AN INVESTIGATION INTO APPLYING ONTOLOGIES TO THE UK RAILWAY INDUSTRY

by

Jingfu Wei

A thesis submitted to the University of Birmingham for the degree of

DOCTOR OF PHILOSOPHY

Birmingham Centre for Railway Research and Education

School of Engineering

University of Birmingham

May 2021

ABSTRACT

The uptake of ontologies in the Semantic Web and Linked Data has proven their excellence in managing mass data. Referring to the movements of Linked Data, ontologies are applied to large complex systems to facilitate better data management. Some industries, e.g., oil and gas, have attempted to use ontologies to manage its internal data structure and management. Researchers have dedicated to designing ontologies for the rail system, and they have discussed the potential benefits thereof. However, despite successful establishment in some industries and effort made from some research, plus the interest from major UK rail operation participants, there has not been evidence showing that rail ontologies are applied to the UK rail system.

This thesis will analyse factors that hinder the application of rail ontologies to the UK rail system. Based on concluded factors, the rest of the thesis will present corresponding solutions. The demonstrations show how ontologies can fit in a particular task with improvements, aiming to provide inspiration and insights for the future research into the application of ontology-based system in the UK rail system.

ACKNOWLEDGEMENTS

This thesis presents four years of work at the Birmingham Centre for Railway Research and Education (BCRRE), which cannot be completed without the support from the following people:

- Dr John Easton, for his excellent guidance and supervision with high level of expertise and patience, without whom, I could not have completed this thesis
- Dr Lei Chen and Prof Clive Roberts, for their expertise and valuable ideas to improve my work
- My family members, particularly my parents and my wife, for their unwavering love, support and encouragement throughout my life, teaching me the right from wrong
- My Chinese friends at BCRRE, for letting me have the sense of belonging in a foreign country
- Jun Hoe Chan, Jonathan Shi, Kieran Saunders, Avigail Gazit, Char Lena, Charlotte Green, Katherine Ellis, Liban Hannan, Olivia Renshaw and Ibrahim FitzGibbon, for being great friends who gave me their company and useful advice when I encounter difficulties, helping me to adapt to the life in the UK
- Marcus Young and other collaborators from the University of Southampton, for their collaboration and inspiration to help me completing Chapter 7

- My other PhD colleagues, and staff members at the BCRRE and the UOB, for helping me resolving problems during my study at the University of Birmingham

I am deeply grateful to aforementioned people for their invaluable support and encouragement throughout my study and my life. I also highly appreciate the funding provided by the School of Engineering to cover the tuition fee at the University of Birmingham. I would also love to extend my thanks to Anne O'Connell who edited this thesis for conventions of language, spelling and grammar.

Table of Contents

List of Figures

List of Tables

ABBREVIATIONS

AI – Artificial Intelligence

API – Application Programming Interface

BBC – British Broadcasting Company

C4R – Capacity for Rail

CIS – Customer Information System

CSV – Comma-Separation Values

D2R – Database to Relational

DfT – Department for Transport

DSS – Decision Support System

EPSRC – the Engineering and Physical Sciences Research Council

ESB – Enterprise Service Bus

FA –Factor Analysis

GPS – Global Positioning System

HTML – HyperText Markup Language

HTTP – HyperText Transfer Protocol

ICT – Information and Communications Technology

IDC – International Data Corporation

IEEE – Institute of Electrical and Electronics Engineers

IL – Integration Layer

IOHN – Integrated Operations in the High North

IRS – International Railway Standard

ISBM – Information Service Bus

ISO – International Standard Organization

IT – Information Technology

KMO – Kaiser–Meyer–Olkin

NaPTAN – National Public Transport Access Nodes

NLP – Natural Language Processing

NRE – National Rail Enquiries

OBDA – Ontology-Based Data Access

OGC – Open Geospatial Consortium

ORBIS – Offering Rail Better Information Services

ORR – Office of Rail and Road (formerly Office of Rail Regulation)

OWL – Web Ontology Language

PCA – Principal Components Analysis

RaCoOn – Railway Core Ontology

RAIB – Rail Accident Investigation Branch

railML – Railway Markup Language

RAM – Random-Access Memory

RDF – Resource Description Framework

RDG – Rail Delivery Group

RDO – Railway Domain Ontology

RSSB – Railway Safety and Standards Board

SHACL – SHApes Constraint Language

SMIS – Safety Management Intelligence System

SPARQL – SPARQL Protocol and RDF Query Language

SPSS – Statistical Product and Service Solutions (originally stood for Statistical Package for the Social Sciences)

SVM – Support Vector Machine

SWRL – Semantic Web Rule Language

T2F – Track to the Future

TAM – the Technology Acceptance Model

TfGM – Transport for Greater Manchester

TF-IDF – Term Frequency – Inverse Document Frequency

TfL – Transport for London

TIPLOC – Timing Point Locations

TMS – Traffic Management System

TOC – Train Operating Company

TOPS – Total Operation Processing System

TRUST – Train Running Under System TOPS

TSI – Rail Technical Specification Interoperability

UAT – User Acceptance Testing

URI – Uniform Resource Identifier

URL – Uniform Resource Locator

USP – Under Sleeper Pad

XML – eXtensible Markup Language

XMLS – eXtensible Markup Language Schema

W3C – World Wide Web Consortium

WKT – Well-Known Text

WWW – World Wide Web

1 INTRODUCTION

1.1 OVERVIEW

Large complex systems exist in many industries nowadays. As the name suggests, they are complex, consisting of multiple sub-systems and databases. Meanwhile, stakeholders tend to create their own data silos within such a system, which is unavoidable as a result of diverse IT infrastructure suppliers. Due to the nature of data silos, data retrieval and consumption have become increasingly difficult. Developers and data scientists often have to meet the challenge of efficiently querying and retrieving various heterogeneous, disparate and diverse data irrespective of their source and structure. To address the problem of disparate data sources, many organisations have invested in data integration to enhance system performance and facilitate information sharing.

Additionally, many organisations have also embraced the Big Data era, craving more value from existing data (Dong and Srivastava, 2013). The vast amount of data creates further difficulty when the data needs to be extracted and analysed. Thanks to modern relational databases, data can be stored in a highly structured manner, easing the data retrieval process. However, because of the sheer volume of Big Data, unstructured data and databases inevitably exist. Linked Data, however, can benefit analysis of Big Data

by improving data retrieval, and researchers have investigated Linked Data techniques to address this problem.

This thesis will provide insights into current Linked Data applications in large complex systems, using the UK railway system as a case study to discuss Linked Data and the challenges thereof, analysing the further development and adoption of Linked Data technologies.

1.2 The Need to Improve integration

Data silos are repositories of data that are isolated from the rest of an organisation; they tend to exist in large organisations and have an impact on the productivity and performance of the organisation (Fredsall, 2015). An entrepreneur pointed out that 'silos defeat collaboration and stymie value creation' (Scott, 2018); the collaboration between different units within an organisation can be affected by the confusion and inefficiency brought by data silos. Data and information retrieval is strained by data isolation, despite the effort to develop more advanced data storage techniques (Gardner, 2005). Although it has been admitted that demolishing data silos can improve the efficiency, accessibility and performance of information systems (Fredsall, 2015), the integration level has only been slightly improved (Scott, 2018; Speiser and Harth, 2010).

1.3 Data Heterogeneity in the UK Railway System

Efficient railway operation is linked to the efficient exchange of data. As a result of privatisation in the 1990s, the UK railway system has been fragmented. This fragmentation creates a complicated structure as illustrated in Fig. 1; that is to say, there are many stakeholders and participants responsible for different aspects of railway operation, such as regulation, operation, maintenance, IT and ICT systems, etc. As a result of separating responsibilities across multiple parties, data silos have been created by an increasing number of privately owned IT systems across the domain (Tutcher, 2015b). Although there are more than 130 national information systems that are supported by over 20 suppliers in the UK railway system, little effort has been made to enhance the exchange and sharing of data and knowledge (Brewer, 2011). This has created both technological and operational barriers that obstruct effective railway operation, and has made data sharing and exchange between different industry stakeholders complex, especially while the railway industry shifts towards data-driven decision-making processes (Wei, 2018).

Most data is proprietary and difficult to be accessed for future usage within an organisation although it is supposed to be easily accessible (Köpf, 2010); meanwhile, proprietary formats lead to difficulty in access and analysis by other parties, inevitably increasing the cost of performing data analysis (Umiliacchi and Henning, 2008). Consequently, data silos and disparate formats lead to data heterogeneity, which potentially makes useful data inaccessible to other systems.

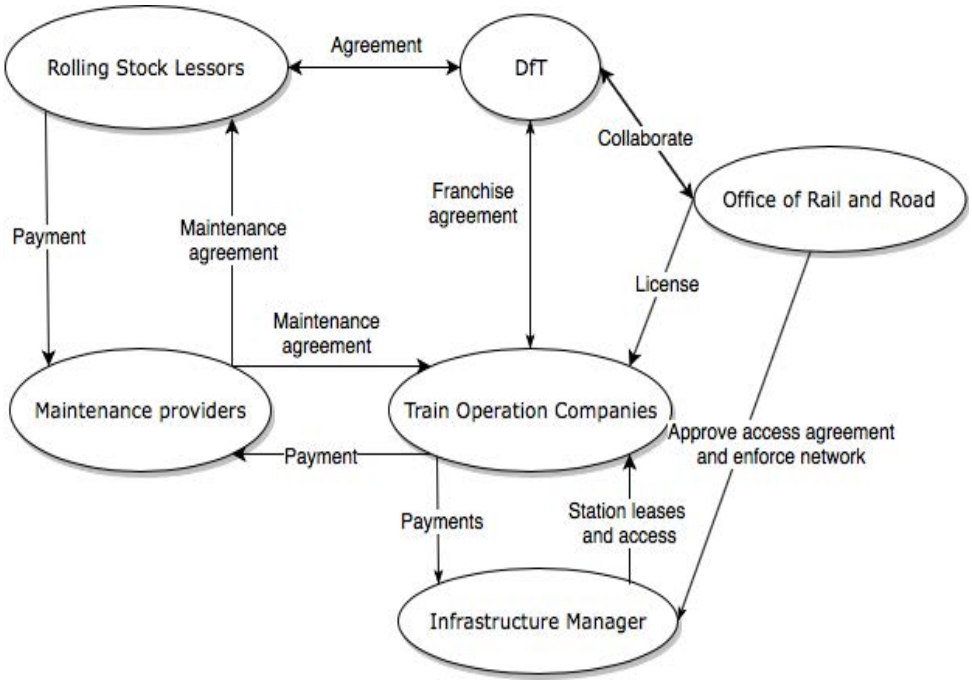


FIG. 1 HIGH-LEVEL STRUCTURE OF THE UK RAILWAY SYSTEM (COMPETITION COMMISSION, 2007)

An example is that to modify a railway asset, many information sub-systems within a system must be modified, especially when many systems have been purchased and maintained in isolation; Fig. 2 reveals the complexity of the operating model. It is worth mentioning that processes and staff are not

included in Fig. 2 because of complex variation; as a result, there is a strong need to improve interoperability and integration to improve the operation model (Durk, 2013). Meanwhile, many rail assets are safety-critical so that extreme care is required to ensure everything is right. On top of that, data heterogeneities must be dealt with too. Extra resources must be supplied. It is reasonable to envisage that if systems were integrated, such a task would become much easier and faster. An integrated system can create more opportunities to improve the railway operation by establishing a more efficient data transfer environment to help data analysis, modification and storage.

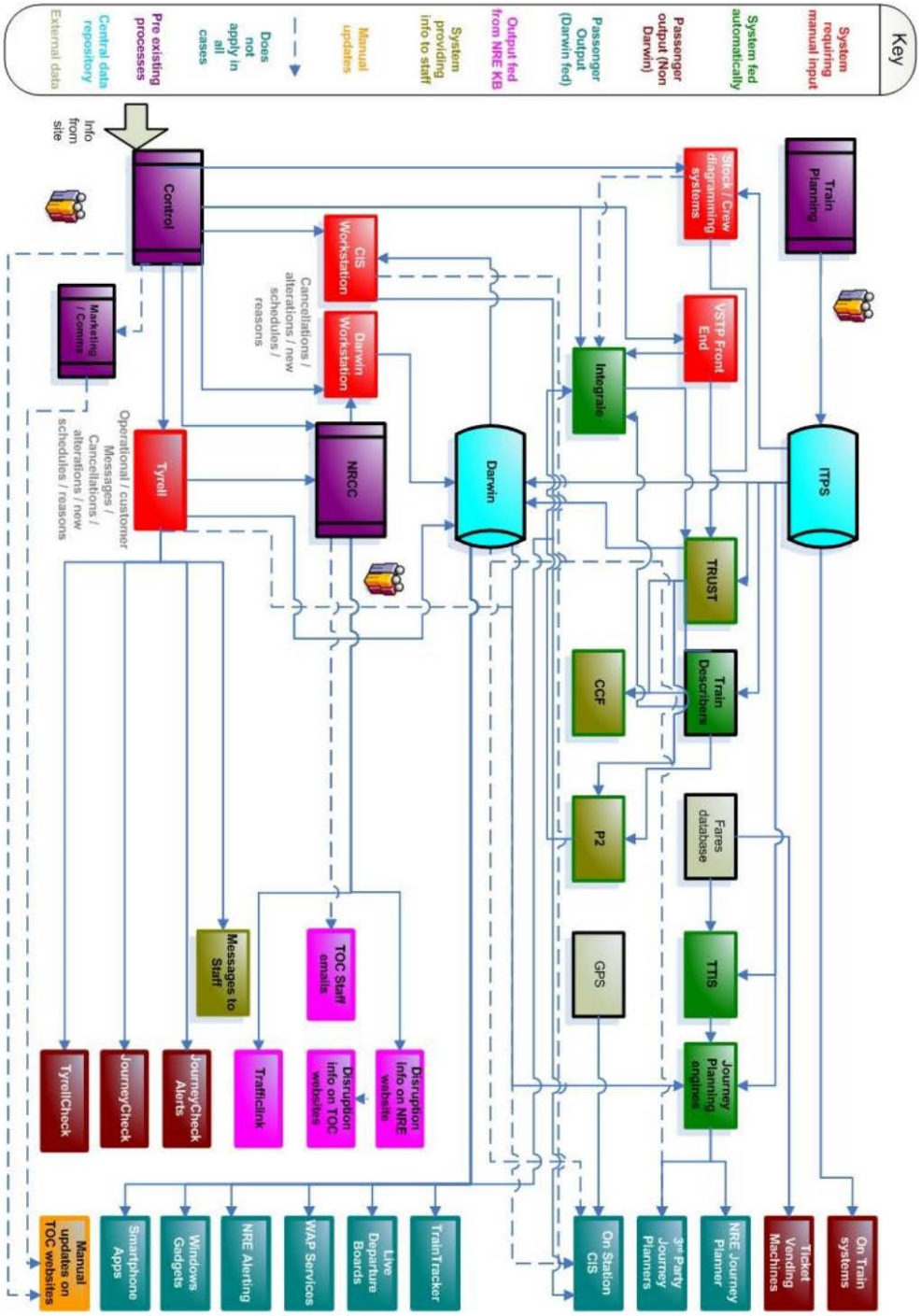


FIG. 2 ILLUSTRATION OF UK RAILWAY OPERATING MODEL (DURK, 2013)

Another example is that it is difficult to incorporate condition monitoring data into legacy systems such as the Traffic Management System (TMS) as it is often the case that data was generated context-independently and

distributively. It takes many resources for developers and researchers to understand where the data comes from and what it means. The lack of standardised models has created barriers to data exchange, leading to unnecessarily increased cost and effort in the development of railway operation in the UK.

1.4 Efforts Already Made

Large organisations and various industries should notice the potential of Linked Data technologies such as ontologies (Ebrahimipour and Yacout, 2015). Presently, systems are integrated on a system-to-system basis (Morris, 2017). However, the increasing need for data integration has been addressed by the UK government, which published the Rail Technical Specification Interoperability (TSI) to facilitate harmonious and smooth data interchange while ensuring high-level data interoperability (Department for Transport, 2011a). In addition, a vision of the future, in which an ontology-based framework would be beneficial to data exchange within the industry, was presented in the 2012 Rail Technical Strategy (TSLG, 2012), while Network Rail, as the major railway infrastructure manager in the UK, has also addressed its interest in the potential benefits brought by the development of generalised and standardised information architectures using ontologies to facilitate data integration, reuse and sharing (Network Rail, 2013). A recent government policy paper has also stated that the system operator should aid the railway industry, government and other funders in making

better decisions by increasing the quality and transparency of information and analysis (Department for Transport, 2017). Some researchers have developed ontologies in order to describe abstract concepts and business processes (Köpf, 2010; Morris et al., 2014; Roberts et al., 2011; Tutcher, 2015b, 2017; Umiliacchi and Henning, 2008).

The research on ontology-based applications has also gained a lot of attention because of the advantages that ontologies possess (Bizer et al., 2009; Choi, 2014; Compton et al., 2012; Ebrahimipour and Yacout, 2015; Lewis et al., 2006; Tutcher et al., 2017); it has proved that ontology-based data integration could improve current railway operation systems in the UK. Much research has been devoted to modelling railway concepts and knowledge (Köpf, 2010; Lu et al., 2006; Mohan and Arumugam, 2005; Tutcher et al., 2017; Umiliacchi and Henning, 2008), with regard to aspects such as integrated optimisation of track usage, condition monitoring data analysis, better route utilisation, better operation planning, better decision-making processes and faster customer information (Lu et al., 2006; Morris et al., 2014; Roberts et al., 2011).

However, despite this interest, there has been little work carried out on implementing a system. The lack of demonstration and understanding how ontology can help existing railway operation has discouraged the industry from taking further steps to adopt an ontology-based data framework to facilitate the future development of ICT infrastructure, hence this thesis.

1.5 Research Questions and Thesis Structure

As per the brief introduction in the previous section, the following research questions are proposed, and their corresponding introduction have been tabulated in Table 1.

TABLE 1 RESEARCH QUESTIONS AND THEIR CORRESPONDING CHAPTERS

Research question	*Brief introduction*	*Chapter*
What work has been completed to demonstrate the usefulness of ontologies in large complex systems overall?	It is necessary to understand how ontologies fit in complex systems and what they can bring to industries where large complex systems exist, including the railway industry.	Chapter 2
Given the fact that both the rail industry and research community are interested in ontology-based applications, why is there no sign that an ontology-based system has been implemented within the rail industry with an appropriate system architecture?	Ontologies have gained much attention. However, no evidence shows there is a system using ontologies established in the industry. The proposed question needs further investigation and discussion.	Chapter 4
In spite of research into ontology-based applications, little has been done to demonstrate their scalability, especially with high-velocity data. Therefore, can we	Based on the conclusions drawn from Chapter 4, it is not clear whether an ontology-based data processing system is capable of handling data at industry level in terms of volume and velocity. In order to	Chapter 5

understand to what extent an ontology-based data processing system can perform with industry-level data in the UK railway industry?	popularise ontologies in the UK rail industry, it is necessary to reveal the scalability and performance of ontologies.	
Given the fact that ontologies can integrate data, how can we use ontologies to manage unstructured data in the railway industry?	Another conclusion drawn from Chapter 4 is that many professional developers are not aware of the practical usage of ontologies, while some of them reckoned that relational models could achieve similar effects. Therefore, it is necessary to demonstrate the benefits of ontologies while managing unstructured data in comparison to relational models.	Chapter 6
Many ontology models can only be manipulated by relevant professionals; how can we enable those who are not familiar with ontologies to use them?	The investigation elaborated in Chapter 4 also revealed that many complained about the lack of supporting tools for using ontologies. Some also suggested that learning how to use them from scratch takes time, which might not be necessary. There is a strong need for developing tools that allow non-professionals to use ontologies.	Chapter 7
How can we reproduce some manual processes using ontologies to achieve more digitalised and more effective processes in the railway industry?	Some responders pointed out that using ontologies could be appealing if it could improve existing processes in the rail industry. Yet, there has been no demonstration of it. Replacing existing manual processes with the usage of	Chapter 8

ontologies still awaits discussion.

The rest of this document will aim to answer the proposed problem. Chapter 2 will present a comprehensive review on the state-of-art literature and latest situation of data policies of UK railway industry. An investigation of the factors that discourage the professionals working in the industry from using ontologies will be elaborated in Chapter 4. The following chapters will present solutions accordingly for each identified factor obtained in Chapter 4.

1.6 CONTRIBUTIONS

This thesis has made the following contributions to answer the proposed research questions:

1. Identify the factors that might have discouraged the adoption of ontology-based solutions for the UK railway industry by a survey which was also supported by a thorough review of current literature.
2. Identify example solutions for each identified factor based on the literature to provide demonstrations for future practice for future railway-related ontology-based application development, bridging the gap between theoretical ontology-based solutions and practical applications for the UK railway industry.

Following papers were published:

- Wei J. Scalability of an Ontology-Based Data Processing System. InIET Conference Proceedings 2018 May 16. The Institution of Engineering & Technology.

This paper discussed and demonstrated to which extent ontologies can perform in a practical railway business environment with reference to industrial level of data volume.

- Armstrong J, Rempelos G, Wei J, Preston J, Blainey S, Easton J, Roberts C. Developing a generalised assessment framework for railway interventions. InComputers in Railways XVII: Proceedings of the 17th International Conference on Railway Engineering Design and Operation (COMPRAIL 2020) 2020 Sep 7 (pp. 127-138). WIT Press.

The joint effort to establish an ontology-based intervention assessment framework was elaborated in this paper. It proved the worthiness of ontologies in the field of data management, and improvements of manual processes in data preparation by applying a unified data schema with semantics using an ontology-based data description framework.

2 LITERATURE REVIEW

2.1 LINKED DATA AND THE SEMANTIC WEB

2.1.1 LINKED DATA

Berners-Lee developed the first version of the World Wide Web[1] (WWW) in 1989 to solve a problem of information management where different information was stored on different computers (World Wide Web Foundation, 2008). Before the advent of WWW, users had to log onto different computers to retrieve information and had to learn how to interact with different programmes on each computer; therefore, Berners-Lee proposed three technologies which have laid the foundation of present Web: HyperText Markup Language (HTML), Uniform Resource Identifier (URI) and HyperText Transfer Protocol (HTTP). HTML, URI (commonly named Uniform Resource Locator, URL) and HTTP were designed to allow the display, location and transmission of information on the Web, respectively (World Wide Web Foundation, 2008). The isolation of data led to poor enterprise performance; the advent of WWW made information available and greatly improved connectivity, but it still had some drawbacks.

As its name suggests, HTML is a markup language that is oriented to structure textual documents instead of data (Heath and Bizer, 2011); it makes machines display pre-defined content, as illustrated in Fig. 3. This limitation

[1] Commonly known as the Web.

means that only humans can ‘understand’ the content, while machines only ‘display’ the content as instructed. Despite embedded data in HTML documents and structured data being made available by Web APIs (Application Programming Interfaces), only a small amount of entities can be represented in a restricted way (e.g. using HTML tags to describe tags); relationships between entities and attributes thereof are difficult to model and express (Berners-Lee, 2006; Heath and Bizer, 2011).

Admittedly, HTML documents are linked by hyperlinks, but hyperlinks help users to navigate instead of extracting more information from data. Because of the scatter of data and unlinked entities, when humans need to retrieve or access data on the Web, search engines can only search based on given keywords. Keyword-based searching can return false results as a result of the ambiguity derived from the nature and complexity of human language. To address this problem, the concept of Linked Data emerged to enable machines to understand the relationship between pieces of data, and the design of the Resource Description Framework (RDF) was developed to support Linked Data (Berners-Lee, 2006; Bizer et al., 2008, 2009; W3C, n.d.). An example is shown in Fig. 4; it is impossible to discover the relationship between entities – Birmingham, the University of Birmingham and University in HTML – but with Linked Data, it is possible to describe the relationships between them.

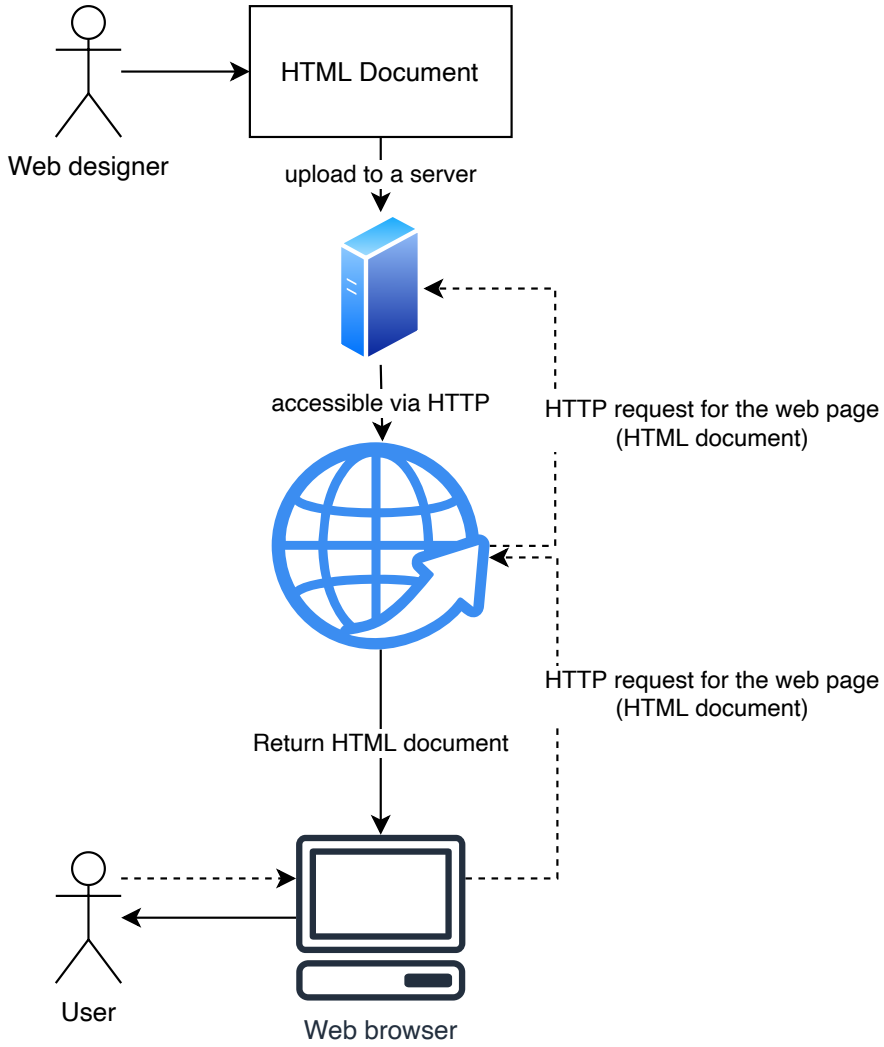


FIG. 3 ARCHITECTURE OF THE WEB

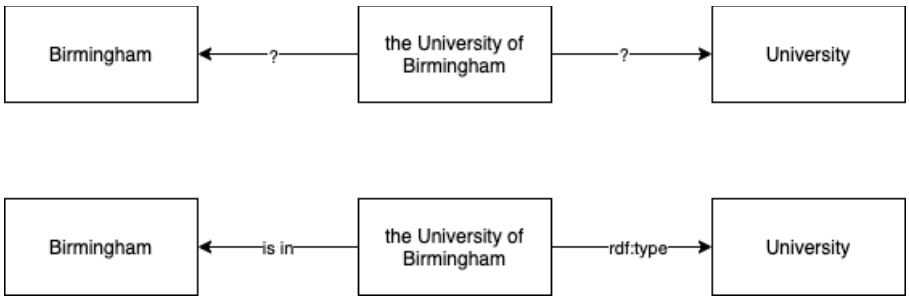


FIG. 4 LIMITATIONS OF REPRESENTING RELATIONSHIPS BETWEEN ENTITIES

To facilitate Linked Data, Berners-Lee (2006) outlined several 'rules' for data publishing to establish a global data space, including:

1. *Use URIs as names for things*
2. *Use HTTP URIs so that people can look up those names*
3. *When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL)*
4. *Include links to other URIs, so that they can discover more things*

He also elaborated on his opinion of Linked Data at the 2009 TED Conference, that Linked Data is '*an extremely simple technology but it allows everyone to put everything on the Web that interconnects, facilitating better data and knowledge reusing and sharing, and enhancing data availability and accessibility so that scientists are not stymied*' (Berners-Lee, 2009). Bizer et al. (2009) concluded that the aim of Linked Data is to create links between data from different sources regardless of the origin or geographical location; data should be published so that it is machine-readable and open to linkage with other data sets both from within and externally. Soon afterwards, Heath and Bizer (2011) stated that the rationale for Linked Data is the necessity of sharing and reusing data on the Web, facilitated by the use of hyperlinks and well-structured data, respectively, transforming data islands into a global data space; they emphasised the importance of using URIs to name things and making URIs dereferenceable so that they do not only identify classic

HTML documents but also discover and retrieve a description of the resource identified by the URI.

To link URIs and construct sophisticated models on the Web, RDF was designed to model human knowledge and represent entities (Gibbins and Shadbolt, 2011; RDF Working Group, 2014). Unlike traditional HTML documents on the Web, Linked Data is formed by structured data in RDF and HTTP, allowing navigation between different data sources by following RDF links (Bizer et al., 2008). According to returned RDF descriptions that link to other RDF URIs, the agent could consequently discover new resources and keep tracing new URIs in the same or different namespaces (Heath and Bizer, 2011). For example, when the word 'London' is mentioned, we know it is the capital city of the UK, but from a computer's perspective it is simply a plain string. With Linked Data, a computer can dereference the RDF link to achieve the same result so that it can perform further operations based on the linkage, such as retrieving the weather forecast for the capital of the UK upon a user's query. It enhances machines' intelligence to enable them to capture a real-world concept in a machine-readable method, and the interconnection between these concepts can guide the intelligent agent to locate the 'correct' thing. Thus, Linked Data can form a 'Web of data'; in other words, data is connected according to its relationships and properties, which has laid a solid foundation for the Semantic Web (W3C, n.d.).

2.1.2 THE SEMANTIC WEB

Although the idea of the Semantic Web was proposed years before that of Linked Data, the Semantic Web became possible when the concept and principles of Linked Data were established at a later date. The Semantic Web has been described as '*a new form of Web content that is meaningful to computers will unleash a revolution of new possibilities*' (Berners-Lee et al., 2001), which enables computers to understand the meaning of a given piece of data so that they can perform adaptive operations based on our semantics and underlie a more meaningful Web, forming the next-generation Web (Berners-Lee, 2009; Berners-Lee et al., 2001; Che, 2006; Gruber, 2009; Shadbolt et al., 2006; World Wide Web Consortium, 2012). Linked Data forms an important part of the Semantic Web.

