedia.org/wiki/File:FRBR-Group-1-entities-and-basic-relations.svg (visited on 2023-05-27) (cit. on p. 21).

[Web51] — *File:FRBR-Group-2-entities-and-relations.svg*. URL: https://en.wikipedia.org/wiki/File:FRBR-Group-2-entities-and-relations.svg (visited on 2023-05-27) (cit. on p. 21).

[Web52] German National Library. *RDA*. URL: https://www.dnb.de/EN/Professionell/Standardisierung/Standards/_content/rda.html?nn=147858 (visited on 2023-05-21) (cit. on pp. 21, 64).

[Web53] Wikipedia. *Resource Description and Access*. German. URL: https://de.wikipedia.org/wiki/Resource_Description_and_Access (visited on 2023-05-24) (cit. on p. 21).

[Web54] WHATWG community. *HTML*. URL: https://html.spec.whatwg.org/multipage/ (visited on 2023-05-27) (cit. on pp. 21, 63).

[Web55] World Wide Web Consortium. *Extensible Markup Language (XML)*. URL: https://www.w3.org/XML/ (visited on 2023-05-26) (cit. on pp. 22, 64).

[Web56] Wikipedia. *Extensible Markup Language*. URL: https://de.wikipedia.org/wiki/Extensible_Markup_Language (visited on 2023-05-27) (cit. on p. 22).

[Web57] Library of Congress. *SRU – Search/Retrieval via URL*. URL: https://www.loc.gov/standards/sru/ (visited on 2023-05-21) (cit. on pp. 22, 64).

[Web58] — *CQL: The Contextual Query Language*. URL: https://www.loc.gov/standards/sru/cql/index.html (visited on 2023-05-21) (cit. on pp. 22, 63).

[Web59] Open Archives Initiative. *Open Archives Initiative*. URL: https://www.openarchives.org (visited on 2023-05-21) (cit. on pp. 22, 64).

[Web60] — *The Open Archives Initiative Protocol for Metadata Harvesting*. URL: https://www.openarchives.org/OAI/openarchivesprotocol.html (visited on 2023-05-21) (cit. on pp. 22, 64).

[Web61] Library of Congress. *MARC Standards*. URL: https://www.loc.gov/marc/ (visited on 2023-05-27) (cit. on pp. 22, 64).

[Web62] — *MARC Proposal No. 2023-04*. URL: https://loc.gov/marc/mac/2023/2023-04.html (visited on 2023-06-04) (cit. on p. 23).

[Web63] German National Library. *MAB*. German. URL: https://web.archive.org/web/20181121033620/https://www.dnb.de/DE/Standardisierung/Formate/MAB/mab_node.html (visited on 2023-05-27) (cit. on pp. 23, 64).

[Web64] Wikipedia. *Maschinelles Austauschformat für Bibliotheken*. URL: https://en.wikipedia.org/wiki/Maschinelles_Austauschformat_f%C3%BCr_Bibliotheken (visited on 2023-05-27) (cit. on p. 23).

[Web65] Library of Congress. *Metadata Object Description Schema: MODS*. URL: http://www.loc.gov/standards/mods/ (visited on 2023-05-27) (cit. on pp. 23, 64).

[Web66] — *Metadata Authority Description Schema (MADS)*. URL: http://www.loc.gov/standards/mads/ (visited on 2023-05-27) (cit. on pp. 23, 64).

[Web67] Dublin Core™ Metadata Initiative (DCMI). *DCMI: Dublin Core™*. URL: https://www.dublincore.org/specifications/dublin-core/ (visited on 2023-05-25) (cit. on pp. 23, 63).

[Web68] — *DCMI: Home*. URL: https://www.dublincore.org (visited on 2023-05-25) (cit. on pp. 23, 63).

[Web69] Library of Congress. *Overview of the BIBFRAME 2.0 Model*. URL: https://www.loc.gov/bibframe/docs/bibframe2-model.html (visited on 2023-05-28) (cit. on pp. 23, 63).

[Web70] Wikipedia. *BIBFRAME*. URL: https://en.wikipedia.org/wiki/BIBFRAME (visited on 2023-05-28) (cit. on p. 23).

[Web71] Library of Congress. *LC Catalog*. URL: https://catalog.loc.gov/ (visited on 2023-05-28) (cit. on p. 24).

