

The wind in our sails: Developing an accessible, transparent, reusable, and maintainable knowledge graph in the field of Dutch maritime data

MSc Thesis by Stijn Schouten

2677642, IS, s5.schouten@student.vu.nl

Abstract. Digital sources are more prevalent than ever. Effectively using these sources can be challenging. Researchers are spending time collecting, interpreting, and aligning different sources in their research. A knowledge graph can accelerate research by providing a single connected source of truth that humans and machines can query. During two design-test cycles, four datasets are converted into a knowledge graph. The focus during these cycles is on creating a sustainable and usable approach and artifact for maritime researchers and other enthusiasts.

The development of a sustainable and usable knowledge graph is a dynamic process. The two design iterations provided well-needed feedback to improve the artifact on these aspects. It could be concluded, at the end of the second cycle, that the research goals were met.

Research results of this project are openly published and transparent, such that other researchers can easily continue the development of the knowledge graph.

1 Introduction

Digital humanities research is a multidisciplinary field in which computer science and complex humanities related questions are combined. Digital humanities researchers utilize computational processing power to create and present insights that were not possible before in traditional humanities research[1]. Due to large-scale digitalization efforts in recent decades, the volume of digitally available records is enormous[2]. For instance, the Amsterdam Archives has scanned over 8 million pages in its archive at the moment of writing.¹ Another example, one more relevant to this project, is the VOC databases and their digital records. The Vereenigde Oostindische Compagnie (VOC) was a large trading company founded in 1602 in the Netherlands[3]. The VOC created and curated several written logbooks. These logbooks covered, for instance, the sailors who ventured on the journey to the East Indies and who might never return.² The logbooks

¹ <https://alleamsterdamseakten.nl/overhetproject/> retrieved at: 17-04-2021

² <https://www.nationaalarchief.nl/onderzoeken/zoekhulpen/voc-opvarenden>
retrieved at: 17-04-2021

also cover the cargo shipped by these journeys. Some relatively innocent, such as spices, while others, less so. For instance, the notorious trans-Atlantic slave trade rampant in the 17th century.³ These logbooks were eventually converted to a digital format by efforts of archives and research institutes.

However, these digitalization projects come at a cost. Lora M. Hughes describes that digitizing, storing, and publishing data is a costly endeavor. The usefulness of such projects is not apparent immediately[4]. It takes time for researchers to compose stories about the people who lived then or to incorporate the digitized data in some greater coherent theory. Therefore, the focus in digitizing projects are becoming more result-driven[2].

Research questions

In this research project, a knowledge graph is iteratively designed and developed. The design and development are some of the more practical goals of this research. Moreover, the owners/curators of the data would be interested in the designed artifact, a useful knowledge graph. This artifact, which will be discussed in the next section, can be used by digital humanities researchers and other enthusiasts to ‘accelerate’ their research projects. Meaning, the artifact can be seen as a tool that can drive result-driven research and help answer the research questions of these digital humanities researchers.

The wind in our sails project is a research project. Therefore, the process of creating the artifact is heavily monitored and curated. The decisions made during the process are described and weighed against the alternatives. The process of creating a sustainable knowledge graph could be helpful in other conversion and extension projects. These theoretical and practical implications are boiled down to the following research question:

RQ: How can a sustainable knowledge graph be developed which is useful for researchers in the field of Dutch maritime data?

The main research question aims to create an artifact, the knowledge graph, that fulfills the set requirements of sustainability and usefulness. The chosen approach is an iterative design science method that involves two design and test cycles. This approach allows for multiple rounds of feedback during the research project.

There are two characteristics found in the main research question. First, the result and the approach should be *sustainable*. Second, the designed artifact should be *useful* for researchers. These characteristics are captured in the following sub-research questions:

SRQ1: What is a sustainable method of developing a knowledge graph related to Dutch maritime data?

³ <http://resources.huygens.knaw.nl/boekhoudergeneraalbatavia> retrieved at:
17-04-2021

SRQ2: How can a useful knowledge graph be developed for Dutch maritime history and how can that be measured?

Measuring and determining the most sustainable approach can be subjective. Therefore, four quality attributes are defined that can be used to measure, steer, and lead the design to a sustainable solution. These design requirements are set in cooperation with the project stakeholders and are as follows:

- *Transparency:* Clear and well-described design decisions and rules.
- *Accessibility:* Results and intermediate results should be accessible.
- *Re-usability:* Re-usable methods and semantic models.
- *Maintainability:* Future-proof knowledge graph and infrastructure.

The usefulness variable of the second sub-research question is also challenging to describe and measure. In this research project, competency questions are used as a derivative of the usefulness variable. ‘Scientific competency questions’ can be used to measure and steer the development of IT artifacts[5]. The gathered competency questions are explored in the methodology section of the thesis.

2 Background & Related work

To complete research questions and projects, digital humanities researchers need access to rich and available knowledge sources. Publishing and managing these sources in an open, available, and structured manner boasts the feasibility of such research projects[4]. Data can be shared in several procedures, such as an email containing CSVs or a central SQL database that hosts numerous tables. The user is responsible for collecting the different datasets, interpreting, refining, and finally creating the queries and filtering operations needed to answer their research question. This complex process is not always straightforward; some researchers can interpret variables differently from others, and combining different datasets can take up much time[6]. Another concern with traditionally sharing data is the problem of machine interoperability. A computer can not access the data directly in most cases. The user first has to download and load it into the application of choice. Moreover, a computer can not easily construct and define relations between two data entries. Thus, the human researcher is responsible for much of the intelligence behind the data-engineering.

2.1 Knowledge Graph

A knowledge graph can alleviate these problems. In the semantic journal [7] a knowledge graph is defined as follows:

”Knowledge graphs are large networks of entities, their semantic types, properties, and relationships between entities.”

The entities in this definition are the things in the world around us—for instance, the city of Amsterdam. One semantic classification of Amsterdam is that of a city. This classification means that Amsterdam falls in the same class as Berlin, for instance. Amsterdam has a certain number of residents; that number is a property of the instance of Amsterdam. The Netherlands, as a country type, is also an entity. Two entities can be related to each other via a relationship. In this case, we can define the relationship ‘hasCapital’ and link The Netherlands to Amsterdam.

Instances are connected via a property or a relationship to some object. This serialization of three distinct data points is called a Resource Distribution Framework (RDF) triple and is the backbone of most knowledge graphs. The reasoning engine within a knowledge graph also plays a vital role. The reasoning engine takes RDF triples and an ontology and creates implicit statements. An ontology is considered a formal and explicit definition of a particular domain’s shared conceptualization[8]. It describes the relation between semantic types, properties, and relationships. For instance, an ontology could describe that the property of ‘hasCapital’ is the inverse of ‘isCapitalOf’. After all, this holds as Amsterdam is the capital of the Netherlands. It further implies Amsterdam as a Capital City due to the domain (starts at) and range (ends at) functionality of reasoning languages. The reasoning engine creates these implicit RDF triples in the graph without user interference. This notion of explicit and implicit is illustrated in figure 1.

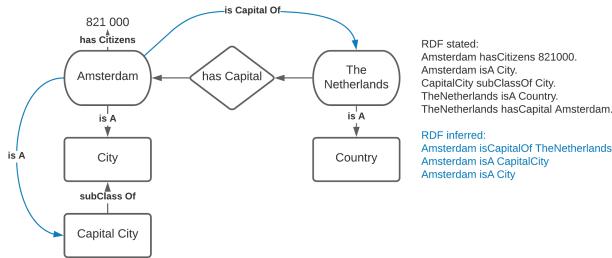


Fig. 1. Visualization of a knowledge graph and its reasoning engine

2.2 Semantic Web

A knowledge graph can be of assistance in structuring data in a uniform way while also making it computer interpretable. However, the shareability problem would not be inherently solved as only the format changed. This shareability issue is solved by web-transforming the knowledge graph. A web-enabled knowledge graph is also known as a semantic web database. In the semantic web database, instances are expressed in Uniform Resource Identifiers. For instance,

Amsterdam is identified as <https://dbpedia.org/resource/Amsterdam> in the DBpedia knowledge graph.⁴ Anyone in the world can now re-use and connect to the instance of Amsterdam by simply absorbing the URI in their semantic web database.

To instantly browse and connect to semantic web databases on the web, SPARQL endpoints can also be traversed. SPARQL endpoints function as gateway ports to and from the semantic web database. With the use of SPARQL endpoints, multiple semantic web databases can be queried simultaneously[9].

In short, knowledge graphs and semantic web technology drastically reduce the time a researcher has to spend browsing and preparing different datasets. Queries can be answered quickly, even with the most complex questions. The semantic web enables a more open infrastructure in which knowledge graphs can point to other knowledge graphs and query results due to standard SPARQL endpoints. Creating a knowledge graph and making it available on the web can solve many of the previously stated challenges of digital humanities research. The advantages of following this approach are illustrated in figure 2.

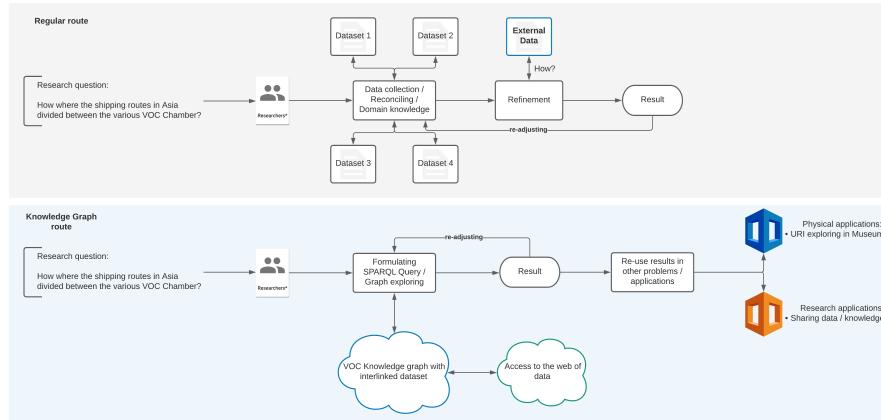


Fig. 2. Visualization of utilizing knowledge graphs in research: The top part describes the regular route of answering a research question. The bottom part describes the knowledge graph route, an overall improved method of answering research questions.

2.3 Literature review

Developing an accessible, transparent, reusable, and maintainable knowledge graph in the field of Dutch maritime data can be challenging. Therefore, existing work and literature are used to understand and build knowledge of the development of such artifacts. First, the four requirements are examined. The accessible,

⁴ <http://wikidata.dbpedia.org/about>

transparent, reusable, and maintainable design guidelines appear closely related to the FAIR principles. The FAIR principles describe four principles that serve as guidelines for data and research publishers. The guidelines describe that publishers should provide Findable, Accessible, Interoperable, and Reusable (FAIR) data and research results. Adhering to this principle entails that the added value of research projects is significantly increased after publishing[10].

The research project "Wikidata as a FAIR knowledge graph for the life sciences" of Waagmeester et al.[11] describes the development of knowledge graph on the Wikidata platform. Wikidata provides many interactivity methods for researchers, both graphically and computationally, thus making the solution *accessible* and *interoperable*. The solution is released under a creative commons zero agreement, making the graph freely available for any use. However, that does not automatically entail that the solution is *reusable*. Poor design decisions could be made during the process, rendering the solution poorly *reusable*. Finally, an observation is made regarding the sustainable growth of the artifact after project completion. The authors presume that further growth of biomedical graphs within Wikidata is reliant on community efforts.

The work 'The creation of knowledge graphs' provides multiple technical approaches in the development of knowledge graphs[12]. In short, dedicated mapping languages and restructured mapping languages for converting datasets to knowledge graphs were compared. These mapping languages provide a general, albeit challenging, approach to converting multiple datasets into knowledge graphs. The languages and applications discussed are not extensive. Therefore it can be fruitful re-do some of the analysis of the discussed tools and other tools alike for The wind in our sails research project.

In an attempt to encompass both the higher-level properties of knowledge graph engineering and the detailed, technical side of these projects, Milos Jovanovik proposes a new Linked Data application development methodology.[13] This methodology is, in the study, compared to other existing Linked Data methodologies. The methodology focuses on allowing data- and domain engineers to produce high-quality linked data aligned with the semantic web. Furthermore, the methodology is centered around the re-usability of methods and conversion methods utilizing a modular approach.

In conclusion, the design guidelines of developing an accessible, transparent, reusable, and maintainable knowledge graph are closely related to the FAIR[10] principle. The work of Waagmeester et al.[11] provides us with tips and pointers on how to develop a knowledge graph according to the FAIR principle. Dimou[12] dives more into the technical side of knowledge graph engineering and inspires further research and comparison of these approaches. Finally, Jovanovik[13] brings it together with his linked data application development methodology, which focuses both on FAIR principles and the technical implementation.

2.4 Related projects

In the literature review, general projects are discussed [11, 12]. However, these projects are of a different domain than maritime history. In this subsection, two relevant projects are examined in more detail.

De Boer et al.[14] started in 2014 with Dutch Ships and Sailors (DSS). In this project, the researchers harmonized four different datasets to unify the scattered Dutch maritime data[14]. Additional work in this domain has been completed by Entjes[15] in 2015. Although Entjes did not explicitly mention the use of competency questions within his research, the research is driven by a similar methodology.

The work of Entjes[15] and De Boer et al.[14] is reflected upon by utilizing the defined design guidelines. The projects are both designed with a high level of *transparency* and *accessibility*. All the design decisions made during the Dutch Ships and Sailors (DSS) project are stored on the project website⁵ and include information such as datasets conversions, methodologies, and much more. By including the project team's thought process, the decisions can be followed and understood, even as an outsider. Furthermore, the project website also hosts a semantic server that can be used to access the knowledge graph directly. All graphs in the database can even be downloaded separately.

However, the other guidelines should be considered as well. In DSS, multiple general-purpose ontologies and models are used, and a specific model is created for the maritime activities. This design decision has definite upsides as it is generally recommended to re-use as many models and ontologies to enable interoperability with other graphs on the web.⁶ Still, these general-purpose ontologies are, as the name describes it, general-purpose. Thus, concepts as naming or things as actors can be captured well. But, domain-specific concepts are not captured in these models. This design decision makes the re-usability of DSS regarding the semantic web high (due to the utilization of general-purpose ontologies) but regarding domain-specific projects low as it is unlikely that the concept of Ship in DSS is used between projects outside of the DSS sphere.

The third implementation of this domain has been completed during the 2019 Time Machine conference.⁷ During this conference, Beretta et al.[16] presented the OntoMe application.⁸ As part of the presentation, the Dutch maritime datasets were selected and converted to a knowledge graph. The project is designed with a high level of maintainability as the poster proposed a method of maintaining and distributing ontologies for digital heritage projects. Furthermore, the selected ontology, an adapted version of the CIDOC CRM, is a widely accepted and highly detailed model for digital heritage. Concepts in this model, such as voyages, have a better chance of being re-used in other semantic web projects, thus offering great alignment possibilities. However, it has to be noted

⁵ <https://dutchshipsandsailors.nl>

⁶ <https://www.w3.org/standards/semanticweb/ontology>

⁷ <https://digital-strategy.ec.europa.eu/en/events/time-machine-conference-2019>

⁸ <http://ontome.net>

that unlike DSS, which used widely accepted conventions for naming things (SKOS, DC Terms, etc.), the OntoMe implementation sticks to CIDOC CRM conventions. This design decision can lead to lower connectedness with the semantic web cloud.

Nevertheless, the transparency and accessibility design principles can be improved. Design decisions are not openly published, making it difficult for outsiders to understand and follow the thought process during the design. However, the adjusted semantic model is posted and accessible on the OntoMe platform.⁹ During the interview with the domain expert from the Huygens, it was apparent that the knowledge graph was challenging to access. No direct download links are provided while the SPARQL endpoint seems to be in beta as of writing.

In conclusion, the discussed projects have their strength and weaknesses. Some design decisions can be re-used for this research project, such as the transparent methodology of DSS or the model maintainability of the OntoME project. The lessons learnt are summarized in the table below.

	DSS and its extension OntoME implementation	<i>Wind in our sails</i>
Transparency	Good, clear process.	<i>Improve on previous iterations.</i>
Accessibility	Good, easy access.	<i>Improve on previous iterations.</i>
Re-usability	To be improved.	<i>Improve on previous iterations.</i>
Maintainability	To be improved.	<i>Improve on previous iterations.</i>

Table 1. Overview of previous implementations

2.5 Related tools

The work done by de Boer et al.[14] and Entjes[15] provides an insight into the linked data development of maritime history. Furthermore, the work of Dimou[12] and Jovanovik[13] discusses several approaches briefly. However, it would be beneficial to reflect and explore the approaches discussed and used as a lot can change in software[17, 18].

The first approach consists of tools based around the RDF Mapping Language (RML). This mapping language is discussed in the work of Dimou[12] as well. Applications built on top of this technology include, but are not limited to, CARML,¹⁰ RMLEDitor,¹¹ and Yarrrml (matey).¹² The RML based editors have their function when a researcher has to combine multiple, differently structured datasets and map entities between the sets[19]. However, all RML-based applications lack a workable visual or GUI representation. This downside makes the recommendation to use such an approach difficult for users outside computer

⁹ <http://ontome.net/project/28>

¹⁰ <https://github.com/carm1/carm1>

¹¹ <https://rml.io/tools/rmleditor/>

¹² <https://rml.io/yarrrml/matey/>

and information sciences. The keyword ‘workable’ is of importance here, the RMLEDitor¹³ is a graphical editor, but it currently limits the user to twenty connections and a file size limit of 2 MB. An email requesting a more extensive version of the application is so far not met with a reply.

In the next category, the tools of the DataLegend ecosphere¹⁴ are explored. These tools allows domain experts to (semi)-automatically convert tabular data into RDF. [20] The tooling is well capable of quickly producing high volumes of RDF data with minimal user input. Modeling the data according to a strict ontology and even linking entities within a dataset is more difficult due to the mapping workflow.

GraphDB is developed by the commercial company Ontotext.¹⁵ This approach is mentioned by Jovanovik as well[13]. GraphDB is, like the name suggests, a database system for knowledge graphs. Later installments of GraphDB are pre-installed with OntoRefine,¹⁶ a tool for cleaning, reconciling, and converting tabular data into RDF. The user can do the complete data-processing work within the GraphDB application. These relatively easy-to-use data cleaning and management tools provide a solid workflow in preparing data for RDF conversion.

In ‘The wind in our sails’ project, the GraphDB application is used for the conversion. The tool is transparent due to its interface. Users can directly intuitively influence the mapping. The tool is reusable due to the ability to import and export mapping tables between projects. Maintainability also scores high due to the active development of the application.¹⁷ The only criticism it faces is the lower accessibility. A commercial company develops the application. Thus, it makes sense that large-scale implementations are paid. It is not necessarily a bad thing. In return, support and active development are provided. Something that is not certain with the use of free open-source solutions.