Berners-Lee also depicted the Semantic Web as '*a component of Web 3.0*' (Shannon, 2006). Speaking of the Web 3.0, it is worth mentioning the previous generations, Web 1.0 and Web 2.0. When the WWW was initially invented, it was intended to publish content; in other words, the majority of users acted as content consumers on the Web 1.0 (Cormode and Krishnamurthy, 2008). In comparison to the Web 1.0, the principle of Web 2.0 was 'viewing the Web as a platform' (O'Reilly, 2006), which represents the ability to connect users and for participants to interact in a less restrictive way (Cormode and Krishnamurthy, 2008). The transition from Web 1.0 to Web 2.0 can be concluded as a change of the role that users of the Web played, from pure viewers to participants who can make changes to the

content and interact with the other participants. The popularisation of the Web 2.0 impressed the crowd with an interactive and social online environment, and thanks to the open platform, people can share their thoughts and knowledge, hence the emergence of blogs and online encyclopaedias. It can be seen that the Web 1.0 made information available from one user to other users whereas Web 2.0 has facilitated greater connection and interaction between users, aiming to provide better services to participants. However, despite a more connective and social Web, a problem remains to be addressed: data and information captured in online documents are only understood by humans, predominantly in HTML. When data and information have been published, machines can only perform certain tasks according to humans' instructions in HTML. Machines can do little when the information captured in HTML documents needs to be extracted without human intervention; for example, if a developer wants to request some data from an API, they must find out the correct API to launch a request to. Researchers have addressed this problem and proposed 'the Web 3.0' (Berners-Lee et al., 2001; Hendler, 2009; Nath et al., 2014).

Despite some divergence, the majority have agreed that the Web 3.0 is '*the Web 2.0 with Semantic Web technologies integrated into, or powering, large-scale Web applications*' (Hendler, 2009). The Semantic Web, as a component of the Web 3.0, provides technological infrastructure to link data from multiple websites and databases unambiguously and explicitly. The adaptation of URI for data in RDF has made it possible to trawl documents and

databases until the desired information is located. Therefore, using the same example as above, the story can be rewritten as follows:

1. Some developers need to obtain real-time train timetable data, and they might need to access the right API.
2. They make a query with a software agent for train timetable data.
3. The agent locates several qualified APIs for the developers based on the APIs' supplementary semantic description.
4. They make the choice, and the agent returns detailed information for the chosen API.

Admittedly, these developers might still have to visit a HTML document in order to complete the API registration, etc. However, a higher level of automation will deduct unnecessary effort to filter the desired content, thanks to a machine-readable and understandable model. A graphic comparison of the three generations of the Web is illustrated in Fig. 5.

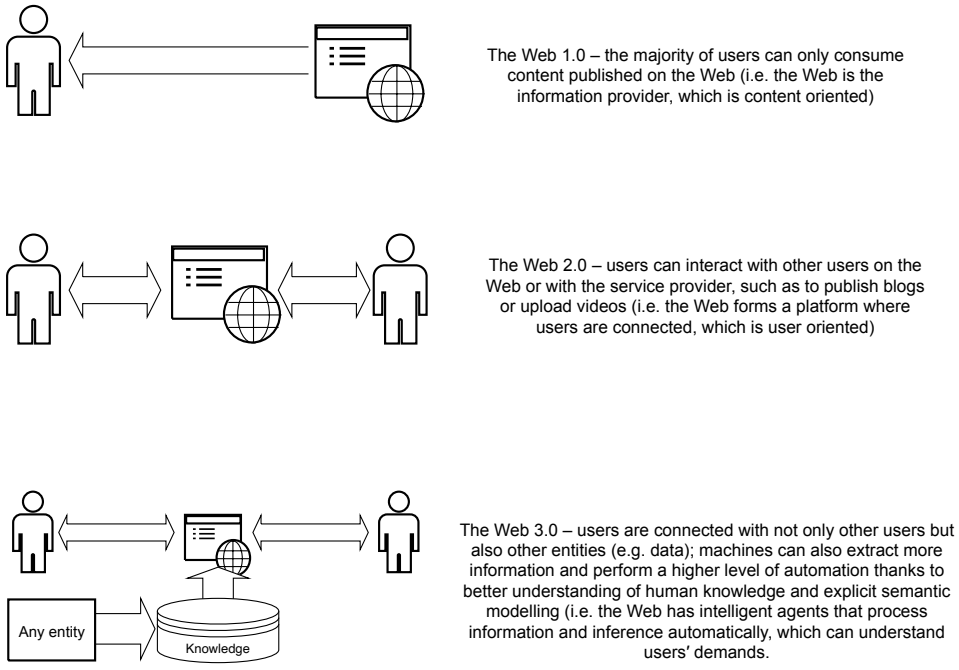


FIG. 5 COMPARISON OF THE WEB 1.0, 2.0 AND 3.0

The Semantic Web is the augmentation of the World Wide Web Consortium (W3C)'s 'Web of Documents', that provides the framework to distinguish ambiguity and hierarchy in heterogeneous data, and methods to describe relationships between entities, allowing machines to comprehend the data and properties thereof. The Semantic Web provides a solid foundation for data integration in a way in which data can be linked contextually to realise and facilitate better performance and efficiency. The idea of Linked Data inspires the transformation of the Web of Documents into the Web of Data, which justifies the goal of the Semantic Web – '*from documents to data and information*' (Shadbolt et al., 2006), facilitating the linkage between concepts. Having machine-readable data on the Web decreases the restriction on the ability of computers to process information. The Semantic Web

provides a way of modelling and representing data on the Web and, by doing so, machines extract not only metadata but also the meaning behind it. Therefore, intelligent agents can handle more complex queries if necessary. A successful commercial application using concepts from the Semantic Web has been realised by Google, which has developed ways to search with a focus on the connections between concepts and entities (Ehrlinger and Wöß, 2016; Singhal, 2012).

For example, supposing the question 'Where is the University of Birmingham?' is asked. If the machine handles this query by searching keywords, it is likely that the user will be navigated to the official website of the University of Birmingham by hyperlinks, but due to a lack of understanding of the information captured from the query, the machine cannot return the result straight away. However, if the information about the address has been represented on the Semantic Web so that the machine has been made aware of the address of the University of Birmingham, the agent could return an accurate answer, as demonstrated in Fig. 6, which matches the user's requirements. It can be seen that the Semantic Web 'answers' users' query instead of 'simply matching' keywords.

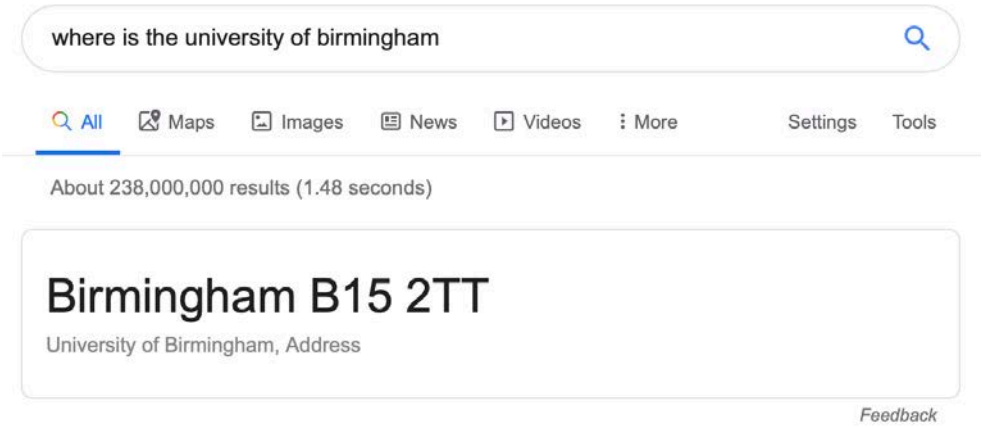


FIG. 6 SCREENSHOT OF A QUERY RESULT ON GOOGLE

To achieve this, an underlying schema and computational model must be provided, that is, the ontology; as illustrated in Fig. 7, ontology plays a central role to form a unified knowledge base that enables advanced knowledge modelling which lays the foundation for Semantic Web applications. Information integration is one of the most challenging issues with regard to the Semantic Web, while the ability of ontologies to express rich semantics is vastly important in this domain (Gaitanou, 2009).

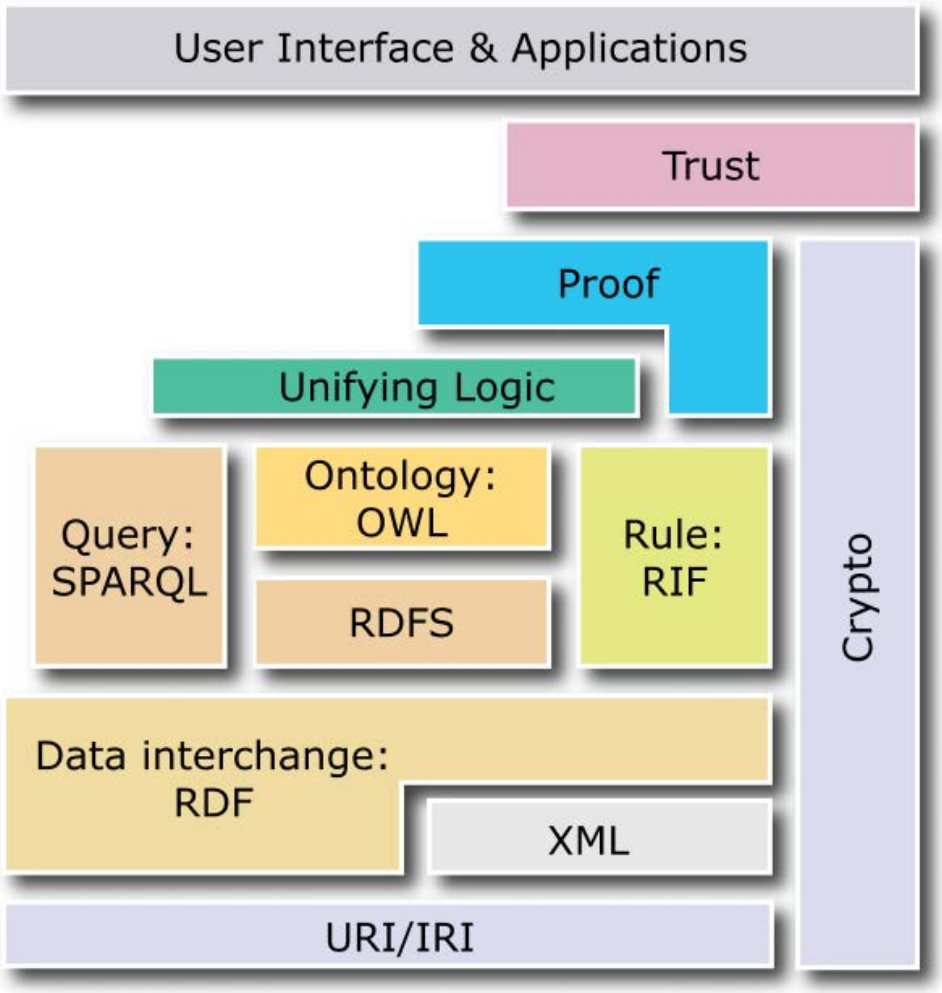


FIG. 7 SEMANTIC WEB STACK[2]

Eventually, once all ontology models are interconnected, a gigantic network, as illustrated in Fig. 8, can be formed. Such a network has inspired multiple industries to proceed with cross-industry knowledge integration (Ashburner et al., 2000; Ebrahimipour and Yacout, 2015).

[2] Sourced from https://www.w3.org/2007/03/layerCake.png; the illustration is the latest work presented by W3C at the point of composing this thesis.

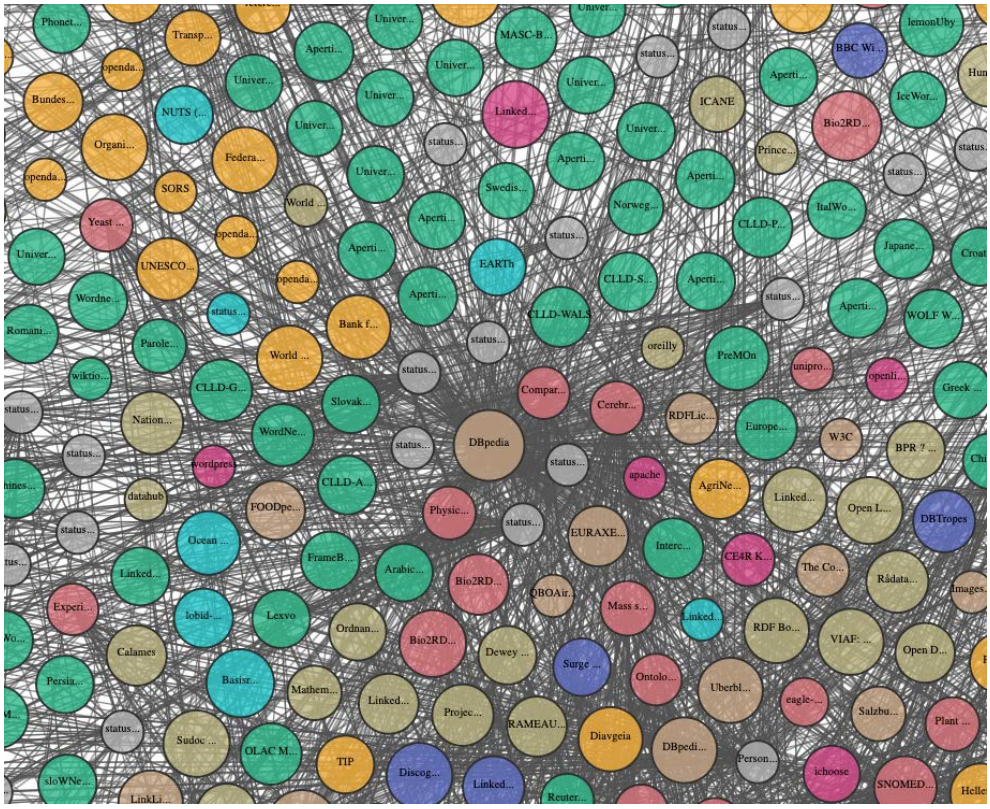


FIG. 8 SNIPPET OF LINKED ONTOLOGY MODELS, EVENTUALLY FORMING LINKED DATA

2.2 ONTOLOGY

2.2.1 ONTOLOGY DEFINITION

Ontology was originally a term used in philosophy, that is the study of existence or being (Philosophy Terms, n.d.). Along with the early development of Artificial Intelligence (AI), the increasing need for automated reasoning required computational models to capture mathematical logics, where the term was adopted (Guarino, 1995). Gruber (2009) stated that in the 1980s, the term ontology was used to represent a modelled world and a component of knowledge systems, while in the 1990s ontology became a standard component of knowledge systems to facilitate interoperability. In 1991, Neches

et al. demonstrated the importance of reusing and sharing reusable components and considered an ontology an important part of assembling a knowledge-based system as a specification mechanism; they also formally gave an initial definition of an ontology as something which '*defines the basic terms and relations comprising the vocabulary of a topic area*'. Gruber (1993) defined ontology as '*an explicit specification of a conceptualization*', adding credit to making ontology a technical term in computer science. In 1997, Borst extended Gruber's definition, proposing that '*ontologies are formal descriptions (specification) of shared knowledge (conceptualisation) in a domain*'; an additional specification is that the conceptualisation should reach a shared view with consensus instead of the individual view, while such a mechanism should be in a formal machine-readable format (Guarino et al., 2009). Eventually, a definition was proposed by Studer et al. in 1998: '*an ontology is a formal, explicit specification of a shared conceptualization*'; a merger of previous definitions, it has been the most prevalent and accepted (Guarino et al., 2009). Guarino et al. (2009) concluded that an ontology is '*a special kind of information object or computational artefact*', a pragmatic method to formally model the structure of a system that comprises generalised, specialised and hierarchical concepts.

An explanation of keywords is provided in Table 2. Overall, ontologies can be described as '*frameworks for representing sharable and reusable knowledge across a domain*', which are capable of modelling high-quality,

coherent and integrated data thanks to their capability for describing relationships, reusability and interconnectedness (Knowledge Hub, 2017).

TABLE 2 GENERAL EXPLANATION OF SOME KEYWORDS IN THE DEFINITION OF AN ONTOLOGY

Keyword	*Explanation*
Formal	An ontology should be machine-readable, interpretable and understandable
Explicit	Axioms and concepts captured in an ontology should be explicitly defined and constrained
Conceptualisation	An abstract, simplified model of some domain knowledge of the world in which relevant concepts are identified
Shared	The knowledge an ontology captures should be consensual and accepted by every contributor in one or multiple domains

An ontology describes a concept and relationships between concepts, becoming a means to represent knowledge for the computing systems. It facilitates:

- Information exchange between humans and organisations
- Interoperability between different systems
- Requirement analysis and system top-level design
- Knowledge reuse

- Explicitly refined domain knowledge
- Separation between domain knowledge and application knowledge underlain by domain knowledge

2.2.2 Different Types of Ontology

Mizoguchi and Nicola (Guarino, 1997; Mizoguchi, 2003) concluded that there are four types of ontology, upper/generic ontology, task ontology, domain ontology and application ontology, and Mizoguchi also categorised ontologies as lightweight or heavyweight. A lightweight ontology is utilised for Web search engines like Yahoo, comprising topic hierarchy and distinction between words with respect to a poorly detailed concept or the principle of concept organisation; a heavyweight ontology is the opposite as it focuses on semantically rigorous relations between concepts and excellent consistency (Mizoguchi, 2003). Yan (2015) elaborated on the definitions of the four types:

- Upper Ontology: An upper ontology mainly denotes the most common and generalised concepts, with attention paid to the most basic attributes and semantic relations such as behaviours, time, things, etc.; it is independent from a specific problem or domains so that it can be shared and applied to numerous fields. All other kinds of ontology can be perceived as special cases of an upper ontology.

- Domain Ontology: A domain ontology is concentrated on a specific domain and relevant knowledge of that; it is less sharable than an upper ontology.
- Task Ontology: A task ontology is designed to capture knowledge of a generic task, describing the relationships between concepts in the task such as planning and fleet management. Some researchers have generalised such an ontology as providing 'human-friendly primitives in terms of which users can describe their own problem-solving process with a high level of descriptiveness and readability' (Ikeda et al., 1998).
- Application Ontology: An application ontology is exploited to represent terms and jargon and their relationships germane to a task in a specific domain. It is dependent on both tasks and domains, focusing on the smallest scope and the most specific use cases (Malone et al., 2010).

The hierarchy is shown in Fig. 9.

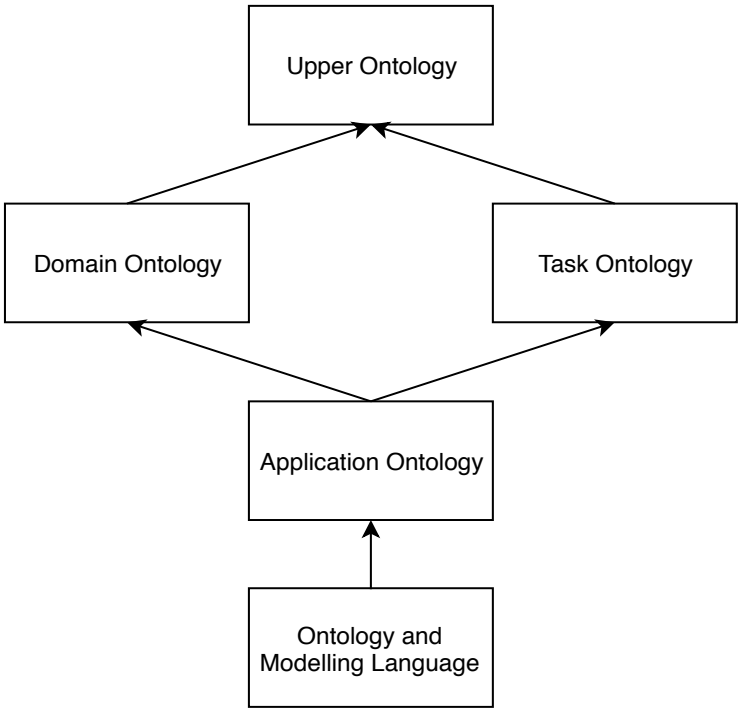


FIG. 9 CLASSIFICATION OF DIFFERENT TYPES OF ONTOLOGY

2.2.3 ONTOLOGY ENGINEERING

Uschold et al. outlined what should be included in an ontology in 1998:

> *An ontology may take a variety of forms, but necessarily it will include a vocabulary of terms, and some specification of their meaning. This includes definitions and an indication of how concepts are inter-related which collectively impose a structure on the domain and constrain the possible interpretations of terms* (Uschold et al., 1998)

It can be seen that the basic components of an ontology must include Classes (i.e., Concepts), Relations, Properties (i.e., Attributes), Instances (i.e.,

Individuals) and Axioms. In company with the continuous development and additional requirements for ontologies, more components can be added to an ontology if necessary; these include Function terms, Restrictions, Rules and Events (Gómez-Pérez et al., 2010). The functionality of those components is summarised in Table 3.

The example shown in Table 3 is depicted in Fig. 10. It can be seen that restrictions have been placed in between some entities, such as *PetrolCar fuelledBy only Petrol*. However, although it might seem intuitive with this single case, modelling an ontology can be a highly complex task that might involve multiple participants, including domain knowledge experts, to complete it, hence the need for a scientific and efficient process.

TABLE 3 DIFFERENT COMPONENTS OF AN ONTOLOGY

Component	*Functionality*	*Example*
Classes	Sets of collections of a concept, kinds of things	Car; car is a collective term used to represent road vehicles that typically have four wheels, hence being a class
Instances	Objects instantiated from concepts (i.e. classes)	BMW 320i (F30); an instance of a car
Relations	The ways in which classes or individuals relate to each other	BMW 320i (F30) is a car manufactured by BMW; the relationship between BMW 320i (F30) (an instance of a car) and BMW (an instance of a car manufacturer) is *carObject-manufacturedBy-carBrandObject*

Properties	Characteristics and parameters that objects have	BMW 320i (F30) is fuelled by petrol
Function terms	Complex structures that can replace a specific vocabulary with certain relations in a formal statement, which can be seen as classes that describe activities rather than entities	Petrol cars can be rear-wheel drive, front-wheel drive or four-wheel drive
Restrictions	Formalised assertions that have to be met in order to ensure captured data is valid	Petrol cars must be fuelled by petrol
Rules	Logic statements that describe logical inference, which forms antecedent-consequent logics	If a car is solely fuelled by petrol, it is a petrol car
Axioms	All assertions in a logical form that incorporate descriptions in their domain of application	A car is a kind of vehicle
Events	Alterations in properties or relations	The BMW 320i (F30) was powered by an N20B20 engine from 2012 to 2015 and a B48B20A engine from 2015 to 2019

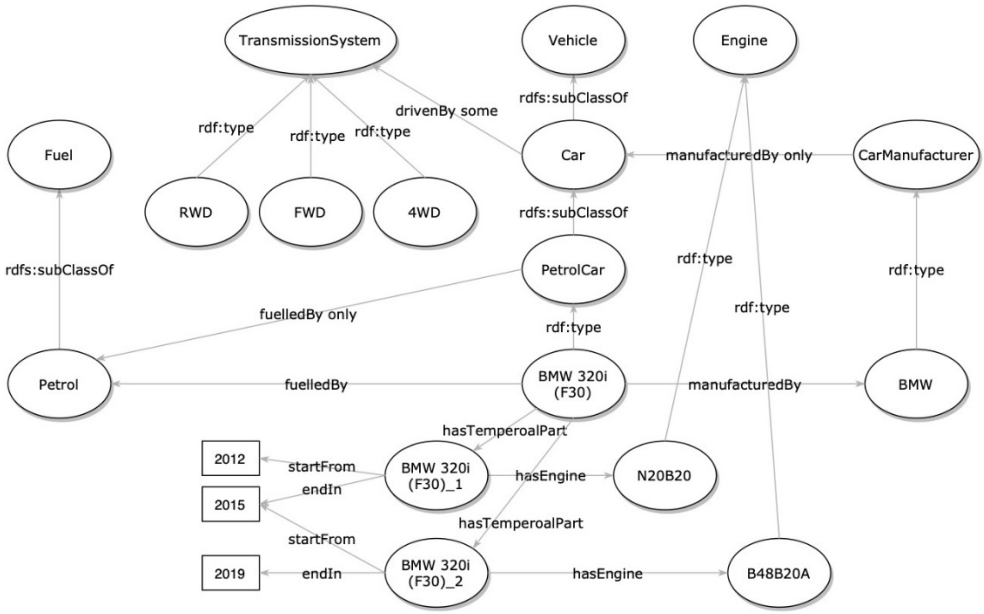


FIG. 10 DEPICTION OF A SIMPLE CAR ONTOLOGY USING DATA FROM TABLE 3 (RESTRICTIONS ARE WRITTEN WITH MANCHESTER OWL SYNTAX[3] FOR READABILITY)

In 2001, researchers from Stanford University proposed a primitive method in seven steps (referred to as 7-Step below) to model ontologies, along with their open-sourced ontology editor – Protégé (Noy and McGuinness, 2001); they also suggested three principles for engineering an ontology:

[3] https://www.w3.org/TR/owl2-manchester-syntax/

1) *There is no one correct way to model a domain – there are always viable alternatives. The best solution almost always depends on the application that you have in mind and the extensions that you anticipate.*
2) *Ontology development is necessarily an iterative process.*
3) *Concepts in the ontology should be close to objects (physical or logical) and relationships in your domain of interest. These are most likely to be nouns (objects) or verbs (relationships) in sentences that describe your domain.*

However, 7-Step lacks an evaluation and maintenance process in the later stage of an ontology's design process; this was addressed by Pinto and Martins (2004) who summarised the different stages of an ontology design:

- Specification
- Conceptualisation
- Formalisation
- Implementation
- Maintenance
- Knowledge acquisition
- Evaluation and documentation

Multiple ontology engineering methods have been developed in later years, based on the results achieved by the aforementioned studies (Noy and McGuinness, 2001; Pinto and Martins, 2004), e.g., NeOn methodology (Carmen Suárez-Figueroa et al., 2012; Pérez et al., 2008), HCOME (Kotis and Vouros, 2006) and Ontology Maturing (Braun et al., 2007).

In more recent years, because of the booming volume of data, many researchers have started to investigate fully automated or semi-automated methods to extract knowledge from unstructured data. An example is using a Graph Embedding algorithm to semantically extract a graph and then using ontology alignment techniques to either complement existing ontologies or build a new one (Cai et al., 2018; Goyal and Ferrara, 2018).

2.2.4 USING ONTOLOGIES TO INFER IMPLICIT KNOWLEDGE

Reasoning capability is crucial to applications underpinned by ontologies (Sirin and Parsia, 2004). Before going deep into reasoning, a famous model should be mentioned. This model is known as DIKW (Data, Information, Knowledge, Wisdom) Hierarchy; it reveals the hierarchy and the relationships between data, information, knowledge and wisdom (Ackoff, 1989), as illustrated in Fig. 11.

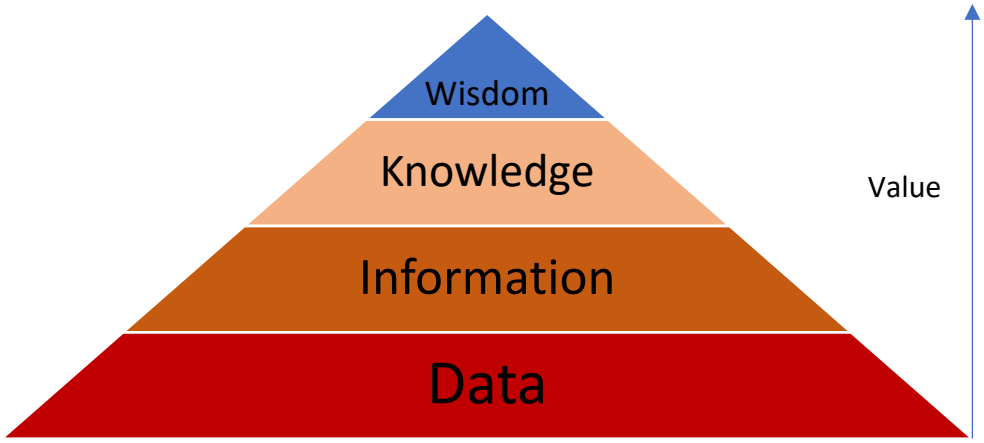


FIG. 11 DIKW HIERARCHY

Rowley (2007), referring to the DIKW pyramid, concluded that '*typically information is defined in terms of data, knowledge in terms of information, and wisdom in terms of knowledge*'. In a nutshell, we can gain knowledge based on information derived from data being absorbed from the real world and logics can therefore be drawn from knowledge. Here, information is the objective fact concluded by analysing relevant data from the real world and information is induced and deduced from perceived data.

However, machines cannot ‘act’ like a human as data is simply a sequence of binary values. The reason why we understand data is because we ‘give’ it meaning (i.e., information) in context (i.e., knowledge). A human can construct links between entities. In other words, a human brain can wave a gigantic network to store knowledge; such a knowledge network can be expressed by triples, i.e., subject-predicate-object structure.

For example, 01/01/1991 is a sequence of digits, which might mean nothing to some people, yet it can be perceived as a date in some people’s view because they have learnt that such a format (i.e. DD/MM/YYYY) can represent a date. If we take it further, it can be perceived as the New Year’s Day, while if we move another step forward, it can be a person’s birthday; such perceptions require prior knowledge (i.e. people around that person have to gain the information, that this person was born on 1st January, 1991, in advance). Such a transformation is demonstrated in Fig. 12.

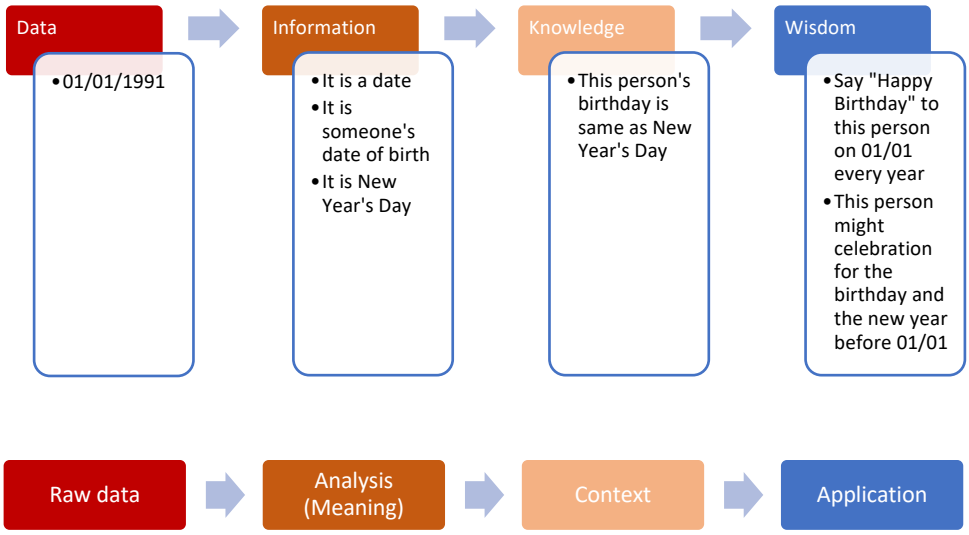


FIG. 12 FROM DATA TO WISDOM

This example demonstrates a flow of how raw data is transformed into wisdom. The whole process can be seen as different levels of understanding, as depicted in Fig. 13, where it is clear to see that it is possible to extract relationships from data and from the relationships it is possible to locate patterns. The patterns can be further concluded as knowledge and wisdom. With data, information and knowledge, it is possible to ensure a thing will be done 'correctly', while wisdom guides people to choose the 'correct' way before doing a thing. This process might seem straightforward to humans, but it is not to machines.

Machines are not capable of inferring new facts based on existing data unless pre-determined logic (e.g., IF-ELSE statement) is provided. However, by using semantic reasoners (i.e. rule engines or inference engines), machines can infer consequent facts based on given rules and axioms (Clark et al., 2011; Sirin and Parsia, 2004).