[Web72] British Library. *Explore the British Library*. URL: https://explore.bl.uk/primo_library/libweb/action/search.do (visited on 2023-05-28) (cit. on p. 24).

[Web73] OCLC, Inc. *WorldCat®*. URL: https://www.worldcat.org/ (visited on 2023-03-31) (cit. on p. 24).

[Web74] Karlsruhe Institute of Technology (KIT). *KVK – Karlsruhe Virtual Catalog*. URL: https://kvk.bibliothek.kit.edu/index.html?lang=en (visited on 2023-05-20) (cit. on pp. 24, 63).

[Web75] Library Service Center Baden-Württemberg and GBV Common Library Network. *K10plus – Kooperationsprojekt BSZ und GBV*. German. URL: https://www.bszgbv.de/services/k10plus/ (visited on 2023-03-30) (cit. on pp. 24, 63).

[Web76] Bibliotheksverbund Bayern. *B3Kat – kurz und bündig*. German. URL: https://www.bib-bvb.de/web/b3kat (visited on 2023-05-20) (cit. on pp. 24, 63).

[Web77] Hochschulbibliothekszentrum des Landes Nordrhein-Westfalen. *Hochschulbibliothekszentrum NRW*. German. URL: https://www.hbz-nrw.de/ (visited on 2023-05-20) (cit. on pp. 24, 63).

[Web78] hebis-Verbundzentrale. *hebis*. German. URL: https://www.hebis.de/ (visited on 2023-05-20) (cit. on pp. 24, 63).

[Web79] Library of Congress. *Library of Congress Name Authority File (LCNAF)*. URL: https://id.loc.gov/authorities/names.html (visited on 2023-05-19) (cit. on pp. 25, 64).

[Web80] ISNI International Agency. *ISNI*. URL: https://isni.org/ (visited on 2023-03-31) (cit. on pp. 25, 63).

[Web81] OCLC, Inc. *VIAF*. URL: https://viaf.org/ (visited on 2023-03-31) (cit. on pp. 25, 64).

[Web82] Prussian Cultural Heritage Foundation, FIZ Karlsruhe – Leibniz Institute for Information Infrastructure, and German National Library. *Deutsche Digitale Bibliothek – Kultur und Wissen online*. URL: https://www.deutsche-digitale-bibliothek.de/?lang=en (visited on 2023-06-13) (cit. on pp. 25, 63).

[Web83] German Lost Art Foundation. *Proveana – Provenance Research Database*. URL: https://www.proveana.de/en/start (visited on 2023-05-20) (cit. on p. 25).

[Web84] German National Library. *Our collection mandate*. URL: https://www.dnb.de/EN/Professionell/Sammeln/sammeln_node.html (visited on 2023-05-21) (cit. on pp. 25, 26).

[Web85] — *Cataloguing media works*. URL: https://www.dnb.de/EN/Professionell/Erschliessen/erschliessen_node.html (visited on 2023-05-21) (cit. on pp. 25, 27).

[Web86] — *Metadata services*. URL: https://www.dnb.de/EN/Professionell/Metadatendienste/metadatendienste_node.html (visited on 2023-05-21) (cit. on pp. 25, 26).

[Web87] — *BIBFRAME – Bibliographic Framework Initiative*. URL: https://www.dnb.de/EN/Professionell/ProjekteKooperationen/Projekte/BIBFRAME/bibframe_node.html (visited on 2023-05-28) (cit. on p. 26).

[Web88] — *MARC 21*. URL: https://www.dnb.de/EN/Professionell/Metadatendienste/Exportformate/MARC21/marc21_node.html (visited on 2023-05-22) (cit. on pp. 26, 27).

[Web89] — *MARCXML*. URL: https://www.dnb.de/EN/Professionell/Standardisierung/Standards/_content/marcxml.html (visited on 2023-05-22) (cit. on p. 26).

[Web90] — *Metadatenherkunft in der DNB und in MARC 883*. German. URL: https://wiki.dnb.de/display/ILTIS/Metadatenherkunft+in+der+DNB+und+in+MARC+883 (visited on 2023-05-22) (cit. on p. 26).