2.6 Related ontologies and models

The Dutch Ships and Sailors (DSS) project and the ‘Archangel & Elbing’ extension share the same ontology. Meaning, an instance of the class ship has the same meaning across the different research projects. In previous sections, the up-and downsides of using the ontology of DSS are discussed. In the third related project, the OntoMe implementation, a different ontology is used. In this project, an extended version of the CIDOC Conceptual Reference Model (CRM) is used. The CIDOC CRM is a high-level formal ontology for integrating cultural heritage datasets around the world.¹⁸ A high-level ontology is an ontology that can capture a wide range of phenomena by avoiding detailed definitions.

¹³ <https://rml.io/tools/rmleditor/>

¹⁴ <https://iisg.amsterdam/en/clariah>

¹⁵ <https://www.ontotext.com/products/graphdb/>

¹⁶ <https://openrefine.org>

¹⁷ <https://graphdb.ontotext.com/documentation/standard/release-notes.html>

¹⁸ <http://www.cidoc-crm.org/version/version-7.1.1>

In The wind in our sails project, the CRM is selected as the ontology of choice. The CRM is a safe choice; it has been proven to work for the OntoME implementation. There exist other high-level ontologies as well, such as DOLCE¹⁹ or COSMO.²⁰ It could be that these ontologies are also suitable for maritime datasets. However, comparing and developing these different conversions would take considerable time and does not fit the research question. Finally, some of the concepts of the OntoMe implementation can be re-used in The wind in our sails project.

3 Methodology

Design science is selected as the central methodology, as the primary goal of the thesis is the development of an artifact: the knowledge graph. In Design Science research, several research methods and frameworks exist[21, 22]. The main takeaway of these theories is the design-test cycle. In this iterative cycle, the researcher uses the designed artifact to solve the given problem while grounding the artifact in a social-technical foundation. The artifact is evaluated and the evaluation results are used in the second iteration of the design-test cycle. The cycle continues until a certain predefined threshold is met.

In The wind in our sails project, two cycles are completed. Two cycles allow for a review and evaluation period while still being time-efficient. Before initiating the first design cycle, it is essential to build the foundation. The foundation, which is explored in the previous section, consists out of three parts. The first layer is the literature review. Here, we have a birds-eye view of the problem- and solution space. Then, on top of that, we have the related projects: a more zoomed-in vision and review of previous iterations. Finally, the practical foundation is built by reviewing the related tools and ontologies.

In the design-test cycle, the approach of Jovanovik[13] is implemented. With the goal to re-use some of the steps during the second design-test cycle. Furthermore, an open and unstructured interview is conducted with a domain expert from the Huygens during the first design-test cycle. The domain expert is responsible for some and involved in most of the curations of datasets used during the research project. An interview does not only help with understanding the datasets but also creates an understanding of the use case. How are domain experts planning on using the artifact and what are the expectations. Not all questions can be asked and answered during a single interview; some questions may arise during the cycle. Therefore, close contact was kept via email. Working together at one of the offices was not possible due to the ongoing pandemic. Finally, the documentation provided by the authors or of the previous projects is examined during desk research.

The first review period has two goals. First, errors, shortcomings, and other issues are discussed. Second, the review period enables the start and direction

¹⁹ <http://www.loa.istc.cnr.it/dolce/overview.html>

²⁰ <http://ontolog.cim3.net/wiki/COSMO.html>

of the second design iteration. An unstructured discussion session with digital humanities experts and the domain expert underlines the review period.

Finally, the design is tested once again in the second review period. In this review period, a workshop is conducted with several stakeholders from the Huygens ING. In this workshop, the developed artifact can be user-tested, thus validating the research results. Aside from the workshop, a short data story is written to illustrate the usefulness of the created knowledge graph. A summary of the research methodology is captured in figure 3.

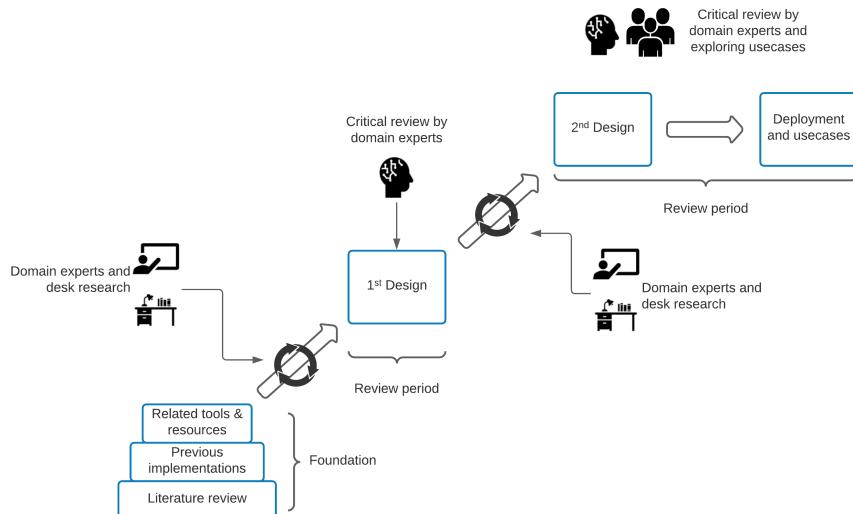


Fig. 3. Visual representation of the thesis methodology

Completing the constructed design framework aims to answer the main- and sub-research questions. The *sustainability* aspect can be fulfilled by evaluating the design and approach with the design guidelines. *Usefulness* can be measured with the completion of the competency questions. [5] In this research project, the competency questions are derived from two sources: the Huygens ING domain expert and a related project[15]. The competency questions are listed below along a source indicator (D for domain expert and R for related project):

- (D) Which VOC Chamber was accountable for the highest number of slaves transported?
- (D) How were the shipping routes in Asia divided between the various VOC chambers (for example, did ships from a specific chamber only sailed on certain routes)?
- (D) What was the average value of cargo on VOC return voyages per crew member/ship's ton, and how did this evolve over time?

- (D) To what extent was the value per ships' ton on return voyages correlated with the skipper's track record (how much experience, i.e., how many previous voyages / how quickly did a ship get to Asia on an outbound voyage)?
- (R) How did the price of goods change over time?
- (R) Search for specific information in large datasets, such as names.
- (R) Can external factors, such as war, be linked to the intensity of shipping?

4 Initial design

In the first design cycle, the analysis of the domain knowledge will take more time and effort than the second design iteration. Documenting challenges and other factors contributing can contribute to lowering the 'learning curve' for future iterations. The structure of the iterations is derived from, as discussed in the methodology, the work of Jovanonik[13]. The final two steps of the process are not completed entirely, as the results are not hosted on the web.

4.1 Domain and data knowledge

In the first step of the process, Jovanonik describes two distinct actions. First, the researcher has to understand the data and its context. Second, the researcher has to develop an understanding of ontologies and models used in the domain. As the second part has already been discussed in previous sections, this step will focus merely on comprehending the data and its context. Three VOC related datasets are included in the first design iteration and are addressed below:

Dutch Asiatic Shipping (DAS)

The Dutch Asiatic Shipping (DAS) dataset is a large dataset containing over 8700 voyages from the Netherlands to the Dutch Indies and back. The core details are noted from every voyage, such as the VOC Chamber who outfitted and contracted the ship or the departing and arrival date of the ship at the ports.²¹ This dutiful noting of events during the voyages and the eventual digitalization resulted in a vast normalized dataset of 72 columns spread out over nine tables. Most of the columns and values could be interpreted directly due to the logical naming schematic—for instance, the departing date of a ship is denoted with the column "voyDepartureEDTF". Other, more challenging, columns could be answered by interviewing one of the domain experts. The availability of two voyage IDs in the dataset (voyId and voyNumber), is an example of a challenging attribute. At first, the choice was made to use the latter voyNumber as the indicating value for the voyage, but it was pointed out that the voyId would be more relevant as it appeared in other datasets. Furthermore, ships also had two identifiers, but for another reason. It was explained that the same ship could change names in between or even during voyages. A remark denoted this event:

²¹ <http://resources.huygens.knaw.nl/das> retrieved at: 13-05-2021

"Ship was renamed on its third voyage from Barbara Theodora to Torenvliet." This possibility means that the ship's name is not persistent and thus should not be used in any identification. A final striking attribute of the DAS set is the onboard category. In this table, different categories listing from onbI to onbVI are linked to the voyage and are counts of crew and passengers during specific events. These events include, but are not limited to, boarding, disembarking, and even casualties. These events are not directly understandable from the column names, but an explanation was provided.²²

The core concepts of the DAS dataset can be summarized in a couple of points:

- Voyages are leaving and arriving on a specific date from and to a specific port. Some voyages are recorded to call at the port of The Cape of Good Hope (SA) for refilling supplies and crew;
- During these voyages, events occurred, such as a storm or a capture of an enemy vessel. These events are stored in the voyParticulars column;
- Voyages took place on ships, and these ships can have different names over time, have different types, have several masters responsible over different voyages, were built at shipyards, and were outfitted by VOC Chambers.

Bookkepper-General Batavia (BGB)

The following dataset that is added to the iteration is the Batavia Bookkeeper-General(BGB). In Batavia (now Jakarta ID), the bookkeeper and its clerks kept track financially of the goods coming into the Dutch Indies and leaving the Dutch Indies. These records were mainly lost over the years. However, a sizeable portion of the books was retrieved and eventually digitized and published.²³

There are overlapping features between the previously discussed DAS and the current BGB dataset. There are also recorded voyages from one particular port to another port on a particular date in this dataset. Aside from these voyages, the core of the BGB is the cargo transported on these voyages. As the goal of the BGB was to track the movements and transactions of ships financially, extensive detail has been put into the cargo record. It includes the specific item transported, its value, its unit of measurement, and even some specific detail about the item. These details range from ordinary things to fascinating details, such as "intended for the kings of Cheribon".

Due to the earlier 'practice' on the DAS dataset, knowledge about the BGB set could be acquired more easily. Nevertheless, some columns and values remained ambiguous. It seemed that during voyages, multiple entries of the same good could be transported. These were stored separately due to fiscal responsibilities. Consequently, the 'carOrder' (the order of goods within a transport) should be used to uniquely identify goods on the ship to avoid duplicates. Another unusual aspect of the BGB set was that the duplication of two or more ships used on the same voyage. At first, it was thought to be an error of some

²² <http://resources.huygens.knaw.nl/das/help/onboard> retrieved at: 13-05-2021

²³ <https://bgb.huygens.knaw.nl>

sort. However, it was explained that ships could be used in convoy from one port to another. To keep it ‘simple’, the bookkeepers lumped together the different cargo of the ships. A final intriguing aspect of the BGB dataset is that the value of the goods is stored in two currencies. The currency used in the Republic at that time was the Gulden. However, the trading partners in Batavia and its surroundings preferred quality silver coins more than in Europe at that time. The bookkeepers in Batavia gradually increased the value of Guldens until the currency was outbalanced between the two regions. As a counter-measure, a different variant of the Gulden was introduced in Batavia, the so-called ‘light’ Guldens, while the regular Guldens were dubbed ‘heavy’ [23]. Note that the explanation above is just a (mis-)interpretation of the events by the researcher. It was further explained that small denominations of the Gulden (Stuivers and Penningen) could be converted to Guldens via a simple calculation.²⁴

The core concepts of the BGB dataset can be summarized in a couple of points:

- Like the voyages in DAS, voyages leave and arrive on a specific date to and from a specific port. Unlike DAS, the voyages in BGB do not have a stopover at the Cape of Good Hope (SA);
- These voyages transported several goods with a specific value, unit of measurement, and specification;
- Voyages partook on ships that have different names over time. Note that the information captured about ships is lacking in comparison to DAS.

Places

The Places dataset is a single table containing all the relevant geographical places. The table consists of a naming schema that firsts list the place and then the country code. Moreover, the geographical position of the place is also present in the dataset. This dataset is highly relevant in this research project due to its prevalence in other datasets. In the DAS and BGB datasets, any reference to a place is recognized and disambiguated and is linked to the Places table.

4.2 Data and ontology modeling

In the second phase of the first iteration, the gathered knowledge about the domain and the datasets is used in the modeling phase. This phase aims to map the identified columns and concepts to the existing model and add new attributes to that model if one of these concepts does not fit the model. The conceptual model as a mindmap is illustrated in figure 4.

Initially, in the mindmap, four main components have been identified. Note that the creation of the mindmap was before any conversions of datasets. Following the mindmap, the components have to be mapped to some existing or new ontological concept. Furthermore, the mindmap only illustrates the components

²⁴ 1 Gulden is worth 20 Stuivers and 1 Stuiver is worth 16 Penningen

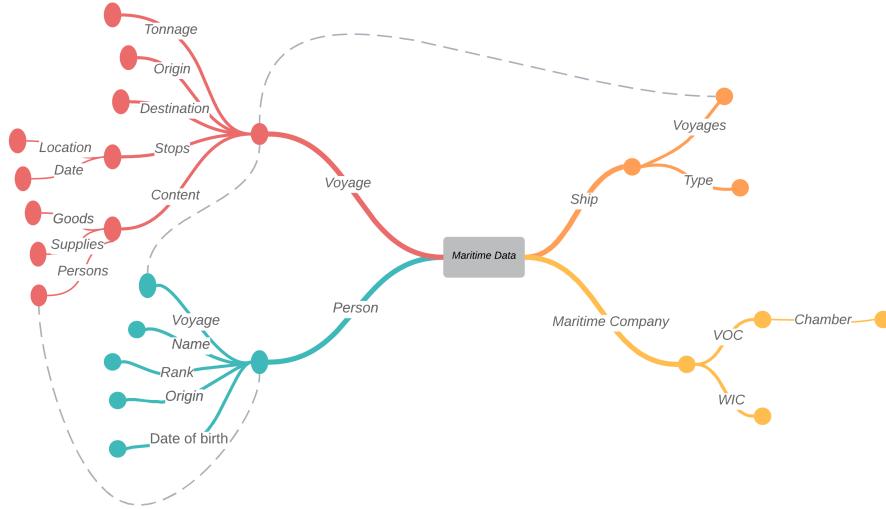


Fig. 4. Mind map of the conceptual model

and their relations but not the interrelation between components. These interrelations include, but are not limited to, a voyage taking place on a ship, a sailor participating on a voyage, and the VOC chamber outfitting the ship before the voyage. With these high-level concepts defined, the lower-level concepts can be specified.

The starting point of the lower-level, detailed concepts is the voyages. A voyage is the journey a ship undertakes, moving from one place to another. In this journey, sailors, soldiers, and other personnel are employed while ensuring that the goods transported on the ship safely reach their destination. The goal is first to reuse CIDOC CRM classes and properties in the modeling phase, if possible. Then, if that is not possible, reuse classes and properties used by the OntoMe implementation[16]. Finally, new classes and properties are created to accommodate the envisioned data model. These newly created properties and classes are integrated into the CIDOC CRM model as a subclass of existing classes. Therefore, allowing interoperability with other CIDOC CRM extensions and models.

Displaying the complete data- and ontology model, including the design decisions, will not benefit the readability. Therefore, a branch, starting from the voyage, is illustrated in figure 6 below. The complete mapping figures are included in the Appendices A-C.

The **voyage** instance was initially defined as a subclass of CRM: E7 Activity. This definition was changed later in the design iteration to a subclass of CRM: E9 Move to accommodate the linking of goods (physical objects) to the voyage via the property CRM: P25 moved. **ships** are defined as a subclass of CRM: E22 man-made object and are originating as well from the OntoMe implementation.

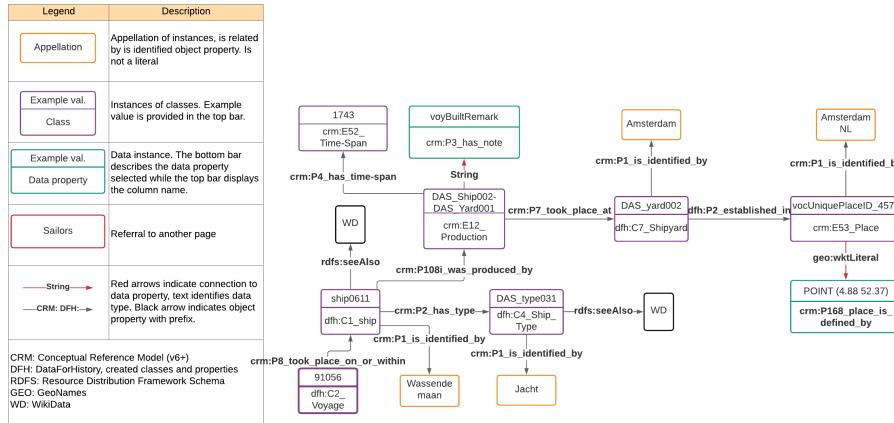


Fig. 5. Voyage to ship branch of mapping according to CIDOC CRM

The CIDOC CRM: P8 took place on or within is used as property between the **voyages** and the **ships**. In the property description of CRM: P8 took place on or within, the relation is even used as an example: ‘It describes a period that can be located with respect to the space defined by an E19 Physical Object such as a ship or a building.’, validating the use of this property. The **name of the ship** is defined using the property CRM: P1 is identified by, which links the ship instance to a CRM: E41 Appellation instance. There are several benefits of having the name of something as a URI instead of a literal value. For once, it allows to find and connect things with the same name. Furthermore, the URI name can also link to other entities, such as the origin of a name.

Ship types are defined as a subclass of CRM: E55 Type, and the linking between ships and the ship types occurs with the property CRM: P2 has type. The production of ships is modeled as a CRM: E12 Production instance with outgoing links to a CRM: E52 Time-Span for the date and the created class of **shipyard** for the production place. In the OntoMe project, the shipyard was modeled as a subclass of CRM: E40 Legal Body. However, linking together a Production and a Legal Body requires a new property. It does not exist in CIDOC CRM. Thus, in this research project, the shipyard is modeled as a subclass of CRM: E53 Place; a place where they make ships. The connection between a place and a production does exist in CIDOC CRM, namely the property CRM: P7 took place at. However, that design decisions lead to an issue. The **shipyard** is located in a **place**. There are properties in CIDOC CRM that can relate one CRM: E53 Place with another CRM: E53 Place. Such as CRM: P121 overlaps with or CRM: P10 falls within. While they both allow one place to fit in another place, the scope notes of these properties do not align completely with the model. Therefore, the property DFH: P2 established in was created to denote the relationship between a place and another place but with a temporal aspect.

Design issues, decisions, and solutions are spread out throughout the first design iteration. Going over them one by one, just like above, does not fit the thesis. Therefore, the most impactful issues are listed below. A complete overview of the model can be found in the Appendices A-C.