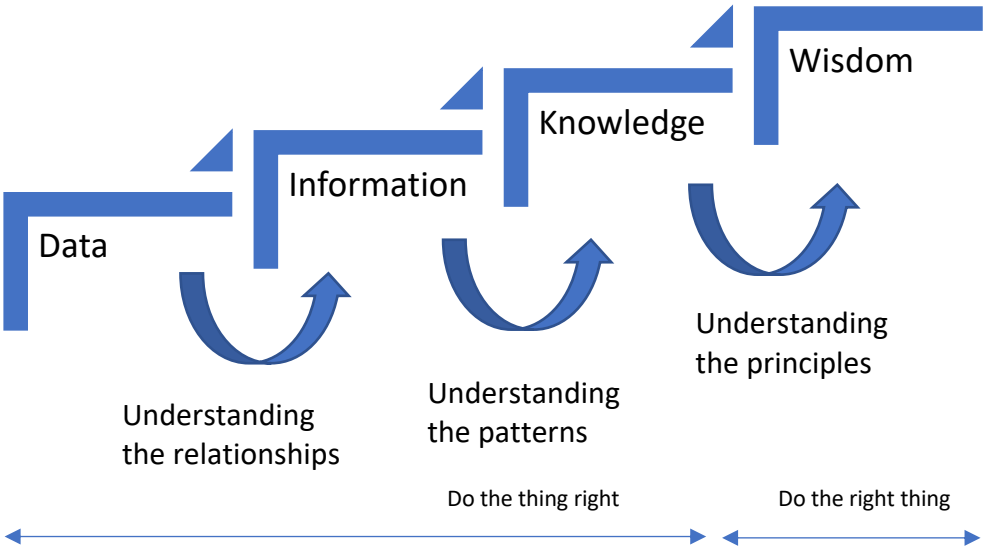


FIG. 13 DIFFERENT LEVELS OF UNDERSTANDING WHILE TRANSFORMING DATA TO WISDOM

A reasoner is a tool that aids machines in understanding the description logics defined in ontology rules and ontologies themselves, and inferring resultant facts; it usually also provides other functions such as an ontology consistency check, identification of subsumption, individual classification and so on (Shearer et al., 2009; Sirin and Parsia, 2004). Some reasoners which have been implemented are: HermiT, an optimised OWL reasoner that can cope with large-sized complex ontologies (Shearer et al., 2009); and Pellet, Pellet 2 and their commercial implementation, Pellet 3, which can also efficiently handle OWL, providing additional nominal support and explanations of the inference (Clark et al., 2011; Sirin and Parsia, 2004).

It is reasonable to envisage the potential this knowledge network has – with a gigantic network that allows machines to infer knowledge based on the data or information fed to them, much more can be achieved.

2.2.5 Tools for Ontologies

Presently, there are three main types of tool that help users to work with ontologies: programming frameworks, ontology editors and triple stores; some are open-sourced and free to use whereas some are commercially licensed.

2.2.5.1 Programming Framework

As a result of the increasing need to work with ontologies, many frameworks have been implemented, which allow developers to develop applications based on ontologies, or help developers to manage ontologies more easily. Some recent developments of programming frameworks[4] for ontologies are shown in Table 4.

It is worth mentioning that although most of those frameworks are designed to interact with ontology models at coding level, they also support in-memory storage as well as on-disk storage, and usually support SPARQL. In other words, they can work similarly to dedicated triple stores with compromised query performance.

2.2.5.2 Triple Store

A triple store[5] is a special database for storage and retrieval of RDF and OWL data, typically in conjunction with support for SPARQL and HTTP. In theory,

[4] Some frameworks were intended to work with RDF, providing little support on OWL vocabularies.

[5] https://www.w3.org/2001/sw/wiki/Category:Triple_Store

many modern triple stores are ‘quad stores’ that not only store triples (i.e., subject-predicate-object) but also store named graphs, then becoming quads (i.e., graph-subject-predicate-object). Because quad stores are essentially triple stores, the term ‘triple stores’ will be used below. In this section, some dedicated triple stores are shown in Table 5. It is worth noting that although many frameworks can be used as local triple stores, they are excluded from this section.

2.2.5.3 EDITOR

If data publishers are not satisfied with existing ontologies, they can build their own ontologies or extend existing ones. However, it is not always an easy task to create an ontology by coding, hence the rationale for ontology editors. Some popular ontology editors are shown in Table 6.

TABLE 4 SOME PROGRAMMING FRAMEWORKS FOR USING ONTOLOGIES

Name	*Description*	*Programming language*
Owlready2[6]	Owlready2 is an open-sourced Python package that can create ontologies with Python code or load OWL as Python objects and make inferences via HermiT and Pellet 2. It allows direct access to OWL-based ontologies. Owlready2 also includes an optimised triple store based on SQLite3 that supports large ontologies (Lamy, 2016).	Python3
Ontospy[7]	Ontospy is another open-sourced Python library that includes several command-line interfaces for OWL ontology inspection, documentation and visualisation. However, ontology editing is not supported by Ontospy (Pasin, 2019).	Python3
RDFLib[8]	RDFLib is an open-sourced Python package for working with RDF. It provides several useful APIs to help developers parse or serialise RDF in various formats, manipulate graphs and create an in-memory or persistent RDF store – Berkeley DB. However, RDFLib does not support OWL vocabularies. Both Owlready2 and Ontospy are underlain by RDFLib.	Python3
dotNetRDF[9]	dotNetRDF is an open-sourced .NET library for editing, managing and querying RDF, also providing several .NET APIs for interacting with other triple stores. It can also work as a stand-alone triple store to store and retrieve RDF data.	.NET C#
Jena[10]	Jena is an open-sourced Java framework for building Semantic Web and Linked Data applications developed by the Apache Software Foundation. Jena can work with both RDF and OWL; it has the most comprehensive support for working with ontologies, including reasoning support. Jena can also be used to construct a stand-alone triple store that has SPARQL endpoints. Many APIs are provided by Jena; these help developers to interact with external storage or applications.	Java

[6] More details are available at https://pythonhosted.org/Owlready2/
[7] More details are available at http://lambdamusic.github.io/Ontospy/
[8] More details are available at https://rdflib.readthedocs.io/en/stable/
[9] More details are available at https://www.dotnetrdf.org
[10] More details are available at https://jena.apache.org/index.html

RDF4J[11]	RDF4J is another open-sourced Java framework for processing and managing RDF data. Like Jena, it also provides support for reasoning at RDFS level and APIs for interacting with other applications. RDF4J can also be used to store RDF data and form SPARQL endpoints.	Java

[11] More details are available at https://rdf4j.org

TABLE 5 BRIEF INTRODUCTION TO SOME TRIPLE STORES

Name	*Description*
Stardog[12]	Stardog is a commercial triple store that provides fast SPARQL query, OWL reasoning, intuitive user interaction, etc. Inclusion of the latest Pellet 3 reasoner enables Stardog to run reasoning with fine performance at different reasoning levels.
Sesame[13]	Sesame is an open-sourced RDF database that supports RDFS-level reasoning. OWL vocabularies are not natively supported by Sesame. However, as a result of Sesame Sail API, some third-party stores have been built upon Sesame through the AP, which has made it possible to handle OWL data and reasoning at OWL level.
OpenAnzo[14]	OpenAnzo provides both commercial and open-sourced versions of a triple store. The most notable feature of OpenAnzo is that it is not only a triple store but also a service-oriented semantic middleware platform that facilitates the creation of complex applications based on W3C semantic technologies such as OWL and RDF.
OpenLink Virtuoso[15]	OpenLink Virtuoso also has both commercial and open-sourced editions. It has a built-in OWL reasoner that supports the latest OWL vocabularies. A notable feature of OpenLink Virtuoso is RDB2RDF, which converts data stored in relational databases to RDF directly.
Oracle Database 19c[16]	Oracle Database 19c has complete support for RDF storage, which enables easy integration of an RDF database to other Oracle database products, targeting large complex systems with find performance. The database has complete support on OWL and reasoning thereof.

[12] More details are available at https://www.stardog.com
[13] More details are available at https://www.w3.org/2001/sw/wiki/Sesame
[14] More details are available at http://www.openanzo.org
[15] More details are available at http://virtuoso.openlinksw.com
[16] More details are available at https://www.oracle.com/database/technologies/

TABLE 6 SOME POPULAR ONTOLOGY EDITORS

Name	*Description*
Protégé[17]	Protégé is an open-sourced and pluggable ontology editing and knowledge management platform developed by the Stanford Center for Biomedical Informatics Research at the Stanford University School of Medicine (Musen, 2015). It provides a series of tools that enable interactive and intuitive ontology editing and maintenance. The compatibility of reasoners enables users to run an inference about concepts captured in the ontology. The later addition of support for SWRL has enabled more sophisticated reasoning functions. The main reasons why Protégé is popular is because of its high level of flexibility, scalability and extensibility (Escórcio and Cardoso, 2007), which facilitates the construction of large ontologies. It also provides several APIs for programming knowledge-based tools and applications in Java (Alatrish, 2012).
TopBraid Composer[18]	TopBraid Composer has three commercial editions, Free Edition, Standard Edition and Maestro Edition, developed by TopQuadrant. TopBraid Composer is a professional Eclipse-based tool for ontology development and semantic applications. Benefiting from its built-in rule engine, it can help users to ensure the consistency of the ontology with W3C's SHACL. It also has a complete suite to help users build and publish Semantic Web applications.
OWLGrEd[19]	Developed by the Institute of Mathematics and Computer Science at the University of Latvia, OWLGrEd is a free-to-use graphical ontology editor that enables users to edit and visualise an ontology with only a 'few mouse clicks'. It also provides many export options to facilitate visualisation sharing. However, in spite of powerful visualisation and interactive editing, OWLGrEd cannot run reasoning due to the lack of reasoners.
Apollo[20]	Apollo is an open-sourced knowledge modelling software in

[17] More details are available at https://protege.stanford.edu/about.php
[18] More details are available at https://www.topquadrant.com/products/topbraid-composer/
[19] More details are available at http://owlgred.lumii.lv/
[20] More details are available at http://apollo.open.ac.uk/

	Java developed by the Knowledge Media Institute at the Open University. Although it can only natively import from ontologies coded in Apollo Meta Language, Apollo is pluggable, and can be adapted to different formats via plugins. The built-in consistency checker ensures the consistency of the ontology during the editing process. However, there is no built-in reasoner nor external reasoners, hence its incapability for reasoning.
OntoStudio X[21]	OntoStudio X is a professional ontology development environment based on Microsoft Excel 2019, developed and supported by Semefora Systems GmbH. It aims to provide industry-leading semantic processing capability with high performance. OntoStudio X also provides a set of modelling tools for ontologies and rules with functions for the integration of heterogeneous data sources. OntoStudio X also has programming interfaces (available in Java and Python) that enable users to deploy self-developed modules, ensuring high extensibility. It also has a built-in reasoner.

[21] More details are available at https://www.semafora-systems.com/ontobroker-and-ontostudio-x

2.2.6 Some Existing Ontologies

Several domain ontologies have been published and exploited. For example, Gene Ontology[22], a computational model for biological systems, consists of three independent ontologies: biological process, molecular function and cellular component, providing structured, controlled and precise vocabularies and classifications that cover several domains in biosciences; it has become an important tool for researchers to turn data into knowledge (Ashburner et al., 2000). Gene Ontology contains '1395 component terms, 7291 function terms and 8479 process terms', allowing annotation of genes, attributes thereof and relationships between genes (Smith and Kumar, 2004). The role of ontology in the bioinformatics domain has reshaped bioinformaticians' opinion towards ontologies, that it should be a mainstream product and be available to the bioinformatics community (Bodenreider and Stevens, 2006). In more recent research, Groß et al. 2016) identified the need to interlink various ontologies used in the bioinformatics domain and investigated mapping approaches, focusing on a review of automated annotation that keeps ontology-based mappings in the presence of evolving ontologies; they found over 500 ontologies being used at the time in bioinformatics, covering many sub-domains, realising more automated and explicit data processing in the bioinformatics domain.

[22] More details are available at http://www.geneontology.org

Moreover, the BBC, one of the most influential broadcasting organisations in the world, also makes extensive use of ontologies to be ready for the advent of the Semantic Web[23]. The usage of ontologies enables the BBC to link its topics and share the content it creates, facilitating better business management, data storage and sharing to other parties, and most importantly, enabling its audience to have a better experience (BBC, n.d.). The ontologies published by the BBC laid the foundation for a Linked Data Platform that enables both internal and external developers to interact with the BBC's open data to bring more inspiration to their creative work. The design of the BBC ontologies contains massive reuse of existing ontologies, amongst which the most notable one is DBPedia[24], an open-sourced knowledge base. DBPedia predominantly extracts and maps semi-structured and unstructured data from Wikipedia[25], the sixth most popular website and the most widely used online encyclopaedia, to transform data published on Wikipedia into Linked Data format to facilitate a more comprehensive view of over 5 million entities with support in multiple languages (Lehmann et al., 2015).

In the railway industry, Morris et al. (2014) reviewed some ontologies designed for the rail domain, including upper and domain ontologies; they found that the application of ontologies can bring better data integration, and better value can be obtained by combining data from various sources

[23] More details are available at https://www.bbc.co.uk/ontologies
[24] More details are available at https://wiki.dbpedia.org/about
[25] https://www.wikipedia.org/

within the rail domain. Railway Core Ontology (RaCoOn) is one of the ontologies aiming to improve rail data integration, particularly focusing on railway asset monitoring applications; it is a great example of enhanced management across large complex systems (Tutcher, 2015b; Tutcher et al., 2017).

Another successful ontology established in the rail domain is Railway Domain Ontology (RDO) produced by the InteGRail project[26]; it is a method of constructing a machine-interpretable conceptual model of domain concepts and physical components into practice, offering major participants in the rail industry a unified standard to exchange data between one information system and another (Köpf, 2010).

All the aforementioned ontologies are domain ontologies because they cover the certain scope of knowledge in a domain. There have been also some task ontologies designed for purposes such as text classification, document classification and data mining. Cheng et al. (2004) demonstrated how to incorporate user context and preference in the form of an ontology to classify unstructured documents into useful categories, and Fang et al. (2007) introduced an ontology-based Web method for automatic classification and ranking of documents. Another study presented a way in which the concepts of machine-learned functions are captured by an ontology, assisting general

[26] http://www.integrail.info

users to access learned models and enhancing the reusability of the models obtained (Xu et al., 2016).

2.2.7 Some Ontology-Based Applications

Much research has been devoted to exploiting the full potential of ontologies, with many developments using ontologies. Overall, there are two main aspects: heterogeneous data integration, and knowledge modelling and management. In this thesis, the following section will focus on data integration, while applications with ontology-based knowledge modelling will be briefly introduced in this section.

Ontology-based management systems have been used for digital information management, using ontologies to organise human knowledge and logic (Brochhausen et al., 2011; Fensel, 2002; Studer et al., 1998). For example, traditionally, keyword matching systems offer limited information-sharing functionalities with little support for information maintenance (Fensel, 2002), while the Semantic Web resolves the ambiguity and implicitness led by keywords by using ontologies to enhance the capability to understand users' exact questions and requirements (Berners-Lee, 2006; Berners-Lee et al., 2001; Che, 2006). The introduction of the Semantic Web inspired further development with ontologies. Fensel (2002) proposed an ontology-based knowledge management tool that processes heterogeneous, distributed and semi-structured documents to facilitate automated information extraction and information maintenance, in which ontologies provide machine-

readable semantics for both explicit and implicit information with inference, improved information access in large intranets, and knowledge sharing and reuse for customer relationship management. The EU co-funded project Advancing Clinico-Genomic Trials on Cancer – Open Grid Services for Improving Medical Knowledge Discovery (ACGT)[27] also delivered the ACGT Master Ontology (MO) and technical infrastructure thereof, including an Ontology-based Trial Management Application (ObTiMA) that utilises ACGT-MO to facilitate semantic integration in the context of multi-centric, post-genomic clinical trials (Brochhausen et al., 2011). Additionally, Cheng et al. (2004) attempted to transform traditional keyword document matching to knowledge-based document matching by analysing the content of unstructured or semi-structured documents in conjunction with a domain ontology to classify documents semantically. Furthermore, another group of researchers devoted to bioinformatics also noticed that many pieces of literature exist in the form of free text, which makes information retrieval and processing more difficult; thus, they developed the ontology-based MEDLIE document classification tool to help professionals to search information in a domain-independent manner (Camous et al., 2007). Additionally, Wang et al. (2006) applied ontologies to image annotations to help machines better understand users' queries for images and improve the overall performance of image retrieval.

[27] https://cordis.europa.eu/project/rcn/79480/factsheet/en

Attempts have also been made to used ontologies with fault diagnosis, thanks to their flexible and comprehensive knowledge modelling capability. Some researchers have used an ontology-based approach in prognostics and health management to ease diagnosis activities and minimise the impact on the global performance of a system; it has been designed to exploit domain knowledge and data to provide a more holistic view of the incident and identify the cause of abnormality (Medina-Oliva et al., 2014). Some other researchers have developed an ontology-based diagnosis system to carry out predictive railway maintenance to decrease disruption by enhancing the capability of diagnosing a mission-critical fault while keeping maintenance costs as low as possible, reducing life-cycle spending and gaining a better return on investments (Umiliacchi et al., 2011). Zhou et al. (2015) developed a method for intelligent fault diagnosis based on ontology and FMECA (Failure Mode, Effects and Criticality Analysis), transferring FMECA into a machine-interpretable form to enhance automation capability for more intelligent diagnosis and more rapid and accurate solutions, facilitating knowledge sharing between different wind turbine suppliers to reduce overall costs.

Ontologies can capture real-world concepts, hence their potential in decision-making. Many projects have attempted to implement ontology-based enhanced Decision Support Systems (DSS). It is of vast importance that a DSS delivers relevant, reliable, precise and accurate information to its users, while the ontology can establish the infrastructure to realise such a goal (Blomqvist, 2014). Decision support has been deemed as one of the main

objectives of ontology-based knowledge management systems, and Bastinos and Krisper (2013) proposed an outline of how to model decisions in ontologies.

Ontology-based decision support has been investigated in various domains. In the medical sector, because multiple factors have to be taken into account to make a decision, researchers have developed ontologies to help practitioners to diagnose disease and make decisions (Farooq et al., 2011; Haendel et al., 2018; Zhang et al., 2014); Zhang et al. (2014) proposed an ontology-driven decision support approach that distinguishes patients with mild cognitive impairment, to assist physicians in conjunction with a set of rules and machine learning techniques. In the rail domain, Lu et al. (2006) were inspired by Semantic Web technologies to propose ontologies to be used with intelligent DSS in the railway system, that improved rail data presentation and queries as well as the linkage of global databases; Lewis (2015) utilised an ontology to integrate heterogeneous data as well as knowledge to aid decision-making processes for the rail industry. In addition, Chang (2008) proposed an ontology-based approach to manage product design knowledge to realise consistent and accurate decision support for error detection and analysis. Abanda et al. (2011) noticed the complexity in making land delivery decisions in Zambia; they discussed the extent to which ontologies can be used in DSS development to can facilitate the land delivery process.

Use of ontologies in Natural Language Processing (NLP) applications has been also investigated given that ontologies can capture real-world concepts. In fact, the above-mentioned ontology-based document classifications could be seen as some examples of ontology-based NLP applications. In 1995, Mahesh and Nirenburg attempted to represent the meanings of text in order to facilitate natural language interpretation and generation by using an ontology for NLP; they also drew some initial conclusions on the usefulness of ontologies for NLP:

- Reduce ambiguities with constraints given by the ontology based on sectional preferences for relations between concepts
- Infer from input text with the knowledge contained therein to further reduce ambiguities and fill slots while necessary
- Infer from the topology to improve searching for the shortest path between concepts, enabling metonymy and metaphor processing (Mahesh and Nirenburg, 1995)

Other researchers have designed rich lexicon models with ontologies in OWL to capture semantics in a domain based on human knowledge, while facilitating greater lexicon sharing and easier NLP analysis (Cimiano et al., 2007). More recently, along with the increasing popularity of machine learning techniques, ontologies have also been used in more NLP applications together with machine learning techniques. For example, one research project demonstrated an ontology-based approach to classify sentiment from

unstructured data (i.e. free text), using knowledge contained in ontologies to improve the performance of a Support Vector Machine (SVM) classifier (Thiyagu and Sendhilkumar, 2011). It can be seen that NLP has benefited from using ontologies as a result of their flexible knowledge modelling and managing capability.

Ontologies also have long history of being a useful set of tools to integrate heterogeneous and unstructured data, facilitating better data accessibility and integration (Ashburner et al., 2000; Chandrasekaran et al., 1999; Cruz and Xiao, 2005; Ebrahimipour and Yacout, 2015; Köpf, 2010; Morris et al., 2014; Saa et al., 2012; Tutcher, 2015b; Tutcher et al., 2017; Udrea et al., 2007; Xiaomeng et al., 2015). Heterogeneous data and data silos hinder collaboration between departments or organisations, creating difficulties for data sharing and reuse, especially when many stakeholders are involved (Verstichel et al., 2011). Dill (2019) discussed the issues brought by heterogeneous and unstructured data; despite its great value, it is difficult to analyse such data due to its heterogeneity plus various problems such as the greater cost of data cleaning and filtering, and a difficult and complex data retrieval process. W3C has published several ontologies to facilitate data integration, such as Semantic Sensor Network Ontology (Haller et al., 2017; Compton et al., 2012), Organisation Ontology (World Wide Web Consortium, 2014), W3C Geographical Ontologies (Lieberman et al., 2007), etc., which have also been adapted to implement practical applications. For example, as part of the UK government's open data scheme, an organogram for

government offices was implemented with reference to an Organisation Ontology in the form of RDF[28] to provide a clearer picture of the UK government to the public. There are also some commercial services available for data integration solutions with ontologies (Stardog Union, 2017).

In terms of industry-wide application, the oil and gas industry is a decent example. In 2004, an attempt was made to integrate data using ontologies in the oil and gas industry with regard to machine interpretability and interoperability based on ISO 15926, with an investment of £2.5 million; it aimed to deliver approaches to establishing information pipelines for information exchange and integration that are compliant with the Semantic Web standard, as illustrated in Fig. 14 (IIP Steering Group, 2008). Integrated Operations was designed to optimise production, operation and vending processes; production and the average oil recovery rate increased by 5%–10% and 14%, respectively, while the operation and maintenance costs reduced by 25%–40% (IIP Steering Group, 2005).

[28] https://ckan.publishing.service.gov.uk/dataset/staff-organograms-and-pay-government-offices

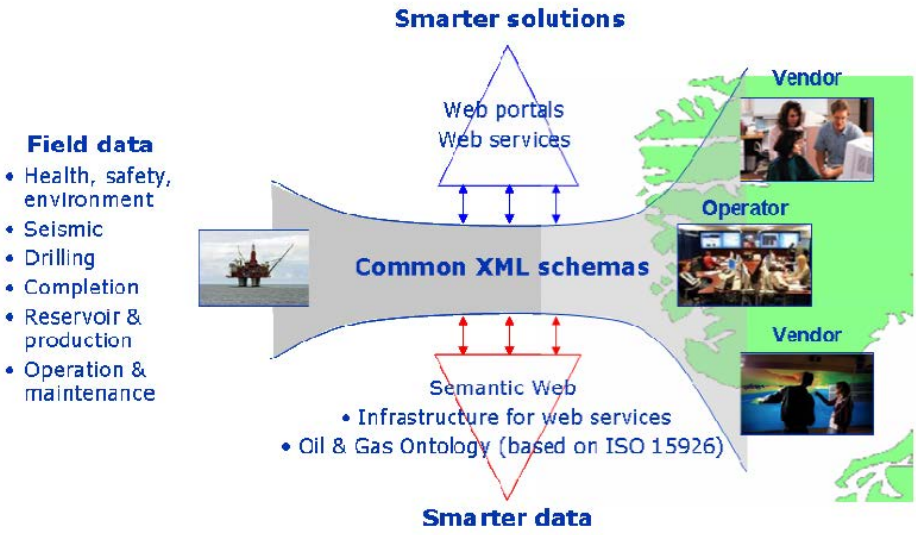


FIG. 14 THE INFORMATION PIPELINE. THE IIP PROJECT HAS SUPPORTED THIS THROUGH THE DEVELOPMENT OF TERMINOLOGIES, TAXONOMIES AND ONTOLOGIES FOR THE CORE E&P PROCESSES (IIP STEERING GROUP, 2008)

A succeeding project, Integrated Operations in the High North (IOHN), was launched in 2008 and completed in 2012. It identified the need for collaboration across interdisciplinary, geographical and organisational boundaries; in order to meet the need, sharable information and knowledge models are essential to ensure interoperability (Verhelst, 2012). IOHN delivered a data integration solution based on ISO 15926 for Integrated Operations, realising common and standardised data formats that allow systems across disciplines to exchange and retrieve data between 22 stakeholders in the oil and gas upstream industry (Verhelst, 2012). Fig. 15 shows the architecture of IOHN; it can be seen that users can access all information from a common Information Service Bus (ISBM) upon a set of ontologies. Fig. 16 illustrates early adoption and development of ontologies in the oil and gas industry. The successful application of ontologies in the oil and gas industry has

proven that it is possible to utilise ontologies to manage and integrate data in large complex systems.

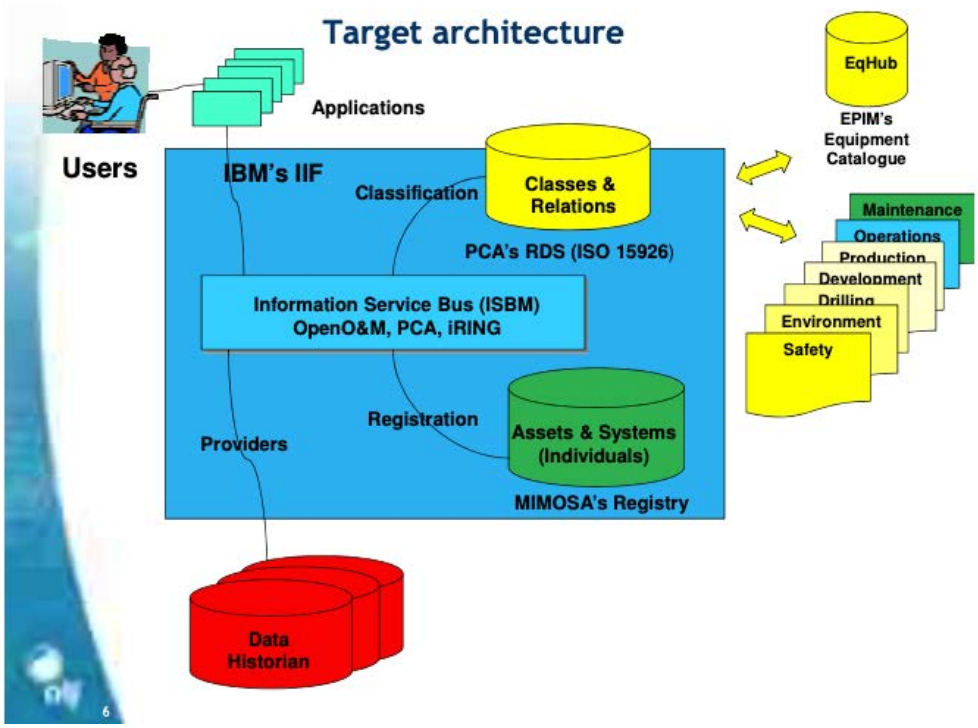


FIG. 15 IOHN ARCHITECTURE (THORE, 2010)

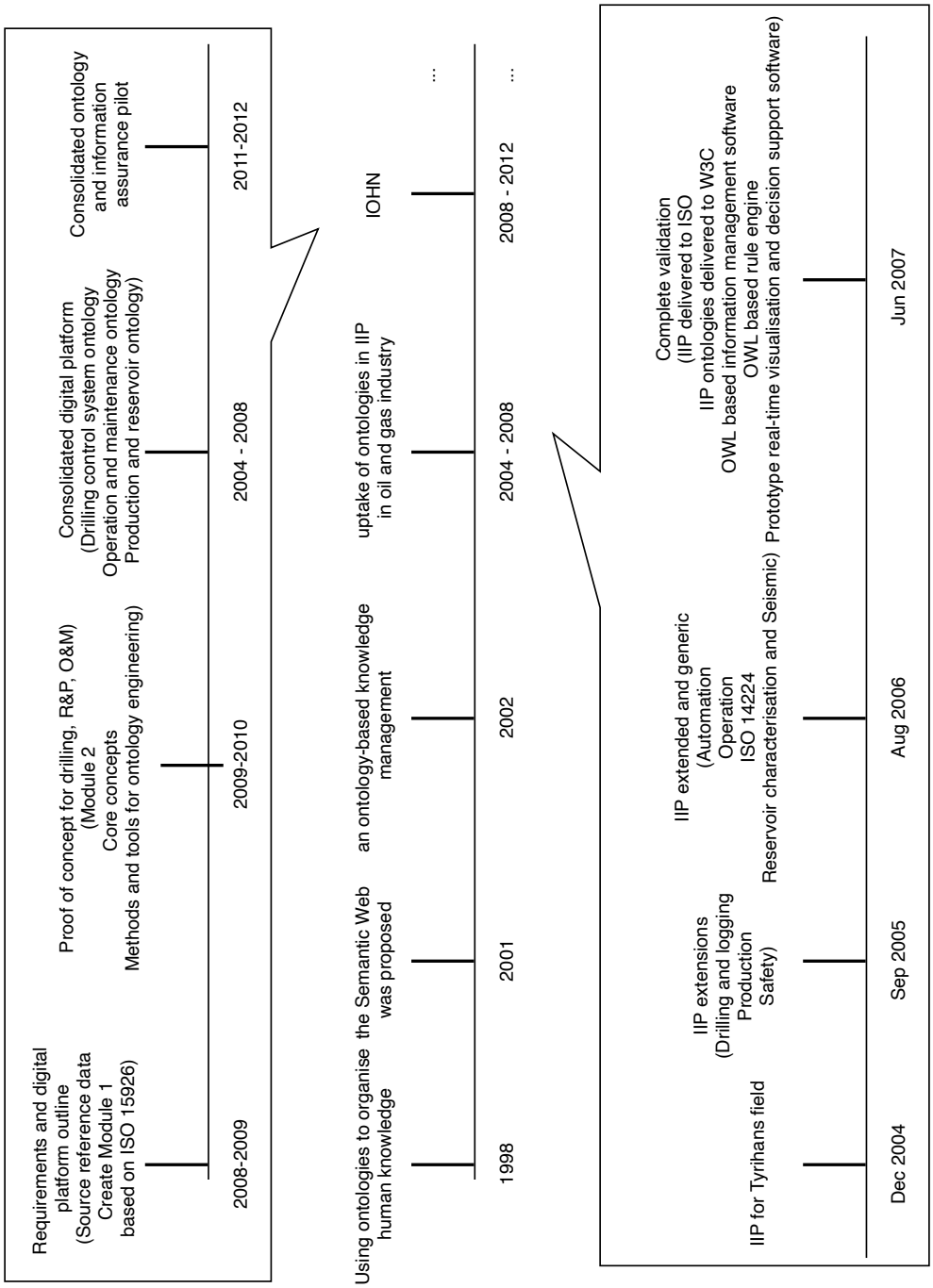


FIG. 16 TIMELINES OF EARLY ADOPTION OF ONTOLOGIES IN THE OIL AND GAS INDUSTRY

2.3 Increasing Need for Data Integration With Ontologies in the UK Rail Industry

2.3.1 Current State

The UK railway industry involves a wide range of participants because of privatisation which also creates additional complexities. Data silos have been found existing in the industry (Morris, 2017) with many legacy systems remaining (Easton et al., 2010). Data silos exist in such large complex system as a result of stakeholders' different goals, priorities and responsibilities, impeding overall high-level data integrity and productivity (Rouse, n.d.). This is especially applicable to the UK railway industry because of the existence of systems supported by dozens of suppliers (Tutcher, 2015b). As briefly mentioned in Chapter 1, the UK rail industry had over 130 information systems in 2011, supported by approximately 20 suppliers, amongst which many were legacy systems that were expensive to maintain and inefficient, leading to the difficulty in information sharing and exploitation as well as the whole-system technology upgrade plan (TSLG, 2012).