[Web91] Library Service Center Baden-Württemberg and GBV Common Library Network. *About us*. German. URL: https://www.bszgbv.de/organisation/projekt/ (visited on 2023-05-23) (cit. on p. 26).

[Web92] GBV Common Library Network. *About the Head Office*. German. URL: https://www.gbv.de/informationen/Verbundzentrale/ueber_die_VZG (visited on 2023-05-23) (cit. on p. 26).

[Web93] Library Service Center Baden-Württemberg and GBV Common Library Network. *Homepage of the K10plus Wiki*. German. URL: https://wiki.k10plus.de/ (visited on 2023-05-23) (cit. on p. 26).

[Web94] Library Service Centre Baden-Württemberg. *K10plus-Recherche*. German. URL: https://www.bsz-bw.de/K10plus.html (visited on 2023-05-23) (cit. on p. 26).

[Web95] GBV Common Library Network. *Holdings statistics of the union catalogue*. German. URL: https://www.gbv.de/informationen/Verbundzentrale/Datenbankstatistik/Datenbankstatistik_1540 (visited on 2023-05-23) (cit. on p. 26).

[Web96] Library Service Center Baden-Württemberg and GBV Common Library Network. *K10plus Wiki: Open Data*. German. URL: https://wiki.k10plus.de/display/K10PLUS/Open+Data (visited on 2023-05-23) (cit. on p. 26).

[Web97] Universitätsbibliothek Regensburg. *RVK Online*. German. URL: https://rvk.uni-regensburg.de/regensburger-verbundklassifikation-online (visited on 2023-05-29) (cit. on pp. 27, 64).

[Web98] Library Service Center Baden-Württemberg and GBV Common Library Network. *K10plus Wiki: Authority Data*. German. URL: https://wiki.k10plus.de/display/K10PLUS/Normdaten (visited on 2023-05-29) (cit. on p. 27).

[Web99] German National Library. *Erfassungshilfen für Personen und Familien*. German. URL: https://wiki.dnb.de/pages/viewpage.action?pageId=90411361 (visited on 2023-05-29) (cit. on p. 27).

[Web100] Wikimedia Foundation. *Wikidata:Introduction*. URL: https://www.wikidata.org/wiki/Wikidata:Main_Page (visited on 2023-05-30) (cit. on p. 28).

[Web101] — *Help:Items*. URL: https://www.wikidata.org/wiki/Help:Items (visited on 2023-05-30) (cit. on p. 28).

[Web102] — *Wikidata:Data access*. URL: https://www.wikidata.org/wiki/Wikidata:Data_access (visited on 2023-05-30) (cit. on p. 28).

[Web103] — *Help:Statements*. URL: https://www.wikidata.org/wiki/Help:Statements (visited on 2023-05-30) (cit. on p. 28).

[Web104] — *Nicolaus Copernicus (Q619)*. URL: https://www.wikidata.org/wiki/Q619 (visited on 2023-05-30) (cit. on p. 28).

[Web105] Wikipedia. *Graph Theory: Applications*. URL: https://en.wikipedia.org/wiki/Graph_theory#Applications (visited on 2023-04-06) (cit. on p. 30).

[Web106] Python Software Foundation. *Python Graph Libraries*. URL: https://wiki.python.org/moin/PythonGraphLibraries (visited on 2023-04-15) (cit. on p. 30).

[Web107] Naveh, Barak and Contributors. *JGraphT: a Java library of graph theory data structures and algorithms*. URL: https://jgrapht.org/ (visited on 2023-04-08) (cit. on p. 30).

[Web108] Toyota, Gustavo. *VisualSQL*. URL: https://visualsql.net/ (visited on 2023-06-10) (cit. on p. 45).

[Web109] SQLeo. *SQLeo*. URL: https://sqleo.sourceforge.io/ (visited on 2023-06-10) (cit. on p. 45).

[Web110] Chart.io, Inc. *Choosing a Visual SQL Query Builder? Here's What You Should Know*. URL: https://chartio.com/learn/visual-sql/choosing-a-visual-sql-query-builder/ (visited on 2023-06-10) (cit. on p. 45).

[Web111] FinancesOnline. *20 Best Visual SQL Query Builders of 2023*. URL: https://financesonline.com/20-best-visual-sql-query-builders/ (visited on 2023-06-10) (cit. on p. 45).