- In the first design iteration, two new classes are created for modeling the departure and arrival of ships at a port. In the OntoMe project, these activities do not have a class. They are connected via the created property P1 had departure place. This solution does seem more elegant, as it directly connects the voyage to a place. However, it is more difficult to state something about the departure or arrival of the ship in this manner. Properties of properties do not exist in RDF. Therefore, the classes of DFH: C10 Departing and DFH: C11 Arriving are created and modeled as subclasses of a CRM: E7 Activity. The property CRM: P9 consists of is used to connect these activities. This decision allows for easily linking dates, notes, and other events to the departure and arrival of a ship.
- In one of the data cleaning and reconciling projects of the Huygens, a match was made between masters of ships in the DAS file and the personnel logs of VOC Opvarenden (VOCOP). Not all persons could be identified, but the identified ones are linked with their VOCOP ID. This linking between datasets is captured with the CRM: P48 has preferred identifier.
- Another modeling decision stems from modeling economic goods, such as copper, spices, and others, transported on a ship. These goods have a certain value, expressed in Guldens Zwaar and Guldens Licht. However, to express the value of a Thing in CIDOC CRM, one has to create a Purchase activity. In our datasets, it is unclear who or when these goods are purchased. We only know the value of these goods. Therefore, the property DFH: P9 has monetary amount was created. This property allows for linking Things (economic goods) to a monetary amount without the need for a Purchase activity.
- A final proposition made in the modeling is the utilization of stopovers during voyages. These stopovers are adopted from the OntoMe project and integrated into the first design iteration. Arriving and Departing activities are linked in the same manner as they are linked to a voyage. However, the departure and arrival place are the same during a stopover. Hence the place is linked directly to the stopover.

4.3 Transformation into linked data

In the previous section, the conceptual mindmap is converted step-by-step into a detailed, thought-out model. The datasets are mapped to the model in the next phase, thus creating the knowledge graph. In the first design iteration, a pre-processing, time-saving step was completed before moving on to the mapping and conversion. The datasets were delivered as a relational schema. Meaning, the datasets were spread out over different tables while being connected via a key - foreign key relation. In a relational databases, this structure is very sensible.

It saves data storage. For instance, a ship can move multiple goods, so saving a single key instead of the whole ship entry makes the database leaner and easier to maintain. This database process is called normalization.²⁵

Nonetheless, we are creating a different structure than a normalized database. In RDF, there is no such thing as duplicate data. Remember that RDF is just statements in the form of triples. When making a statement: ‘The Amsterdam is a ship’ twice, the graph engine will only register one. The reason is that ‘The Amsterdam is a ship’ is registered as URIs: unique web identifiers. The graph engine records instances with an identical URI as the same instance. If the complete triple is already entered, that information already exists as far as the graph engine is concerned. Utilizing this feature of RDF, the several normalized tables are denormalized and combined into a single table per dataset. This pre-processing of the datasets will save time in the mapping; only one table per dataset has to be converted. This process is illustrated in figure 6.

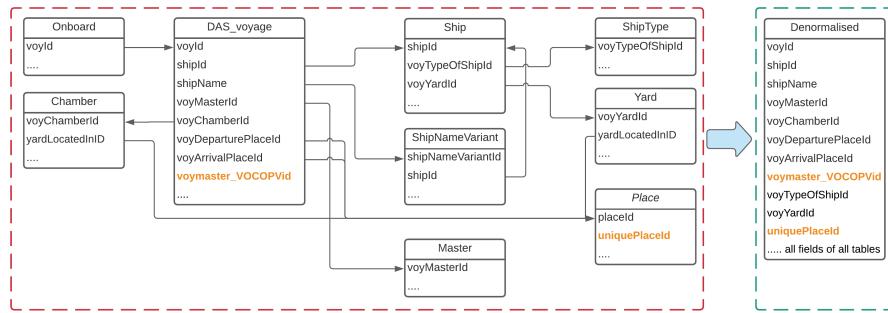


Fig. 6. Denormalizing example of the DAS tables

In principle, this pre-processing step can be completed with any relational database application. It requires the user to import the files into the database and create links from and to the different tables. Then, it is simply a matter of creating a SQL query that outputs every table and its links in a single view. In The wind in our sails project, MS Access was used due to its availability (pre-installed on most PCs) and ease of use.

In the mapping process, some columns are adjusted or newly created. Similar to the previous section, a small sample of the process is illustrated. In table 2, the places mapping is displayed.

In the mapping of the Places table, the *PlaceID* is mapped to the class of CRM: E53 Place. The PlaceID is used in the creation of the URI, consisting of /place/PlaceID. The *name of the place* is mapped to the CRM: E41 Appellation class. This URI starts with an /appellation/ identifier and is followed by the

²⁵ https://en.wikipedia.org/wiki/Database_normalization

Data Set Source	Name	Mapped To or property used	Note
Places	Existing uniqueStandardizedToponymID	CRM:E53.Place	
Places	Existing uniqueStandardizedToponymCountryCode	CRM:E41.Appellation	
Places	Existing URI	rdfs:seeAlso	
Places	Created GeoWKT (LNG + LAT)	geo:wktLiteral	Combined two values to create WKT

Table 2. Places mapping table

name and country code of the place. The URI pattern, including the type, is a way to create meaningful and insightful URIs, in line with standards.²⁶ Note that underscores replace spaces to avoid unintuitive URI encoding. The spacebar character is encoded as a %20; an underscore is better readable. The existing URI pointing towards a GeoNames instance is transferred into the knowledge graph. Finally, the created column *GeoWKT* is a composition of the existing latitude and longitude values. The reason being that this GeoWKT format can be utilized in the visualization part of the knowledge graph, more on that in the next section.

The most impactful mapping decisions are highlighted below. The complete mapping tables, including a general URI approach, can be found in the Appendices D-F.

- **The denotation of time** plays a central role in the maritime model. It relates to many concepts. Such as the arrival or departure of ships. Furthermore, time can be used in calculations. For example, to derive the total length of a voyage. In the datasets, the most precise level of time is on a date level. In some entries, this precision varies. It can happen due to unreadable records such that the listed date can only be guessed. Moreover, it can happen that a departure activity happened around midnight, effectively spanning two dates. In the initial phases of the conversion, the dates were sanitized and formatted according to ISO standards. The reasoning behind this decision is as follows: an involved related researcher knows that some dates might be ambiguous. Thus they can account for that in calculations. In contrast, another researcher might not know this and assumes that all the dates are in the same format. This assumption can lead to problems in their calculations and overall lower usability of the artifact. However, in consultation with the domain expert, the original dates could enrich the dataset. It was decided that the original dates were added as an extra attribute for researchers to consult while retaining data usability.
- Converting datasets into a knowledge graph is essentially the creation of triples in RDF format. Part of this process is deciding which entity is related to which. Another essential part of this process is the creation of the entities. Entities are formatted as URIs. Whenever two entities have the same URI, then these entities must be the same. Therefore, it is critical to determine and **create unique URIs for unique things** and use multiple URIs for different things. An example is provided by the conversion of

²⁶ <https://www.pldn.nl/wiki/Boek/URI-strategie#URI-patroon>

goods shipped on the voyages. The good copper can be shipped on different voyages and in different quantities for different prices. It will make it impossible to link this entry-specific information if a single entity is used for all these goods. Therefore unique URIs are created for every good transported. However, repeated information is stored in a type classification, eliminating data pollution. The goods URI is built out of the following components: 1386 (Product Type) - 99405 (Voyage) - 19 (Order). The Order component is added as voyages could ship the same good in different entries because of accounting responsibilities.

- Creating new and unique URIs is essential for unique entities, as demonstrated in the previous point. **Nevertheless, for entities that are the same, the same URI should be used** to avoid data pollution. Identifying entities across different datasets is rather tricky[24] and it is not captured in this research project. Fortunately, other researchers have performed data recognition and linking on these datasets [25, 26]. For example, the BGB dataset is linked at some entries to DAS voyages. In these cases, the DAS voyage entity is reused, effectively linking the two datasets. The implications of entity recognition and linking is illustrated in the Appendices P.

4.4 Data interaction and visualization

In previous steps, the different datasets are converted and connected. The second part of the puzzle is making the knowledge graph accessible. Researchers should be able to query the created knowledge graph and wrangle the retrieved data to achieve their research goal. However, the usability of the knowledge graph would be significantly improved if the graph would be accessible via a straightforward interface. In The wind in our sails project, the Metaphacts platform²⁷ is selected as the interface. The reason is two-fold. First, the integration with GraphDB is seamless. The application comes in a pre-configured Docker container, allowing integration between different systems as well. Second, the Metaphacts platform hosts very usable graph exploration tools. These tools can help in answering the defined competency questions. In the following section, some examples are illustrated alongside their purpose related to this research project. Note that the interface can be hosted on the web, thus allowing improved findability.

The starting page of the application is a simple home page with two search functions and some pre-configured pages. The first search function can be used to search for names—for instance, the user types in Amsterdam. The semantic search component then returns multiple results, as Amsterdam can point to the place, chamber, and ships. The second search function is similar, but it is focused on date search. Utilizing this function already solves one of the competency questions: ‘Search for specific information in large datasets, such as names.’ The home page, including the search functions, is illustrated in figure 7 below.

The pages and tools can be created and configured using HTML and CSS. These options allow for great configurability and maintainability, as these are

²⁷ <https://metaphacts.com/product>

Welcome to the start page of this demo project.

In this test environment, the VOC knowledge graph is loaded and a few example queries and results are demonstrated.

Pages and queries:

- Map component containing all start- and end points of the voyages
- Table of all voyages with their departing and arriving point
- SPARQL result of grouping by Arriving Place and counting occurrence for each chamber

[Help Page](#)



Fig. 7. Starting page of the Metaphacts platform

widespread (markup) languages. The configuration of a search function using pre-installed components is relatively straightforward, as shown in the listing below.

Listing 1.1. Code example of the semantic search component

```

1 <div id="first">
2   <semantic-simple-search
3     query='
4       SELECT ?resource ?label ?labelDisplay ?typeLabel WHERE {
5         FILTER REGEX(str(?label), ?_token_--)
6         ?resource a ?type.
7         ?resource crm:P1_is_identified_by ?label.
8         ?type rdfs:label ?typeLabel.
9         BIND ( REPLACE( str(?label), "http://example.com/appellation/" , "" ) as ?labelDisplay )
10        FILTER ( langmatches(lang(?typeLabel), "en") )
11      } LIMIT 200'
12     placeholder='Search for an appellation.. (use - instead of spaces)'
13     template='<span style="color:blue">{{labelDisplay.value}}</span> ( {{typeLabel.value}} ) '
14   >
15   </semantic-simple-search>
16 </div>
```

Another feature is the ability to connect the created knowledge graph with the web of data. Entities that are reconciled at some places can be used to create a service SPARQL query. This query can include results outside of the knowledge graph, e.g., the web of data. An example is created for the ship entities in the knowledge graph. Every time a user enters the entity webpage of a ship, the application will send out a live request to Wikidata. Wikidata then returns a picture (if available) and all the facts it has about the ship. Updates on Wikidata would be immediately reflected on the platform as well. Creating this interface answers one more competency question: ‘Can external factors, such as war, be

linked to the intensity of shipping?'. External factors, as demonstrated, can be linked to the knowledge graph. The interface is demonstrated in figure 8 below.

Apellation of ship	Build on date	Departed on date	Departed from	Arrived on date	Arrived at	Qualified by VOC Chamber
Duyken	91056	1595-04-02	Tweel_NL	1596-06-06	Pulau_Enggano_ID	
Duyken	91069	1598-05-01	Tweel_NL	1598-11-26	Banten_ID	
Duyken	91087	1599-12-21	Tweel_NL	1600-09-01	Banten_ID	
Duyken	95828	1597-02-25	Bal_ID	1597-08-11	Tweel_NL	
Duyken	95834	1599-01-12	Bal_ID	1599-07-19	Tweel_NL	
Overijssel	91056	1595-04-02	Tweel_NL	1596-06-06	Pulau_Enggano_ID	
Overijssel	91069	1598-05-01	Tweel_NL	1598-11-26	Banten_ID	
Overijssel	91087	1599-12-21	Tweel_NL	1600-09-01	Banten_ID	
Overijssel	95828	1597-02-25	Bal_ID	1597-08-11	Tweel_NL	
Overijssel	95834	1599-01-12	Bal_ID	1599-07-19	Tweel_NL	

WikiCommons picture (if available)

Wikidata Table

Property	Object
GND ID	7665079-9
VIAF ID	152407791
Library of Congress authority ID	n00041450
Bibliothèque nationale de France ID	1358024m
IdRef ID	052604160
Instance of	barque
Firebase ID	/m/02571f
Commons category	Duyken (ship, 1595)
number of mats	3
length	19.9

Fig. 8. Linked web implementation of the knowledge graph

There are many more components and pages created in the Metaphacts platform. These include graph exploration, table components, map components, and date searches. Screen captures of these components are included in the appendix (G).

4.5 Review of initial design

Modeling datasets into an existing ontology is a delicate process, even with high-level ontologies, such as CIDOC CRM. There is a constant push and pull between modeling from a data perspective and modeling from an ontology perspective. In a perfect world, the ontology and the data are a match. In this case, there exists a class and property for every envisioned concept. Nevertheless, this is not the case when using a general ontology, as it defeats its purpose. Therefore, the envisioned concepts should be fitted within the scope of the ontology. An economic good can not be modeled as a CRM:E21 Person, as it does not fit the scope definition of the class person. The CRM is an extensive ontology with almost 100 classes and over 200 properties. Discovering and understanding these scope notes takes time and experience. This experience is borrowed from two domain experts. The domain experts have an impressive track record with modeling projects into the CRM ontology. Hence, their knowledge and expertise during the first review period are greatly appreciated.

One of these misplacements in the first design iteration is the use of voyage. The voyage, as a class, was created as a subclass of the CRM:E9 Move. This choice stems from a modeling perspective; it links together the voyage with a physical object via the property CRM:P25 moved. In the CRM, the Move class is further defined as a subclass of the Activity class. The Activity class is defined as such:

"This class comprises actions intentionally carried out by instances of E39 Actor that result in changes of state in the cultural, social, or physical systems documented. This notion includes complex, composite and long-lasting actions such as the building of a settlement or a war, as well as simple, short-lived actions such as the opening of a door."

Reading the scope note, the choice of modeling a voyage as a subclass of Move does not seem like a wild choice. After all, the voyage is carried out by multiple actors, moving things from one place to another. However, for the advanced modeler, such as the two domain experts, the class Activity has a slightly different meaning. CIDOC CRM was initially created as an ontology for registering museum artifacts. Typically, a move indicated that someone moved an artifact from one place to another, in a single continuous activity. The voyage seems out of place, knowing this. Connecting the graph to other CRM graphs would create a world where an instance of a voyage is the same as moving a painting between musea.

This difference in semantics between classes is not unique to this project. The domain experts foresaw this and started developing an extension for the CRM ontology. This extension, focusing on social, economic, and legal life, allows for modeling concepts outside of the museum artifact realm.²⁸ Within extension, the class Phase exists. The use of a phase was explained as follows:

"A teacher is still a teacher, even when they are not teaching. When that person is changing jobs, they are not a teacher anymore, thus ending that phase. Teaching, explaining something to students, is an activity."

Consequently, the voyage can be defined as a phase. During a voyage, lots of activities can happen, such as raising the sails. However, during the night, the activities dwindle. There is no continuity; thus, while the sailors are still participating on a voyage, no direct activity is involved. Other extension classes and properties are used throughout the second iteration to close the gap between the data and the ontology.

We can take a step back to a more meta-level by checking the research questions' progress. Any forthcoming discrepancies can be fixed in the second design iteration.

- **Usable:** The solution is approaching usable. In section 4.4, some of the competency questions could be answered with the interface. However, not all competency questions can be answered at this moment. The first design iteration did not include the very extensive VOC Opvarenden (VOCOP) dataset. This dataset includes rich records of the sailors who ventured during these voyages. These records will be included during the second design iteration. Thus, fulfilling the usable requirement.

²⁸ <http://ontome.net/project/64>

- **Sustainable:** is not as strictly defined as the usability requirement. We can only review the approach and the decisions made during the first design iteration and determine where the sustainability can be improved. The approach, thus far, is *transparent* due to the extensive documentation. The Metaphacts integration allows for an *accessible* environment where the knowledge graph can be accessed interactively. By reviewing different approaches and methods, the current approach seems reasonably *reusable* as well. The reusability will only increase during the second iteration due to the utilization of the CRM extension. *Maintainability* is a quality attribute that could be improved during the second iteration. The knowledge graph could be maintained on GraphDB, as that information is not heavily reliant on other sources. This is not the case for the ontology, as The wind in our sails projects uses multiple external ontologies. Thus, whenever one of these ontologies changes, the administrator has to adjust the graph manually. Finally, the research of Waagmeester et al.[11] shows that sustainably growing the knowledge graph is primarily because of community involvement. In the current design iteration, neither concern is addressed. These shortcomings, including the current architecture, are illustrated in figure 9

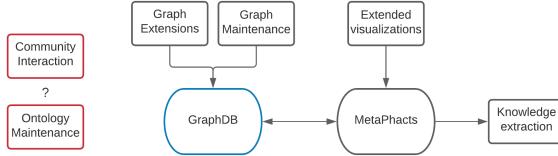


Fig. 9. Architecture of the first design iteration

5 Second iteration of the design

In the second design iteration, the results of the first review period are processed, and any shortcomings regarding the research questions are hopefully solved. This iteration will be noticeably compacter due to an improved understanding of the source data, improvements rather than additions, and fewer datasets converted.

5.1 Domain and data knowledge

The design thus far consists of voyages (DAS), cargo moved on voyages(BGB), and the places visited (Places). However, several competency questions are related to the people involved in these activities. In the second design iteration, attention is spent on adding persons to the design to solve these competency questions.

VOC Opvarenden (VOCOP)

The VOC and its chambers kept extensive records of its contracted crew members (Opvarenden in Dutch). Researchers over the years have digitized these records. As a result, over 700,000 records are stored in the Dutch Archives.²⁹ These records include personal data, ranks, wages, and references to voyages. The voyages overlap with the voyages of the DAS dataset.

Whenever a sailor signed up on one of the voyages, their name and origin were registered. Keen and lucky sailors could sign up again after their return. Personal identification numbers did not exist in these times. Hence, their name and origin provided enough identification, leading to more records than people. Identifying a person and understanding their career, for instance, would require entity recognition within these records. Data stored in these records that helped researchers identify these entities. One of these data points is the reason a contract ended. If the sailor would be unfortunate enough to perish during the voyage, that person could not rejoin a new voyage. Petram et al. [26] clustered entries belonging to the same person with some level of confidence. This process links together multiple entries to the same person, leading to better utilization and results. This process of entity recognition has also been completed for the places listed within the records. Just like names, places can be written differently. Due to normalization efforts, various appellations can be reduced to the same place.

Another aspect of the VOCOP data set is the debt letters. These letters are linked, where possible, to the contract entries. If the sailor would not make it back safely, someone would still be entitled to their wages; a Golden Age life insurance. The debtors mainly were familial relations or spousal. In other cases, an orphanage was listed as the debtor. At first sight, this might seem like a benevolent decision. However, in reality, teenagers were forced to enlist at the VOC while the orphanage raked in their wages.

The core concepts of the VOCOP dataset are summarized in a couple of points:

- There are many records of listings that include rich details. Due to entity recognition efforts, some of the listings can be related. This is also the case for some of the places used.
- The sailors provided rights to their income for the ones who remained in the form of debt letters.
- The records include references to the voyages from the DAS dataset.