Isolated data silos lead to less interoperability between systems. Typically, the interoperability between systems in the UK rail industry mainly relies on intermediate interpreters (Roberts et al., 2011). However, when something changes in the data model, it could be a complex task to upgrade intermediate software, especially when data heterogeneity exists in the system because it is difficult to ensure all data sources have been changed in legacy

systems accordingly. It is not only inefficient but also expensive, especially in terms of condition monitoring.

Disparate data sources have raised the level of data heterogeneity and lowered interoperability; meanwhile, the existence of diverse IT suppliers in the UK railway industry makes it almost impossible to compel all stakeholders to implement a generic and centralised repository to facilitate greater data sharing and reuse in the UK railway system. Improving railway operation still necessitates joint effort on data integration and sharing.

It is also worth mentioning that in the UK, both railway infrastructure managers and TOCs monitor the condition of their assets; they tend to develop and maintain their own condition monitoring systems separately and keep the data in a proprietary binary format, making data integration more complex and sustaining the longevity of legacy systems (Easton et al., 2010). This is particularly common in the UK railway industry as a result of the privatisation mentioned above. On top of that, although the longevity of condition monitoring systems can reduce cost in some ways, old systems are difficult to incorporate into newer systems. It is reasonable to imagine that different condition monitoring system suppliers might have different ways of collecting and using data with their own standards. For example, the Total Operating Processing System (TOPS) was developed in the 1960s in its own programming language to collect information from locomotives where TOPS was deployed, generating plain text-based data and obscure results (i.e.

unstructured data) (Easton et al., 2010), making it difficult to incorporate generated data into other systems.

Moreover, the Department for Transport (2011b) stated that when new stakeholders joined the industry, existing legacy systems, such as TOPS, could be found to be unsuitable for their organisational structures. However, despite being suggested that although new information systems are important for more efficient railway operations, many legacy systems are still in use and have inhibited future development of the railway system in the UK because of the incapability to incorporate new technologies; and for local railway operators, systems have to be designed individually, creating barriers for data exchange between systems (Department for Transport, 2011b). Although legacy systems should be replaced, and introducing the new technology will generate more value for investments, it is still almost impossible to replace existing legacy systems in a foreseeable future because of the cost and the risk of interrupting existing operation.

Newly designed systems, that were built to adapt to legacy systems, tend to be supplied by different IT suppliers and are only accessible within an organisation, thus more and more data silos appeared, leading to increasing complexity in data interoperation and exchange. As a result of this complexity in the UK railway industry, it is difficult to reuse most data and it is usually difficult to process proprietary formats with systems supported by different parties; thereby, a lot of potentially useful data cannot be accessed by other

systems outside an organisation. Poor integration of data has made it more difficult to allow data to be exchanged across the boundaries of organisations. A higher level of data integration will benefit the industry from aspects such as cost reduction, improved publication of travel information, smarter decision-making, more transparent and consistent data sharing and so on. The UK rail industry has acknowledged this issue and has shown great interest in improving the current data management strategy (Department for Transport, 2011b, 2018; Network Rail, 2013; Network Rail Limited, 2017a; TSLG, 2012).

Moreover, over the last two decades, the number of passengers using the UK's railway has doubled and is still gradually increasing (Network Rail Limited, 2017a), and increasing demand and popularity has led to continuously increasing investments in railway infrastructure and rolling stock. Network Rail is spending £130m every week on improvements for passengers through its Railway Upgrade Plan (Network Rail Limited, 2017b). The attention from the UK rail industry is increasingly focused on digitalisation that aims to deliver more trains, reduce crowding, provide better connections and improve performance and safety for passengers (Network Rail Limited, 2017a). A digital railway relies on smooth and flexible data transaction and usage, but the full potential of data utilisation has not currently been reached. Therefore, in 2012, Offering Rail Better Information Services (ORBIS), one of the largest rail infrastructure digital transmission projects in Europe, was launched to assist Network Rail to achieve safer, more efficient,

and more cost-effective railway operations by providing better data access to existing asset information. This project delivered several outcomes to the industry, facilitating smoother data transmission as well as the progression of digital railway; via ORBIS, effort was also put into establishing a standardised infrastructure model while semantic data models could provide a long-term solution enabling the entire industry to gain access to asset data and information (Morris, 2017; The Institution of Engineering and Technology, n.d.; Tutcher, 2015a). Network Rail thereafter proposed a future plan to improve current data usage, including the introduction of a high-level data model to facilitate seamless data exchange and transparent data sharing (Network Rail, 2013; Network Rail Limited, 2017a). Furthermore, the Department for Transport (DfT), Network Rail, the Office of Rail and Road (ORR) and the Rail Safety and Standards Board (RSSB) jointly published an up-to-date policy paper to establish a Joint Rail Data Action Plan (Department for Transport, 2018); it focuses on the following aspects:

- Data transparency – establish open and clear categories for rail datasets and identify sets that are commercially sensitive
- Data use and access – clarify the access, use and ownership of different datasets by enlarging the knowledge reserve and developing more understanding of rail data
- Data standards and quality – produce a standardised format and enhance the quality of open datasets with high level of accuracy

- Data value and principles – propagate open data across the industry and discover more value and potential thereof
- Rail culture and information/data skills – promote data and information sharing both within and outside the industry to enhance efficiency, performance and customer experience, facilitating more co-operation with other partners outside the industry and identifying the data and information skills required by the innovation

Officials believe that despite the effort made before 2018, there is still much more that could be done collectively by further collaborations between a range of rail agencies (DfT, the Rail Delivery Group (RDG), ORR, RSSB, Transport Systems Catapult, Transport for London and Transport Focus) to develop higher-quality and more open rail datasets and greater railway dataset sharing between railway participants and other parties outside the rail industry (Department for Transport, 2018). Applying Linked Data technologies is part of the effort; for example, a standardised data model and architecture could be developed to consolidate Network Rail's and RDG's open data platforms, and GPS information could be linked to a specific train to improve the accuracy of location data and predict arrival time more accurately.

In respect of standardised data models and architecture, standards exist to facilitate data exchange and manipulation, such as Railway Markup

Language (railML)[29]. railML can describe railway concepts such as infrastructure, timetable, etc. in the form of XML, and version 3 has become part of International Railway Standard (IRS) 30100 (Morris, 2017; Nash et al., 2010). However, segregated information systems still make it difficult to manipulate and query data across the industry. For instance, full track geometry is collected by unattended track geometry measurement systems running on in-service trains; it is saved in large databases every 0.2 m along the track being monitored ; but apart from sensors mounted on a rail vehicle, a lot of disparate sources of data exist, making the data not be utilised as they should be (Weston et al., 2015).

In addition, research shows that, despite the existence of data exchange standards in the rail industry, many of them remain proprietary and have been supplied for point-to-point interfaces rather than a generalised context between bespoke systems (Morris, 2017). An effort has been made to facilitate greater data sharing, for instance by the Open Rail Data[30] scheme. As part of Open Rail Data, Network Rail launched National Rail Enquiries (NRE) which published two supplementary data feeds (Durazo-Cardenas et al., 2016):

[29] Details are available at https://www.railml.org/en/. railML is an open-sourced and common data format that enables mutual railway data exchange between systems in the form of systematic XML. It can also describe railway related data.

[30] Details are available at https://wiki.openraildata.com/index.php?title=Main_Page. Open Rail Data in the UK consists of several schemes that are supported by Network Rail, RSSB, ORR, ATOC, Transport for London (TfL), Transport for Greater Manchester (TfGM), High Speed Two Ltd, the British Transport Police and OpenStreetMap.org; it publishes data that has been made available by the rail and transport industry.

- Knowledge (KB) – provides information regarding station facilities, ticket price, line status, etc. via feeds that are encoded in eXtensible Markup Language (XML);
- Online Journey Planner (OJP) – provides real-time journey planning updates and disruption updates with the aid of a combination of timetabled information, live train running information from DARWIN, customer location and ticket pricing to deliver a variety of journey planning services as an API.

DARWIN is the UK railway industry's official train running information engine; as well as being part of NRE it was also implemented later to provide an integrated data stream and updated real-time operational information to the public in combination with the ability to take feeds directly from every Train Operating Company (TOC)'s Customer Information System (CIS) (National Rail Enquiries, n.d.). The implementation of DARWIN established an integrated and consistent passenger information infrastructure to enable almost all stations across the country as well as digital devices to display coherent departure and arrival information (Rail, 2015). DARWIN has improved the level of accuracy of information presented to passengers; it also feeds information to almost all stations in the country, providing forecasts to facilitate a higher level of operation automation (Open Rail Data Wiki, n.d.).

In accordance with the comparison between TRUST and DARWIN[31], TRUST[32], a system operated by Network Rail to monitor train operation, focuses on what has already happened more (Open Rail Data Wiki, n.d.; Safety Central, n.d.). Although TRUST is much older, it is still under active development and support, contributing to better data sharing across the industry (Open Rail Data Wiki, n.d.). However, the UK industry is still seeking more measures to improve the situation. A recent government report has suggested that in order to improve means of working between the government and the rail industry and facilitate better data usage, a new data portal will be deployed alongside AI technologies (Williams Rail Review, 2019).

Admittedly, the Open Rail Data scheme has facilitated greater data sharing. It has enabled developers both within and outside the industry to gain access to both real-time and historical railway operation data. However, despite using a supplementary XMLS that helps users to understand the data they receive, many terms and jargon are not familiar to users; therefore, users have to spend time and effort to understand what data they have received and where it comes from (i.e. context), resulting in semantic heterogeneity. A lack of rich semantic information means that developers have to program and adhere to pre-defined logic. This potentially makes it more difficult to realise a higher level of automation. Apart from data heterogeneity, there are still disincentives for information sharing because of the nature of

[31] Details are available at https://wiki.openraildata.com/index.php/TRUST_vs_Darwin
[32] Details are available at https://safety.networkrail.co.uk/jargon-buster/trust/

business activities – railway stakeholders tend to just keep information to themselves. It has been suggested that companies that operate rail maintenance and reporting systems have sold relevant reports back to interested stakeholders, creating more difficulties in data sharing as sharing raw data might vastly affect their business; the most likely scenario is for data sharing to be mandated by the government (Tutcher, 2015b).

DfT specifically united a few stakeholders of the UK rail industry to make a new data usage plan to facilitate future railway improvements and upgrades. This Joint Rail Data Action Plan (Department for Transport, 2018) was later replaced by the Rail Data Council as the formal programme governance (Rail Delivery Group, 2020). Thereby, it can be noted that both the industry and the research community have realised that the current poor condition of data integration across the industry might impede progress in railway development, hence being dedicated to improving data interoperability. For example, in order to fulfil growing demand for quality and transparency of information while ensuring the stability and reliability of the network, the Technical Strategy Leadership Group has suggested exploiting new technologies, such as common data architectures and protocols that enable information sharing between Safety Management Intelligence System (SMIS), and other industry information systems such as Network Rail asset databases (aligning with the Information theme in the Rail Technical Strategy) (The Technical Strategy Leadership Group, 2017).

It is worth mentioning that an XML-based data exchange system might work well within an organisation but be problematic for exchanging data between organisations. For instance, the imperial units (e.g., miles and chains) are predominantly used in the UK to describe track mileage whereas metric units are more often used in European countries. Therefore, when describing a cross-channel service, data being transmitted from the UK side might be wrongly interpreted by the continental European side if communication has not been properly established; such a case has caused serious consequence in the past (Easton et al., 2010). To date, there is no evidence stating that such semantic heterogeneity has been addressed, although it is notoriously famous for its negative impacts on adding complexity to large complex systems (Cruz and Xiao, 2005).

2.3.2 Ontology and Data Integration

Information might be stored in distributed databases or in files, spreadsheets, etc. in large organisations, leading to incomplete, inaccurate and inconsistent data retrieval and processing with increasing but unnecessary complexity, cost and effort, eventually causing difficulty with data interoperation (Parent and Spaccapietra, 2000). As a result, data heterogeneity persists in the UK railway industry.

It is necessary to integrate heterogeneous data because of the difficulty of data interoperation as data sources might use disparate syntax, schema and semantics (Bishr, 1998). Cruz and Xiao (2005) investigated different types of

data heterogeneity and causes thereof, which are shown in Table 7; they also proposed a solution for tackling syntactic, schematic, and semantic heterogeneity by integrating semantic data using ontologies.

TABLE 7 DIFFERENT TYPES OF HETEROGENEITY (CRUZ AND XIAO, 2005)

Type	Cause
Syntactic heterogeneity	Use different models or languages
Schematic heterogeneity	Structural differences
Semantic heterogeneity	Different meanings of interpretations of data in various contexts

The fragmentation of rail data leads to difficulties in transforming data into knowledge, thus the need for a knowledge model to integrate data; ontology is a decent candidate (Chang, 2008; Tutcher, 2015b; Tutcher et al., 2017). Ontologies can be applied in many scenarios, data integration being one of the earliest applications identified (Siegel and Madnick, 1991). Collet et al. (1991) initially identified that a common metadata vocabulary would be the most useful basis for the context in systems with multiple databases; thus, it would be beneficial to develop knowledge bases to integrate information sources, to facilitate smooth access and coherent modification across multiple databases. Siegel and Madnick (1991) identified that it would be impossible to understand the meaning of all information while remaining in the current context as a result of the increased scope of data, hence the great

importance of integrating disparate databases and context knowledge with metadata. In order to so, it is vital to represent and manipulate the context using semantic knowledge and common vocabularies to resolve conflicts; they stated that ontology could be applied to establish knowledge bases where '*component systems must provide semantic mappings to that ontology*' (Siegel and Madnick, 1991).

Given the fact that ontologies capture essential relationships between concepts, it is possible to commence automatic reasoning about data (Knowledge Hub, 2017). With reasoners (Clark et al., 2011; Tsarkov and Horrocks, 2006), ontologies can infer new facts based on the existing model in the way a human might (e.g. two men have the same father so that they are brothers) (Wei, 2018). Reasoning enables more automation and more advanced operation in various systems, such as decision-making systems and knowledge management systems (Ashburner et al., 2000; Tutcher, 2015b). Such a trait enables automatic reasoning about data, which allows less programming logic but more data logic, hence a decrease in software size (Chandrasekaran et al., 1999; Guarino et al., 2009). In addition to reasoning, ontologies are highly structured and flexible, which allows easy coherent navigation between concepts and comprehensive representation of any data (Knowledge Hub, 2017); this enables smoother data integration regardless of the origin and format of the data, decreasing the difficulty of tasks such as data integration data mining, data analysis and knowledge management. As a result, it is possible to integrate a range of heterogeneous

data into an ontology or a set of ontologies, making the heterogeneous data accessible across the domain (Ebrahimipour and Yacout, 2015). As found in Chapter 2, ontologies have been widely used in a range of industries, such as biosciences (Ashburner et al., 2000) and the oil and gas industry, etc. (Ebrahimipour and Yacout, 2015; Leal, 2005). Previous studies and projects for the railway system in Europe, such as InteGRail, IT2Rail, C4R, ONTIME and the development of RaCoOn, have demonstrated the usefulness of ontologies which can be seen as tools for data integration in the railway industry, benefiting not only ICT systems but also the application of new technologies to legacy systems. Ontologies can also include semantic rules which allow computer systems to infer new facts based on existing facts (triples), which enables better domain knowledge management and decision-making. In the study of RaCoOn, it has also been systematically discussed how an ontology can be modelled to target the railway system (Tutcher, 2015a) and how to use ontologies to realise data integration to complement existing condition monitoring systems (Morris, 2017). Meanwhile, there are commercial solutions available for data integration in the rail system, too. For instance, ERTMS Solutions has proposed railway IT integration solutions (i.e. ODASE platform) to eliminate data stored in siloed sub-subsystems which cannot exchange information with each other (ERTMS Solutions, n.d.). Thus, it can be noted that using ontologies to integrate data for the railway system has become one of the key priorities for both the research community and industry.

Using ontologies can assist coherent and consistent data manipulation across multiple databases, benefiting large complex systems. Introducing an Enterprise Service Bus (ESB) can facilitate data manipulation across databases in the domain (Schmidt et al., 2010); however, despite ESBs, a lack of semantic mappings can still result in ambiguities and confusions between ESBs maintained by different organisations or departments, as shown in Fig. 17.

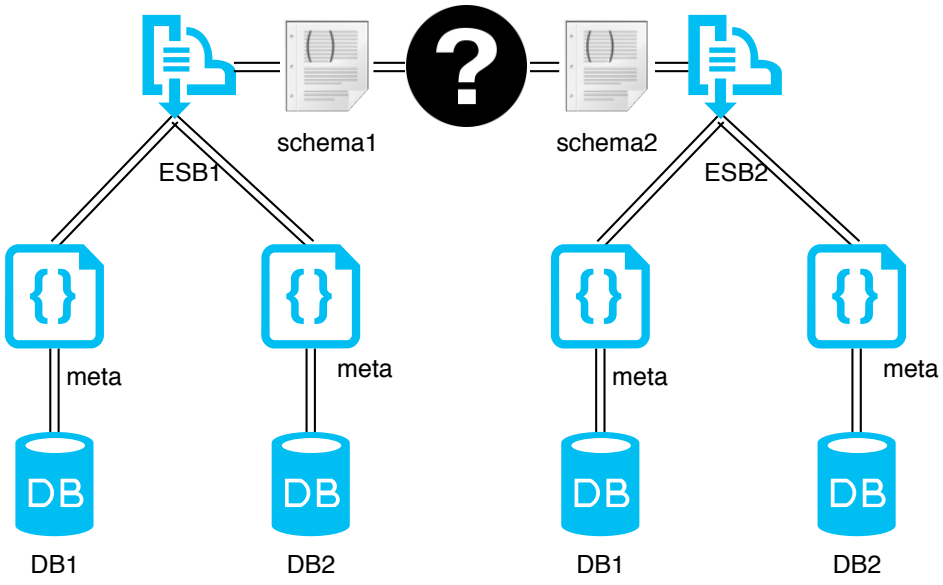


FIG. 17 A LACK OF SEMANTIC MAPPING RESULTS IN AMBIGUITIES

Meanwhile, technologies applied in the Semantic Web enable data to be linked, bringing a broader view to its users, while ontologies, as the backbone of the Semantic Web and Linked Data, have continuously gained significant attention to be used to integrate heterogeneous data along with the development of Web technologies (Brochhausen et al., 2011; Guarino, 1995,

1998; Lamy, 2016; Liu et al., 2011; Lodemann and Luttenberger, 2010; Tutcher et al., 2017; Udrea et al., 2007; Verstichel et al., 2011; Wei, 2018; Xiaomeng et al., 2015). An ontology can explicitly define the concepts and properties thereof (Jacquette, 2014). Entities and their relationships can be described by a set of classes and properties, i.e. vocabularies, with RDF and OWL which also provide a high level of expressivity in the modelling domains of interest (Bizer et al., 2009).

Data exists in many forms. For example, a value of '1' can represent an integer value of one or a Boolean value true. When such a value is stored in a database, it might create confusion when performing data retrieval and modification, especially when IT systems are maintained by multiple parties. Traditionally, data has been stored without a context, which adds difficulties while retrieving and changing data. Technologies applied in the Semantic Web enable users to have a broader view of the data and, as the underlying back-end. Therefore, there are reasons to believe that users can benefit from a flexible, unified, structured and meaningful data description framework that is backed by an ontology or a set of ontologies with rich semantics (Guarino et al., 2009). Flexible and contextual annotation ensures context-awareness in data analysis and integration (Kalibatiene and Vasilecas, 2011).

As a conclusion, an ontology stores the concept per se, in other words, not only the data but also the information and context where the data is generated and described. On top of that, with given rules, ontologies can be

referred to, to allow machines to infer new facts based on existing facts they contain, being capable of modelling basic human logic to enable machines to 'comprehend' our knowledge in a digital way. Ontologies also enable advanced semantic representation in a way in which concepts can be refined, to the extent where machines can understand the exact 'meaning' behind given semantics, eliminating ambiguity and enhancing the user experience. It has also been suggested that inference also facilitates data integration by making data more explicit, hence realising simplified information systems with reduced programming logic while interpreting data (Tutcher et al., 2017). Thus, ontologies have been the focus of research into linking heterogeneous data (Bodenreider, 2008; Tutcher et al., 2017; Udrea et al., 2007; Umiliacchi et al., 2009; Verstichel et al., 2011; Xiaomeng et al., 2015).

2.3.3 Ontology-Based Data Integration in the Railway Industry

Despite the existence of non-ontology-based data integration platforms, such as ORBIS, the problem of resolving data obscurity and heterogeneity remains; ontology can resolve this issue. Semantic data integration with ontologies has been demonstrated as useful in other domains, such as the oil and gas industry (Ebrahimipour and Yacout, 2015), biosciences (Bodenreider, 2008) and so forth. Therefore, research has been conducted into the use of ontologies for integrating heterogeneous data in Europe for railway operation, too (Capacity for Rail, 2017; Easton et al., 2010; Köpf, 2010; Morris, 2017; Morris et al., 2014; TSLG, 2012; Tutcher, 2015b; Umiliacchi et al., 2009).

The European project InteGRail was launched in 2005 to deliver RDO on a case study basis, developing the concepts of Intelligent Maintenance, Monitoring and Decision Support with lower cost; the project was completed in 2009 and in its final report, the rationale for RDO was specifically elaborated (Köpf, 2010):

> *The Railway system is very complex and produces continuously huge quantities of data, most in proprietary formats, which are difficult to understand, elaborate and share. As a consequence, most data are archived for "future use" and never looked at, unless a specific need occurs. Vice versa, a lot of useful information could be extracted from available data, if this could be effective (bring to good results) and easily feasible (at low cost).* (Köpf, 2010)

A lack of data understanding and reuse can make the transformation from data to information more difficult. InteGRail addressed this problem and gave a feasible solution: developing a standardised knowledge model (i.e. RDO) in OWL to refine existing data so that computers can perform automated data extraction and unambiguous data-to-information transformation, eventually facilitating semantical information interchange and sharing in a scenario where multiple stakeholders and participants form more complex railway systems with huge heterogeneity and numerous data silos.

A generic information exchange format can be beneficial because it could reduce the associated difficulties brought about by isolated data storage and ambiguous data interpretation; the use of ESB in the rail section was also proposed, which was believed to be essential to software engineering in railway applications (Köpf, 2010). InteGRail addressed the gap between railway infrastructure manager and operators, thus the proposal of an ontology-based data sharing solution to establish an IT infrastructure enabling the railway system to be managed as a single system. Researchers working on this project used Database to Relational (D2R) tools to map existing data to the Linked Data format that was proposed in the InteGRail report then produced a real-time and consistently updated copy of existing data which is available and accessible to other systems; the benefit of this approach is that legacy systems, which might be relied on by other systems, can be kept as they are but the performance has to be compromised in order to produce a copy of the data in the format proposed in a real-time manner (Spanos et al., 2012). Consequently, the solution proposed by InteGRail is suitable for non-time-critical tasks, such as Network Statement Checker (Köpf, 2010; Morris, 2017). Despite its inferior performance, it does not mean that the solution proposed by InteGRail is inferior; data interoperability, adaptability, consistency and transparency are of importance not only to modern railway systems but also for future railway systems (Network Rail, 2013) so it is cogent that the ontology-based approach proposed by InteGRail would be beneficial to railway operation.

Similar to InteGRail, the more recent project IT2Rail found that using ontologies will save costs and improve overall performance; ontology-based data integration could also enable more intelligent ways of introducing new applications into legacy systems as well as improved decision-making, predictive maintenance and smart fault diagnosis (Gogos and Letellier, 2016). Both projects intend to achieve the architecture illustrated in Fig. 18.

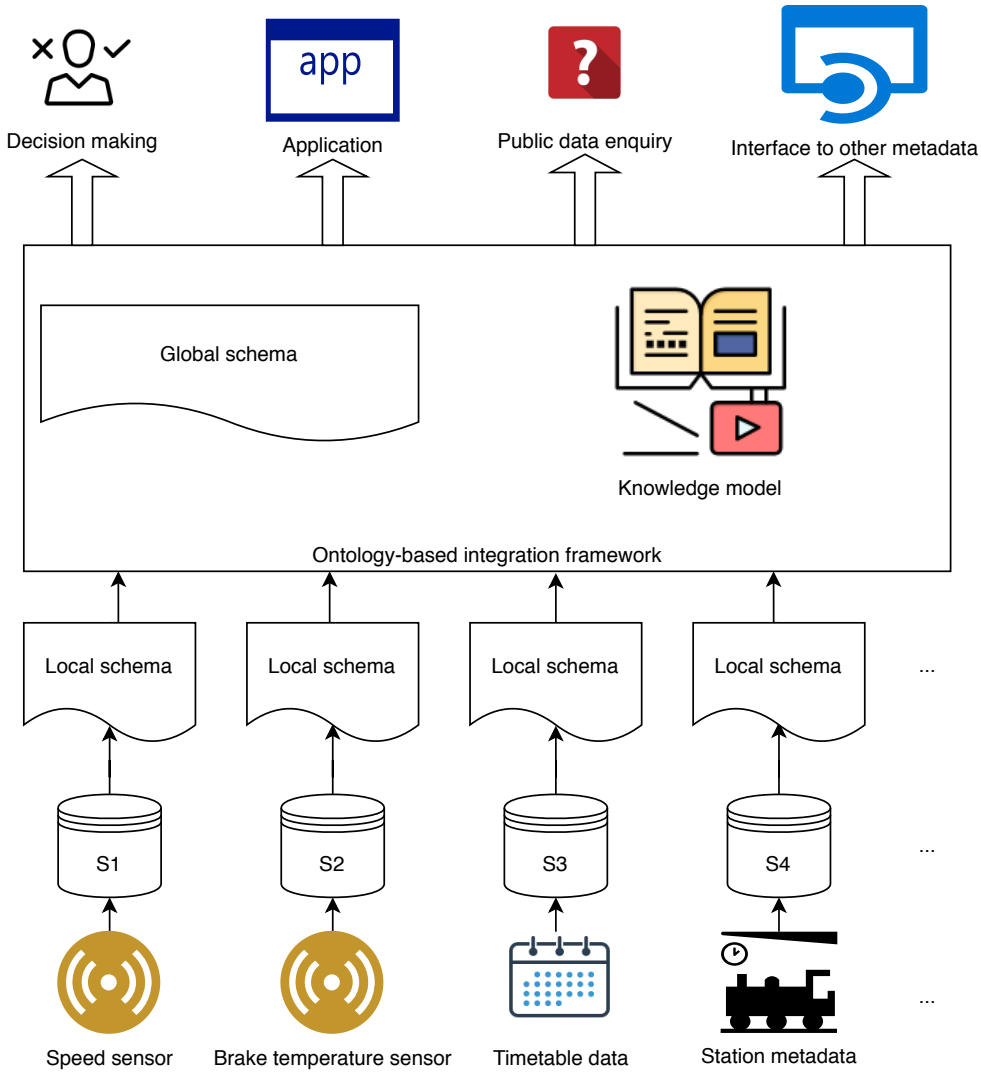


FIG. 18 ARCHITECTURE ENVISAGED BY RESEARCHERS

Another collaborative project, Capacity for Rail (C4R), noticed the vast importance of a sharable and standard data model for the railway operation, too; participants of this project reckoned that a lack of data exchange and management practices between stakeholders has made the rail industry fall behind other large-scale infrastructure industries, e.g. the oil and gas industry (Capacity for Rail, 2017; Technische Universität Dresden et al., 2016). In order to increase the capacity, C4R Deliverable 3.4.1 suggested that it is necessary to address the problem where real-time operational data lacks cross-border support, that is to say, data format and granularity are not standardised, hence the difficulty in the data exchange and sharing process; ubiquitous data also requires further integration with a standardised model, not only integrating data within the rail industry but also data from other transportation modes (Technische Universität Dresden et al., 2016). In the following Deliverable 4.3.2, it was suggested that such a model should be a semantic model; an ontology-based data model could fit such a purpose (Capacity for Rail, 2017). The reason why a semantic model is vital is because information loss due to semantic heterogeneity is common as a result of the existence of data silos. The report presented the example of Network Rail's Corpus database which provides a list of location codes (i.e. STANOX, TIPLOC and NLC codes) whereas geographical coordinates are provided with reference to Timing Point Locations (TIPLOC) codes by the National Public Transport Access Nodes (NaPTAN) database; they essentially refer to the same concept (e.g., same location) but are presented with different codes,

as shown in Fig. 19 (Capacity for Rail, 2017). In this example, it can be seen that despite the same concepts, the two databases use different means to describe them; such description is problematic, for instance, when a STANOX code is given: it has to retrieve the assigned TIPLOC code thereof from Network Rail's Corpus database first, then use the TIPLOC code to query the corresponding coordinates against NaPTAN's database. Semantic integration establishes a linkage between ICT systems, open data resources, and TMS- and non-TMS-related railway systems, as illustrated in Fig. 20.

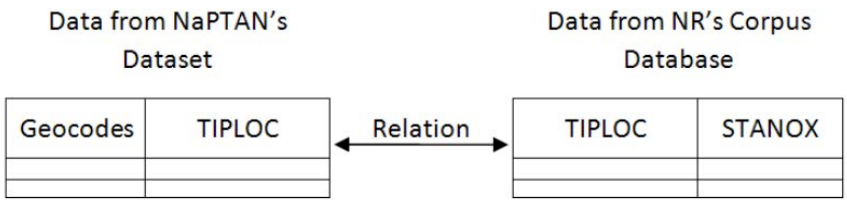


FIG. 19 DATA FROM NAPTAN'S DATABASE IS RELATED TO DATA FROM NETWORK RAIL'S DATABASE

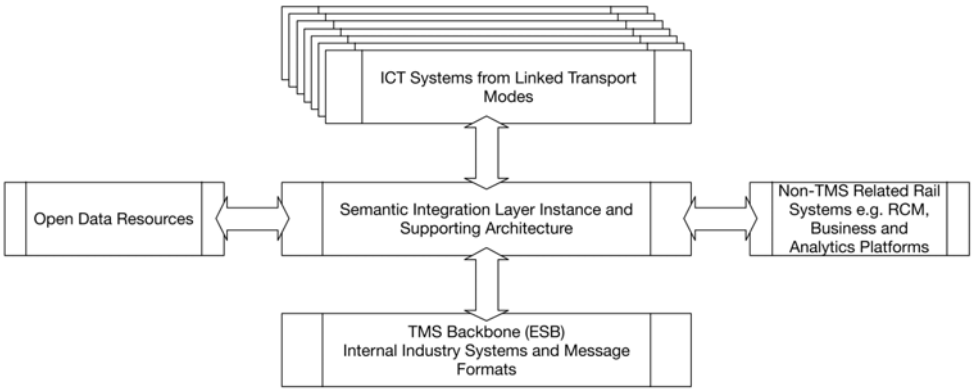


FIG. 20 DATA INTEGRATION STRUCTURE AROUND A SEMANTIC INTEGRATION LAYER (CAPACITY FOR RAIL, 2017)

C4R demonstrated that the whole industry is moving towards a more integrated and digitalised railway operation with ESB models, elaborating on the RaCoOn model which decouples software applications from the data to

enable generic data interpretation and interlink disparate data sources. C4R also suggested and proved the rationale of using ontologies to establish ESB architecture in the UK railway industry: the utilisation of open data from external public sources to support railway operation, such as situational awareness during disruptions and decision-making (Capacity for Rail, 2017).