[Web112] Wikipedia. *List of relational database management systems*. URL: https://en.wikipedia.org/wiki/List_of_relational_database_management_systems (visited on 2023-06-10) (cit. on p. 46).

[Web113] American National Standards Institute, Inc. (ANSI). *American National Standards Institute – ANSI Home*. URL: https://www.ansi.org/ (visited on 2023-05-26) (cit. on p. 63).

[Web114] GBV Common Library Network. *Welcome*. German. URL: https://en.gbv.de/ (visited on 2023-05-28) (cit. on p. 63).

[Web115] International Organization for Standardization (ISO). *ISO – International Organization for Standardization*. URL: https://www.iso.org/ (visited on 2023-05-26) (cit. on p. 63).

[Web116] National Information Standards Organization. *National Information Standards Organization | NISO Website*. URL: http://www.niso.org/ (visited on 2023-05-26) (cit. on p. 64).

[Web117] Hitzler, Pascal et al., eds. *OWL 2 Web Ontology Language Primer (Second Edition)*. W3C Recommendation, 11 December 2012. URL: https://www.w3.org/TR/owl2-primer/ (visited on 2023-05-13) (cit. on p. 64).

[Web118] Library Service Centre Baden-Württemberg. *Home – BSZ*. German. URL: https://www.bsz-bw.de/ (visited on 2023-05-28) (cit. on p. 64).

[Web119] Library of Congress. *Z39.50 Maintenance Agency Page*. URL: https://www.loc.gov/z3950/agency/ (visited on 2023-05-26) (cit. on p. 64).

Abbreviations

ANSI	American National Standards Institute [Web113]
B3KAT	the union catalogue of the German library networks BVB and KOBV [Web76]
BIBFRAME	Bibliographic Framework [Web69]
BK	Basisklassifikation [Facharbeitsgruppe Sacherschließung 1995]
CQL	Contextual Query Language [Web58]
DC	The Dublin Core Metadata Element Set, in short: Dublin Core [Web67]
DCMI	Dublin Core Metadata Initiative [Web68]
DDB	German Digital Library [Web82]
DNB	German National Library [Web13]
FRAD	Functional Requirements for Authority Data [IFLA WG FRANAR 2008]
FRBR	Functional Requirements for Bibliographic Records [IFLA SG FRBR 2009]
FRSAD	Functional Requirements for Subject Authority Data [IFLA WG FRSAR 2010]
GBV	Gemeinsamer Bibliotheksverbund, the joint library network of the German states Bremen, Hamburg, Mecklenburg-Vorpommern, Niedersachsen, Sachsen-Anhalt, Schleswig-Holstein, Thüringen, and the Foundation of Prussian Cultural Heritage [Web114]
GND	“Gemeinsame Normdatei”, the Integrated Authority File of the German National Library [Web11]
hbz	library network of the German states North Rhine-Westphalia and Rhineland-Palatinate [Web77]
hebis	library network of the German state of Hesse and the region of Rhine Hesse [Web78]
HNA	historical network analysis
HTML	Hypertext Markup Language [Web54]
HTTP	Hypertext Transfer Protocol [Web29]
IFLA	International Federation of Library Associations and Institutions [Web49]
ISNI	International Standard Name Identifier [Web80]
ISO	International Organization for Standardization [Web115]
KB	knowledge base
K10plus	the union catalogue of the German library networks GBV and SWB [Web75]
KPE	Kalliope Union Catalogue [Web14]
KVK	Karlsruhe Virtual Catalogue [Web74]

LoC Library of Congress [Web36]

LCNAF Library of Congress Name Authority File [Web79]

MAB “Maschinelles Austauschformat für Bibliotheken” [Web63]

MADS Metadata Authority Description Schema [Web66]

MARC Machine-Readable Cataloguing [Web61]

MODS Metadata Object Description Schema [Web65]

NISO National Information Standards Organization of the USA [Web116]

OAI Open Archives Initiative [Web59]

OAI-PMH OAI Protocol for Metadata Harvesting [Web60]

OWL the Web Ontology Language recommended by the W3C [Web117]

RDA Resource Description and Access [Web52]

RDBMS relational database management system

RDF Resource Description Framework [Web30]

RDFS RDF Schema [Web32]