5.2 Data, ontology modeling, and transformation into linked data

The inception of the model, displayed in figure 4, does not change in the second iteration of the design. The changes in the second iteration are in the details. One of the more critical changes is related to the structure of the voyages. In

²⁹ <https://www.nationaalarchief.nl/onderzoeken/zoekhulpen/voc-opvarenden>

summary of the previous iteration: a **voyage** is linked to a starting point via a departing activity. In some cases, a **stopover** occurred during the voyage. Finally, the voyage ends with the arrival of the voyage at its destination. Persons and goods were directly linked to the voyage; it was assumed that these were involved during the whole duration of the voyage. However, in the VOCOP dataset, this is not the case. Persons could change ships during their stopover, effectively participating in two voyages. The first iteration could not adequately convey this concept.

In an attempt to correctly convey this concept, a new class is introduced. The **leg** class is, similar to the voyage, a **phase**. It indicates a continuous part of the voyage from and to a harbor. A voyage can consist of multiple legs, thus eliminating the need for a stopover entity. The time between the end of the first leg and the beginning of the second leg is the stopover. Finally, the sailors are linked to the departing and arriving properties to indicate (dis)-embarkment activities. In the figure below, this construct is illustrated, along with the properties used.

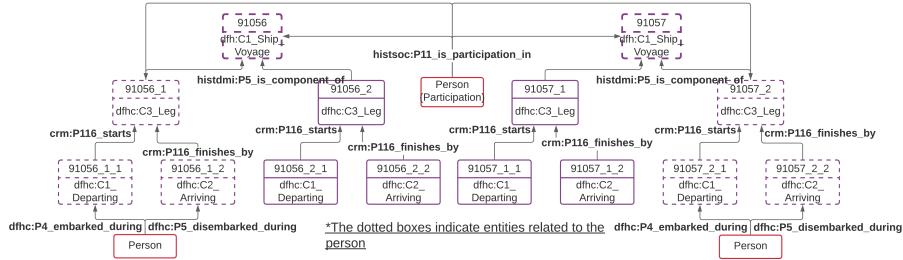


Fig. 10. Overview of the voyage construct

This construct might seem excessive, there exist many links between the sailors and voyage-related entities. Nevertheless, this construct allows researchers to analyze the participation on different abstraction levels. A simple search can be performed that returns all sailors who participated on a voyage, regardless if they changed ships. Another search can be performed on a leg level to determine who was on board during a specific part of the voyage. Lastly, a query can be constructed that returns who embarked on the ship and when. A direct link from the voyage to its starting- and ending point has also been included. This link could be created with the re-use of the existing OntoMe ontology. Direct links can improve the accessibility of the graph, as long as the indirect and direct paths do not conflict. This construct is also prepared and repeated for goods transported on the ship. An important assumption is that sailors disembark the ship at the end of the first leg and embark on the ship again when it leaves. This assumption is also expressed in the graph. In contrast, it is assumed that the goods are not unloaded during a stopover unless stated otherwise.

Other concepts are revisited as well. They are included in Appendices H-J. For instance, the onboard categories are updated. At first, the onboard categories were mapped via a custom data property to the voyage. However, this lean relation is a bit too short-sighted. Instead, we can model this onboard concept in a more semantic approach. These onboard components are the headcounts during a specific part of the voyage. This observation can be modeled as a CRM: E16 Measurement; it measures quantitative properties determined by observation. Furthermore, the measurement can be expressed in a CRM: E54 Dimension, e.g., a number. Expressing the onboard relation in this way creates more, maybe unnecessary, relations. However, it allows for other entities to relate better to this concept. Such as, who counted the number of people, or when the count occurred.

The VOCOP dataset has been modeled, mapped, and converted to RDF in the second design iteration. The CIDOC CRM extension (SDHSS) prefix for economic, social, and legal life has been utilized for the VOCOP dataset. (Re-)Using this extension lowers the number of classes and properties created, as it hosts some very useful concepts. The complete mapping can be found in Appendices K-L. In addition, the updated mapping tables of the other datasets are also included in Appendices M-N. An selection of modeling and mapping decisions is summarized below:

- The persons are related to the voyage via an SDHSS: **C15 participation** entity. This participation indicates the continuous participation of a person in an activity. Persons did not participate passively; they had responsibilities and work to do. These responsibilities, in the form of rank descriptions, were strictly defined within the VOC. The property SDHSS: **P12 is participation with social quality** is used to relate the practiced rank and thus occupation of that person.
- In the VOCOP dataset, only the median wages of the ranks are stored. Back then, the wages and the rank descriptions (aside from a reorganization in 1784) were relatively stable[26]. The **wage** is related to the **rank** via the property DFHC: **P11 monetary compensation**.
- Every **rank** has a list of responsibilities attached. For instance, a sailor is responsible for cleaning, loading and unloading, and more. There exist qualitative attribute definitions in the SDHSS. These include C15 **Skill** for physically demanding skills or C29 **Know-how** for mentally demanding jobs. However, we can not be confident of the skill or know-how of a person. They might be a lousy employee, failing their described tasks. Thus a new entity is introduced: DFHC: **C5 Occupational Responsibility**. This class mixes mentally demanding tasks, such as leading a crew, and physically demanding skills, such as hoisting the sails. The proven capabilities of a person can be converted to either C15 Skill or C29 Know-how.
- Personal relations can also efficiently be modeled within the SDHSS extension. The entity SDHSS: **C3 Social Relationship** links together two persons, a target and source. The class of SDHSS: **C4 Social Relationship Type** is then used to define the relationship. For the debt letters, a

relationship type of beneficiary and recipient is initially chosen. However, the debt letters can also indicate the relationship between the beneficiary and the recipient, adding another relationship type to the social relationship.

- The normalized **places** are linked to the **Places** mapping of the first iteration. For transparency and data completion sake, the original appellation of the place is included. The **place** mentioned at the **end of the contract** is not normalized yet. Meaning the names of these places vary wildly. Drawing meaningful conclusions of these entities can be challenging, as it is unknown which entities are identical. Therefore, this appellation is included in the graph but as a data property and not as an entity. In future iterations, these place appellations can be converted to entities after recognition and linking processes.

OntoME

The changes made during the second design iteration aim to improve the *reusability* of the artifact by utilizing an extended ontology. Creating mapping figures and tables improves *transparency* concerns. Converting the VOCOP dataset ensures that the *usability* aspect is met due to completing the competency questions. However, there were some discovered concerns regarding the *Maintainability* of the solution. In the first iteration, the ontology was designed using the widespread Protegé³⁰ application. This application has some very user-friendly options to start creating an ontology. Nevertheless, it lacks automatic alignment and reusing options with other CRM-based ontologies. The OntoMe³¹ application was created for projects like these. It natively supports CIDOC CRM and its common extensions. Users can create extensions themselves and connect them easily with others. The power of OntoMe is in the reusing of ontologies. The application has similarities with Github,³² allowing cooperation and discussion between users on the creation and utilization of classes and properties. Hosting the envisioned ontology, created during the cycles, in OntoMe would solve the *Maintainability* issue.

5.3 Data interaction and visualization

Changes in the ontology have a cascading effect on the Metaphacts front-end part of the artifact. Prebuilt components which require a specific class or property to function suddenly stop working. All components are checked and adjusted to accommodate the revised data model. These adjustments ranged from simply changing a class identifier to completely revamping a component in exceptional situations.

³⁰ https://protegewiki.stanford.edu/wiki/Main_Page

³¹ <http://ontome.net>

³² <https://github.com>

In the second design iteration, new concepts are added as well. For instance, the VOC Opvarenden (VOCOP) dataset is successfully integrated into the artifact. This dataset contains information about the personnel employed on the voyages. In an attempt to make this information visually appealing, two templates have been added. Whenever a user browses a voyage, the voyage template is used. Within this template, the activities during the voyage are displayed on a table and within a map. Furthermore, the participants of that voyage are listed on a table, including their rank. Finally, the goods shipped on the voyage are illustrated as well. The instance of a person is also templated. The template includes an activity table with occupied rank, a timeline, and an activity map. The activity table of a person illustrates the career of that person. He can start as a simple sailor and can occupy the rank of a master later in his career. This added functionality answers one of the competency questions. Screencaptures of these added templates are included in Appendices O.

Geovistory

Community interaction is not necessarily a research goal. However, it can have clear benefits. As the research of Waagmeester et al.[11] points out, sustainable growth of the knowledge graph can be realized with community involvement. An example where the knowledge graph can become larger is the extraction of knowledge within the voyage particulars. Each voyage has a short paragraph attached to it with the most notable events during that voyage. These particulars can fill in missing information or even generate entirely new knowledge. An example of notable events is extracted from voyage 93545:

“At Portsmouth from 02-05 till 09-05-1722, where one seafarer went over to the MIDDLEWOUD (2498); at Principe from 30-07 till 30-08-1722, where two seafarers and two soldiers deserted. From 11-09 till 16-09-1722 at Cape Lopez; from 21-10 till 10-11-1722 at Benguela; from 26-12-1722 till 18-01-1723 at Saldanha Bay. The ship was laid up at Batavia in 1730.”

This particular voyage includes five more legs than initially stated. Adding these legs creates a complete picture of the voyage and thus a larger knowledge graph. Unfortunately, going over these particulars one by one is rather time-consuming.

A proposal that captures this issue is the utilization of the Geovistory platform.³³ This platform is another interface that can interact with the knowledge graph. However, it differs from Metaphacts. Geovistory is focused on converting text sources to knowledge. It hosts a text editor that can recognize entities and propose linkage to existing entities within the knowledge graph—an appropriate use case for extracting knowledge from the voyage particulars. Furthermore, the platform is accessible for anyone with a user account. With the correct configuration, it is possible to enable the crowd-sourcing of knowledge extraction. Extending the proposed architecture with this platform will improve the developed artifact over time. The final architecture is illustrated in the figure below.

³³ <https://docs.geovistory.com/>

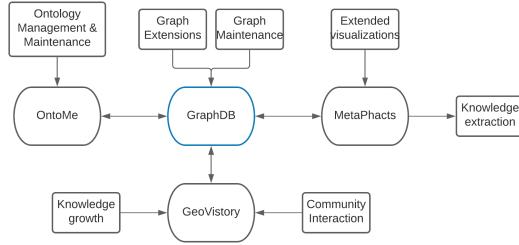


Fig. 11. Final proposed architecture

5.4 Review of the second iteration design

With the completion of the second design iteration, most of the issues discovered during the review period are solved during this iteration. These include, but are not limited to, better utilization of the CIDOC CRM model and its extensions, ontology management, community involvement, and the conversion of participants (VOCOP). In the second review, these improvements are explored and related to the primary research goal. The review consists of a workshop with five domain experts from the Huygens. The expertise of the domain experts ranged from IT infrastructure and development to data science, all with an affinity for history, of course. Second, the usability is tested with a short data-story.

Workshop

The workshop was conducted near the end of the research project and consisted of six participants. These participants were employed or were closely related to the main stakeholder, the Huygens ING. The roles of the digital-humanities experts varied from maritime historians to infrastructure specialists—a broad range of experts who are affiliated with the maritime domain. During the workshop, four topics were discussed:

- **Concept of Dutch Asiatic Shipping:** In the first topic, the DAS Concept (appendix H) was demonstrated to the domain experts. Notable feedback included that data provenance, such as the change of a usable date, should be included. However, the main graph is not the place for that as it pollutes the graph and thus diminishes the usability. A solution would be to have a separate graph for provenance linked to the main graph. This separate graph can then be optionally included in searches.
- **Utilization of CIDOC CRM and its components:** The choice to use CIDOC CRM and its extensions was discussed in the second topic. In general, the choice was received positively. The domain experts are involved in other digital heritage projects that use CIDOC CRM. Utilizing the same ontology means that The wind in our sails project has a good probability of being interoperable with these other projects. Another point was raised with CRM's complex method of expressing simple events; it was expressed

as verbose. This verbosity does have its benefits as it can capture completely different events in the same method. Finally, the extensions of CIDOC CRM were discussed. New extensions are developed by groups or individuals who are merely focused on their projects without re-using or considering existing extensions. There was consensus that the OntoMe application could alleviate this issue.

- **Metaphacts platform as interface:** In the third topic, the interactive Metaphacts interface was demonstrated. Overall, the impressions were positive. The practical goal of this research project was to create a single interconnected data source. The completion was confirmed by illustrating the voyage template (Appendix O). This template includes the voyage information of DAS, the economic goods information of the BGB, and the participant information of VOCOP. Furthermore, enthusiasts who want to gather specific knowledge and domain researchers who want to answer research questions can use the intuitive components of the platform—concluding that the implementation of the research project is deemed usable. However, there was also some negative feedback. Developing a data story straight from Metaphacts does not seem possible. This missing feature is not critical but could be helpful for further implementations.
- **Proposed infrastructure:** Finally, the proposed infrastructure, illustrated in figure 11, was discussed. The proposal has some exciting features, such as the community involvement possibility of Geovistory. It was noted that the community-driven data must be kept in a separate data layer as uncontrolled data lead to pollution of the graph. A downside of the proposal is the limited use of open-source software. Nevertheless, the proposed infrastructure is further researched outside of the realm of this project.

The workshop provides a clear end to the research project. The feedback was overall positive but also provided criticism. The developed artifact will be transferred to the stakeholders after the finalization of the thesis. Feedback received after the transfer is outside the scope of this research project.

Data story

Initially, one of the goals in The wind in our sails research project was to create a usable knowledge graph in the field of Dutch maritime data. Discussion on what makes a knowledge graph useful stems mainly from the perspective of maritime historians, as they are the primary stakeholders. They provided the competency questions which are used to test and improve the artifact. However, the developed artifact is not only intended to be used by maritime historians. Other enthusiasts can be drawn in by their research of an event or a long-lost forefather who had a remarkable VOC career. Determining if the knowledge graph is useful for these enthusiasts is challenging. There is no clear picture of who these people are and what their research questions entail. Nevertheless, one subject has been found, and their research question is explored in a short data story. The data story is included in Appendices Q.

In short, the data story is about the life of Hendrik Prins. The story starts with limited knowledge about the life of Hendrik Prins. Only the name and place

of origin are known. By exploring and querying the knowledge graph using the SPARQL-endpoint, more and more can be told about the life of Hendrik. For instance, his career from sailor to weapons master is illustrated. Furthermore, the events that happened during his voyages can also be retrieved. Even his demise is captured in this data story. Quantitative analysis is also possible using the artifact. The time-span of Hendrik Prins's voyages are plotted on a boxplot and are compared to similar time-spans. It turns out that the voyages of Hendrik Prins were overall faster. Creating stories like these can spark the interest of non-historians. The ease of writing such a data story can hopefully inspire others to write similar exciting papers.

However, limitations of the knowledge graph were also discovered during the construction of the data story. These limitations are inherent to the nature of the data and are difficult to solve. Nevertheless, they are relevant for the main research question, as it affects the usability of the artifact.

For example, modern data processing tools, such as Pandas,³⁴ can efficiently work with large amounts of data. The Pandas framework stores dates in a nanosecond resolution, limiting the usable timespan from 1677 to 2262.³⁵ Some of the records within the artifact are outside of these bounds, resulting in an error. There is a reasonable probability that researchers might run into similar issues utilizing different toolkits, thus, limiting the overall usability of the artifact.

The second limitation is within the data. It is logical that over the years, logbooks were lost or became unreadable. The researcher has to take this into account when analyzing results. This issue can be solved by uncovering new sources and appending them to the knowledge graph. Unfortunately, there is no possibility to make observations of your own, for obvious reasons.

In short, no solid conclusion can be made if this is the case for every enthusiast who wants to extract knowledge from the artifact. However, it does not bear any critical implications for the main research question. Further research could be performed that focuses more on this group and determines the usability with a survey, for example.

6 Results

The complete project is made available on GitHub³⁶ to promote the transparency and accessibility of the artifact. The GitHub repository includes the docker container with a preconfigured repository and preconfigured Metaphactory interface. Furthermore, the (intermediate) results in the form of triples and OpenRefine projects are included. The repository structure is explained in an included readme file.

³⁴ <https://pandas.pydata.org/>

³⁵ https://pandas-docs.github.io/pandas-docs-travis/user_guide/timeseries.html#timeseries-timestamp-limits

³⁶ <https://github.com/stijnschout3nthewindinoursails>

6.1 Results in numbers

In the table below, the scale of the project is illustrated. The complete docker package is over 9 GB and houses over 47 million triples.

Concept	Size in triples	Entities
Knowledge graph	47,749,878	
Named graph: vocop	14,042,325	
Named graph: das	650,328	
Named graph: bgbdas	1,602,837	
Named graph: places	35,026	
Named graph: bgbnodas	2,434,662	
Inferred triples	28,984,700	
Ontology size		2,417
Classes existing		161
Properties existing		251
Classes created		6
Properties created		21

Table 3. Results in numbers: the complete knowledge graph consists out of the named graphs and the inferred triples

6.2 Evaluation of competency questions

In the table below, the competency questions are listed, along with a brief statement about the solvability of each question.

Question	Answer
Which VOC Chamber was accountable for the highest number of slaves transported?	Can be answered by querying voyages that transported slaves
How were the shipping routes in Asia divided between the various VOC chambers (for example, did ships from a specific chamber only sailed on certain routes)?	Can be answered. Already a preconfigured component in Metaphacts
What was the average value of cargo on VOC return voyages per crew member/ship's ton, and how did this evolve over time?	Can be answered. Goods, participants and prices are available
To what extent was the value per ships' ton on return voyages correlated with the skipper's track record (how much experience, i.e., how many previous voyages / how quickly did a ship get to Asia on an outbound voyage)?	Can be answered with data research, the components are available in the knowledge graph. Data paper provides an example.
How did the price of goods changed over time?	Can be answered, goods are linked to voyages which have dates
Search for specific information in large datasets, such as names.	Can be answered with search components of Metaphacts
Can external factors, such as war, be linked to the intensity of shipping?	If the external factors are captured in the same ontology, then yes.

Table 4. Competency questions including statement regarding solvability

7 Discussion & future work

The path for future work, appending more datasets and improving the model, is paved by the used methodology. By following the framework of Jovanovik[13], which focused on reusable components, further iterations could be completed by other researchers. The transparent documentation aids researchers in understanding the decisions made during the design. Other datasets could include the Northern Musterolls³⁷ or the Archangel and Elbing voyages³⁸ in the Baltic sea.

However, even with many changes in the second design iteration, the processing time of the second design iteration was much shorter than the first iteration. Whereas the first iteration took around six weeks to complete, the second design iteration finished in two weeks. There is a steep learning curve in understanding the concepts, tooling, and methods used. Hopefully, other researchers will initiate new iterations and utilize the results of this research project to face a lesser steep learning curve.

There are also limitations to the research project. In the second design iteration, the proposed infrastructure is expanded with the use of OntoMe and Geovistory. However, unlike the completed integration of Metaphacts and GraphDB, the two extensions are not integrated entirely. The two extensions are web-based and, therefore, difficult to integrate without direct support. In future iterations, the

³⁷ <https://www.noordelijkscheepvaartmuseum.nl/nl/collectie-en-onderzoek/onderzoek/monsterrollen>

³⁸ <http://resources.huygens.knaw.nl/archangel/app/inleiding>

proposed architecture, including extensions, could be implemented on a web-server. However, this responsibility is outside the scope of this research project and lies with the curators and creators of the data.