RaCoOn was developed and tailored to describe general concepts and data thereof for the railway, establishing a systematic and reusable method to provide machine-interpretable conceptualisation of part of the railway domain (Tutcher et al., 2017). The initial study of RaCoOn (Tutcher, 2015b) focused on describing railway infrastructure and signalling with reference to railML and ISO 15926. It was then modularised and extended to a set of ontologies to represent concepts in depth for railway sub-domains including timetables, rolling stock and infrastructure, as well as cross-domain support. The whole model was arranged in three layers hierarchically, as illustrated in Fig. 21 (Morris, 2017); the top layer incorporates cross-domain support and fundamental vocabularies while the middle layer is responsible for representing core concepts of railway systems in the UK and the bottom layer is used to describe the specific division of railway system, for example, infrastructure. This structure is the same as the hierarchy depicted in Fig. 9. The cross-domain ontologies can be seen as upper ontologies that provide general concepts that are commonly referred in other domains, too, such as place; the core ontologies can be seen as a domain ontology which provides

general railway-related concepts; task ontologies are specifically tailored to describe timetables, rolling stock and infrastructure, respectively.

It is worth noting that constraints are parts of RaCoOn; they are used to verify the validity and consistency of captured data (Tutcher, 2015a). Such verification is of vast importance in large complex systems to ensure only healthy data is being processed, which is beneficial for preventing databases being corrupted.

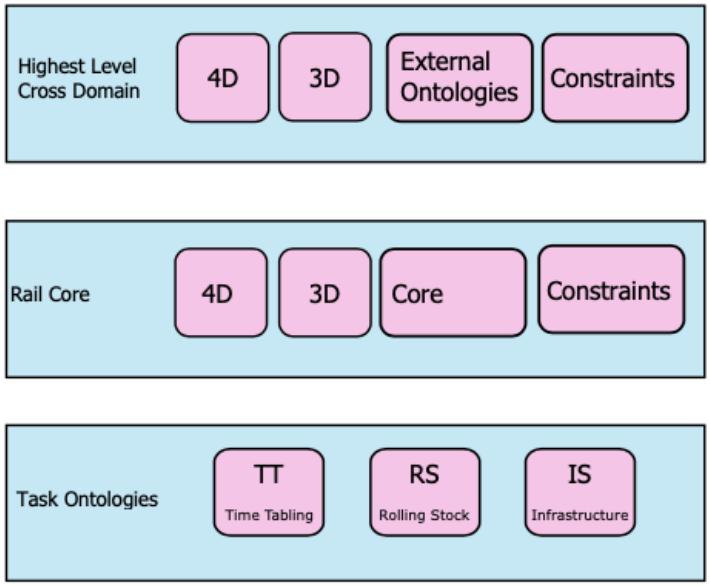


FIG. 21 HIERARCHY OF RAIL CORE ONTOLOGIES (MORRIS, 2017)

RaCoOn was modelled with reference to different observations about the persistence of entities through time (Tutcher, 2015a), thus the existence of 3D and 4D ontologies in RaCoOn. 3D and 4D ontologies represent objects in three and four dimensions, respectively. The temporal description forms the fourth dimension in comparison to 3D ontologies; in other words, 3D

ontologies represent individual entities with only spatial parts and exist at each moment of their existence, whereas 4D ontologies describe objects with both spatial and temporal concepts regardless of space–time (Hales and Johnson, 2003; Verdonck et al., 2014). This is the reflection of two philosophical concepts – endurantism and perdurantism. Hales and Johnson (2003) explained the terms:

- **Endurantism** is defined as '*objects have three spatial dimensions and move through time, wholly presenting at each time at which they exist, that is to say, objects here now will be here now*'. For example, a person will be always the same person regardless of time although his/her properties (e.g., age, hairstyle, job, education background and wealth) might be changed.
- **Perdurantism** is contrary to endurantism, that is '*objects are not wholly present at each time at which they exist, which are composed of temporal parts, that is to say, objects here now might be only partially perceived whereas other parts might have been or not yet encountered*'. For example, a person's education experience can be seen as the aggregate all temporal parts, e.g., from primary school to university.

Hales and Johnson (2003) systematically discussed these two main theories about the persistence of objects through time from a philosophers' perspective: that both are correct, just different views to observe objects. However,

when it comes to data modelling, most data models are synchronic (i.e., they model the data at one instant), which means that there is no support for representing data that changes over time; such models work well in most scenarios. However, exceptions exist (Tutcher, 2015a). For instance, version control is required to track changes of data; while most version control systems do so by storing historical states of the system at a fixed interval, i.e., a frozen moment, which is synchronic, in a railway system, the condition monitoring system needs to track certain data and changes thereof over certain time, i.e., the aggregate of all temporal parts of certain data, which is diachronic. Therefore, there is still a necessity for corresponding data models that are capable of describing diachronic data explicitly, plus a need to represent diachronic data during railway operations, hence the existence of the 4D version of RaCoOn.

The introduction of the 4D and 3D versions enables RaCoOn to be compatible with different scenarios. In a productive environment, it depends on which file is being imported to the ontology models. Although the 4D version is more explicit for representing temporal parts of data, the 3D version is more lightweight with fewer entities (i.e., smaller size while representing data), thus being suitable for working with large chunk of data; the 3D version is especially useful when using reasoners to infer results as the cost of computation can be increased dramatically together with the proliferation of triples being fed to the reasoner. Yet, although the 3D version might work perfectly within a single system, when exchanging ontology models between

systems, loss of the historical changes of data is inevitable if the 3D version is used; therefore, the 4D version is more suitable when it is necessary to track the history of data and model exchange between systems (Morris et al., 2015).

An important part of RaCoOn worth mentioning is the separation of the T-Box and A-Box. Bergman (2009) analysed the difference between a T-Box (Terminological Knowledge Box) and an A-Box (Assertions Box): the T-Box is similar to a schema or taxonomy that represents a domain whereas the A-Box contains assertions about individuals with relevance to their classes plus the attributes of instances and relationships between them. Both T-Box and A-Box are subject to Set Theory. The separation of T-Box and A-Box is of importance, making it easier to handle and reuse data (i.e. instances), making the conceptual model (i.e. T-Box) as simple and expressive as possible so that it becomes easier to change, map or interlink separate ontologies rather than mix them, and making evaluations for both data and conceptual models easier and faster, which is particularly useful in mass data processing (Bergman, 2009). On the other hand, the maintenance of ontologies can become very complex owing to their size at some point during their lifecycles; splitting the T-Box and A-Box can reduce such complexity. RaCoOn is an ably compliant principle for decreasing computational cost and complexity while ensuring a high level of expressivity. Morris et al. (2015) demonstrated an example of how RaCoOn achieves this: using different reasoners to infer

facts separated at different levels of complexity (e.g. OWL-DL and RDFS) based on parts of RaCoOn.

These traits and applicability of RaCoOn in the UK rail industry have been discussed and demonstrated in a report proving that RaCoOn can be applied to the rail industry in the UK to help to facilitate better data integration and usage. The application of RaCoOn can protect software systems from changes to physical systems in the real world, hence reducing development costs and improving overall system stability. When processing a large volume of raw data, the researchers stored raw data separately, only keys to it being mapped to RaCoOn; this enables relevant services to extract data only when it is required, avoiding the waste of computational resources (Tutcher et al., 2014). In a RaCoOn-based train demonstrator, the same group of researchers demonstrated how 4D ontology helps the system (Tutcher, 2015a; Tutcher et al., 2014): it provides a systematic method for tracking the history of a train with detailed and integrated information; in conjunction with ontological reasoning, it becomes easier to infer a train's location. There are also other applications that can be achieved with RaCoOn (Morris et al., 2014):

- **Forward planning:** As mentioned in the previous text, data is stored in many silos. Data silos make it more difficult to make decisions. Ontologies can bring data as a whole together to facilitate the decision-making process.

- **Maintenance timing:** well-timed maintenance can save money, which is always being sought by the infrastructure manager. In order to do so, operational data must be extracted from the actual network on time. As a result of heterogeneity in the network, it is not always approachable. Using RaCoOn to represent the physical asset and describe its data to make the data available to analysis systems can facilitate data retrieval and integration. Together with reasoners, more automation can also be achieved.
- **Train identification:** usually, most trackside sensors capture data with timestamps, which might not be problematic when the amount of data is small. On the other hand, data analysis is carried out by TOC maintenance departments, requiring manual data retrieval from the infrastructure manager to identify the train with issues. The proliferation of data and the existence of manual processes lead to cost wastage on data collection and integrations as well as the time used in the identification of problematic trains, eventually impacting efficiency. Ontologies make data have contexts so that there is always a way to trace back where the data revealing a problem is from and when it was generated from the trackside sensors. In this way, professionals can have faster access to problematic trains.
- **Predictive maintenance:** Umiliacchi et al. (2011) have demonstrated that predicted maintenance is the most cost-effective way to perform railway maintenance in comparison to planned maintenance

and condition-based maintenance. Planned maintenance means assets are visited and fixed at a fixed interval regardless of their condition whereas predictive maintenance is the contrary: the repair job is predicted by a computer system based on data generated from a series of sensors. The readings from those sensors are fed to a mathematical model to be analysed to determine whether the maintenance is necessary. However, underlying data with regards to maintenance could be spread everywhere across the industry; therefore, many predictive maintenance methods are based on an asset by asset, system by system policy. When a new asset is introduced, development of a new predictive system is likely required, with little reuse of previous systems. A lack of means to describe physical assets has increased the efficiency and cost whereas applying RaCoOn could implement a domain-wide system with general support for describing physical assets in order to bring relevant information together when it is required.

- **Customer information:** Although the DARWIN data feed provides real-time arrival and departure data to almost all stations across the country as well as Web services as mentioned in section 2.3.1, the integration of GPS units and electrified track circuits could improve DARWIN's accuracy (ON-TIME, 2013). RaCoOn can bring different types of data together in a standardised way to represent them. In addition, when a multimodal journey is being planned, information

from different transportation modules should be considered altogether; here, data integration realised by RaCoOn is beneficial.

The same report also proposed a supplementary architecture to support ubiquitous railway data processing for TMS with RaCoOn; it incorporates a high-velocity database as a cache to ease the pressure on the semantic database, as shown in Fig. 22. This architecture integrates data that is external to TMS, feeding up-to-date data directly to TMS to facilitate the decision-making process and, most importantly, it establishes an ICT infrastructure that ensures data consistency and accuracy.

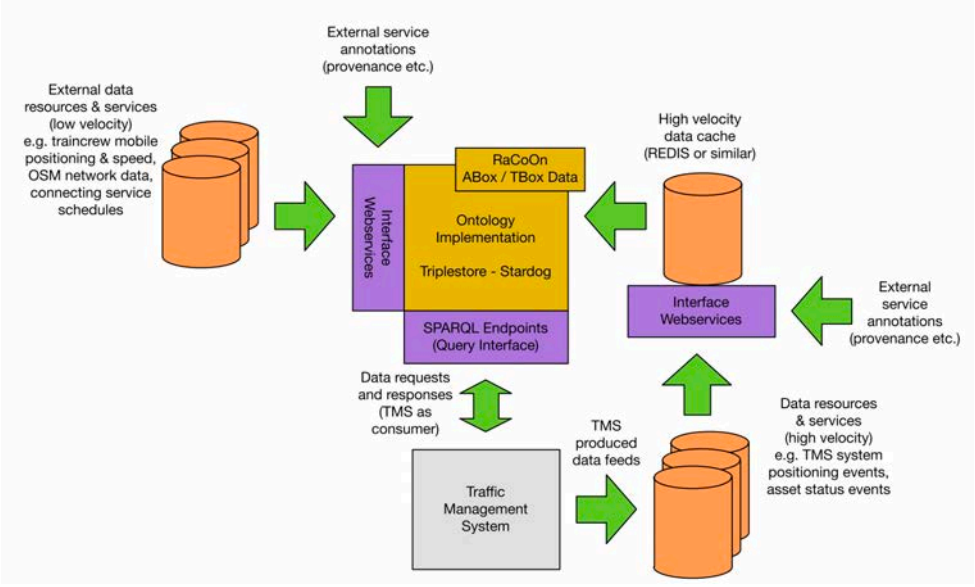


FIG. 22 SYSTEM ARCHITECTURE FOR ONTOLOGY-BASED UBIQUITOUS DATA PROCESSING (CAPACITY FOR RAIL, 2017)

In conclusion, using ontologies is beneficial to railway systems (Capacity for Rail, 2017; Gogos and Letellier, 2016; Köpf, 2010; Technische Universität Dresden et al., 2016), although it might depend on whether stakeholders are

willing to fund and commit to both technological and organisational changes across the industry (Köpf, 2010), especially while some benefits, such as cost reduction, might not be presented to some participants (Gogos and Letellier, 2016). Based on research, poor integration between ICT systems can cost 1%–2% of annual revenue (i.e., approximately £82 million to £164 million); the additional cost can be saved with continuous effort to improve the data management and integration strategy. It has been suggested that integrated data resources will have a more critical role in the railway industry in the next 5 years; this could be achieved by applying new technologies, including but not limited to ontology (Capacity for Rail, 2017).

However, despite the attention and the response from the industry and the research community, plus proven benefits brought by ontologies and their increasing popularity, there is still no evidence indicating that there is an ontology-based integration solution, not to mention an ontology-based system that has been applied to the railway system. There has not been any investigation with respect to factors that hinder the uptake of ontologies for the UK railway industry, which still requires investigation. Moreover, although many studies have produced ontology models and demonstrators (Gogos and Letellier, 2016; Köpf, 2010; Tutcher et al., 2014, 2017), none of them discussed the scalability of an ontology-based data processing system, and there is no discussion of how to enable people who are not familiar with ontologies to interact with them. Using ontologies to develop a system is still a specialist task, thus the need for a generic software solution to interact

with ontologies and the rules thereof. Moreover, as discussed in Chapter 2, ontology-based integration is beneficial to the railway industry but using ontologies to manage unstructured data and reproduce existing manual data processes in order to achieve a higher level of digitalisation and efficiency needs further investigation.

2.4 Using the Triple Store – Why Not Use a Relational Database?

First, it is clear that databases are used to make real-world information available to computers, providing a reservoir to store data. Usually, a data model has to be provided in the beginning in order to enlighten computers about 'what is stored'. An example is a relational data model.

The relational data model, ever since it was proposed by Codd in 1970, has become a 'standard' approach to managing data by representing information and its supporting database, the relational database, has been the priority choice for application owing to its ability to effectively manage a large volume of data (Martinez-Cruz et al., 2012). An example of a relational model is illustrated in Fig. 23 (Paredaens et al., 1989).

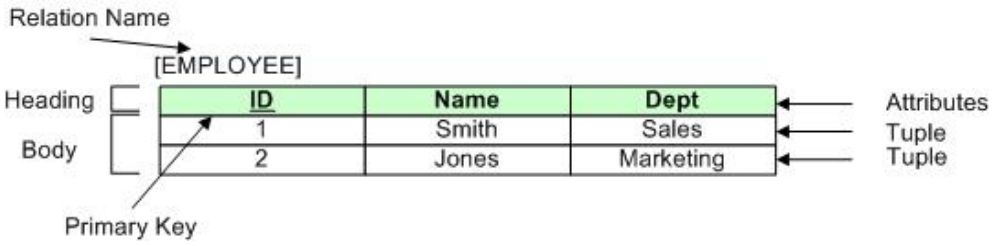


FIG. 23 DATA STRUCTURE OF A RELATIONAL MODEL

It can be seen that in a relational model, data is presented in tuples with reference to the relation headings, while the headings are attributes of the relation name (i.e., the name of the data table). The ability to reference attributes (i.e., columns in the opinion of database software) by name and tuples (i.e., rows) with the primary key enables data organisation without the requirement for a physical storage model (Paredaens et al., 1989). With relational models, as long as the table and its primary key are provided to the machine, the machine can locate the specific row(s) on which to perform data extraction operation based on the value(s) of the primary key. In other words, it is not essential to make the computer understand how data is organised conceptually, unlike from a human's point of view. Therefore, relational models have successfully separated human concepts and data storage logic (Martinez-Cruz et al., 2012).

However, despite being the most popular way to manage data, relational models and their storage provider, i.e. relational databases or SQL databases, still have problems (Gyorodi et al., 2015). Two of the most notable problems with the relational model are the worse support for the type of system (Taylor et al., 1989) and a higher level of data fragmentation (Gray, 1997).

This makes it very difficult to manage data across multiple data tables, not to mention across different databases, which can be expensive. As a result of data fragmentation, it is not hard to realise that when the data schema changes, it can be very complex to ensure the new schema has been applied properly to all tables. Admittedly, relational databases are highly suitable for storing data that can be tabulated with a fixed model (schema); yet, in today's context, where data is highly connected, relational models cannot hold relationships between concepts or data, and a lack of support for rich semantics also creates confusion sometimes when the volume of the data is large (Neo4J, 2018).

Ontologies, on the other hand, solve the aforesaid problems, providing rich semantics and the ability to model structure of concepts, eliminating ambiguity and removing the barrier between tables, respectively (Jacquette, 2014). Ontologies are independent of the implementation so that they can be operated at a high level of abstraction (Dillon et al., 2008). Ontology models work similarly to relational models, that is independent of the storage and, owing to their ability to describe the relationship between entities, ontology models have become great candidates to conceptualise the information.

The example illustrated in Fig. 23 can be represented in a graph rendered based on a simple ontology model, as shown in Fig. 24. It can be seen that in the ontology model, there is no 'primary key' and the headings, to some

extent, can be seen as the ‘predicate’ in terms of triple structure. Instances are presented not in the form of tuples but with relations (i.e., predicates). This enables further inference and extensibility. For example, if departments, sales and marketing, are modelled as instances of the concept ‘Department’, it is possible to use department to locate an employee instead of locating the table and using a primary key to locate the employee. This benefit might seem trivial, but in the context of Big Data, this can greatly save computational costs. Meanwhile, it is possible to extend the model to incorporate more ‘tables’ (i.e., classes) to realise data integration.

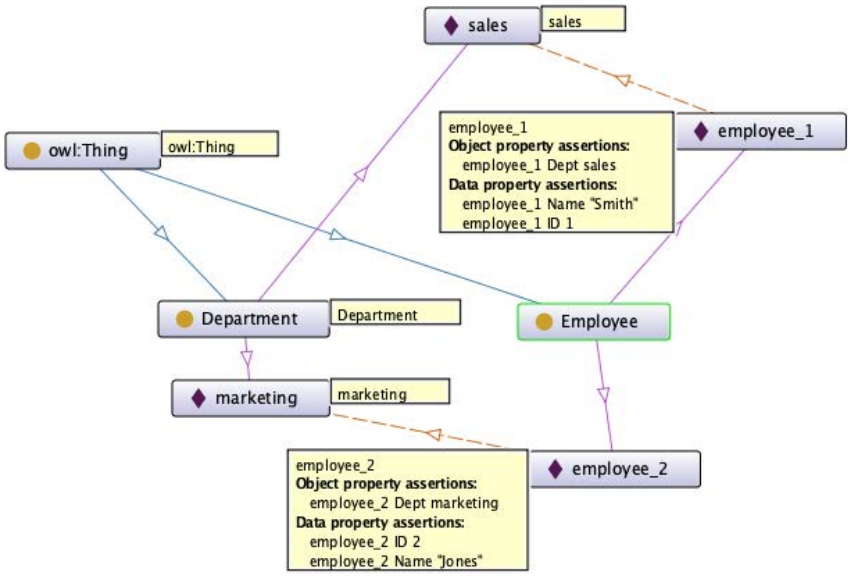


FIG. 24 EXAMPLE ILLUSTRATED IN FIG. 23 PRESENTED IN THE FORM OF AN ONTOLOGY

Railway data is often highly connected in the operational context, relating to many aspects such as maintenance, risk assessment and condition monitoring (Umiliacchi et al., 2009). It is necessary to adopt ontologies in order

to have a more holistic view of the whole system; as such, using ontologies to establish a standardised framework to facilitate rail data analysis could be beneficial.

3 RESEARCH METHODOLOGY

This chapter will introduce the methodology of this study and the research approach. Research philosophy is the key to establishing an appropriate approach for the research, which is defined *as a system of beliefs and assumptions about the development of knowledge* (Saunders, Lewis, & Thornhill, 2009). There are several different kinds of research philosophy. Amid them, pragmatism aims to deliver practical solutions that might underlie future practice for a problem, normally putting emphasis on practical solutions (Saunders, Lewis, & Thornhill, 2009). Pragmatism research is normally initiated by sensing the wrongness, driving the reflective process of inquiry and eventually re-forming the belief when the problem has been resolved (Elkjaer and Simpson, 2011), hence being a suitable research philosophy underlying this research. Meanwhile, it has been concluded that research into information systems might involve social and natural sciences engineering and management research, which is often related to practical applications (Alfaries, 2010). Combining the nature of the aim of this study, pragmatism underpinned and guided the construction and execution of this research.

This thesis investigates the hypothesis proposed in Chapter 1 and the problems pertaining to the hypothesis in Chapter 2. In order to complete this task, Chapter 4 was planned to understand factors that discourage further development of ontologies first before proceeding to discussions of future development directions. Through existing literature, it was possible to establish a

hypothesis and a survey could be conducted to justify and validate the hypothesis in Chapter 4. Mixed research was adopted because both qualitative and quantitative data were necessary to conduct the study. Reasons are as listed based on existing literature:

- Significant value could be gained from qualitative research while investigating one's attitudes and behaviour (Hammarberg, Kirkman & De Lacey, 2016). Kothari (2004) stated that qualitative research is of great importance in behavioural study and valuable to discover the underlying motives. Therefore, it is helpful to apply qualitative research to identify and validate rail professionals' attitudes towards ontology-based solutions in the UK railway industry.
- Quantified data is necessary to justify the result concluded subjective attitudes to enhance the credibility of the research. According to Saunders, Lewis and Thornhill (2009), the Internet questionnaire is one of the data collection techniques most used in social research to obtain quantifiable data.

The remaining parts of the research used practical solutions to demonstrate the potential future practice of ontology-based solutions in the UK railway industry, adapted to action research methodology that focuses on practical improvements (Avison et. al, 1999). The design of the software architecture and flow were empirical conclusions based on existing literature and knowledge, which can be an empirical study (Benbasat and Zmud, 1999).

This research was underpinned by pragmatism. It used a mixed research approach to identify factors that might discourage railway professionals from using ontologies first; subsequently, action and empirical research was applied to this research to address practical issues and demonstrate solutions for future practice towards greater adoption of ontology-based solutions in the UK railway industry.

As a conclusion, the research was started with an investigation into factors that hinder further adoption using the mixed research method. By analysing both qualitative and quantitative data, it was possible to conduct a survey to conclude these factors and validate them. Based on the identifications and existing literature, the research thereby identified some possible solutions and demonstrated how they contributed to those factors from different aspects using action and empirical research methodology. Ultimately, the demonstrations intended fill in the gap between theories of existing ontology-based techniques and practical applications in the UK railway industry, targeting to bring more insights into ontology-based applications for the UK railway industry.

4 Investigation of Deterrents to Ontology-Based Applications

4.1 Background

In Chapter 2, it was seen that it is beneficial to use ontologies in large complex systems such as railway operations. Heterogeneous data and data silos hinder collaboration between departments or organisations, creating difficulties for data sharing and reuse, especially when many stakeholders are involved (Verstichel et al., 2011). Dill (2019) discussed the issues brought about by heterogeneous and unstructured data; despite the great value the data might have, increasing the size of data makes it difficult to analyse due to its heterogeneity, which has also created various problems such as the increased cost of data cleaning and filtering, and a difficult and complex data retrieval process. W3C has published several ontologies to facilitate data integration, such as Semantic Sensor Network Ontology (Compton et al., 2012; Haller et al., 2017), the Organisation Ontology (World Wide Web Consortium, 2014), W3C Geographical Ontologies (Lieberman et al., 2007), etc., which have also been adapted to implement practical applications. For example, as part of the UK government's open data scheme, an organogram for government offices was implemented with reference to the Organisation

Ontology in the form of RDF[33]. Collection of such data might require additional effort if there is no integration. There have been commercial services for data integration solutions with ontologies, too (Stardog Union, 2017).

Although the benefits brought by ontologies have been proved and demonstrated by various projects, and ontologies have become more and more popular in both academia and industry, there is no evidence of a mature ontology-based system being applied in the UK rail industry, nor any discussion of the reasons why there has been so little adaptation of ontologies, while stakeholders have shown little interest. Despite the review in Chapter 2 showing that there is an aspiration to establish a more integrated and intelligent data infrastructure for the UK rail industry, there is still no clear sign that the uptake of ontologies is in place. It is rational to assume that there must be factors that put professionals off; nonetheless, there has been little discussion of the deterrents to ontology-based applications in the industry. To raise the uptake level of ontologies, it is necessary to address the reasons why ontologies have not been adopted in the UK railway industry, despite existing research and conceptualisation.

To investigate factors that deter railway professionals to use ontologies, it is necessary to understand their true attitudes towards ontologies. Meanwhile, Saunders, Lewis and Thornhill (2009) suggested that in-depth interview can

[33] More detail is available at https://ckan.publishing.service.gov.uk/dataset/staff-organograms-and-pay-government-offices

collect qualitative and quantitively data of one's behaviour, while Internet survey is a valid alternative to in-depth interview. Balancing the efficiency and convenience of conducting a survey with aforementioned method, online survey was chosen to investigate and conclude UK railway professionals' true attitudes towards ontologies, because such method allowed the survey candidates to complete the questionnaire at their own convenience regardless of spatial and temporal restriction.

Therefore, this chapter describes a survey[34] that was conducted to investigate the factors that discourage professionals who work in both academic research and in software development in the industry. A conclusion of the factors that hinder the popularisation of ontologies will be given, followed by stating existing literature to justify the survey result to support following elaborations on solutions for each identify factors in the remaining chapters.

This chapter aims to achieve the following objectives:

1) Reveal whether professionals working in the UK rail industry have heard of ontologies
2) Address factors that stop professionals working in the industry being interested in using ontologies
3) Discuss whether professionals are willing to use ontologies in the future

[34] The full questionnaire can be found at https://www.smartsurvey.co.uk/s/VO077/

4.2 THE SURVEY

The questionnaire was circulated between members of the mailing list Open Rail Data Talk[35], which is mostly used by professional users of Open Rail Data[36] to exchange ideas and ask questions. Users on the mailing list have a higher level of understanding of how much more can be achieved by making between use of data, hence their being candidates for this survey.

The survey consisted of four parts which focus on the general perception of ontologies, future plans to use ontologies, and factors why ontologies will not be used in future development or research by relevant personnel, including those who are not familiar with ontologies using ontologies after knowing the benefits.

In order to investigate the reason(s) why ontologies are not applied in the industry, several questions were proposed. The full list of questions is shown in Table 8. Q1 aims to draw out the demographics of responders, and Q2 aims to investigate whether the candidate has heard of ontologies, which was used to provide an answer to objective 1). Q3 and Q4 are both follow-ups to Q2 depending on the answer to Q2. Both of them aim to discover whether professionals are interested in ontologies. Q5, Q6 and Q7 continue from Q3, to reveal if there is any potential will from professionals to use

[35] https://groups.google.com/forum/#!forum/openraildata-talk

[36] https://wiki.openraildata.com/index.php?title=Main_Page Open Rail Data in the UK consists of several schemes that are supported by Network Rail, RSSB, ORR, ATOC, TfL, TfGM, High Speed Two Ltd, British Transport Police and OpenStreetMap.org, which publish data that has been made available from the rail and transport industry.

ontologies in the future and, if not, to reveal what factors discourage them from using ontologies. Q5, Q6 and Q7 also aim to achieve objective 2). The rest of the questions are successors of Q4, designed to achieve objective 3).

TABLE 8 FULL LIST OF QUESTIONS

Question number	***Question***
1	What is your role (e.g., developer, manager)?
2	Have you heard of ontologies?
3	If so, have you or your team members used them?
4	If not, would you be interested in learning about ontologies and the benefits thereof?
5	Do you or your team members have any plan to use ontologies in the future?
6	What factors have put you off continuing to use ontologies?
7	If there were tools available that allow people who do not know much about ontologies to work with them, would you continue using ontologies?
8	Would you be interested in using ontologies in the future after knowing their benefits?
9	Would you be interested in using tools that allow people who are not familiar with ontologies to use them?

Q6 choices	**Choice statement**
1	We can achieve the same result without using ontologies
2	We have to learn more to use ontologies
3	It is difficult to learn
4	We lack professionals who can use ontologies well
5	Ontology applications require industry-wide collaboration
6	Other

The proportion of responders' roles is shown in Fig. 25. They are student, researcher, manager and developer, which account for 10%, 20%, 7% and 63%, respectively. The survey results are tabulated in Appendix A.

Amongst all responders, 60% have heard of ontologies, while only half of them have used ontologies. However, amongst those who have used ontologies, only researchers and managers intend to use ontologies in future projects. This particularly addresses the issue proposed in Chapter 2 Section 3, that is, in spite of the will to use ontologies and the research conducted on them in the UK railway industry, there has been no commercial establishment underlain by ontologies, while it seems that developers are not as interested in using ontologies in comparison to researchers and managers. For those who have not heard of ontologies, half are interested in learning about them. On top of that, candidates who have not heard of ontologies but show interest in them are willing to try ontologies in the future after learning the basis of ontologies. Almost all responders who revealed an interest in ontologies demonstrated interest in using supportive tools that enable people who are not familiar with ontologies to interact with ontologies and the rules thereof.

In order to understand why ontologies are not popular amongst professional developers in the industry, factors which might discourage people using ontologies (i.e., Q6 in the survey) have been concluded and are illustrated in Fig. 26. The answers 'We can achieve the same result without using ontologies' and 'We have to learn more to use ontologies' rank as the most significant factors.

Many responders particularly pointed out in the comments that existing relational databases are more mature and easier to use and quicker to deploy in commercial projects. For example, a researcher specifically commented that:

> *The existing toolset for relational DBs is more comprehensive than that for ontologies – if you want to use a relational DB, there is a range of frameworks, databases and management tools to suit every possible deployment and budget. The ecosystems for ontologies are at this moment not as developed, though it is improving fast.*

Some responders suggested that the term 'ontology' is not commonly seen so that they have to learn from scratch; especially when a similar result can be achieved with relational databases, it does not seem necessary to learn how to use ontologies.