RVK Regensburger Verbundklassifikation [Web97]

SNA social network analysis

SoNAR research project “SoNAR (IDH), Interfaces to Data for Historical Social Network Analysis and Research” [M. Bludau et al. 2020; Menzel, M.-J. Bludau, et al. 2020; Web9]

SPARQL Simple Protocol And RDF Query Language [Web33]

S–P–O subject–predicate–object

SQL Structured Query Language [Web34]

SRU Search/Retrieval via URL [Web57]

SWB Südwestdeutscher Bibliotheksverbund, the joint library network of the German states Baden-Württemberg, Sachsen, Saarland, and further institutions [Web118]

T-PRO Thesaurus of Provenance Terms [Web40]

URI Uniform Resource Identifier [Berners-Lee, Fielding, and Masinter 2005]

VIAF Virtual International Authority File [Web81]

WWW World Wide Web

XML Extensible Markup Language [Web55]

Z39.50 ANSI/NISO Standard “Information Retrieval: Application Service Definition and Protocol Specification” [Web119]

ZDB German Union Catalogue of Serials [Web12]

ZEFYS the newspaper information system of Berlin State Library [Web16]

Appendix A

Research Data

The research data created during the work on this thesis were generated, documented, and made accessible for at least 18 months according to the "Leitlinie zum Umgang mit Forschungsdaten in Abschlussarbeiten" https://www.ibi.hu-berlin.de/de/studium/rundumdasstudium/fdm-fuer-studierende/leitlinie_forschungsdaten_finale_version_dez_21-1.pdf .

The research data is stored in two datasets in the open repository Zenodo:

- https://doi.org/10.5281/zenodo.8036824 (open access) with the following contents:
  - a BibTeX file `masters_thesis.bib` with the bibliographic metadata of all publications and websites cited throughout the thesis
  - a markdown file `statistics_of_data_sources.md` containing statistics of the data sources reviewed in Chapter 4
- https://doi.org/10.5281/zenodo.8036903 (closed access for licensing reasons) containing:
  - a zip file `Lit.zip` with full texts of a part of the cited publications

The research data is documented via a data management plan, which is stored in the cloud storage service *HU-Box* of the Humboldt-Universität zu Berlin and can be viewed and downloaded via the following URL: https://box.hu-berlin.de/d/7aabcdcda32a42bda4dd/

The PDF file of this thesis is available under the same URL.

Humboldt-Universität zu Berlin
Philosophische Fakultät
Institut für Bibliotheks- und Informationswissenschaft

Name: Schneider Vorname: Thomas

Matr.Nr.: 624025

Eidesstattliche Erklärung zur

- ○ **Hausarbeit ***
- ○ **Bachelorarbeit ***
- ◉ **Masterarbeit ***
- ○ **Abschlussarbeit im Bibliotheksreferendariat ***

***** Die eingereichte PDF-Datei ist mit den Printexemplaren identisch.

Ich erkläre ausdrücklich, dass es sich bei der von mir eingereichten schriftlichen Arbeit mit dem Titel

Modelling and Automated Retrieval of Provenance Relationships

um eine von mir erstmalig, selbstständig und ohne fremde Hilfe verfasste Arbeit handelt.

Ich erkläre ausdrücklich, dass ich *sämtliche* in der oben genannten Arbeit verwendeten fremden Quellen, auch aus dem Internet (einschließlich Tabellen, Grafiken u. Ä.) als solche kenntlich gemacht habe. Insbesondere bestätige ich, dass ich ausnahmslos sowohl bei wörtlich übernommenen Aussagen bzw. unverändert übernommenen Tabellen, Grafiken u. Ä. (Zitaten) als auch bei in eigenen Worten wiedergegebenen Aussagen bzw. von mir abgewandelten Tabellen, Grafiken u. Ä. anderer Autorinnen und Autoren (Paraphrasen) die Quelle angegeben habe.

Mir ist bewusst, dass Verstöße gegen die Grundsätze der Selbstständigkeit als Täuschung betrachtet und entsprechend der Prüfungsordnung und/oder der Fächerübergreifenden Satzung zur Regelung von Zulassung, Studium und Prüfung (ZSP-HU) geahndet werden.

Datum 14.06.2023 Unterschrift