Another limitation regarding provenance was discussed during the workshop. Data-altering improvements can also be captured in the artifact. If these changes are stored in the knowledge graph, other researchers could backtrack and understand the changes made in the data, thus improving the transparency of the artifact. However, provenance information should be stored in a separate data layer to avoid graph pollution.

8 Conclusion

The developed artifact, a semantic web knowledge graph, aims to support historians and other enthusiasts in their research. While the methodology also aims to develop the artifact sustainably, such that the artifact can be used and expanded over the years. These goals are captured in the main research question of the project:

“How can a sustainable knowledge graph be developed which is useful for researchers in the field of Dutch maritime data?”

In section six, the gathered competency questions are listed along with a statement about solvability. It can be concluded that the artifact can be deemed usable. Close to all competency questions could be answered by using the knowledge graph. The last competency question: “Can external factors, such as war, be linked to the intensity of shipping?” is debatable. The question could be answered if there was a dataset available that contained these events. A future iteration could include converting such a dataset in the same model to answer this specific research question conclusively.

Furthermore, the participants of the workshop further confirmed the usability. A sidestep has been made by evaluating the artifact with the data story (Appendix Q). During this evaluation, it was apparent that the artifact was useful for a broader range of people even though the sample size of that evaluation was one.

Limitations that hinder the usefulness of the artifact are also discovered. These limitations are inherent to the data itself and include missing knowledge and unsuitable frameworks.

The results regarding the sustainability of the methods used and the developed artifact are multifaceted. There is no single answer that defines a sustainable artifact and process. In the related project section, a conclusion is made about the sustainability of the previous iterations. Using the positive evaluation results and re-using these concepts in this project improves the sustainability. For instance, the transparent nature of the DSS project is re-used by providing precise and numerous documentation. Another concept of the OntoMe implementation is re-used by utilizing the same maintainable approach.

Developing a sustainable knowledge graph consists of making design decisions. On a high or detailed level, design decisions influence the sustainability aspects. Picking between these design options can often lead to dilemmas where there is no clear answer. The choice between data usability and data correctness is one of these dilemmas. Does the data need to be accurate, even if the data is not helpful and can not be processed? The original unaltered data is stored and the cleaned data in this project, fulfilling both sides of this dilemma. Another dilemma was faced during the first review period. In this review, the utilization of CIDOC CRM was examined. The goal is to re-use existing concepts of an ontology. This goal can conflict with the formal definition or use-case of that ontology. Which in turn diminishes the sustainability as different projects might use the concept differently.

The two design cycles make the project stand out as the development of the artifact benefits immensely from receiving feedback. Design decisions made during the cycles are not separated from the researcher. Personal beliefs, experience, and other factors might steer the artifact in a different direction. Therefore, the transparent and accessible nature of the project is of utmost importance. It means that other researchers can not only continue with further iterations but also backtrack decisions and take a different path.

Acknowledgements

I like to acknowledge several people who helped me in completing this research project. First, as someone far removed from the historical domain, Lodewijk Petram helped me throughout the project by answering questions and challenging me in some of the concepts. I'd also like to thank Francesco Beretta and Vincent Alamercury from the University of Lyon, who helped me understand the world of CIDOC CRM a bit better. They were also kind enough to introduce me to the OntoMe application and to participate in an interesting review session. The same thanks go out to the Digital Humanities experts from the Huygens who joined and actively participated in the workshop!

I graciously used a trial license of GraphDB and Metaphacts. Thanks again to the support team of Metaphacts, Sebastian and Wolfgang, who helped me get the software to run.

Special thanks to my thesis supervisors, Victor de Boer & Marieke van Erp, who helped me throughout the thesis project. They were actively involved in answering questions, returning feedback, and thinking along. Even in an ‘online thesis project’.

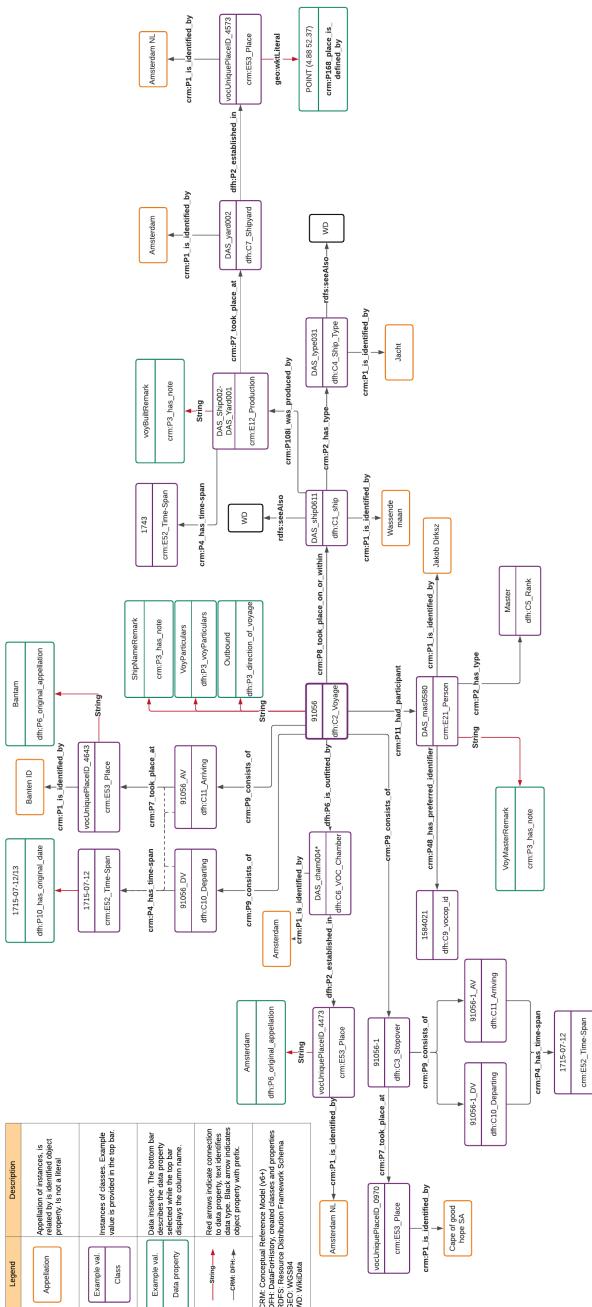
Finally, I'd like to thank friends, family, and girlfriend who supported me during these few months. They also joined my thesis presentation, which was very welcome! The enthusiasm surrounding the project was incredibly motivating. Thanks everyone, and I hope you, the reader, has enjoyed my thesis.

References

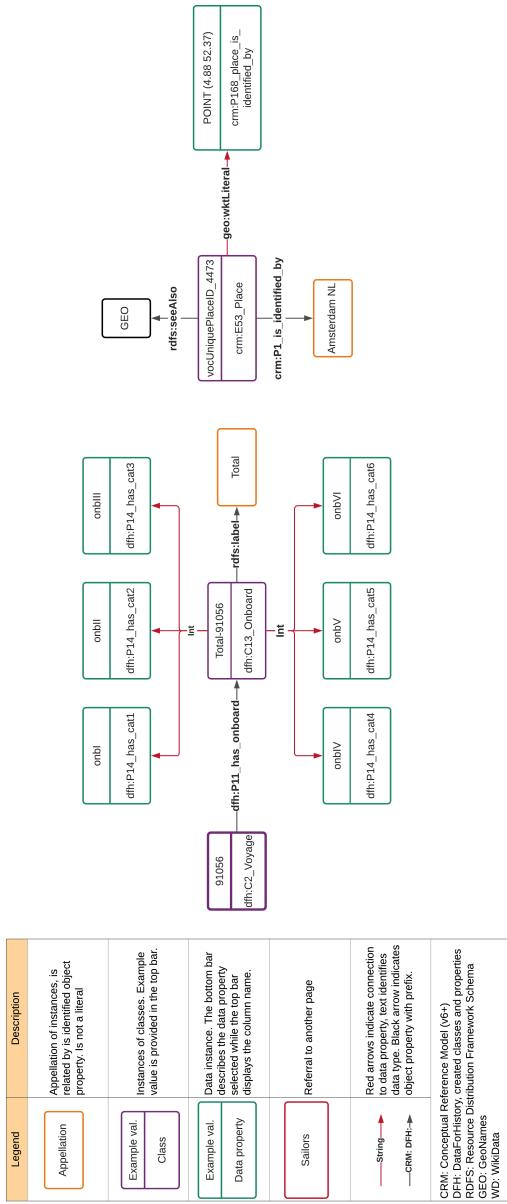
- ¹D. M. Berry, ed., *Understanding digital humanities*, OCLC: ocn701020028 (Palgrave Macmillan, Hounds Mills, Basingstoke, Hampshire ; New York, 2012).
- ²Claire Warwick, M. M. Terras, and J. Nyhan, eds., *Digital humanities in practice: Digitisation and digital resources in the humanities*, OCLC: ocn707962666 (Facet Publishing in association with UCL Centre for Digital Humanities, London, 2012).
- ³O. Gelderblom, A. de Jong, and J. Jonker, “The Formative Years of the Modern Corporation: The Dutch East India Company VOC, 1602–1623”, en, *The Journal of Economic History* **73**, 1050–1076 (2013).
- ⁴L. M. Hughes, ed., *Evaluating and measuring the value, use and impact of digital collections*, eng, OCLC: 774166752 (Facet Publ, London, 2012).
- ⁵K. Azzaoui, E. Jacoby, S. Senger, E. C. Rodríguez, M. Loza, B. Zdravil, M. Pinto, A. J. Williams, V. de la Torre, J. Mestres, M. Pastor, O. Tabouret, M. Rarey, C. Chichester, S. Pettifer, N. Blomberg, L. Harland, B. Williams-Jones, and G. F. Ecker, “Scientific competency questions as the basis for semantically enriched open pharmacological space development”, en, *Drug Discovery Today* **18**, 843–852 (2013).
- ⁶B. Haslhofer, A. Isaac, and R. Simon, “Knowledge Graphs in the Libraries and Digital Humanities Domain”, arXiv:1803.03198 [cs], arXiv: 1803.03198, 1–8 (2018).
- ⁷M. Kroetsch and G. Weikum, “Special issue on knowledge graphs”, *Journal of Web Semantics*, 1–4 (2016).
- ⁸C. Feilmayr and W. Wöß, “An analysis of ontologies and their success factors for application to business”, en, *Data & Knowledge Engineering* **101**, 1–23 (2016).
- ⁹Elsevier, *How AI and knowledge graphs can make your research easier*, en.
- ¹⁰M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axtон, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne, J. Bouwman, A. J. Brookes, T. Clark, M. Crosas, I. Dillo, O. Dumon, S. Edmunds, C. T. Evelo, R. Finkers, A. Gonzalez-Beltran, A. J. Gray, P. Groth, C. Goble, J. S. Grethe, J. Heringa, P. A. 't Hoen, R. Hooft, T. Kuhn, R. Kok, J. Kok, S. J. Lusher, M. E. Martone, A. Mons, A. L. Packer, B. Persson, P. Rocca-Serra, M. Roos, R. van Schaik, S.-A. Sansone, E. Schultes, T. Sengstag, T. Slater, G. Strawn, M. A. Swertz, M. Thompson, J. van der Lei, E. van Mulligen, J. Velterop, A. Waagmeester, P. Wittenburg, K. Wolstencroft, J. Zhao, and B. Mons, “The FAIR Guiding Principles for scientific data management and stewardship”, en, *Scientific Data* **3**, 160018 (2016).
- ¹¹A. Waagmeester, G. Stupp, S. Burgstaller-Muehlbacher, B. M. Good, M. Griffith, O. L. Griffith, K. Hanspers, H. Hermjakob, T. S. Hudson, K. Hybiske, S. M. Keating, M. Manske, M. Mayers, D. Mietchen, E. Mitraka, A. R. Pico, T. Putman, A. Riutta, N. Queralt-Rosinach, L. M. Schriml, T. Shafee, D. Slenter, R. Stephan, K. Thornton, G. Tsueng, R. Tu, S. Ul-Hasan, E. Willighagen, C. Wu, and A. I. Su, “Wikidata as a knowledge graph for the life sciences”, en, *eLife* **9**, e52614 (2020).

- ¹²A. Dimou, *Knowledge Graphs and Big Data Processing*, en, edited by V. Janev, D. Graux, H. Jabeen, and E. Sallinger, Vol. 12072, Lecture Notes in Computer Science (Springer International Publishing, Cham, 2020).
- ¹³M. Jovanovik, “Linked data application development methodology”, PhD thesis (Nov. 2016).
- ¹⁴V. de Boer, M. van Rossum, and R. Hoekstra, “The Dutch Ships and Sailors Project”, DHCommons Journal **1** (2015).
- ¹⁵J. Entjes, “Linking Maritime Datasets to Dutch Ships and Sailors Cloud - Case studies on Archangelvaart and Elbing”, PhD thesis (VU Amsterdam, 2015).
- ¹⁶F. Beretta, V. Alamercury, S. Derkx, L. Petram, and J. Schneider, *Geohistorical FAIR data: data integration and Interoperability using the OntoME platform*, Time Machine Conference 2019, Poster, Oct. 2019.
- ¹⁷R. Schaller, “Moore’s law: past, present and future”, IEEE Spectrum **34**, 52–59 (1997).
- ¹⁸N. Wirth, “A plea for lean software”, Computer **28**, 64–68 (1995).
- ¹⁹P. Heyvaert, D. Chaves-Fraga, F. Priyatna, O. Corcho, E. Mannens, R. Verborgh, and A. Dimou, *Conformance test cases for the rdf mapping language (rml)* (May 2019), pp. 162–173.
- ²⁰R. Hoekstra, A. Meroño-Peñuela, A. Rijpma, R. Zijdeman, A. Ashkpour, K. Dentler, I. Zandhuis, and L. Rietveld, “The dataLegend ecosystem for historical statistics”, en, Journal of Web Semantics **50**, 49–61 (2018).
- ²¹R. J. Wieringa, *Design science methodology for information systems and software engineering* (Springer, 2014).
- ²²Hevner, March, Park, and Ram, “Design Science in Information Systems Research”, MIS Quarterly **28**, 75 (2004).
- ²³J. Helleman, “Valutaproblemen bij de VOC”, MCA **4** (2011).
- ²⁴P. Christen, *Data matching: concepts and techniques for record linkage, entity resolution, and duplicate detection*, Data-centric systems and applications, OCLC: ocn809643173 (Springer, Berlin ; New York, 2012).
- ²⁵B. Hendriks, P. Groth, and M. van Erp, “Recognising and linking entities in old dutchtext: a case study on voc notary records”, Collect & Connect Leiden (2020).
- ²⁶L. Petram, M. Koolen, R. van Koert, M. Wevers, and J. van Lottum, “Data on the Maritime Workforce of the Dutch East India Company in the 18th Century”, to be published.
- ²⁷D. Nadeau and S. Sekine, “A survey of named entity recognition and classification”, en, Lingvisticae Investigationes **30**, 3–26 (2007).

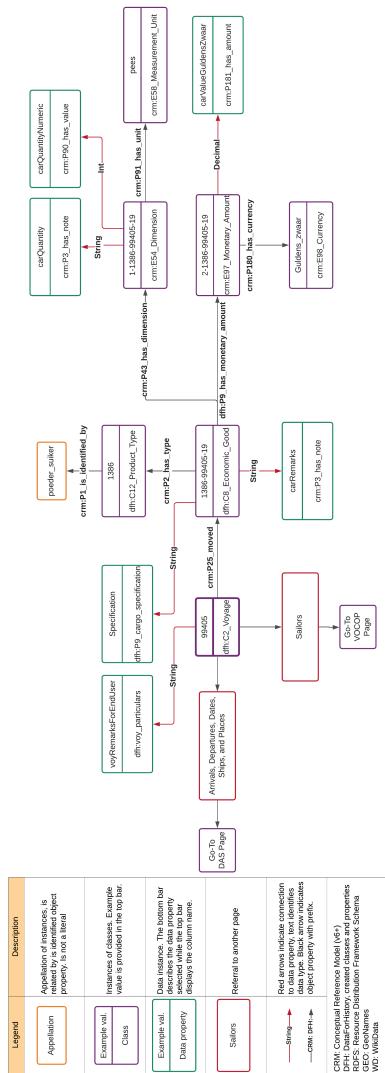
A First design iteration: DAS graph



B First design iteration: DAS onboard & places graph



C First design iteration: BGB graph



D First design iteration: DAS conversion table

Data Set	Source	Name	Mapped To or property used Note
DAS (das_Voyage)	Existing voyId	DFH:C2_Voyage	
DAS	Created Direction	DFH:P3.direction_of_voyage	Combined 'heenreis' and 'terugreis'
DAS (das_Voyage)	Existing shipID	DFH:C1_Ship	
DAS (das_Voyage)	Existing voyMasterID	CRM:E21_Person	
DAS (das_Voyage)	Existing voyMasterRemark	CRM:P3_has_note	
DAS (das_Voyage)	Existing voyChamberID	DFH:C6_VOC_Chamber	
DAS (das_Voyage)	Existing voyDepartureEDTF	CRM:E52_Time-Span	
DAS (das_Voyage)	Existing voyCapeArrivalEDTF	CRM:E52_Time-Span	
DAS (das_Voyage)	Existing voyCapeArrivalEDTF.Remark	CRM:P3_has_note	
DAS (das_Voyage)	Existing voyCapeDepartureEDTF	CRM:E52_Time-Span	
DAS (das_Voyage)	Existing voyCapeDepartureEDTF.remark	CRM:P3_has_note	
DAS (das_Voyage)	Existing voyArrivalDateEDTF	CRM:E52_Time-Span	
DAS (das_Voyage)	Existing voyArrivalDateEDTF.remark	CRM:P3_has_note	
DAS (das_Voyage)	Existing voyChamber2ID	DFH:C6_VOC_Chamber	
DAS (das_Voyage)	Existing voyParticulars	DFH:P4_voy_particulars	
DAS (das_Voyage)	Existing voymaster_VOCOPVid	CRM:E21_Person (preferred ID)	
DAS	Created StopoverC	DFH:C3_Stopover	To indicate a stopover at the Cape
DAS	Created CapeUniquePlaceId	CRM:E53_Place	To set the place of the stopover
DAS	Created DepartingValue (voyId + _DV)	DFH:C10_Departing	To create the entity for departing
DAS	Created ArrivingValue (voyId + _AV)	DFH:C11_Arriving	To create the entity for arriving
DAS	Created DepartingValueCape (StopOverC + _DV)	DFH:C10_Departing	To create the entity for departing at Cape
DAS	Created ArrivingValueCape (StopOverC + _AV)	DFH:C11_Arriving	To create the entity for arriving at Cape
DAS (ship)	Existing voyTonnageMin	DFH:P7_minimum_tonnage	
DAS (ship)	Existing voyTonnageMax	DFH:P8_maximum_tonnage	
DAS (ship)	Existing voyTypeOfShipID	DFH:C4_Ship_Type	
DAS	Created Yard_Joined (ShipId + YardId)	CRM:E12_Prodution	To create the entity of ship building
DAS (ship)	Existing voyBuiltRemark	CRM:P3_has_note	
DAS (ship)	Existing voyBuiltY	CRM:E52_Time-Span	
DAS (ship)	Existing voyYardYardID	DFH:C7_Shipyard	
DAS (shipName)	Existing shipNameVariant	CRM:E41_Appellation	
DAS (shipName)	Existing shipNameVariantRemark	CRM:P3_has_note	
DAS	Created externalShipName	rdfs:seeAlso	Reconcillement on shipname with WikiData
DAS (shipType)	Existing voyTypeOfShip	CRM:E41_Appellation	
DAS (shipType)	Existing voyTypeOfShipExternalID	rdfs:seeAlso	
DAS	Created voyMasterFullName	CRM:E41_Appellation	Combination of all the seperate nameparts
DAS	Created voyMasterRank	DFH:C5_Rank	To indicate the rank of the person (Master)
DAS	Created onbCategoryVoyage (voyageId + Category)	DFH:C13_Onboard	To create a unique identity of category and voyage
DAS (onboard)	Existing onbI	DFH:P14_has_cat1	
DAS (onboard)	Existing onbII	DFH:P15_has_cat2	
DAS (onboard)	Existing onbIII	DFH:P16_has_cat3	
DAS (onboard)	Existing onbIV	DFH:P17_has_cat4	
DAS (onboard)	Existing onbV	DFH:P18_has_cat5	
DAS (onboard)	Existing onbVI	DFH:P19_has_cat6	
DAS (yard)	Existing yardLocatedIn_„UniekeToponiemenVOCPOPV	CRM:E41_Appellation	
DAS (yard)	Existing yardLocatedIn_„UniekeToponiemenVOCPOPVid	CRM:E53_Place	