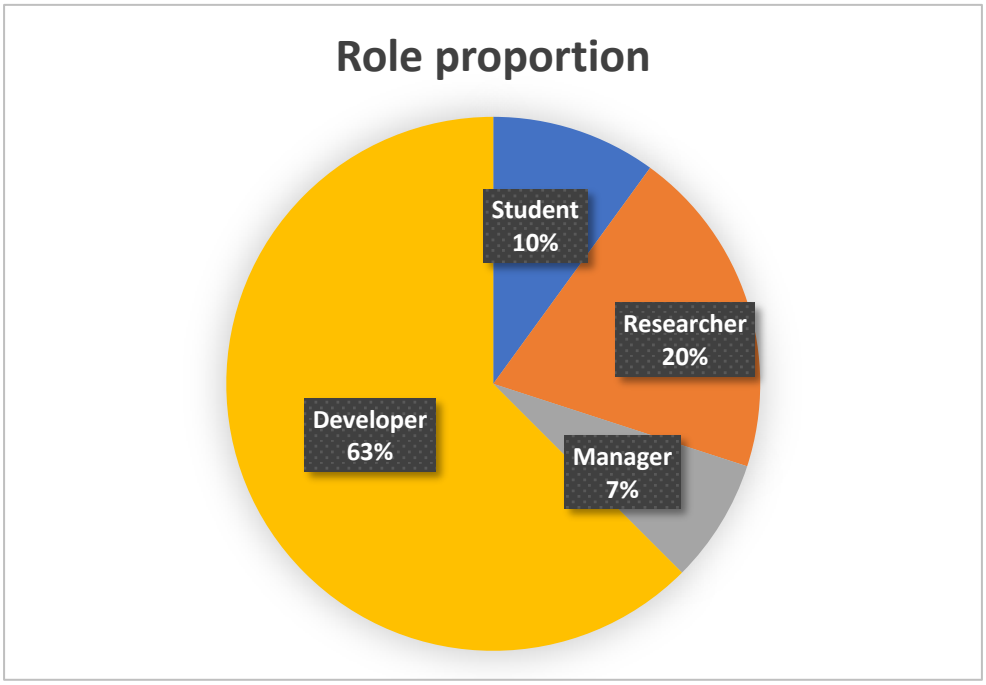


FIG. 25 PROPORTION OF ROLES

Meanwhile, the answers ‘We lack professionals’ and ‘Ontology applications require industry-wide collaboration’ rank as the second most selected factors. Other factors ranked third were time and the availability of tools. Although the option ‘ontology is difficult to learn’ ranks last, almost 40% of responders to Q6 agreed with it.

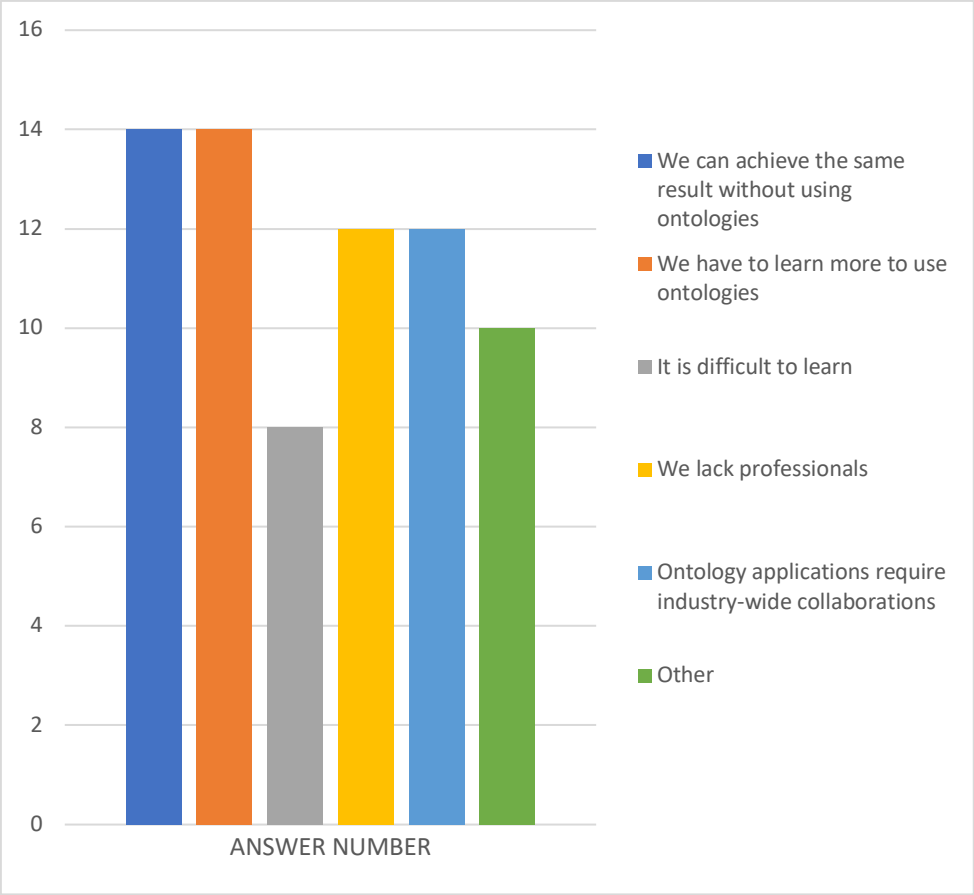


FIG. 26 ANSWERS TO QUESTION 6 (FACTORS THAT DISCOURAGE PEOPLE FROM USING ONTOLOGIES IN THE RAIL INDUSTRY)

4.3 DISCUSSION

Despite the popularity amongst researchers, there have been proof showing that ontologies are not as popular in practical development in the UK railway industry. According to the results from the survey, developers are generally not interested in ontologies and half of those who have used ontologies have no intention to use them in future projects. Conclusions of their comments can be summarised as:

a) Despite having been deployed in other industries, little has been done to demonstrate how to use ontologies to structure data in the railway industry.
b) A lack of supplementary tools for using ontologies repels potential users.
c) There has been little investigation into replicating the current manual process with ontologies in a commercial environment in the railway industry in the UK.

In accordance with the review made in Chapter 2, structured data is beneficial to industries where large complex systems reside, hence the need for better usage of the railway data, more and more data is being structured (Network Rail, 2013). Successful establishments of ontology-based information management strategies in the oil and gas industry have proven the benefits of using ontologies to manage all kinds of data that might be necessary to the business activity (Verhelst, 2012). Despite the benefits, based on the survey result, railway professional developers in the UK tend

to stick with existing in-use technologies, such as relational databases, to structure data such as documents. Some of them reckon that they can achieve the same result without using ontologies so that they do not have to invest time and effort into learning ontologies, and the deployment of existing relational databases is quicker than that of triple stores. However, some famous capabilities, such as semantic description, data integration, and consistent taxonomies, could be highly difficult to be achieved with relational data models (Spanos, et al., 2012). Ontologies can structure data with less cost so that it is worth further investigation with regards to the UK railway industry specifically. Some survey candidates also suggested that it would be better to make them aware that why ontologies are better and how to structure the railway-related data before formally diving into ontologies. Therefore, Chapter 5 will elaborate on an ontology-based approach to structure textual data.

Another significant factor is that some also commented that ontologies are useful, but a lack of tools and professionals discouraged them from using ontologies. According to the result from Q6, it can be seen that the gap between different options is small. Overall, it can be concluded that ontologies are still unfamiliar to many responders, while there are not enough supporting tools to help them to get started. Therefore, the lack of supplementary tools is hindering the development of ontology-based applications in the UK railway industry. The availability of supporting tools was specifically mentioned by some responders who think a lack of supporting tools makes them

prefer using relational databases that are easier to deploy. Some researchers stated that deploying ontology-based applications takes longer and harder due to the immature software ecosystem, which was addressed by some researchers (Morris, 2015). It has been suggested that well-designed software can compensate for the deficiency in ontologies, and the lack of relevant IT skills could be overcome.

Additionally, there have not been enough professionals specialised in ontologies in the industry, either. Most ontology-related literature assumes that potential readers are proficient in ontologies so that they can understand and apply the theory or apply it in the future. The literature shows that some tools have been introduced (Stanford Center for Biomedical Informatics Research (BMIR), 2017; Zhou et al., 2015). However, the problem has still not addressed, that is these tools are oriented towards ontology professionals. Moreover, when ontologies are built, they have to be stored in a knowledge base which is often in the form of a triple store Some respondents to the survey stated that unlike those for triple stores, there are many frameworks and tools already available for relational databases. It is much easier for developers to deploy a relational database with a ready-to-use framework, so they prefer to stay with relational databases. Some RDF store providers have addressed this issue. For example, Stardog, a leading data integration services and RDF store provider (Stardog Union, 2017), has just

rolled out its Python package[37] to help Python developers use their products. According to the response from respondents, everyone has an interest in ontologies if there are tools available to those who are not professional ontology users. Therefore, there are reasons to believe that more supporting tools and frameworks will attract more people to use ontologies, which should be addressed by more research and development. Future research should emphasise the availability of non-ontology professionals, especially those who know little about IT technologies including traditional IT technologies (Morris, 2015). Another issue is that despite having a potential interest, developers tend not to use ontologies as ontology modelling requires industry-wide collaboration which tends to be exhausting owing to the involvement of domain experts and relevant IT personnel across the industry, possibly from multiple organisations. For example, ontology modelling requires a certain level of professional knowledge; while many railway maintenance experts do not necessarily master such knowledge; however they have profound knowledge of railway maintenance. In order to build a maintenance model with ontologies, an ontology expert has to extract the knowledge from maintenance experts because domain experts cannot use ontologies and their rules to model due to a lack of tools for them. Yet, it is very difficult to involve maintenance experts in every step of the ontology design process, i.e. requirement management, goal and scope definition, competency questions, information gathering and elicitation, collating the preliminary

[37] https://pypi.org/project/pystardog/

information and conclusion[38] (Chungoora, 2019). The time-consuming and tedious preparation process leads to a situation where some developers would rather stick with traditional methods to develop models and apply them to relational databases, despite having to face more complicated maintenance and updates in later stages. This additionally justifies the necessity to enable non-ontology professionals to work with ontology rules and to use ontology in their daily jobs. More details and a demonstration will be given in Chapter 6.

On the other hand, due to the limited number of ontology experts working in the UK railway industry, ontologies are not familiar to developers and relevant decision makers working in the rail industry is that many are still not aware of their benefits or an intuitive example that can inspire them, so there is little motive for developers to learn ontologies. The Technology Acceptance Model (TAM), which is often to be used to predicant whether users will use the technology, suggests that users will be more likely to attempt new technologies when their existing job performance could be enhanced with less effort invested and easy to use (Davis, 1993). Therefore, there are reasons to believe that a proper demonstration of how ontology-based approaches could improve their daily work routines is highly necessary. Meanwhile, replacing the manual working process on data management might attract more attention from railway professionals as the industry still suffers

[38] https://www.udemy.com/course/practical-knowledge-modelling

from data silos and heterogeneous data (Capacity for Rail, 2017), so it is worth investigating. The relevant study will be discussed in Chapter 7.

It is also worth noting that a lack of understanding of how well ontology models perform in industry-level tasks also makes the rail industry hesitate (Capacity for Rail, 2017). It is important to understand the extent to which new technology will perform in practice before it is deployed in the industry (Wei, 2018). This issue has been addressed in the literature where an ontology-based data processing system could perform well with industry-level data in a non-real-time manner (Wei, 2018). However, some survey candidates were still concerned with the performance so that additional research should be performed.

4.4 Result Reliability and Validity Test

The results were converted to digits and fed into Statistical Product and Service Solutions (SPSS) software, as illustrated in Fig. 27[39]. Each single-choice question was presented in the form of a variable in SPSS, that is a numeric with a nominal measure, and the multiple-choice question (Q6) was divided into six variables, each depicting a selection for Q6. Comments and other answers for Q6 were defined as strings.

[39] For Q1, 1, 2, 3 and 4 denote 'Student', 'Developer', 'Manager' and 'Researcher'; for the rest, 1 represents 'Yes', 2 denotes 'No' and 3 means 'N/A'.

41 : Q6_3

	Q1	Q2	Q3	Q4	Q5	Q6	Q6_1	Q6_2	Q6_3	Q6_4	Q6_5	Q6_6	Q7	Q8	Q9	Comment	OtheranswerforQ6
1	2	3	2	3	N/A		3	3	3	3	3	3	3	3	3		
2	2	3	2	3	N/A		3	3	3	3	3	3	3	3	3		
3	2	3	2	3	N/A		3	3	3	3	3	3	3	3	3		
4	2	3	2	3	N/A		3	3	3	3	3	3	3	3	3		
5	2	3	2	2	2,4		2	1	2	1	2	2	1	3	3		
6	3	1	1	3	1	N/A	3	3	3	3	3	3	3	3	3		
7	2	2	3	1	2	1,2,3,4,5	1	1	1	1	1	2	1	3	3		
8	2	2	3	2	3	N/A	3	3	3	3	3	3	3	3	1		
9	2	1	1	3	2	1,2,5	1	1	2	2	1	2	1	3	3		
10	2	1	2	1	3	N/A	3	3	3	3	3	3	3	1	1		
11	4	1	1	3	1	6	2	2	2	2	2	1	1	3	3	The existing toolset for Relation DBs is more comprehensive than that for ...	Time: I'll deploy a Relational DB quicker than an ontology right now
12	2	1	2	3	2	3,4,5,6	2	2	1	1	1	1	1	3	3	It's not a term that comes up often in the rail data we work with	
13	2	1	2	1	2	N/A	3	3	3	3	3	3	3	1	1		
14	2	1	2	3	1	1,2	1	1	2	2	2	2	1	3	3	We can achieve the same result without using ontologies	
15	2	2	3	1	2	1	1	2	2	2	2	2	1	1	1	It's not a term I came across before so wasn't considering using it.	
16	2	1	2	1	2	2	2	1	1	1	1	1	1	1	1		
17	4	1	1	3	1	1,2,3,4,5,	1	1	1	1	1	2	1	3	3	Although they are useful, but they are obscure and difficult to model. Mos...	We have professional tools and frameworks for relational DB.
18	2	1	2	1	3	N/A	3	3	3	3	3	3	3	1	1		
19	4	1	1	3	1	1,2,3,4,5,	1	1	1	1	1	2	1	3	3	It is not a must in development as the same result can be achieved by relat...	There is no point to use ontologies when there is no tool available.
20	4	1	1	3	2	1,4,5,6	1	2	2	1	1	1	1	3	3		The selection of tools is very limited. Most of time we have to develop tool...
21	2	2	3	2	3	N/A	3	3	3	3	3	3	3	3	3		
22	2	2	3	2	3	N/A	3	3	3	3	3	3	3	3	3		
23	3	2	3	2	3	N/A	3	3	3	3	3	3	3	3	3		
24	1	2	3	2	3	N/A	3	3	3	3	3	3	3	3	3		
25	2	2	3	2	2	2,4	2	1	2	1	2	2	1	3	3		
26	1	1	1	3	1	N/A	3	3	3	3	3	3	3	3	3		
27	2	2	3	1	2	1,2,3,4,5	1	1	1	1	1	2	1	3	3		
28	2	2	3	2	3	N/A	3	3	3	3	3	3	3	1	1		
29	2	1	1	3	2	1,2,5	1	1	2	2	1	2	1	3	3		
30	2	1	2	1	3	N/A	3	3	3	3	3	3	3	1	1		
31	4	1	1	3	1	6	2	2	2	2	2	1	1	3	3	Existing tools support comprehensive manipulation with relational DB, be...	Time: I'll deploy a Relational DB quicker than an ontology right now
32	2	1	2	3	2	3,4,5,6	2	2	1	1	1	1	1	3	3	The term ontology is not commom to see in the rail industry	
33	2	1	2	1	2	N/A	3	3	3	3	3	3	3	1	1		
34	2	1	2	3	1	1,2	1	1	2	2	2	2	1	3	3	Similar result can be achieved in other ways	
35	2	2	3	1	2	1	1	2	2	2	2	2	1	1	1	It is not something often heard about	
36	2	1	2	1	2	2	2	1	1	1	1	1	1	1	1		
37	4	1	1	3	1	1,4,5,6	1	2	2	1	1	1	1	3	3	Lack of training and supportive tools put me off	Tools and frameworks work well at the moment with relational DB and othe...
38	2	1	2	1	3	N/A	3	3	3	3	3	3	3	1	1		
39	4	1	1	3	1	1,2,3,4,5,	1	1	1	1	1	2	1	3	3	The same result can be achieved in other ways	Deploying something with relational DB is quicker
40	4	1	1	3	2	1,2,3,4,5,	1	1	1	1	1	2	1	3	3		Tools are very limited

FIG. 27 ANSWERS PRESENTED IN DIGITS TO ALLOW STATISTICAL ANALYSIS

The reliability test used was Cronbach's (coefficient) alpha (α), a well-acknowledged measure of assessing the reliability of a questionnaire survey (Heo et al., 2015). Cronbach's alpha was proposed by Lee Cronbach in 1951 and is computed using the following formula:

$$\alpha = \frac{k}{k-1}\left(1 - \frac{\sum_i^k \sigma_i^2}{\sigma_X^2}\right)$$ (Goforth, 2015)

where k is the number of items being tested, σ_i^2 refers to the variance associated with the item i, and σ_X^2 refers to the variance associated with the observed total scores.

Cronbach's alpha measures the internal consistency of a test, representing the extent to which items in a test measure the same concept of a construct (Tavakol and Dennick, 2011). Cronbach's alpha reflects the split-half reliabilities (Cronbach, 1951) owing to substantial randomness components (Cortina, 1993). Researchers have found that most reports on reliability tests of survey data use Cronbach's alpha (Hogan et al., 2000), and Cronbach's alpha has been suggested to be tested and integrated to validate the questionnaire survey feedback (Fitzner, 2007). Cronbach's alpha can be applied to all kinds of question types including multi-choice questions[40] (Santos and Reynaldo, 2013). To conclude, the higher the value of Cronbach's alpha, the

[40] Options for multi-choice questions are analysed separately; they are often treated as dichotomous questions (variables) individually in SPSS (McCormick et al., 2018).

more consistent the answers for questions listed in a questionnaire, and Cronbach's alpha can indicate the extent to which tests that have been constructed or adapted for research projects are fit for purpose (Cronbach and Meehl, 2017; Taber, 2018), i.e., whether the result concluded from a questionnaire survey is consistent. It is certain that when Cronbach's alpha is high, the reliability level of the given measure instrument is high. However, when it is low, it does not necessarily mean that the measurement instrument is not reliable; in that case, additional reliability tests should be introduced (Heo et al., 2015).

Because the survey used in this study mostly includes nominal questions plus one multi-choice question (i.e., Q6), Cronbach's alpha is a suitable tool to assess the reliability of the survey result. There were 14 items taken into account in SPSS, which excluded comments and other answers for Q6 as they are text-based statements made by the respondents. The result is shown in Table 9 in which it can be seen that the Cronbach's alpha of the proposed survey is 0.787.

TABLE 9 RELIABILITY TEST RESULT GENERATED FROM SPSS SOFTWARE

Cronbach's alpha	*Number of items*
0.787	14

According to the definition of Cronbach's alpha, the extent to which a result is reliable is shown in Table 10. Thus, it can be concluded that the reliability of the conducted survey reached an acceptable level.

TABLE 10 CRONBACH'S ALPHA SCORE AND CORRESPONDING LEVEL OF RELIABILITY (GEORGE AND MALLERY, 2003)

Cronbach's alpha (α)	**Level of reliability**
$\alpha < \mathbf{0.5}$	Not acceptable
$\mathbf{0.5} \leq \alpha < \mathbf{0.6}$	Poor
$\mathbf{0.6} \leq \alpha < \mathbf{0.7}$	Questionable
$\mathbf{0.7} \leq \alpha < \mathbf{0.8}$	Acceptable
$\mathbf{0.8} \leq \alpha < \mathbf{0.9}$	Good
$\mathbf{0.9} \leq \alpha < \mathbf{1.0}$	Excellent

Although the survey reached a certain level of reliability, the validity of the survey result cannot be solely determined based on the reliability level. Taherdoost (2018) suggested that the reliability forms part of the validity of a survey and to further validate it, construct validity tests are mandatory, which describe the extent to which a measure can test what it purports to be measuring (Cronbach and Meehl, 2017). Taherdoost (2018) suggested it can be tested by Factor Analysis (FA).

From the perspective of its practical application to assess construct validity, FA can summarise the description of answers to given items, thereby

achieving a description of the accuracy and correctness of the properties measured and their corresponding results (Abdi and Williams, 2010; IBM, 2019; Liu et al., 2003; Yong and Pearce, 2013); thus, FA can be applied to assess the construct validity of the questionnaire. In addition, the conclusions drawn in section 4.3 are latent factors which were not directly observed from the respondents; FA is useful for assessing whether the observed data matches latent factors as expected (Golafshani, 2003).

Assume that there are k samples and each of them is described by n variables while each variable can be explained by m factors. The mathematic model of FA (Yong and Pearce, 2013) is:

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{pmatrix} \begin{pmatrix} F_1 \\ F_2 \\ \vdots \\ F_n \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

$$X = AF + \varepsilon$$

where matrix A contains the factor loadings (i.e., loading matrix), F contains the random variables and ε contains unobserved stochastic error terms. The loading matrix extracted by FA can be therefore observed so that it is possible to identify loadings with a high value. The corresponding variables of loadings identified as having high values can be grouped as latent factors. Once identified latent factors (components) match the anticipated combination of questions accordingly, it is safe to presume that the proposed questionnaire survey holds construct validity.

SPSS provides multiple methods for performing component extraction in FA, amongst which PCA was selected. PCA is a technique to perform dimension reduction to groups of potentially correlated variables and to reduce the number of data dimensions (Abdi and Williams, 2010). PCA has been widely applied in similar questionnaires (Abdi and Williams, 2010; Barry et al., 2017; Bjørnsen et al., 2017; Crawford et al., 1989; Taherdoost, 2018); thus, it is safe to use PCA as the extraction method for FA in this study. In order to proceed with FA, it is necessary to assess the suitability of the data for FA; this can be tested by the Kaiser–Meyer–Olkin (KMO) test that measures sampling adequacy for each item and for the complete model(Glen, 2016; IBM, 2019; Kaiser, 1974). The lower the proportion, the more suitable it is to perform FA (Kaiser, 1974).

The KMO test score was 0.664, which indicates that FA can be performed (Glen, 2016; Kaiser, 1974) but it would be better to supply more samplings. The FA result is shown in Table 11, which justifies the assumption that there are three topics (dimensions) as there are three components identified.

Conclusion a) generalises Q1, Q2, Q3, Q4 and Q5 with reference to the comments, which is reflected in Table 11 as component 2. This justifies the conclusion that little demonstration of the practical usage of ontologies in the rail industry leads to the unwillingness to use ontologies in the future, regardless of whether the respondent has heard of or used ontologies.

Conclusion b) was concluded from two pairs of answers for Q6 and Q7 based on the two highest loadings, more specifically: '*We lack professionals who can use ontologies well*' and '*If there were tools available to allow people who do not know much about ontologies to work with them, would you continue using ontologies?*' Based on a review of the answers, it can be seen that the respondents are willing to use ontologies if more tools are available to non-professionals. Therefore, conclusion b) can be drawn with confidence.

Conclusion c) generalises answers for Q4, Q8 and Q9 plus some comments from the respondents. The ability of ontologies to handle data was specified in the questionnaire and, as a result, a greater willingness to attempt using ontologies was shown. The respondents showed a willingness to use ontologies after knowing their benefits – they can improve their existing manual process – hence conclusion c).

In conclusion for the reliability and validity tests, it can be seen that the proposed measure fits its purpose with an acceptable level of reliability and validity. Thus, it is safe to presume that the conclusions are statistically valid.

TABLE 11 FACTOR ANALYSIS RESULT

Questions	Component		
	1	2	3
What is your role (e.g., developer, manager)?		−.601	
Have you heard of ontologies?		.912	
If so, have you or your team members used them?		.947	
If not, would you be interested in learning ontologies and the benefits thereof?		−.507	.773
Do you or your team members have any plan to use ontologies in the future?	.503	.631	
We can achieve the same result without using ontologies	.871		
We have to learn more to use ontologies	.915		
It is difficult to learn	.936		
We lack professionals who can use ontologies well	.944		
Ontology applications require industry-wide collaboration	.924		

Other	.820		
If there were tools available that allow people who do not know much about ontologies to work with them, would you continue using ontologies?	.948		
Would you be interested in using them in the future after knowing their benefits?			.973
Would you be interested in using tools that allow people who are not familiar with ontologies to use them?			.973

On top of the statistical validation, some existing literature also suggested that the benefits of ontologies are difficult to be replicated with other existing data models (Spanos, et al., 2012). Based on the review made in Chapter 2 Section 3, ontologies have gained interest from the industry, but professionals still know little about how they can use ontologies to manage unstructured data and extract more information. It is favourable to using new techniques to structure the industry data with unified knowledge models, especially for the UK railway industry where mass and complex data sources exist (TSLG, 2012), to discover more value from existing data (Köpf, 2010). The first conclusion from the survey result has justified this situation.

Second, TAM (Davis, 1993) and existing literature (Morris, 2017) have suggested that lack of tools for non-IT experts could decrease the level of willingness to attempt new technologies so that it is necessary to address this issue. Some survey candidates' responses reflected this point so that poor software ecosystem has been listed in the findings.

Moreover, based on Legris et al. (2003) and Davis (1993), new technologies should prove their capability of enhancing exiting working performance in order to increase the acceptance level. This factor is also proposed by some survey candidates, that it remained ambiguous about how ontologies can help them with their daily jobs. The third finding has reflected this factor.

4.5 CONCLUSION

Ontologies have many benefits, including but not limited to establishing a more intelligent and integrated data infrastructure. However, although many researchers from various organisations have demonstrated the benefits of ontologies, and ontology-based systems have been successfully established in some industries, there is no evidence revealing that an ontology-based application has been developed and adopted in the UK railway industry. This chapter presents a survey designed to respond to the proposed research question:

> *Given the fact that both the UK railway industry and the research community are interested in ontology-based applications, why is there no sign that an ontology-based system has been implemented within the industry?*

All objectives have been achieved:

1) Reveal whether professionals working in the UK rail industry have heard of ontologies
2) Address factors that stop professionals working in the industry being interested in using ontologies
3) Discuss whether professionals are willing to use ontologies in the future

According to the survey conducted amongst professionals working in the industry, ontology-based systems do not appear to draw interest from most of the developers and some of the researchers, who are either not familiar with ontologies or incline towards other more mature methods. Many volunteers commented that ontologies are not as easy to deploy; despite their benefits, they prefer to use relational databases instead. Besides that, the nature of the UK railway industry forms silos that hinder ontology-based applications because they require industry-wide collaboration.

To attract more people and facilitate ontology-based applications in the UK rail industry, more tools should be made available, allowing non-ontology professionals to interact with ontologies. Using ontologies can create more value from data and improve the efficiency of operation (Köpf, 2010), hence providing great value to the industry. It has also been demonstrated that the scalability of an ontology-based application can meet the demand of practical application in the UK railway industry with suitable architectures (Wei, 2018). Once ontology-based applications can handle an industrial level volume of data, and have suitable and easy-to-use frameworks and tools, it is believed that more people will start to learn to use ontologies, which will gain more attention from the industry, too. When deploying ontology-based knowledge storage is as easy as deploying a relational database, more developers will at least attempt to use ontologies in the future.

It has assumed that a mature development of ontology-based applications in the railway industry is unlikely to be attempted until the following findings have been addressed according to the survey results and some literature:

- Despite having been deployed in other industries, little has been completed to demonstrate how to use ontologies to structure data in the railway industry
- The lack of supplementary tools for using ontologies repels potential users
- There has been no investigation into replicating the current manual process with ontologies in a commercial environment in the railway industry in the UK

The reliability and validity tests presented in section 4.3 have demonstrated that the proposed findings are statistically satisfactory. However, additional samples should be provided according to the KMO scores. Some literature supported proposed findings; based on some existing literature and survey results, following chapters will address aforementioned questions and present solutions accordingly. Corresponding chapters are shown in Table 12.

TABLE 12 FOLLOWING CHAPTERS AND THEIR CORRESPONDING TOPICS

Problem	*Chapter number*
Despite having been deployed in other industries, little has been completed to demonstrate how to use ontologies to structure data in the railway industry	Chapter 5
Lack of supplementary tools for using ontologies repels potential users	Chapter 6
There has been no investigation into replicating the current manual process with ontologies in a commercial environment in the railway industry in the UK	Chapter 7

5 USING ONTOLOGIES TO MANAGE UNSTRUCTURED DATA

5.1 BACKGROUND

As discussed in Chapter 2, it is possible to use ontologies to integrate data, and some research dedicated to investigating and discussing how to use ontologies to integrate data from diverse sources in the UK railway industry has already been carried out (Morris, 2017). However, both the industry and the research community in the UK railway industry tend to focus on data management in terms of the asset (Capacity for Rail, 2017; Gogos and Letellier, 2016). There has been little discussion about using ontologies to manage unstructured data.

There are three types of data: structured data, semi-structured data and unstructured data (Taylor, 2018). Structured data is data that has been organised to have a fixed record length and is usually marked up with a data schema (i.e., data model); this type is more easily understood by software agents (e.g., search engines). Unstructured data is a general concept that depicts data captured without following any data model and which is not organised in a pre-defined form. Semi-structured data, as its name suggests, is a mixture of structured data and unstructured data that follows certain hierarchical structures, such as tags in XML, yet does not adhere to the formal data schema in fixed fields. Some examples of the three data types are shown in Table 13.

It is not difficult to understand that the higher the level of data organisation level, the more easily software agents can work with the data. However, the existence of unstructured data is inevitable. The International Data Corporation (IDC) has suggested that the total size of data will reach 175 zettabytes (ZB) in 2025, 80% of which will be unstructured (Reinsel et al., 2018). Another report indicated that as much as 90% of data generated daily is unstructured (Marr, 2019). In the UK railway industry, in order increase automation and reduce the human labour involved, it is necessary to manage unstructured data properly, especially given that although much text-based data has been gradually stored in digital form, much of it is still presented in the form of the free text so that it is still difficult to make the software agents understand exactly what information those documents have captured (Network Rail, 2013), which might be useful for business (Davis, 2019).

TABLE 13 EXAMPLES OF DIFFERENT DATA TYPES (RANKED IN ACCORDANCE WITH THEIR DATA ORGANISATION LEVEL) (TAYLOR, 2018)

Data type	*Example*	*Data organisation level*
Structured data	Data stored in a relational database	High
Semi-structured data	HTML page[41], XML	Medium
Unstructured data	Free (i.e., plain) text, paper-based documents	Low

Ontologies can integrate data in a way that is based on taxonomy, owing to their ability to represent human knowledge. Therefore, the lack of management of unstructured data, more specifically plain-text-based documents, will be addressed and a discussion about how to facilitate management with ontologies will be given, specifically for document query and retrieval. This chapter aims to answer the proposed research question:

[41] Although HTML web pages use tags to annotate the content, tags are solely used to indicate to Web browsers how to render the page and do not capture any meaning of the content in the pages, so HTML Web pages are referred to as 'unstructured data' by some researchers (Malone, 2007). However, HTML tags provide rigorous structure in a way that instructs machines to display content, so they are a mixture of structured and unstructured text, hence HTML documents being considered semi-structured in this thesis (Taylor, 2018).

Given the fact that ontologies can integrate data, how can we use ontologies to manage unstructured data in the railway industry?