Table 5. DAS mapping table

E First design iteration: BGB conversion table

Data Set	Source	Name	Mapped To or property used	Note
BGB (cargo)	Existing	carVoyageId	DFH:C2.Voyage	
BGB (cargo)	Existing	carProductId	DFH:C12.Product.Type	
BGB (specification)	Existing	carSpecificationName	DFH:P9.cargo_specification	
BGB (cargo)	Existing	carUnit	CRM:E58.Measurement.Unit	
BGB (cargo)	Existing	carQuantity	CRM:P3.has_note	
BGB (cargo)	Existing	carQuantityNumeric	CRM:P90.has_value	
BGB (cargo)	Adjusted	carValueGuldens *	CRM:P181.has_amount	Converted/added stuivers and penningen to guldens
BGB (cargo)	Adjusted	carValueLichtGuldens *	CRM:P181.has_amount	Converted/added stuivers and penningen to guldens
BGB (cargo)	Adjusted	carRemarks	CRM:P3.has_note	
BGB	Created	carProductVoyageId (Product + voy + cargo)	DFH:C8.Economic_Good	To create a unique identifier for goods transported
BGB	Created	.Dimension (1-carProductVoyageId)	CRM:E54.Dimension	The leading number indicates different dimensions
BGB	Created	.MonetaryAmount (2-carProductVoyageId)	CRM:E97.Monetary_Amount	The leading number indicates different MA's
BGB (place)	Existing	vocPlaceID	CRM:E53.Place	
BGB (product)	Existing	ProductNaam	CRM:E41.Appellation	
BGB (voyageship)	Existing	shipId	DFH:C1.Ship	
BGB (voyageship)	Existing	DAS.voyage *	DFH:C2.Voyage	Used if available
BGB (voyageship)	Existing	DAS.shipID *	DFH:C1.Ship	Used if available
BGB (ship)	Existing	shipNaam	CRM:E41.Appellation	
BGB (unit)	Existing	unitNaam	CRM:E41.Appellation	
BGB (voyage)	Existing	voyDeparturePlaceId *	CRM:E53.Place	Used vocop places id
BGB (voyage)	Created	voyDepartureDate (Year + Month + Day)	CRM:E52.Time-Span	Combined date values to create ISO date format
BGB (voyage)	Existing	voyArrivalPlaceId	CRM:E53.Place	Used vocop places id
BGB (voyage)	Created	voyArrivalDate (Year + Month + Day)	CRM:E52.Time-Span	Combined date values to create ISO date format
BGB (voyage)	Existing	voyRemarksForEndUser	DFH:P4.voy.particulars	

Table 6. BGB mapping table

F URI mapping table

URI	Use
/voyage/	Ship Voyages
/ship/	Ships
/appellation/	Names and dates
/person/	Persons
/rank/	Rank and Occupation
/chamber/	VOC Chamber
/place/	Geographical Place
/shiptype/	Shiptype
/shipyard/	Shipyards
/production/	Production of ships
/stopover/	Stopover - used in first iteration
/participation/	Participation of a person during a voyage/leg
/leg/	Continious part of a voyage
/endofexistence/	Reason a participation ends
/departure/	Departure event
/arrival/	Arrival event
/relation/	Social relationship(type) between two persons
/monetaryamount/	Dimension of Monetary Amount
/number/	Numbers
/measurement/	Measurement of event
/dimension/	Dimension of measurements or physical things
/unit/	Expressed in unit
/point/	To denote arrival or departure; used in first iteration
/good/	Economic goods
/currency/	Currency

G Screen captures of Metaphacts platform: first iteration



Fig. 12. Search function of dates including the activity happening on that date

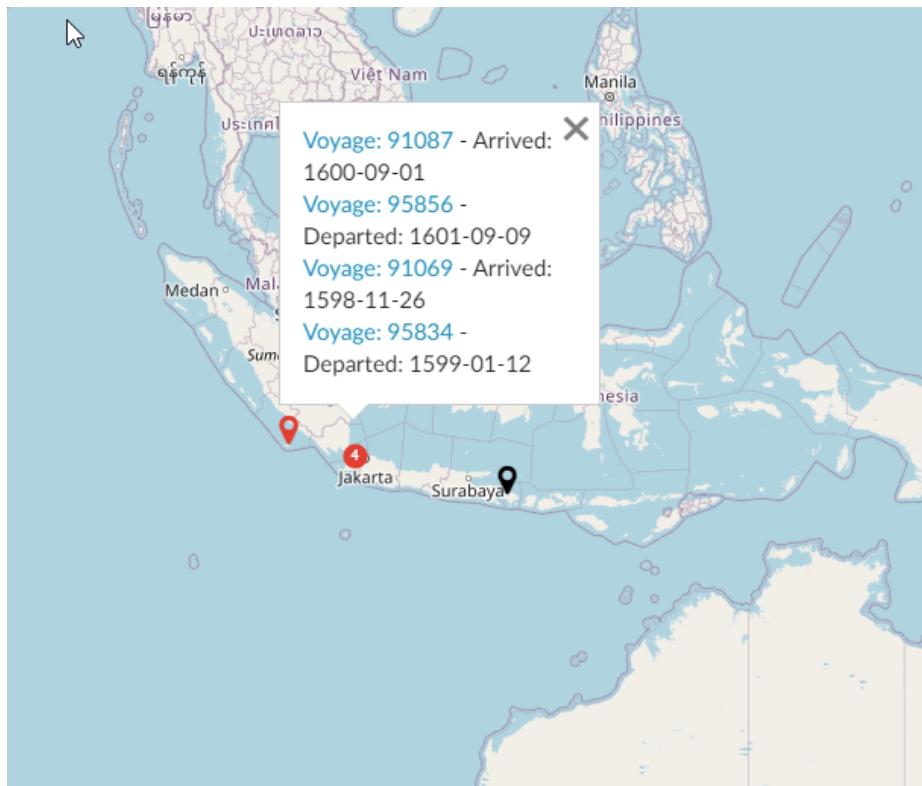


Fig. 13. Activity map of a certain ship



Fig. 14. Activity map of all the voyages registered

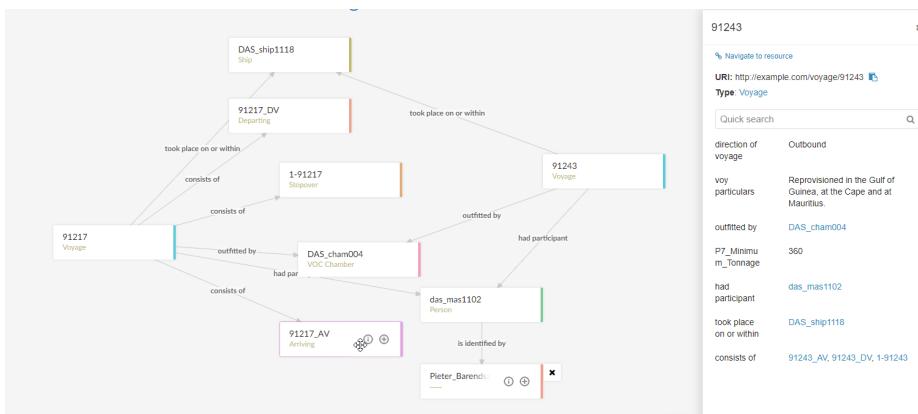


Fig. 15. Graph exploration interface

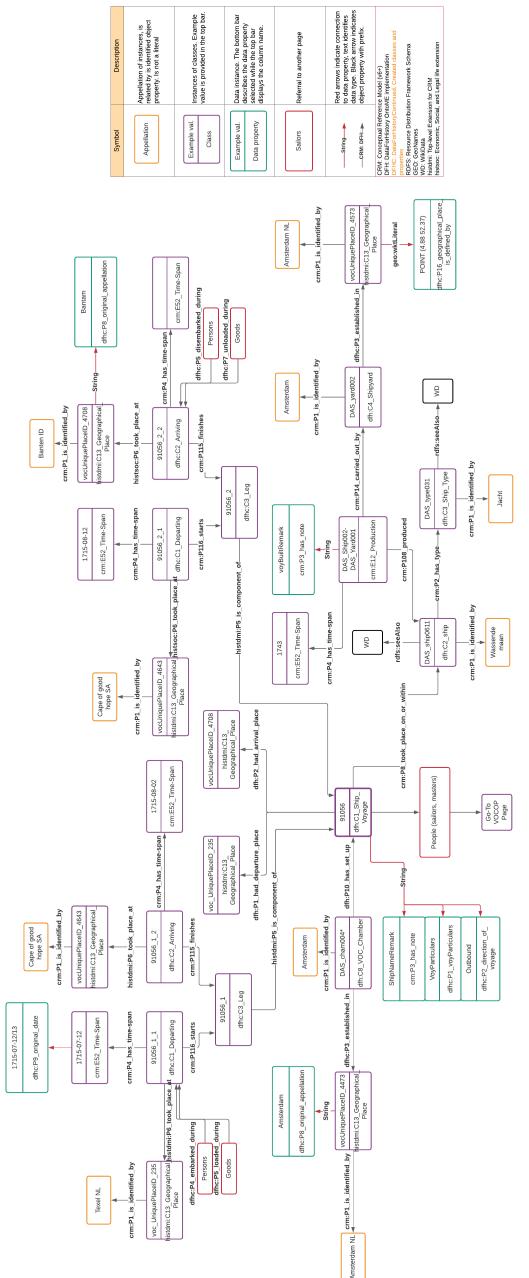
Quick search		<input type="text"/>	Group columns ▾
> vocChamberLabel		<input type="checkbox"/> voyage	departingName ↑ arrivingName
> VOC_Chamber_Amsterdam (3759)		<input checked="" type="checkbox"/> vocChamberLabel 3	Count: 3759 Count: 3759
> VOC_Chamber_Zeeland (1692)		<input type="checkbox"/> departingName	Count: 1692 Count: 1692
> VOC_Chamber_Enkhuizen (526)		<input type="checkbox"/> arrivingName	Count: 526 Count: 526
> VOC_Chamber_Delft (519)		<input type="checkbox"/> Clear grouping	Count: 519 Count: 519
> VOC_Chamber_Rotterdam (517)	517 values		Count: 517 Count: 517
> VOC_Chamber_Hoorn (511)	511 values		Count: 511 Count: 511
> VOC_Chambers_Rotterdam_and_Delft (5)	5 values		Count: 5 Count: 5
▼ VOC_Chambers_Hoorn_and_Enkhuizen (3)	98915, 99003, 99154		Count: 3 Count: 3
	99154		Jakarta_ID Plymouth_GB
	98915		Jakarta_ID Texel_NL
« 1 2 »			

Fig. 16. Smart filterable table

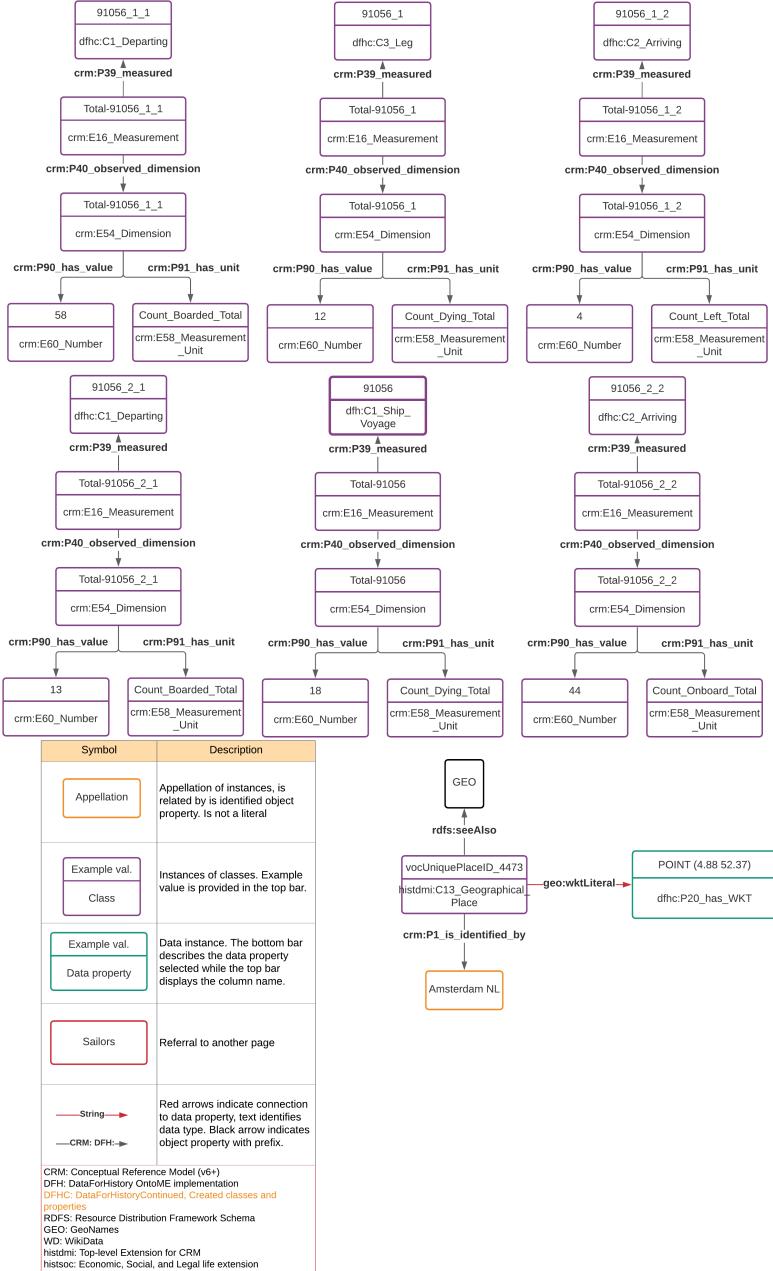
strippedArrivingName	Amsterdam	Delft	Enkhuizen	Hoorn	Rotterdam	Zeeland	Hoorn_Enkhuizen	Rotterdam_Delft
Sri_Lanka_LK	205	4	5	2	1	78	0	0
Banten_ID	52	6	11	6	7	18	0	0
Banda_Aceh_ID	4	0	1	0	0	0	0	0
Madagascar_MG	1	0	0	0	0	0	0	0
Cambodia_KH	1	0	0	0	0	1	0	0
Johore_MY	7	1	0	0	1	2	0	0
Goa_IN	9	1	1	2	2	5	0	0
Malabar_IN	2	0	0	0	0	0	0	0
Mauritius_MU	2	0	0	1	0	0	0	0
Pulicat_IN	14	1	0	2	1	0	0	0
« 1 2 3 4 5 »								

Fig. 17. Table illustrating routes between chambers, answers a competency question

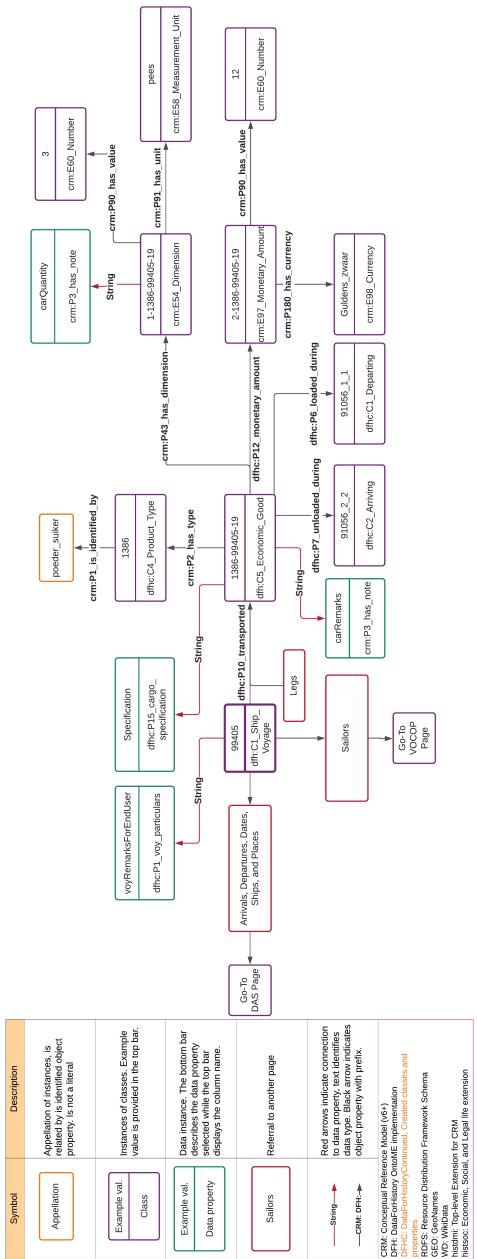
H Second design iteration: DAS graph



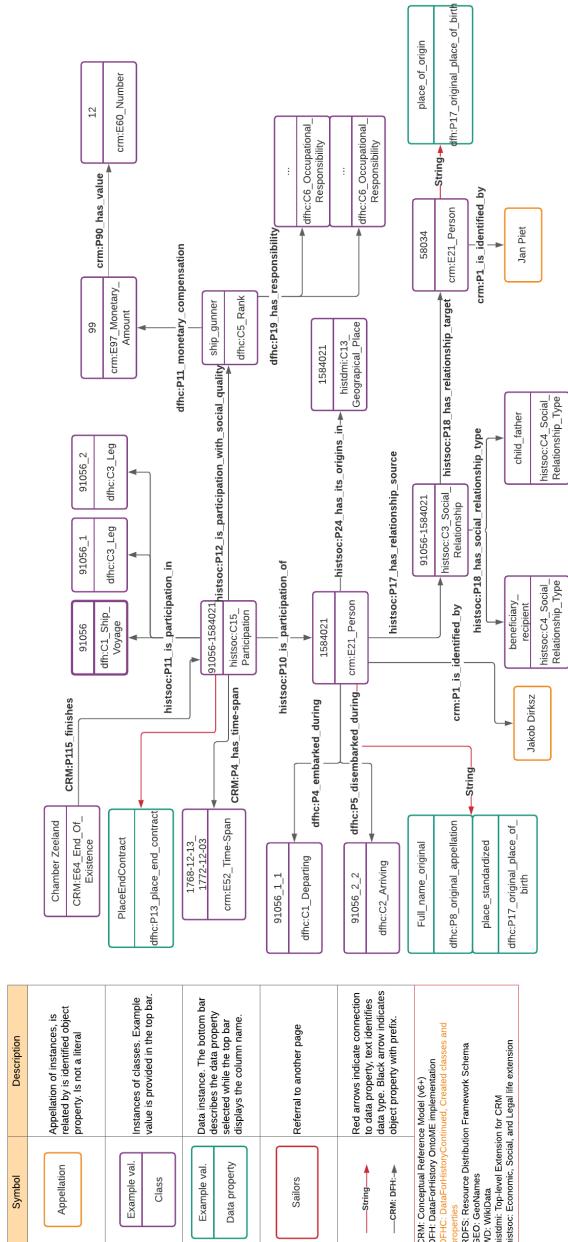
I Second design iteration: DAS onboard & places graph