Objectives that this chapter aims to achieve:

1) Deliver an ontology-based solution to manage unstructured data, using a data source for the case study, which will be demonstrated by showing how the Rail Accident Investigation Brach's accident reports report can be ingested and stored using the proposed approach
2) Discuss whether the proposed solution can improve on the existing solution in terms of accuracy

5.2 Managing Unstructured Documents with Ontologies

Many have defined unstructured data as data that 'might have internal structure but not structured by a pre-defined data model or schema' (Taylor, 2018). As its definition suggests, unlike structured data, there is no fixed pattern that enables software to operate a semantic-based search or automated tasks. Meanwhile, unstructured data can be generated not only by humans, such as textual data, but also by machines, such as images and some sensor readings. Unstructured data might be stored in a variety of locations, for example, in data warehouses, non-relational databases (e.g., NoSQL) or even within software applications. All of these have created additional difficulties in managing unstructured data, yet it is important to emphasise unstructured data as much of its value is still awaiting discovery (Kambles et al., 2017). As discussed in Chapter 2, NLP can benefit from ontology-based approach so that it is reasonable to assume that using ontologies to manage documents for the UK railway industry is benefical.

It has been widely agreed that data has substantial value for railway systems, especially in the era of Big Data (Davies et al., 2019). Information contained in unstructured data, especially textual data, is extremely difficult to recover. For managing documents especially, the traditional taxonomy of documents requires much human involvement. Many machine learning-based classification techniques have been applied to help to reduce the labour involved (Cortes and Vapnik, 1995; Zheng, 2015) and automated document

classification has focused on the kind of objects that can be clustered, too (Camous et al., 2007; Purpura et al., 2019; Yang et al., 2016). However, a problem which remains to be addressed is that most machine learning techniques cannot understand semantics. For example, when a user wants to retrieve a document, traditionally, the matching algorithm can guide the software to match documents based on given keywords; a simple keyword matching system might work well with documents supplied by the same party with consistent word choices, yet it might give disappointing results when multiple words with similar meanings are presented. Such a scenario can be found in the railway industry; for example, the term 'sleeper' used in British English and 'crosstie' used in American English are essentially the same in the railway context, representing a support which might be made of wood, concrete, etc. and holds rail tracks upright and keeps them separated at the correct distance. Therefore, when the user performs a query against the document base with the term 'sleeper', a relevant result might be lost. Mathematic models cannot always comprehend the concept, hence the room for improvement.

Ontologies provide rich semantics (Guarino et al., 2009), enabling diverse contextual annotation by offering a structured knowledge model (Kalibatiene and Vasilecas, 2011). Such a trait ensures context-awareness in data analysis. Moreover, the method for linking unstructured data is also provided by ontologies (Udrea et al., 2007), which is beneficial for eliminating data silos and enhancing data retrieval while processing data across large

complex systems where different data schemes have been established (Choi, 2014). Besides that, it has been specifically pointed out that in order to realise the full benefits of unstructured data, organisations should break down data silos and move towards a sharable data model that can be understood by AI technologies (Davis, 2019). Thus, a question can be proposed:

Is it possible to establish an ontology-based method to enhance existing document classification techniques?

In terms of document classification, semantics are frequently ignored. Such an implicit condition cannot be recognised by traditional methods of statistical and mathematical analysis. It remains a possibility that document analysis can have its basis within the context, hence being more meaningful and accurate. By conjoining various machine learning techniques and ontologies, there are reasons to believe that document analysis and retrieval could be improved in large complex systems.

5.3 Using Ontologies with Machine Learning Techniques to Classify and Query Documents

In this section, some common machine learning techniques for document classification will be discussed, and a solution that combines machine learning techniques and ontologies will be proposed.

5.3.1 Some Common Techniques for Document Classification

As the name suggests, document classification aims to assign one or more categories to a document that could consist of text, images, etc. In this study, the focus will be text-based document classification for the UK railway industry; thus, the term 'document' refers specifically to text-based documents.

Document classification is a well-known problem in many domains (Chen et al., 2006). The core idea of document classification is to extract the features from a document, which can be completed based on documents that have been already labelled (i.e., Supervised Learning) or the results of statistical calculation by an algorithm (i.e., Unsupervised Learning). Unlike library taxonomy that is mostly accomplished manually, digital documents can be analysed by 'smart' software and then categorised; regardless of what method is used to classify documents, a generic workflow is followed by most of classification techniques, as illustrated in Fig. 28. Some commonly seen examples of document classification are shown in Table 14.

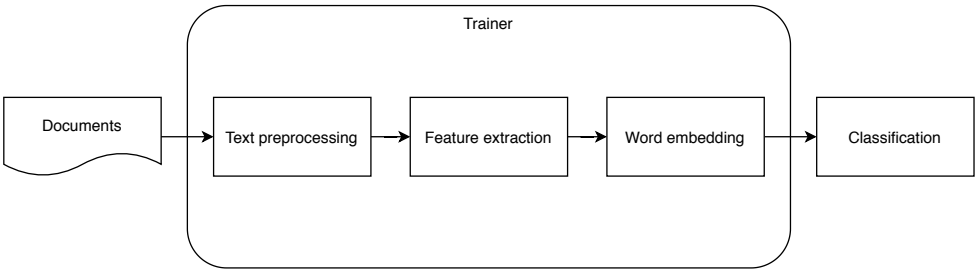


FIG. 28 GENERIC WORKFLOW FOR DOCUMENT CLASSIFICATION

TABLE 14 REPRESENTATIVE EXAMPLES OF APPLICATIONS OF DOCUMENT CLASSIFICATION

Domain	*Application*
Communication	Filter spam emails
Entertainment	Add interest labels to the video
Education	Draw users' portrait to analyse their interest
E-business	Analyse reviewers' sentiment
News	Label the report

As mentioned previously, document classification can be completed in both supervised and unsupervised manners. Classic supervised learning, such as Naïve Bayes (Ting et al., 2011), SVM (Cortes and Vapnik, 1995; Noble, 2006) and K-Nearest Neighbour (Peterson, 2009), requests a labelled training dataset (i.e. documents with known classes), whereas unsupervised learning, on the contrary, infers the category without having a labelled training dataset. Usually, supervised learning can achieve better accuracy, yet it requires labour to label the training data (Ericson and Rohm, 2017). Depending on the size of the dataset, it is possible to manually label part of it as the training set. However, in reality, the document repository might be huge; therefore, unsupervised learning is an ideal choice to reduce manual labour as much as possible. It is worth noting that although semi-supervised learning mixes the advantages of both types, it requires human intervention, too; thus, it is not discussed in this study.

There are some advanced algorithms designed to classify documents. A well-known statistical method, Term Frequency – Inverse Document Frequency (TF-IDF), addresses an issue that the traditional way in which keywords are determined mostly by reference to the term frequency (i.e. how many times a word is mentioned in the document) cannot locate the most relevant keyword(s) (Ramos, 2003). Over 80% of text-based recommending systems in digital libraries used TF-IDF in 2015 (Beel et al., 2016). However, TF-IDF does not consider the relationships between words, only taking a statistical conclusion into account and deducting the weight of frequent words that might be also useful in certain cases (Rajaraman and Ullman, 2011). Graph-based keyword extraction can rectify such an issue. An example is the TextRank algorithm. TextRank can produce graph models for the words presented in one or multiple documents; derived from Google's PageRank algorithm (Page et al., 1998), that is used to rank Web pages in accordance with the results calculated by several iteration processes, TextRank forms a graph using words as nodes instead of Web pages, taking the relationship between words into consideration to enhance a machine's capability of understanding semantic relationships (Mihalcea and Tarau, 2004). However, despite being more complex and computationally expensive, TextRank does not always necessarily achieve better accuracy than TF-IDF as it still relies heavily on word tokenisation.

There are also many other more advanced algorithms (Settouti et al., 2016). Nevertheless, this study does not seek to demonstrate how more

comprehensive algorithms can improve overall accuracy; instead, it aims to investigate how to use ontologies in this context in order to improve existing methods to improve the management of documents. Therefore, only TF-IDF and TextRank are discussed, and TextRank has been selected to classify documents because of its consideration of relationships between terms.

5.3.2 How Can Ontologies Fit into this Context?

As mentioned in section 5.3.1, both TF-IDF and TextRank algorithms rely on accurate word tokenisation. However, it is inevitable that computers will miss phrases or terms; for instance, 'cross tie' is another way to represent 'crosstie' in US English, but if it is separated into the two words 'cross' and 'tie' during the tokenisation process, the original meaning can be lost, hence eventual inaccuracy. Such a tokenisation process can be easily completed by the human brain based on our knowledge, but not by machines. Consequently, low-level ambiguity can lead to inaccuracy in high-level calculations. To eliminate the ambiguity in this context, ontologies can be useful.

Ontologies have a known ability to distinguish terms based on their semantic definition to eliminate semantic ambiguity (Guarino et al., 2009). In 2003, researchers proposed an ontology-based approach to classifying emails; they found that ontologies can be used with learning algorithms (Taghva et al., 2003) to provide a rigorous and robust lexicon for a domain. Therefore, it is possible to use ontologies to enhance the word tokenisation process and eventually to improve the overall performance of algorithms such as TF-IDF and TextRank that rely on word tokenisation. Meanwhile, ontologies are hierarchical and much research investigating methods to enhance traditional machine learning techniques has considered hierarchical information (Nyberg et al., 2010) as a higher level of generalisation can be achieved (Gabrilovich and Markovitch, 2005). It has also been suggested that relationships captured by ontologies could also facilitate syntactical analysis and

improve the overall performance of information retrieval systems (Nyberg et al., 2010). It is also possible to use ontologies to improve the document retrieval process. The principle remains the same because of the ability of ontologies to eliminate semantic ambiguity (Fang et al., 2007), as illustrated in Fig. 29. Therefore, there are reasons to believe that ontologies can help the UK railway industry to manage unstructured documents.

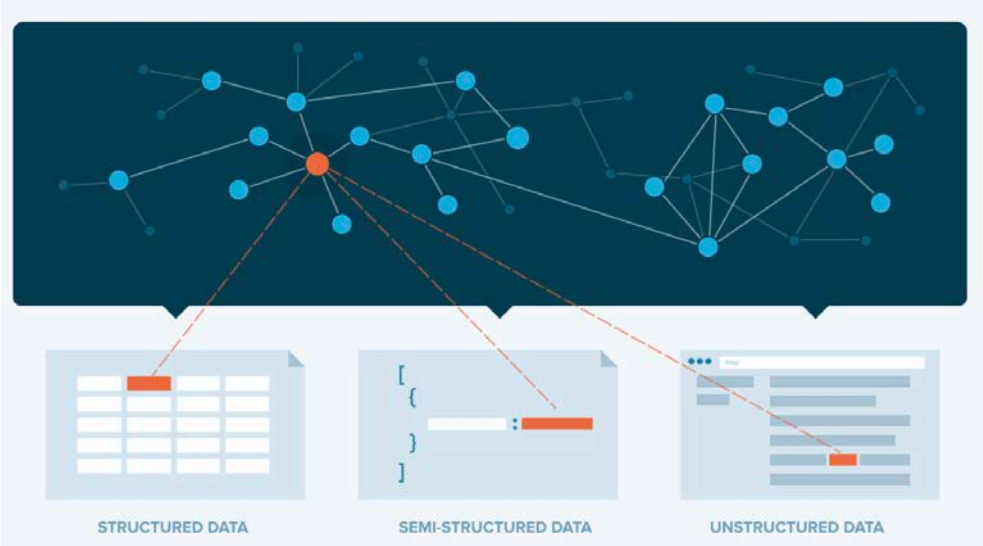


FIG. 29 DATA MAPPED TO THE CONTEXT REGARDLESS OF ITS ORIGINAL FORM

Additionally, ontologies can integrate data stored in different silos as discussed in Chapter 2. Without an integration platform, a user might have to search for documents from a few sources in order to obtain what is needed, as depicted in Fig. 30. However, if there is an integrated solution, the user might only need to search once to obtain the required documents, as illustrated in Fig. 31, hence improved user experience and efficiency.

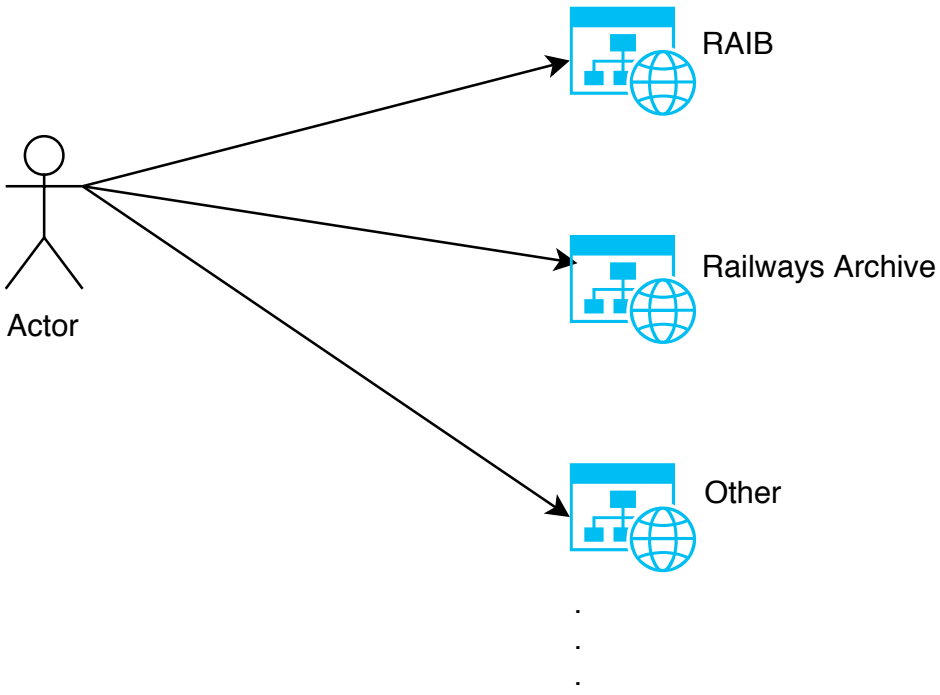


FIG. 30 THE USER MIGHT HAVE TO RETRIEVE INFORMATION FROM A FEW DIFFERENT SOURCES

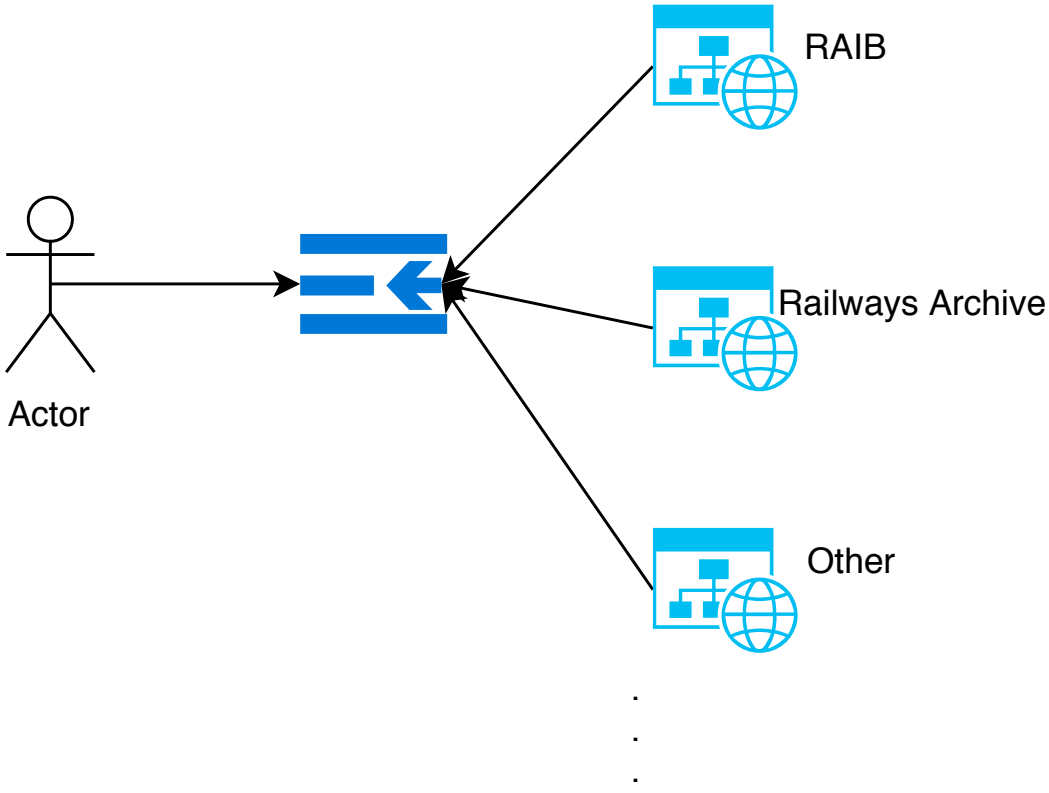


FIG. 31 THE SEARCHING PROCESS IS SIMPLIFIED WHEN AN INTEGRATED SOLUTION IS PROVIDED TO INTEGRATE DATA FROM DIFFERENT SOURCES

Therefore, in this study, a framework that enhances document classification has been designed to help the UK rail industry to manage documents with ontologies; it will be elaborated in the following sections.

5.3.3 Using RaCoOn to Manage Unstructured Data

RaCoOn was produced to accommodate better knowledge management across large complex systems in collaboration with improved data integration within the rail industry (Tutcher et al., 2017). Although RaCoOn provides comprehensive support for modelling rail infrastructure, timetable and rolling stocks, events and documentation were not taken into account. Thus, in order to manage unstructured reports, there is a need to extend RaCoOn. Meanwhile, to facilitate a higher level of automation, it also requires a way to manage machine learning functionality. Since ontologies can encompass knowledge, a machine learning ontology was also modelled to capture machine learning functions; it was based on some common knowledge and research results from machine learning studies, e.g. a suitable algorithm in accordance with the shape and type of given data (Ericson and Rohm, 2017).

An example is provided to demonstrate the RaCoOn extension, that for instance, a runaway accident is caused by a malfunctioning brake. It is illustrated in the form of triples in Fig. 32, where the dashed line represents the inferred relationship. As it is for demonstration purposes only, no specific time and location was added, and no restrictions were put on classes. The inserted triples and properties thereof are in red, while solid lines, dashed

lines and dotted lines denote class properties, instance properties and inferred relationships, and rectangles and ovals represent classes and instances, respectively.

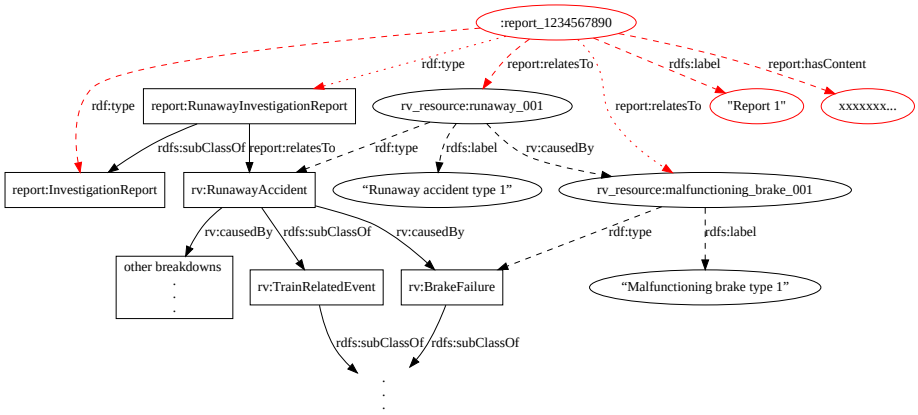


FIG. 32 KNOWLEDGE MODEL OF A RUNAWAY ACCIDENT INVESTIGATION REPORT

This structure is useful because it can link abstract events, the runaway accident and the malfunctioning brake, as a whole in a hierarchical and logical structure, i.e., antecedent–consequence. Meanwhile, the specific report that is related to this particular runaway accident can be linked. The class '*RunawayInvestigationReport*' ensures that all its instances relate to runaway accidents. This fraction of the knowledge model captures the following statement:

> *A runaway investigation report is an investigation report that relates to some runaway accidents. A report called 'Report 1' is an investigation report which relates to a runaway event labelled 'runaway accident*

type 1', caused by 'malfunctioning brake type 1' which is a kind of brake failure.

Such a piece of knowledge can be inferred as '*Report 1 is a runaway investigation report, which investigates the runaway accident caused by "malfunctioning brake type 1"* ' based on our knowledge. Such an inference can be conducted automatically by the reasoner deployed in the triple store.

Suppose that a user wants to retrieve all runaway investigation reports that relate to '*Malfunctioning brake type 1*', a query string can be composed as illustrated in Fig. 33:

```
SELECT *
WHERE
{
    ?report             rdf:type            :InvestigationReport;
                        report:relatesTo    ?runaway_accident.
    ?runaway_accident   rdf:type            rv:RunawayAccident;
                        rv:causedBy         ?problem.
    ?problem            rdf:type            rv:BrakeFailure;
                        rdfs:label          "Malfunctioning brake type 1".
}
```

FIG. 33 EXAMPLE SPARQL QUERY

It is worth mentioning that there is no 'correct' model for composing the query string. The query demonstrated above follows the minimised number of subjects principle, that is the less complex a query (e.g. amount of query variables), the better its performance in theory (Pérez et al., 2009).

Therefore, related events were manually tagged to the reports while corresponding knowledge models (e.g., the example given in Fig. 32) and assertions were inserted into extended RaCoOn as much as possible.

It is necessary to mention that the focus of this study was not ontology learning, i.e., automated knowledge extraction and augmentation from data, but the use of ontologies to facilitate existing learning techniques to manage unstructured data in a more automated manner; therefore, RaCoOn was manually extended to fit the purpose of this study. A high-level generic process is illustrated in Fig. 34.

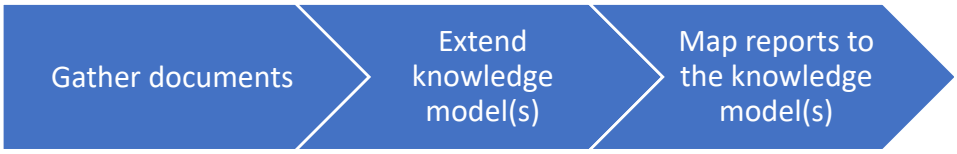


FIG. 34 GENERAL PROCESS OF MAPPING REPORTS TO ONTOLOGIES

Then, a rule can be added as demonstrated in Fig. 35:

```
IF {
    ?report     report:relatesTo    ?event.
    ?event      rv:causedBy         ?problem.
}
THEN{
    ?report     report:relatesTo    ?problem.
}
```

FIG. 35 EXAMPLE RULE (PRESENTED IN IF-THEN FORM FOR READABILITY)

The aforementioned query string can be concentrated as shown in Fig. 36:

```
SELECT *
WHERE
{
    ?report     rdf:type            :RunawayInvestigationReport;
                report:relatesTo    ?problem.
    ?problem    rdf:type            rv:BrakeFailure;
                rdfs:label          "Malfunctioning brake type 1".
}
```

FIG. 36 QUERY STRING FOR THE EXAMPLE MENTIONED ABOVE WHEN THE RULE IS INSERTED

The application of ontologies can ease the modelling process as there is no requirement to explicitly outline '*Report 1 is a runaway investigation report, which relates to "Malfunctioning brake type 1"'*. Ontologies provide sufficient semantic expressivity and schema flexibility to model structures for unstructured data, and there is no need to explicitly refine the implicit relationships between entities. This flexibility enables future easy maintenance and extension of the model without the effort of creating additional models to manage other similar unstructured data from other sources, e.g., documents stored in different document repositories.

The example query string can also facilitate the retrieval process. It can help the user to retrieve specific types of documents. Thus, there are reasons to believe that using ontologies to manage unstructured data can also improve the data query.

Meanwhile, the separation of functional programming and data modelling can be facilitated by adding rules into ontologies, too. This is particularly useful in the rail industry due to the existence of legacy systems as it is very difficult to make changes to their programming logics, plus making changes to old systems carries a risk. Separation is also useful when creating a new data management system as it allows the size of programming logics to be reduced. Maintenance at a later stage can also be easier as there is no need to re-compile the software. Additionally, because ontologies are extendable,

existing models can be reused and extended to fit other tasks, such as safety assessment models, decision support models, diagnosis models, etc.

5.3.4 An Ontology-Based Classification Framework

In this section, an ontology-based classification framework will be presented to help manage unstructured data, especially textual data, to realise the scenario proposed in section 5.3.3. The proposed framework aims to form the mapping layer for unstructured data in the system structure illustrated in Fig. 37, where the unstructured data can be fed to the framework and the framework can complete classification and mapping to the ontology process. The user only has to interact with the UI in order to access the data. This management strategy aims to reduce human involvement while managing unstructured data; by using ontologies, it also aims to structure data with an explicitly defined hierarchy (paradigm). This high-level architecture is Ontology-Based Data Access three-level architecture (Calvanese et al., 2011) and the corresponding mappings are marked with dotted lines in Fig. 37.

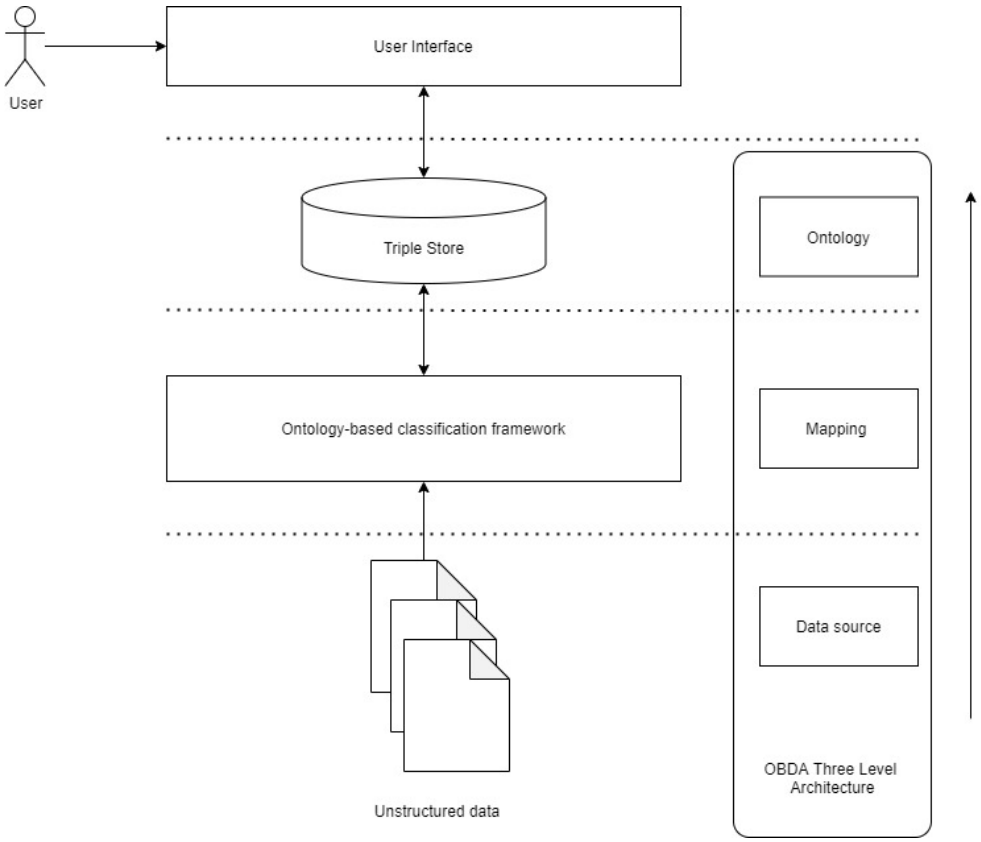


FIG. 37 HIGH-LEVEL SYSTEM STRUCTURE

The architecture of the framework is illustrated in Fig. 38, in which green boxes represent high-level modules, including the machine learning utility, data analyser, triple utility, data set generator, SPARQL parser and relational database mapper. These modules are classified according to their purpose. In some of the green boxes, light blue boxes are the tools included in high-level modules; connected to these are cyan boxes, the sub-modules thereof, differentiated by specific functions. The proposed framework has been implemented in Python 3.6[42], which is designed to provide common tools for interlinking data and use ontologies to simplify and enhance the machine

[42] https://www.python.org

learning training process, especially for text-based document classification. It is worth mentioning that although this study mainly focuses on textual data, relevant APIs for image and numeric data were left for future development and improvement. The design of this architecture refers to several previous studies (Camous et al., 2007; Thiyagu and Sendhilkumar, 2011; Wijewickrema, 2014; Wolstencroft et al., 2006).

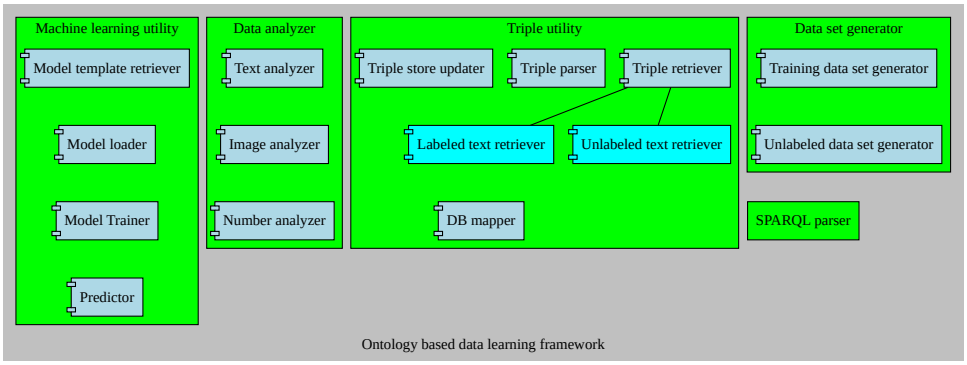


FIG. 38 ARCHITECTURE OF THE PROPOSED ONTOLOGY-BASED FRAMEWORK FOR CLASSIFICATION

In the illustration, there is a green box containing no other components. This is the SPARQL parser, which is capable of parsing the various requests into SPARQL, converting processing results from other modules to SPARQL queries depending on the user and requests made by other systems.

The machine learning utility provides support to machine learning functions. This was achieved by wrapping several common functions (e.g. text encoding, model definition, executable TextRank algorithm) provided by Scikit-learn (Pedregosa et al., 2011) and Keras (Chollet, 2015).

Suppose we have a document D. After the initial pre-processing by the NLP module and classification completed by TextRank, there will be a list of keywords with corresponding scores; an example result is shown in Fig. 39.

```
science - 1.717603106506989
fiction - 1.6952610926181002
filmmaking - 1.4388798751402918
China - 1.4259793786986021
Earth - 1.3088154732297723
tone - 1.1145002295684114
Chinese - 1.0996896235078055
Wandering - 1.0071059904601571
weekend - 1.002449354657688
America - 0.9976329264870932
budget - 0.9857269586649321
North - 0.9711240881032547
```

FIG. 39 EXAMPLE OF OUTPUT FROM TEXTRANK

The data analyser was implemented to analyse the features of data sets with reference to others' work. It has three components, for the analysis of text data, image data and number data. Features can be wrapped up and sent to the model template retriever, which is in the machine learning utility, to retrieve the pre-defined model stored in the triple store.