J Second design iteration: BGB graph



K Second design iteration: VOCOP graph



L Second design iteration: VOCOP conversion table

Data Set	Source	Name	Mapped To or property used	Note
VOCOP (contracts)	Existing	vocop_id	dfnc:P14.original_vocop	Clusters are used, link to original id
VOCOP	Created	VID	crm:E21.Person	Used first entry of each clustered person
VOCOP (contracts)	Existing	full_name_normalized	crm:E41.Appellation	
VOCOP (contracts)	Existing	full_name_original	dfnc:P8.original_appellation	
VOCOP (places)	Existing	StandardizedId	histdm:C13.Geographical_Place	
VOCOP (contracts)	Existing	place_normalized	dfnc:P17.original_place_of_birth	
VOCOP	Created	participation	histsoc:C15.Participation	Combined person + first voyage id (chronological)
VOCOP (contracts)	Existing	outward_voyage	dfn:C1_Ship_Voyage	Existing in DAS
VOCOP (contracts)	Existing	changed_ship_at_cape_voyage.id	dfn:C1_Ship_Voyage	Existing in DAS
VOCOP (contracts)	Existing	return_voyage	dfn:C1_Ship_Voyage	Existing in DAS
VOCOP	Created	embarked	dfnc:C1.Departing	Created arrival en departing entities
VOCOP	Created	disembarked	dfnc:C2.Arriving	for mapping embarking and
VOCOP	Created	embarked_second	dfnc:C1.Departing	disembarking activities to the persons
VOCOP	Created	disembarked	dfnc:C2.Arriving	
VOCOP	Created	embarked_cape	dfnc:C1.Departing	
VOCOP	Created	disembarked_cape	dfnc:C2.Arriving	
VOCOP (ranks)	Existing	rank	dfnc:C5.Rank	as subclass of Occupation
VOCOP (ranks)	Adjusted	rank_description(eng)	dfnc:C6.Occupational_Responsibility	Splitted the multi-valued column
VOCOP	Created	comp_rank	crm:E97.Monetary_Amount	wage entity
VOCOP (ranks)	Existing	median_wage	crm:E60.Number	
VOCOP (contracts)	Existing	reason_end_contract	crm:E64.End_Of_Existence	links to the participation
VOCOP (contracts)	Existing	place_end_contract	dfnc:P13.place_end_contract	non-normalized placenames
VOCOP	Created	time-span	crm:C4.Time-Span	Combined starting and end date
VOCOP	Created	relationship	histsoc:C3.Social_Relationship	to indicate beneficiary relation
VOCOP	Created	defaultrelation	histsoc:C4.Social_Relationship.Type	standard beneficiary - recipient
VOCOP	Adjusted	relation	histsoc:C4.Social_Relationship.Type	translated the relations to en
VOCOP (beneficiary)	Existing	beneficiary_id	crm:E21.Person	recipient
VOCOP (beneficiary)	Existing	full_name	crm:E41.Appellation	
VOCOP (beneficiary)	Existing	place_of_birth	dfnc:P17.original_place_of_birth	non-standardized

M Second design iteration: DAS & Places conversion table

Data Set	Source	Name	Mapped To or property used	Note
DAS (voyages)	Existing voyId		DFH:C2.Ship_Voyage	
DAS	Created Direction		DFH:C2.direction_of_voyage	Combined 'heenreis' and 'terugreis'
DAS (voyages)	Existing shipID		DFH:C2_Ship	
DAS (voyages)	Existing voyMasterID		CRM:E21_Person	Used if no VOCOPID available
DAS (voyages)	Existing voyMasterRemark		CRM:P3.has_note	
DAS (voyages)	Existing voyChamberID		DFH:Cs_VOC_Chamber	
DAS (voyages)	Existing voyDepartureEDTF		CRM:E52.Time-Span	
DAS (voyages)	Existing voyCapeArrivalEDTF		CRM:E52.Time-Span	
DAS (voyages)	Existing voyCapeArrivalEDTF.Remark		CRM:P3.has_note	
DAS (voyages)	Existing voyCapeDepartureEDTF		CRM:E52.Time-Span	
DAS (voyages)	Existing voyCapeDepartureEDTF.remark		CRM:P3.has_note	
DAS (voyages)	Existing voyArrivalDateEDTF		CRM:E52.Time-Span	
DAS (voyages)	Existing voyArrivalDateEDTF.remark		CRM:P3.has_note	
DAS (voyages)	Created voyChamber2ID		DFH:Cs_VOC_Chamber	
DAS (voyages)	Existing voyParticulars		DFH:C1_voy_particulars	
DAS (voyages)	Existing voymaster_VOCOPVid		CRM:E21_Person	Used if available
DAS	Created CapeUniquePlaced		HISTDMI-C13.Geographical.Place	Is used as arrival and departure place of 1st and 2nd leg
DAS	Created DepartingValue (voyId + _DV)		DFH:C1.Departing	To create the entity for departing
DAS	Created ArrivingValue (voyId + _AV)		DFH:C2.Arriving	To create the entity for arriving
DAS	Created DepartingValueCape (StopOverC + _DV)		DFH:C1.Departing	To create the entity for departing at Cape
DAS	Created ArrivingValueCape (StopOverC + _AV)		DFH:C2.Arriving	To create the entity for arriving at Cape
DAS (ship)	Existing voyTonnageMin		DFH:C21.minimum_tonnage	
DAS (ship)	Existing voyTonnageMax		DFH:C22.maximum_tonnage	
DAS (ship)	Existing voyTypeOfShipID		DFH:C3.Ship_Type	
DAS	Created Yard_Joined (ShipId + YardId)		CRM:E12.Production	To create the entity of ship building
DAS (ship)	Existing voyBuiltRemark		CRM:P3.has_note	
DAS (ship)	Existing voyBuiltY		CRM:E52.Time-Span	
DAS (ship)	Existing voyYardYardID		DFH:C4.Shipyard	
DAS (shipName)	Existing shipNameVariant		CRM:E41_Appellation	
DAS (shipName)	Existing shipNameVariantRemark		CRM:P3.has_note	
DAS	Created externalShipName		rdfs:seeAlso	Reconcillement on shipname with WikiData
DAS (shipType)	Existing voyTypeOfShip		CRM:E41_Appellation	
DAS (shipType)	Existing voyTypeOfShipExternalID		rdfs:seeAlso	
DAS	Created voyMasterFullName		CRM:E41_Appellation	
DAS (voyages)	Existing voyMasterRank		DFH:C5_Rank	Combination of all the separate nameparts To indicate the rank of the person (Master)
DAS	Created Participation		HISTSOC:C15	Voyage + VOCOP ID
DAS	Created Leg1		DFH:C3	First leg of the voyage
DAS	Created Leg2		DFH:C3	Second leg of the voyage
DAS (voyages)	Existing Original.Top		DFH:P8	Original appellation of the places
DAS (yard)	Existing yardLocatedIn_UniekeToponiemenVOCPopV		CRM:E41_Appellation	
DAS (yard)	Existing yardLocatedIn_UniekeToponiemenVOCPopVid		HISTDMI-C13.Geographical.Place	
DAS (Onboard)	Created MeasL-6		CRM:E16.Measurement	Functions as the measurement
DAS (Onboard)	Created Dim1-6		CRM:E54.Dimension	Functions as the dimension of the measurement
DAS (Onboard)	Created Unit1-6		CRM:E58.Measurement_Unit	Functions as the unit; e.g. count disembarked
DAS (Onboard)	Existing Num1-6		CRM:E60.Number	Actual number of measurement
Places	Existing uniqueStandardizedToponymID		HISTDMI-C13.Geographical.Place	
Places	Existing uniqueStandardizedToponymCountryCode		CRM:E41_Appellation	
Places	Existing URI		rdfs:seeAlso	
Places	Created GeoWKT (LNG + LAT)		DFH:C20.has_WKT	Combined two values to create WKT

N Second design iteration: BGB conversion table

Data Set	Source	Name	Mapped To or property used	Note
BGB (cargo)	Existing	carVoyageId	DFH:C1_Ship_Voyage	
BGB (cargo)	Existing	carProductId	DFHC:C4_Product_Type	
BGB (specification)	Existing	carSpecificationName	DFHC:P15_cargo_specification	
BGB (cargo)	Existing	carUnit	CRM:E58_Measurement_Unit	
BGB (cargo)	Existing	carQuantity	CRM:P3_has_note	
BGB (cargo)	Existing	carQuantityNumeric	CRM:P90_has_value	
BGB (cargo)	Adjusted	carValueGuldens *	CRM:P90_has_value	Converted/added stuivers and penningen to guldens
BGB (cargo)	Adjusted	carValueLichtGuldens *	CRM:P90_has_value	Converted/added stuivers and penningen to guldens
BGB (cargo)	Adjusted	carRemarks	CRM:P3_has_note	
BGB	Created	carProductVoyageId (Product + voy + cargo)	DFH:C5_Economic_Good	To create a unique identifier for goods transported
BGB	Created	..Dimension (1-carProductVoyageId)	CRM:E54_Dimension	The leading number indicates different dimensions
BGB	Created	..MonetaryAmount (2-carProductVoyageId)	CRM:E97_Monetary_Amount	The leading number indicates different MA's
BGB (place)	Existing	vocPlaceID	SDHSS:C13_Geographical_Place	
BGB (product)	Existing	ProductNaam	CRM:E41_Appellation	
BGB (product)	Existing	Eng	rdfs:label	Automatic translation of product
BGB (voyageship)	Existing	shipId	DFH:C2_Ship	
BGB (voyageship)	Existing	voyages *	DFH:C1_Ship_Voyage	Used if available
BGB (voyageship)	Existing	DAS.shipID *	DFH:C2_Ship	Used if available
BGB (ship)	Existing	shipNaam	CRM:E41_Appellation	
BGB (unit)	Existing	unitNaam	CRM:E41_Appellation	
BGB (voyage)	Existing	voyDeparturePlaceId *	SDHSS:C13_Geographical_Place	Used vocop places id
BGB (voyage)	Adjusted	voyDepartureDate (Year + Month + Day)	CRM:E52_Time-Span	Combined date values to create ISO date format
BGB (voyage)	Existing	voyArrivalPlaceId	SDHSS:C13_Geographical_Place	Used vocop places id
BGB (voyage)	Adjusted	voyArrivalDate (Year + Month + Day)	CRM:E52_Time-Span	Combined date values to create ISO date format
BGB (voyage)	Existing	voyRemarksForEndUser	DFH:P4_voy_particulars	
BGB	Created	Leg	DFHC:C3	Transported goods
BGB	Created	Departure	DFHC:C1	Goods were loaded during
BGB	Created	Arrival	DFHC:C2	Goods were unloaded during

O Screen captures of Metaphacts platform: second iteration

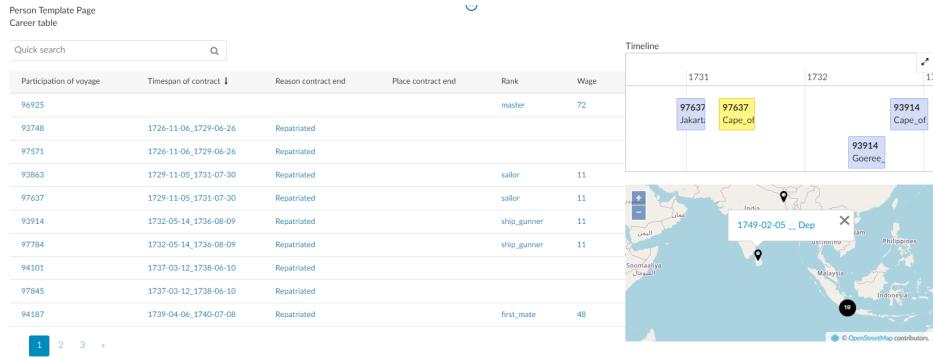
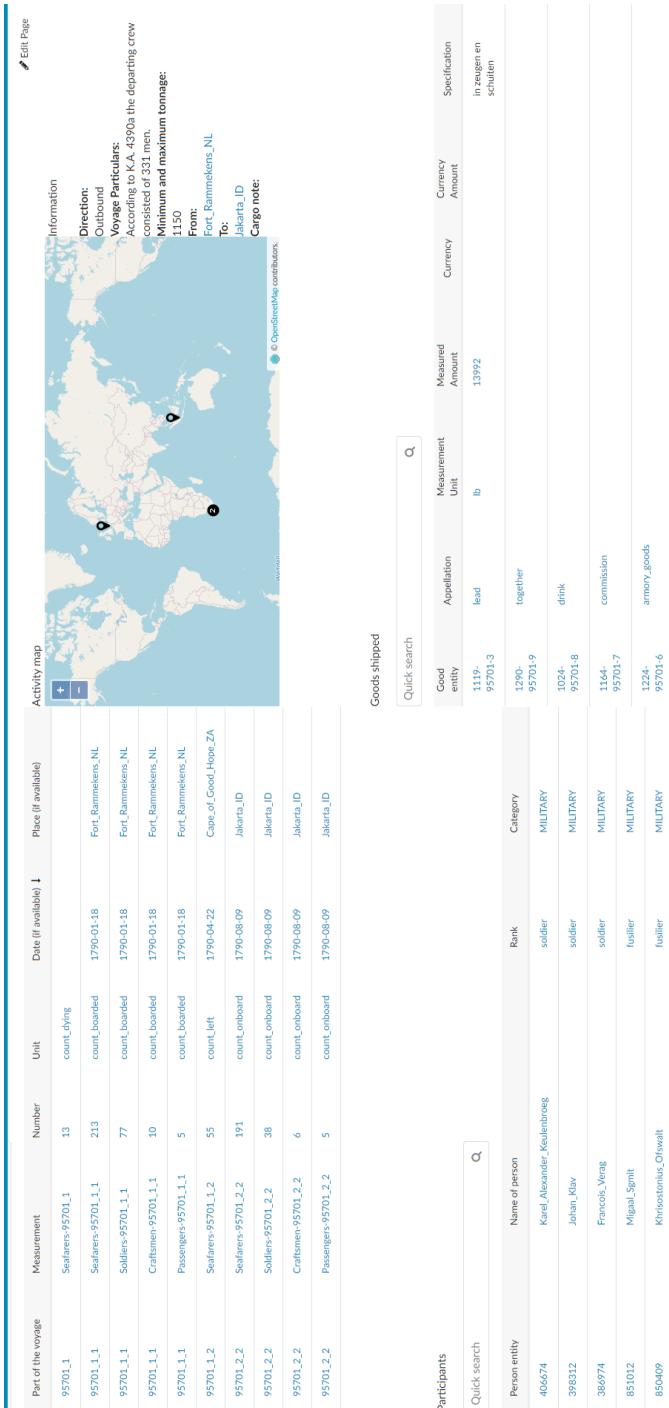


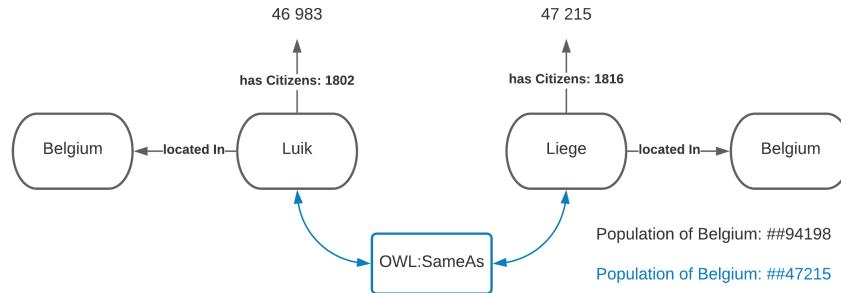
Fig. 18. Career page and activity map of a person

**Fig. 19.** Voyage page

P Entity recognition and linking and its implications

Entity recognition and linking processes consist of identifying and matching entities within the same or even multiple datasets. [24] For instance, the place "Luik" and "Liege" can exist both in the same dataset as a place of origin variable. If RDF conversion took these as face-value, it would create two different "City" instances when actually "Luik" and "Liege" are two names for the same city. This 'double naming' will be problematic if a researcher tries to determine the population of Belgium by taking all its cities and summing the population. The sum will count the city of "Liege/Luik" double if it both has a citizen property attached to it.

These duplicate entity issues would not be a big issue in a few instances, as they can be adjusted manually. However, in larger numbers, this would be taking considerable time and, in some cases, outright impossible. Fortunately, several natural language processing algorithms exist which can be trained to perform this exercise automatically. [27] If an entity matches, with some level of accuracy, with another entity, they could be connected via the property `sameAs`. This `sameAs` property is of a special type within RDF, and it allows connected instances to inherit all properties and class relations of each other. Meaning, the "Liege/Luik" issue described earlier would be solved by querying at most one population statistic (for instance, the most recent). Even with automated algorithms, this process is still complex due to the many variances of names and a wide range of variables to take into account. This issue and its solution are illustrated in the figure below.



Q Data story

Data story VOC

June 30, 2021

0.0.1 The life of Hendrik Prins



The Vereenigde Oostindische (VOC) was one of the largest employers in the 1700-1800s. Naturally, a sizable portion of the Dutch Republics' population was employed by the VOC. This phenomenon does not only sparks the interest of historians but also of other enthusiasts, especially if a forefather was known to be employed by the VOC. We have such a case with a personal relation of mine, the Prins family. By their research, they know that their forefather was named Hendrik Prins, and he had his origins in Delft. Interestingly enough, they still have the VOC emblem of Hendrik.

In this short data story, we will look into the life of Hendrik Prins. We start by just knowing his name and birthplace and use the VOC knowledge graph to explore related voyages, goods, and events during the life of Hendrik Prins.

Data story by Stijn Schouten, written on 26-06-2021 Composed using JupyterLab, using the SPARQL-Wrapper to access the GraphDB endpoint.

Finding Hendrik Prins In the query below, we select all the persons who originated from Delft NL and filter out the Hendrik Prins name.

```
[31]: sparql.setQuery(prefixes+"""
SELECT ?person ?name ?from
WHERE
{
```

```

?person histsocP:P24 <http://example.com/place/vocUniquePlaceID_1192>.
<http://example.com/place/vocUniquePlaceID_1192> crm:P1 ?from.
?person dfhc:P8 ?name.
filter contains(?name, "Hendrik Prins")
} """

```

	person	name	from
0	person/1042325	Hendrik Prins	appellation/Delft_NL
1	person/1087798	Hendrik Prins	appellation/Delft_NL

We find that there are two records with the name Hendrik Prins who originated from Delft. We can take a closer into these persons by retrieving more information.

```

[33]: sparql.setQuery(prefixes +"""
SELECT ?person ?participation ?timespan ?reasonend
WHERE
{
    ?person histsocP:P24 <http://example.com/place/vocUniquePlaceID_1192>.
    ?person dfhc:P8 ?name.
    ?participation histsocP:P10 ?person.
    ?participation crm:P4 ?timespan.
    ?reasonend crm:P115 ?participation.
    filter contains(?name, "Hendrik Prins")
}""")

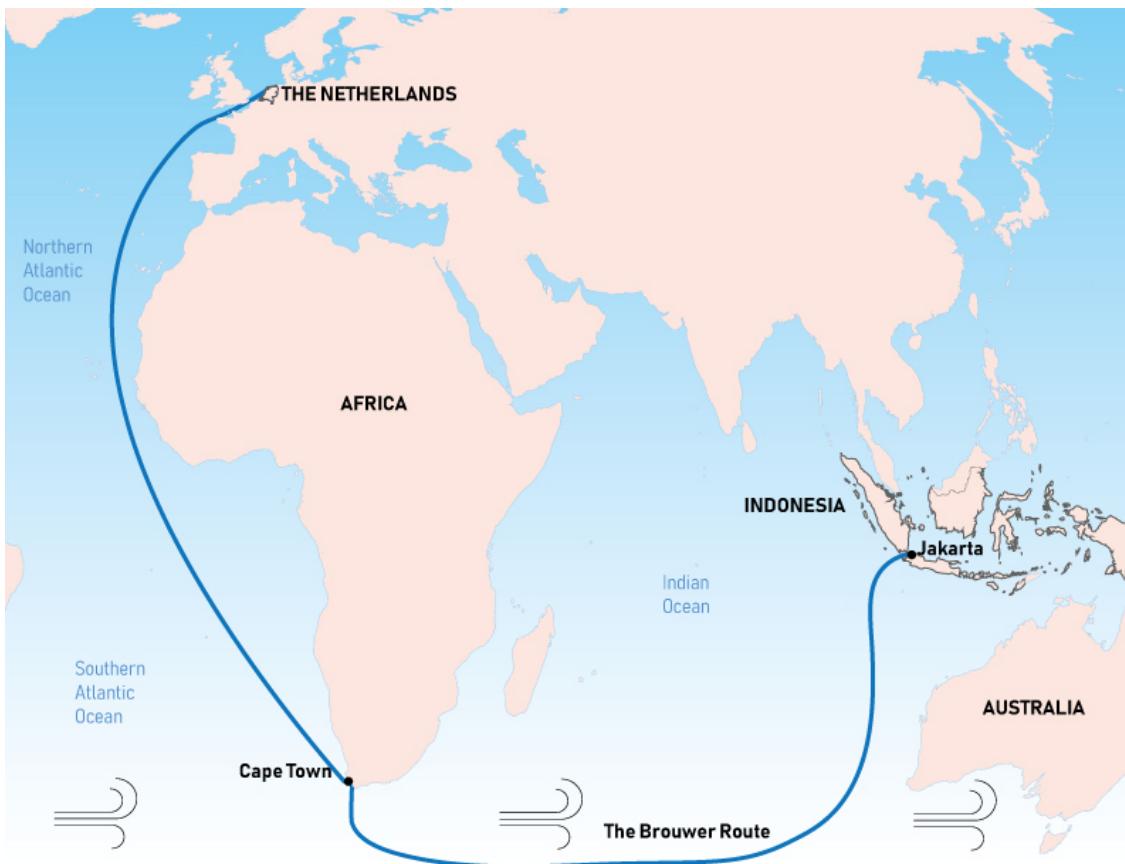
```

	participation	person	timespan	reasonend
0	participation/94248-1042325	person/1042325	appellation/1741-03-29_1743-07-20	endofexistence/Repatriated
1	participation/94327-1042325	person/1042325	appellation/1743-11-27_1745-08-24	endofexistence/Repatriated
2	participation/94412-1042325	person/1042325	appellation/1745-11-11_1746-06-10	endofexistence/Last_record
3	participation/94678-1087798	person/1087798	appellation/1754-03-06_1758-08-19	endofexistence/Deceased

There are four participations linked to the two records. The Hendrik Prins with identifier 1042325 starts his career in 1741 and continues enlisting three times with his last entry in 1746. The Hendrik Prins with identifier 1087798 starts his career in 1754, and it is a short-lived career as his contract ends due to him perishing on his first voyage.

There is a possibility that the first Hendrik Prins could be the same as the second. There is a gap of 8 years between the last enlisting and the enlisting of the second Hendrik Prins. This gap could exist due to missing records or other reasons. The reason for the ending of the first Hendrik Prins does not give a final answer. The last record category is because of reasons unknown.

The three participations of the first Hendrik Prins entry involved the same route. (screenshot) Like many others in that time, the route started from the Netherlands (Goeree) and sailed to Batavia (now Jakarta) via the Cape of Good hope. The precise route is unknown, but looking at the time-frame, there is a good chance the faster Brouwer route is used. The route is illustrated.



Data analysis We can compare the voyages of Hendrik Prins with other similar voyages which have the same starting and ending place. If the departing and arrival dates are known, we can conclude if the voyage of Hendrik Prins was, by comparison, quicker.

```
[35]: sparql.setQuery(prefixes +"""
SELECT ?voyage ?leg ?starttime ?endtime
WHERE {
?voyage pdfh:P1 <http://example.com/place/vocUniquePlaceID_1787>.
?voyage pdfh:P2 <http://example.com/place/vocUniquePlaceID_2357>.
?leg histdmiP:P5 ?voyage.
?start crm:P115 ?leg;
      crm:P4 ?starttime.
?end crm:P116 ?leg;
      crm:P4 ?endtime.
}
""")
```

```
[37]: firstlegs = []
secondlegs = []
for i in dfn.itertuples():
    if int(i[4][-1]) == 1:
```

```

        firstlegs.append({'starttime':date(*map(int, i[2].split('-'))), 'endtime':date(*map(int, i[1].split('-'))), 'voyage':i[3]})

    else:
        secondlegs.append({'starttime':date(*map(int, i[2].split('-'))), 'endtime':date(*map(int, i[1].split('-'))), 'voyage':i[3]})

for entry in firstlegs:
    delta = entry['starttime'] - entry['endtime']
    entry['delta'] = delta.days
fldf = pd.DataFrame(firstlegs)
print(fldf[fldf.delta > 0].describe())

for entry in secondlegs:
    delta = entry['starttime'] - entry['endtime']
    entry['delta'] = delta.days
sldf = pd.DataFrame(secondlegs)
print(sldf[sldf.delta > 0].describe())

```

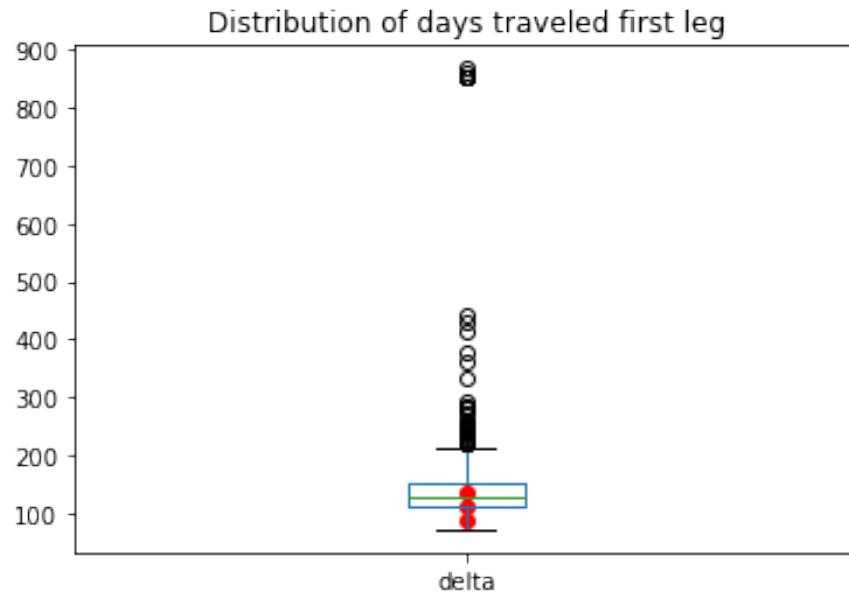
	delta
count	542.000000
mean	148.994465
std	102.697238
min	71.000000
25%	110.000000
50%	127.000000
75%	152.000000
max	868.000000
	delta
count	481.000000
mean	81.446985
std	22.556822
min	30.000000
25%	67.000000
50%	79.000000
75%	90.000000
max	287.000000

There are 542 first legs, from Goeree NL to the Cape of Good Hope SA registered. The mean delta (days traveled) is 148. There are exciting outliers, such as the max value of 868. Meaning a ship took around three years to complete the leg. From the Cape, 481 legs have registered that travel to Jakarta ID. The second leg is overall much shorter, with a mean of 81 days traveled. The outliers are much more confined as well, with a maximum value of 287.

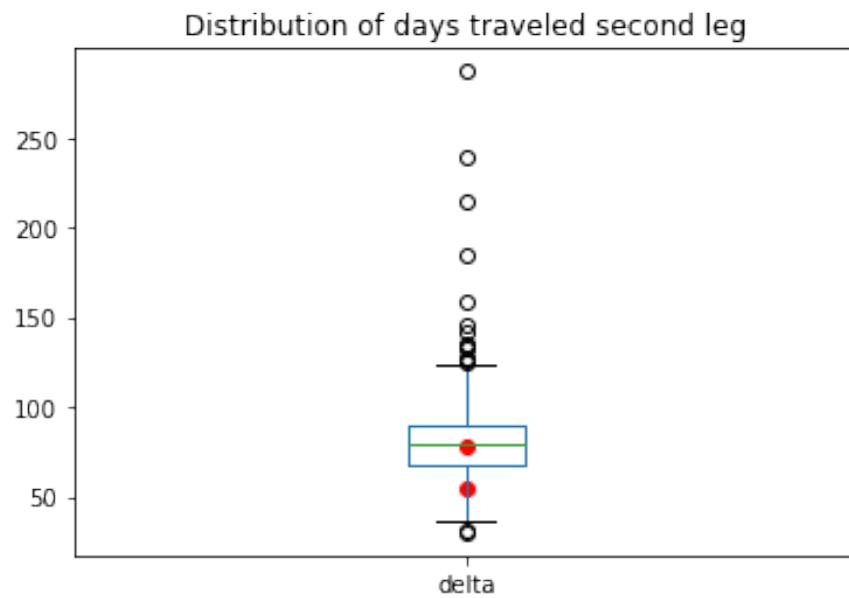
Hendrik Prins sailed three times on the voyages 94248, 94327, and 94412 in the outwards direction, from the Republic to the Dutch Indies. We can compare the delta (days traveled) with the other similar voyages in a boxplot. Note that the second leg misses an entry due to a missing arrival date (voyage 94327).

The voyages of Hendrik Prins were, overall, quicker than the mean.

```
[12]: plot = fldf[fldf.delta > 0].plot(kind='box', title="Distribution of days traveled first leg")
plot.scatter([1,1,1],[87,137,109], color='r')
```



```
[13]: plot = sldf[sldf.delta > 0].plot(kind='box', title= "Distribution of days traveled second leg")
plot.scatter([1,1],[55,78], color='r')
```



Qualitative analysis In the previous section, the artifact is used in data analysis. However, the artifact is also widely deployable for qualitative research. For example, the property P1:Voy Particulars illustrates the highlights of a voyage in a couple of sentences. We can retrieve the related particulars of the Hendrik Prins voyages to discover unusual or interesting facts.

```
[39]: sparql.setQuery(prefixes +"""
SELECT ?voyage ?particular
WHERE {
?participation histsocP:P10 <http://example.com/person/1042325>.
?participation histsocP:P11 ?voyage.
?voyage a cdfh:C1.
?voyage dfhc:P1 ?particular
}
""")
```

		particular	voyage
0	Pieter Jellesz.	was schout-by-nacht of the return fleet. The 117 men aboard the TOLSDUIN included 90 seafarers.	voyage/98050
1		After an accident on a voyage to Amboin the DELFLAND was laid up on 12-07-1746.	voyage/94248
2	The TOLSDUIN called at Duins (17-12-1743 till 06-01-1744) where 11 seafarers and 4 soldiers were pressed by the English. During the voyage one soldier became seafarer.		voyage/94327
3		At the Cape a stowaway, found on the TOLSDUIN, embarked on the EENDRACHT, which set out to Batavia on 22-03-1745 (3309).	voyage/98101

Hendrik Prins sailed under the command on one of the schout-by-nacht' of the Replubic: Pieter Jelleszoon. Furthermore, they were pressed by the English at one point; it must have been thrilling. Finally, the crew of Prins found a stowaway when docked at the cape. Overall, an interesting life on board a VOC ship.

The Wind in our sails project does not only cover the moments of ships and their participants. It also features a connection with the bookkeeper's logs. Information about cargo shipped on the voyages is stored. With a query, we can retrieve the goods that were shipped on the voyages of Hendrik Prins.

```
[41]: sparql.setQuery(prefixes +"""
SELECT DISTINCT ?voyage ?typename ?direction
WHERE {
?participation histsocP:P10 <http://example.com/person/1042325>.
?participation histsocP:P11 ?voyage.
OPTIONAL{?voyage dfhc:P2 ?direction}
?voyage a cdfh:C1.
?voyage dfhc:P10 ?good.
?good crm:P2 ?type.
?type crm:P1 ?typename.
}
""")
```

typename	voyage	direction
nootmuskaat	98050	Return
indigo	98050	Return
kleed	98050	Return
gingam	98050	Return
baftas	98050	Return
salempuris	98050	Return
muris	98050	Return
guinees	98050	Return
bethilles	98050	Return
ongelden	98050	Return
garen	98050	Return
ebbenhout	98050	Return
terindani	98050	Return
bherms	98050	Return
carrots	98050	Return
niquanias	98050	Return
zijde	98050	Return
peper	98050	Return
thee	98050	Return
chauter	98050	Return
koffieboon	98050	Return
borax	98050	Return
cauri	98050	Return
salpeter	98050	Return
garioffelnagel	98050	Return
foelie	98050	Return
photos	98050	Return
benzo	98050	Return
tansjeebs	98050	Return
sapanhout	98050	Return
gomlak	98050	Return
kurkuma	98050	Return
zeildoek	98050	Return
malmal	98050	Return
neusdoek	98050	Return
sits	98050	Return
douriassen	98050	Return
dongris	98050	Return
amirtje	98050	Return
armozijn	98050	Return
scheepsklederen	98050	Return
boek	98050	Return
samen	98050	Return
schrijfgereedschap	98050	Return
tinwerk	98050	Return

linnen	98050	Return
boter	98050	Return
tafelmes	98050	Return
jucht	98050	Return
salmiak	98050	Return
aluin	98050	Return
verf	98050	Return
ijzer	94248	Outbound
spijker	94248	Outbound
kleinigheden	94248	Outbound
gereedschap	94248	Outbound
samen	94248	Outbound
over_te_veel_berekend_in_de_factuur	94248	Outbound
provisie	94248	Outbound
platlood	94248	Outbound
ambachtsgereedschap	94248	Outbound
fluweel	94248	Outbound
equipagegoederen	94248	Outbound
wapenkamergoederen	94248	Outbound
artilleriegoederen	94248	Outbound
zilver	94248	Outbound
dukatons	94248	Outbound
scheepskleederen	94248	Outbound
tarwe	94248	Outbound

In the table above, the goods shipped on the voyages of Hendrik Prins are listed. In the outward voyages, goods for trading were brought along, such as velvet or silver. On the return voyage, the goods shipped were more diverse. From nutmeg to salmiak, the spices and goods flowed rich.

Career and conclusion There is no guarantee that the second Hendrik Prins from Delft is the same person as the first one. Overall, between the third and fourth contract, there is a gap of over eight years. We might check two things if we want to determine if the two entries are the same person. First, we can check if the debt letters of the entries refer to the same person. Second, we might check the career path of the participants to determine if there is a logical connection.

```
[43]: sparql.setQuery(prefixes +"""
SELECT DISTINCT ?voyage ?typename ?direction
WHERE {
?person histsocP:P24 <http://example.com/place/vocUniquePlaceID_1192>.
<http://example.com/place/vocUniquePlaceID_1192> crm:P1 ?from.
?person dfhc:P8 ?name.
?relation histsocP:P17 ?person.
filter contains(?name,"Hendrik Prins")
}
""")
```

Empty DataFrame

Columns: []

Index: []

Unfortunately, both entries do not have a debt letter relation. Thus that trail ends here. The other trail, one of the careers, we can still follow. In the query below, the rank performed during the voyages is displayed.

```
[46]: sparql.setQuery(prefixes +"""
SELECT ?person ?rank
WHERE {
?person histsocP:P24 <http://example.com/place/vocUniquePlaceID_1192>.
<http://example.com/place/vocUniquePlaceID_1192> crm:P1 ?from.
?person dfhc:P8 ?name.
?participation histsocP:P10 ?person.
?participation histsocP:P12 ?rank.
filter contains(?name,"Hendrik Prins")
}
""")
```

	person	rank
0	person/1042325	rank/sailor
1	person/1042325	rank/ship_gunner
2	person/1087798	rank/senior_weapons_master

The rank trail does yield some knowledge. Hendrik Prins started as a sailor, the base rank on a ship. He then moved to a more experienced role, the ship gunner. The ship gunner is still a

sailor but also bears the responsibility of firing the canon in times of turmoil. The second entry of Hendrik Prins bears the role of senior weapons master (or Opperkonstabel in Dutch.) The weapons master is responsible for the weapons on board the ship. A rank that is not easily reached without experience. Thus, there is a good chance, due to missing records, Hendrik Prins had a career from sailor to the senior weapons master in his later years.

The life of Hendrik Prins came to an end during voyage 94678. There is no reason stated, but the life on board of a ship could be harsh. During this particular voyage, seven other seafarers, six soldiers, and one craftsman perished. The table can be viewed below.

Part of the voyage	Measurement	Number	Unit
94678	Passengers-94678	0	count_dying
94678	Seafarers-94678	8	count_dying
94678	Soldiers-94678	6	count_dying
94678	Craftsmen-94678	1	count_dying