The process of matching with ontology was implemented with reference to another similar study (Fang et al., 2007). To analyse the keywords, each keyword forms a node, hence the collection of nodes:

$$W = (w_1, w_2, w_3, \cdots, w_n)$$

where n is the number of keywords and the order of nodes is determined by their scores from high to low. Meanwhile, assume we have an ontology O and $\forall w \sqsubset O$ and:

$$O \to T = (t_1, \cdots, t_m),\ t_m = (c_s, c_p, c_o)$$

$$\omega_m = \frac{1}{distance(c_s, root)^{1/4}}$$

$$t = P(w_s, w_o)$$

where T is the collection of triple models defined in O, m is the number of triples and P represents the predicate of a triple. It can thereafter be deduced $\forall w \sqsubset T$, so that:

$$t_x = (w_s, p, w_o), t_x \in T$$

Matching the node:

$$w_{xn} \text{ with } W_y$$

where w_{xn} is the selected node and W_y is the collection of all other nodes except w_x (i.e., $w_{xn} \cap W_y \equiv W$), using $P(w_{xn}, w_y)$; then, a list of triples T_{xn} with subjective w_{xn} can be obtained. Eventually, a collection of matched triples can be:

$$T_x = (T_{x1}, T_{x2}, \cdots, T_{xn})$$

If T_x has one or more elements, then it is safe to use the ontology to describe the document, otherwise the ontology needs to be revised and extended. Multiply the score (represented using ω_{xn}) of each w_{xn} and w_y pair in T_{xn} as the weight to calculate the most predominant triple and cross-reference the result obtained using the approach proposed by Fang et al. (2007) who used the Earth Mover's Distance (EMD) to calculate the similarity between a document d and an ontology O:

$$EMD(d, O) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} d_{ij}}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}}$$

$$d = \{(\omega_{x1}, T_{x1}), \dots, (\omega_{xn}, T_{xn})\}$$

$$\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} = \min\left(\sum_{i=1}^{m} \omega_{T_m}, \sum_{j=1}^{n} \omega_{Tx_n}\right)$$

$$Simlarity = 1 - EMD(d, O)$$

$$Simlarity > 0.5 \rightarrow \exists(d \in O)$$ (Fang et al., 2007)

The final list of keywords can be refined by semantic normalisation (e.g., replacing synonyms) and classes in ontologies can be assigned to the given document.

The triple utility includes several tools that are used to manipulate the triple store, e.g., to update, parse triples into JSON or XML, and retrieve triples. It also establishes the connection to the triple store. The data set generator generates training sets for model training, as well as sets of unclassified data for prediction if the given learning type is classification.

Based on the proposed architecture, a high-level flow has been designed, that is illustrated in Fig. 40. The whole system has three major components: an ontology-based learning framework, a triple store and a text pre-processor. The proposed framework bridges other components, helping to exchange data and training the model to extract features from given

unstructured data. The triple store (preferably Stardog[43]) provides access to ontologies (i.e., RaCoOn in this study) and the inference function, while the text pre-processor can cleanse and tokenise text-based data.

[43] https://www.stardog.com

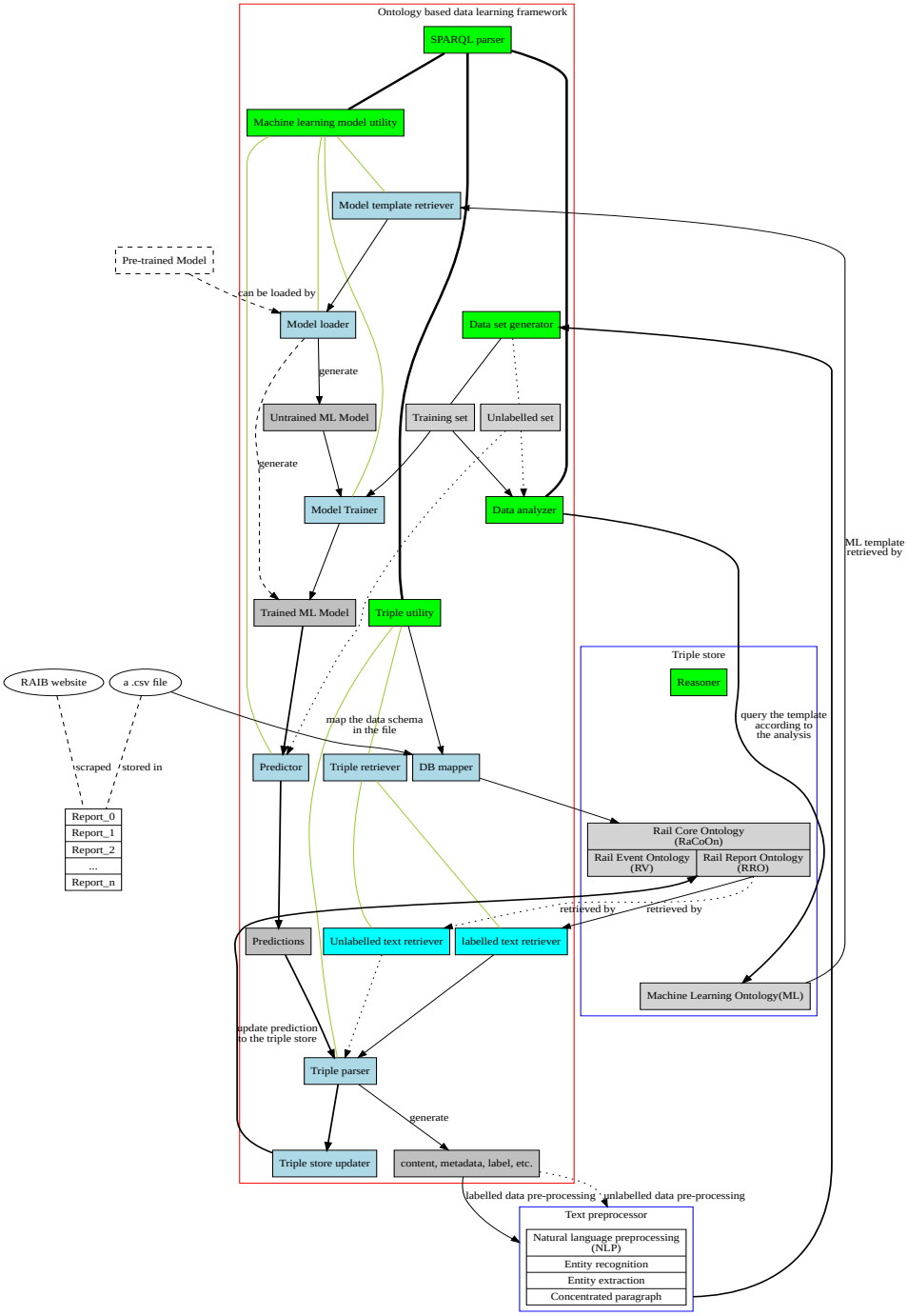


FIG. 40 HIGH-LEVEL FLOW OF THE PROPOSED EVENT LEARNING SYSTEM

5.4 Case Study – Classifying the Event Type for RAIB Investigation Reports

The Rail Accident Investigation Branch[44] (RAIB) publishes investigation reports of UK railway accidents on a regular basis. These reports can be categorised by railway type, including heavy rail, light rail, metros and heritage railways, as well as by report type, comprising investigation reports, bulletins, interim reports, discontinuation reports and safety digests. This study particularly focuses on investigation reports for light rail.

However, no other category is provided by the website. In other words, users can only search for documents based on the aforesaid categories. That means that when a user intends to search for a report based on the event, it will be based on keyword matching. Simple keyword matching works, but the result might be inaccurate. For example, a test was performed to search for reports investigating runaway accidents due to a malfunctioning brake, but reports that had no relevance to a malfunctioning brake were returned, although they were related to runaway accidents. Such an issue might lessen the overall efficiency when tasks such as safety reviews and assessments, infrastructure planning, etc. need to be performed, as users need to take additional steps to filter the retrieved reports. Thus, it is reasonable to

[44] More details are available at https://www.gov.uk/government/organisations/rail-accident-investigation-branch

assume that it would be beneficial if event types could be extracted from those reports.

This section demonstrates how the framework proposed in section 5.3.4 can be applied to help achieve this goal.

5.4.1 Data Preparation

In order to initialise the case study, RAIB reports needed to be collected. Reports published by RAIB are accessed via their website[45], but no option is provided to download the whole repository on the website. Therefore, a scraper was designed using Scrapy[46] to collect reports from the Web portal and make the data ready. When this study was being implemented, 350 reports were collected from the RAIB website.

5.4.2 Ontology-Based Report management

Next, the database mapper mapped the CSV file to the triples conforming to the model defined in the extended RaCoOn for each row, and the mapped result was asserted afterwards. Although only investigation reports were used in this case study, there might be a variety of different types of documents in the real world; thus, it is necessary to assert the mapped result to ensure satisfaction with the hierarchy defined in the given ontologies. After

[45] More details can be found at https://www.gov.uk/raib-reports?keywords=&report_type%5B%5D=investigation-report&date_of_occurrence%5Bfrom%5D=&date_of_occurrence%5Bto%5D=

[46] More details can be found at https://scrapy.org. Scrapy is an open-sourced web crawling framework that is coded in Python.

conversion, the data will be stored in the supplementary triple store, i.e., Stardog in this study. The triple retriever and parser can thereafter extract reports and content thereof and send them to the pre-processor upon request.

The text pre-processor executed the same operation for both labelled and unlabelled content. The first step was to apply generic pre-processing techniques that are commonly seen in NLP, that include noise removal, tokenisation and normalisation. The pre-processing was completed with the help of Natural Language Toolkit (Bird et al., 2009) . The extracted content can be further extracted if more concise content is wanted. This was done to extract entities from the text, forming concentrated content and removing redundancy, such as adjectives and adverbs. Google Cloud Natural Language (Google, 2018), Stanford CoreNLP (Manning et al., 2014) and ReVerb (Fader et al., 2011) could be used to extract entities in the text; Stanford CoreNLP was used because of its efficiency and cost.

The data set generator loaded the processed content and its matching labels, producing training sets and other sets awaiting tagging. Data in training sets was then read by the data analyser (i.e., a text data analyser in this example). The text data analyser formed a query string according to the features of the training sets. For instance, some reports were tagged with more than one label in this case study, hence the analysis result being multilabel classification. Labels can be then mapped to the model defined in the ontology, and

relationships can be refined between labels. New triples can be formed thereafter and inserted back into the triple store.

Additionally, it is worth noting that the reason why labelled data existed was that although TF-IDF and TextRank are classified as unsupervised learning, they can be used in supervised environments by Scikit. Moreover, a pre-trained model can be loaded by the framework if other algorithms are preferred. It is based on whether the user intends using a pre-trained model. If so, the model loader can load the pre-trained model to the framework and generate a trained model and skip the training process; if not, the model template retriever sends the query string generated by the analyser to the triple store, i.e., requests the model template captured in ML Ontology with regard to analysis results, producing a file that contains the source code of the model in Python. The model can thereby be read by the model loader, which can be trained by the given training data set. Once the model is ready to be used for prediction, data in the unlabelled set can be classified, followed by wrapping of predictions into triples by the triple parser. Subsequently, the triple updater instructs the triple store to update newly generated triples.

Although prediction results cannot be as comprehensive, the triple store can still complement them by inferring breakdowns based on given events or vice versa based on an existing knowledge model. The high-level flow is illustrated in Fig. 41.

Finally, new reports can be tagged by the predictor automatically and stored in the triple store as the flow illustrated in Fig. 37. When the user wants to query reports with respect to events, breakdowns or a combination of various events and breakdowns, it can be accomplished by querying the triple store directly. How the text was transformed with reference to ontology models has been illustrated in Fig. 42.

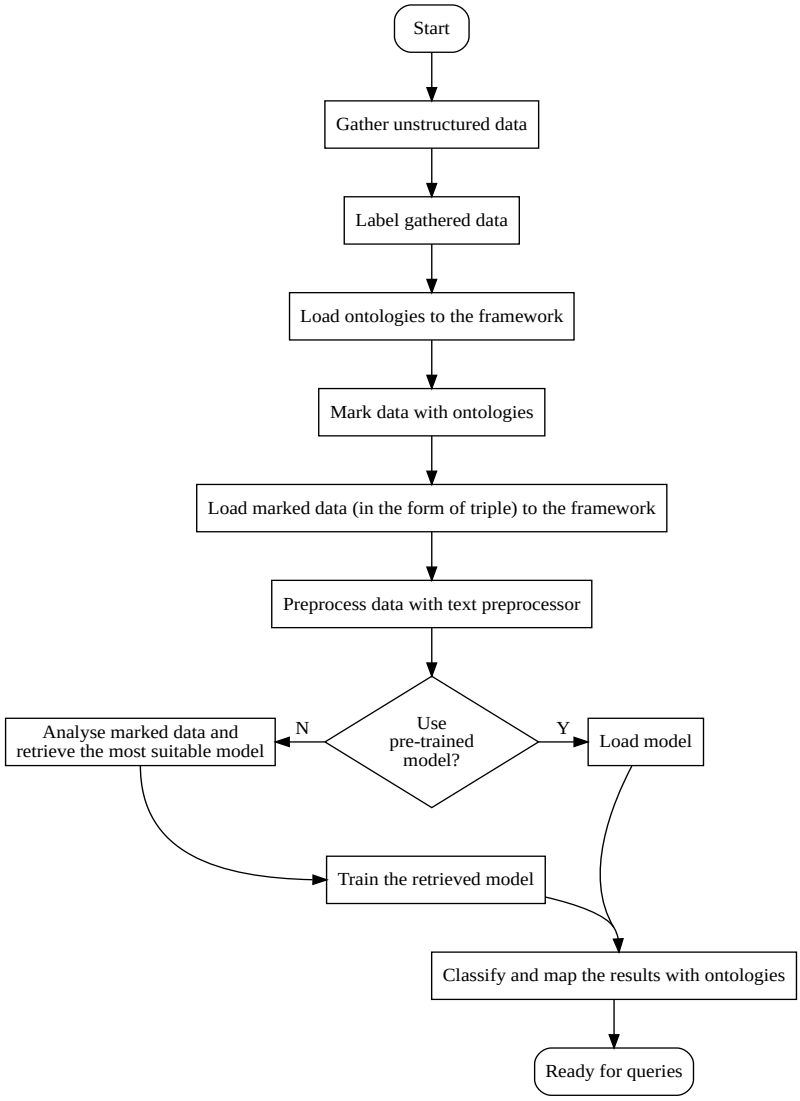


FIG. 41 HIGH-LEVEL FLOW CHART

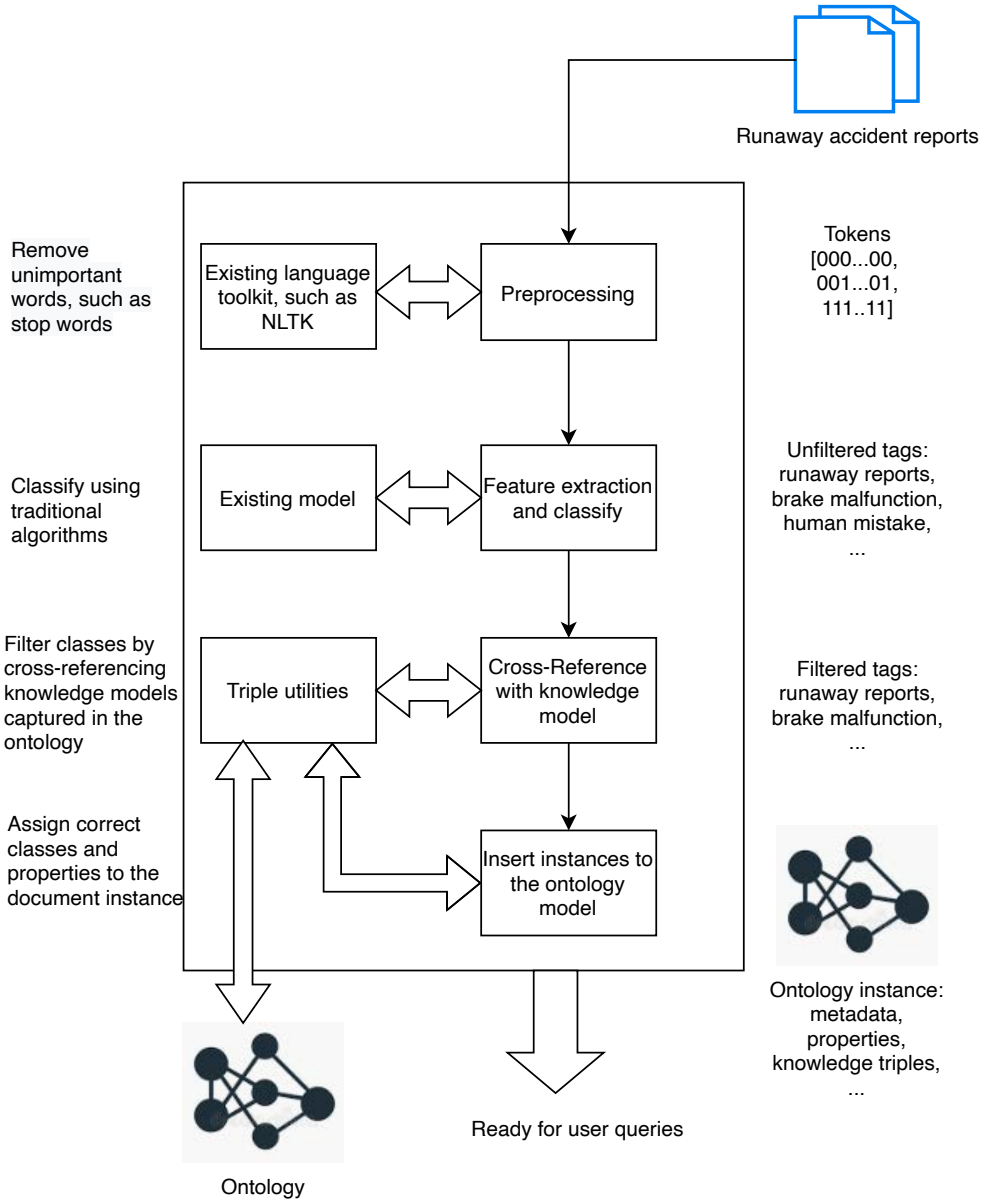


FIG. 42 DEMONSTRATION OF HOW THE DOCUMENTS WERE TRANSFORMED AND MAPPED WITH ONTOLOGIES

5.4.3 Result of Classifying Unstructured Documents

At the point this study was carried out, a total of 350 investigation reports were scraped from the RAIB website. A representative result is shown in this section.

Two rounds of experiments were executed, one solely using TextRank to extract keywords and TF-IDF as the reference, and the other using the proposed framework.

Using the example given in section 5.3.3, there were 27 reports that could be labelled as a runaway accident after manual review. After reviewing these 27 reports, four were found to have relevance to brake failure. Due to a lack of detailed failure information in those four reports, it was assumed that all failures were the same and the corresponding knowledge models were created and inserted into the triple store.

After categorising documents without using the proposed framework to classify runaway accidents, the accuracy (95.7%), precision (73.1%) and recall rate (70.4%) were calculated based on the confusion matrix shown in Table 15.

TABLE 15 CONFUSION MATRIX OF SOLELY APPLYING TEXTRANK ALGORITHM

		Real category	
		Runaway accident reports (Positive)	Other reports (Negative)
Predicted category	Runaway accident reports (Positive)	**19**	7
	Other reports (Negative)	8	**316**

The result after using the framework is shown in Table 16. The corresponding accuracy, precision and recall rate are 96.6%, 77.8% and 77.8%, respectively. All four reports were identified correctly. Consequently, objective 2) is achieved as the proposed solution slightly improves the result.

TABLE 16 CONFUSION MATRIX OF APPLYING THE PROPOSED FRAMEWORK TO MANAGING RUNAWAY INVESTIGATION REPORTS

		Real category	
		Runaway accident reports (Positive)	Other reports (Negative)
Predicted category	Runaway accident reports (Positive)	**21**	6
	Other reports (Negative)	6	**317**

5.4.4 DISCUSSION

It can be seen that the proposed framework can enhance the performance of existing machine learning techniques. Comparing the use of only TF-IDF and TextRank with use of the proposed framework, the accuracy, precision and recall were improved by the new framework and it helped to identify all desired documents when requesting the investigation reports relating to runaway accidents caused by brake failure. It is worth noting that despite a significantly increased recall rate, due to the small number of samples, the true-positive value only increased by 2 while the false-positive value decreased by 1. However, despite the small dataset, it can still be concluded that using ontologies can enhance the performance of traditional classification algorithms and facilitate better management of text documents. This could be due to the presence of ontologies helping text pre-processing, while the internal relationships captured by ontologies can also facilitate entity identification.

Moreover, other than the improved accuracy, precision and recall rate, the knowledge model provides a structure to the unstructured textual data. It is also highly possible that document retrieval could be also improved as ontologies can provide explicit term explanation, and an ontology-based framework for knowledge representation could extract and map unstructured data with the hierarchy (Guarino, 1995). Therefore, it is possible to use proposed framework to manage unstructured documents with semantic annotations.

The prototype should be compatible with any text format in theory. However, it was found that some files encoded in Portable Document Format (PDF) cannot be loaded correctly. It might possibly be because the generation of original files was on different operation systems and by different word processor software, and potentially due to encryptions. This reflects the side effects of unstructured data, that unstructured data could be difficult to be processed so that it justifies the value of continuing the research to map the unstructured data.

5.5 CONCLUSION

In Chapters 2 and 4, it was identified that despite having been deployed in other industries, little has been done to demonstrate how ontologies can be used to structure data in the railway industry; this chapter aimed to address this issue, and has answered the question:

> *Given the fact that ontologies can integrate data, how can we use ontologies to manage unstructured data in the railway industry?*

This chapter demonstrates an ontology-based document classification framework using RAIB reports as a case study to show how RaCoOn can describe documents and in response to objective 1). Ontologies can not only integrate unstructured data but also help to automate classification techniques, enhancing the data analysis process and improving information retrieval by eliminating ambiguities. It is rational to envisage that large complex systems could benefit from enriched data analysis and improved information retrieval as these tasks have been often performed, especially managing unstructured data. The case study has provided a practical demonstration of enhanced model training and information retrieval, focusing on producing a generic framework to help facilitate and use ontologies with machine learning models, also extending RaCoOn to support complex event modelling for the case study. The result of the case study shows that the

proposed approach slightly increases the overall accuracy as discussed in section 5.4.3. Despite this slight improvement, objective 2) can be answered: the proposed ontology-based approach can improve the overall performance of existing learning techniques. This chapter also identified the defects of the existing document keyword matching mechanism in the UK railway industry, and the value of mapping unstructured documents with ontologies. It also presented a feasible solution to realise this task, which aimed to underlie a foundation for future railway unstructured data management.

Nevertheless, the framework can be implemented more generically, with more tools added to facilitate different mapping and learning situations. The framework should be also enriched with the capability to recognise a data schema automatically to enable it to be used by non-data experts. This study only focused on unstructured textual data; images and numeric data (e.g., sensor readings) still await further investigation. Based on the discussion and demonstration made in this chapter, future illustration of the flow and key components coupling could be as illustrated in Fig. 43.

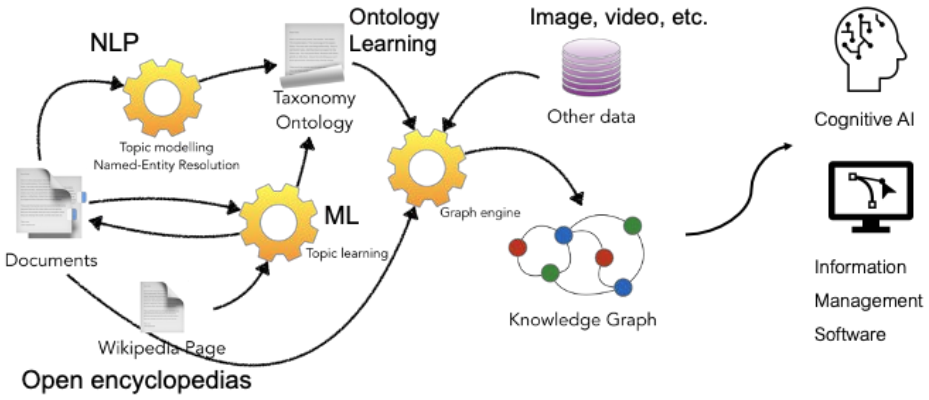


FIG. 43 ULTIMATE FLOW OF THE DATA AND IDENTIFIED KEY COMPOENENTS OF ONTOLOGY-BASED UNSTRUCTRDED DATA MANAGEMENT

Some issues remain unaddressed for both objectives 1) and 2). First, the ontology extension process was completed manually, which might work well while the total size of unstructured data is small but would be impossible in a Big Data environment; consequently, an automated knowledge extraction process for this framework should be developed in order to help the railway industry to better manage its unstructured data. This issue might require additional research into the combination of NLP and ontology applications. Second, owing to the limited data to which access was granted, a larger dataset was not tested. The scalability of this solution remains unknown and awaits further investigation, too. Third, although users can explicitly retrieve unstructured data (in the form of triples) from the triple store, there is no discussion in the literature of how to analyse users' keywords from a plain string and automatically convert them to SPARQL in order to extract data from the triple store.

Another key benefit of ontology-based unstructured data management is that the ontology could underlay future AI technologies. Gruber (2017), who greatly contributed to a series of fundamental works of ontologies, has envisaged that 'humanistic AI' would become a key developing direction for the AI technologies; and being capable to represent human knowledge in a structured and formal way is of great importance for future AI. Ontologies can facilitate this task; along with more and more development of AI-based techniques, there are reasons to believe that the proposed solution has huge potential for the railway industry.

This chapter provides a preliminary insight into the UK railway industry in terms of using ontologies to manage its unstructured documents. After addressing the aforementioned issues, plus the joint effort made by major participants across the industry by modelling a comprehensive data management model in the form of ontologies to not only integrate data but also analyse data accurately, it is practical and beneficial to apply ontologies to manage unstructured data. Systematic validation is required, which should be performed in a tailored experimental environment in the future.

6 Enabling Non-Professionals to Design Rules for an Ontology

6.1 Background

Ontologies have been proven to be beneficial in many industries, and have been widely used in industries such as bioscience (Bodenreider, 2008; The Gene Ontology Consortium, 2001) and oil and gas (Ebrahimipour and Yacout, 2015; Leal, 2005). Due to the increasing demand for knowledge modelling and automated reasoning in industry-level systems, ontologies have naturally come to the attention of scholars, scientists and IT experts in the UK railway industry because of their high level of flexibility and interoperability as discussed in Chapter 2. However, despite the benefits brought by ontologies and the attention drawn to them, the full potential of ontologies has not yet been unleashed in the UK railway industry.

As discussed in Chapter 4, it is clear that developers working for the railway system in the UK rarely know much about ontologies, and those who have used them tend to stick to other 'more mature' technologies which already have an abundance of tools; this saves the cost and time of learning more about ontologies, even though decision makers are generally more interested in using ontologies to facilitate better decision-making. It is almost impossible to make all IT personnel possess sufficient knowledge to work with ontologies; thus, it has led to current dilemma, that 'ontologies seem

brilliant, but we do not know how to use them, so we would rather stick with existing tools'. Therefore, although researchers' proposals have modelled relevant knowledge in the railway industry in the form of ontologies, the industry has not yet taken any step further. It seems that most researchers and ontology developers omit allowing non-professionals to interact with ontologies, which is also justified based on the conclusion drawn from Chapter 4.

Admittedly, based on the survey result detailed in Chapter 4, there are many tools available. Notwithstanding availability, they are mostly code-based; in other words, they require users to have enough knowledge of ontologies. In the meantime, although some ontology editors, such as Protégé and TopBraid Composer, allow users to edit ontology rules, they are heavily coding-based, which requires users to either master the rule language, such as SWRL, or its 'simplified' version, as demonstrated in Fig. 44. In line with the survey, developers working for the UK railway industry still find it is not necessary to learn these, especially when they can achieve similar but inferior results with other methods. However, many professional developers have revealed the will to learn and use ontologies according to the result found in Chapter 4 if more easy-using supplementary tools are available.

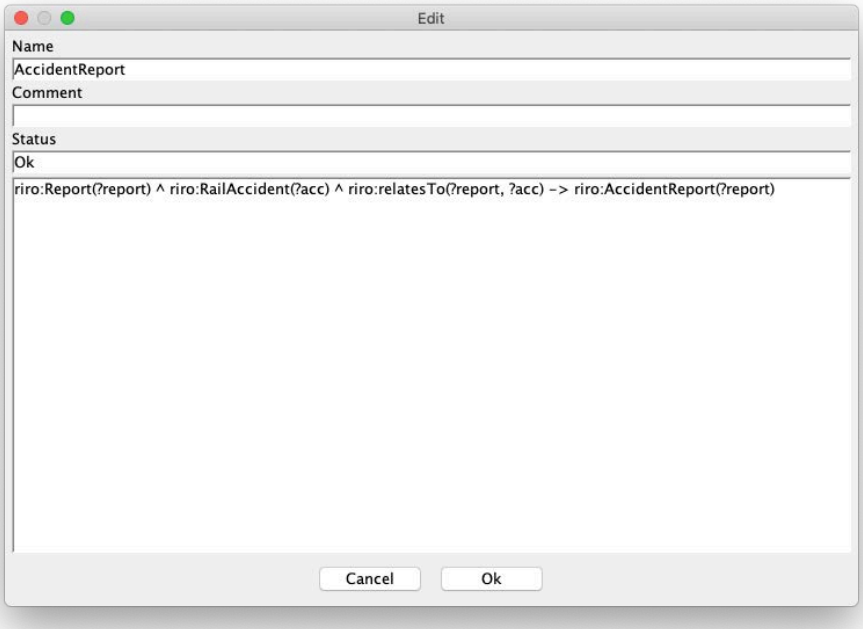


FIG. 44 EXAMPLE OF RULE EDITOR PROVIDED BY PROTÉGÉ

Such a condition has brought a question:

> *Many ontology models can only be manipulated by relevant professionals; how can we enable those who are not familiar with ontologies to use them?*

Little has been done to help non-IT experts working with ontology-based inference on existing data; and in the UK railway industry, decision makers often generate decisions based on data whereas they are not necessarily to be IT expert. Therefore, it is important to allow non-IT experts to discover value from existing data with a holistic view and capability of inference. Supposing there is such a system that allows people without a professional background

to formalise inference rules, e.g. 'If-Then' logic, machines can run some preliminary tests on data, providing initial answers to users' questions. The system does not require the user to have an IT- or ontology-related background, being interactive and intuitive. However, there is no clear evidence that such a tool exists. Most ontology tools require a certain level of understanding of programming while many ontology-based system developers seem to presume that their work is for IT professionals when rules are involved.

This chapter will investigate why some commonly known tools that can edit ontologies are still professional-oriented, and in order to enable non-ontology professionals to use ontologies, a graphic ontology rule designer plus validator will be presented. It aims to inspire other ontology tool developers to allow easier manipulation for non-ontology professionals. To demonstrate this, the following objectives are to be achieved:

1) Deliver an example of a tool that allows non-ontology professionals to manipulate ontology rules, which does not require prerequisite knowledge
2) Discuss how the proposed tool can help professionals working in the UK rail industry to improve their existing process

Contributions of this chapter include:

- Address the necessity of allowing non-ontology railway professionals to edit ontology rules to enrich the current software ecosystem of ontologies

- Present a graphic rule editor that allows users to edit a rule without coding
- Conclude future development directions of supplementary software of ontologies

6.2 The Need to Lower Barriers to Edit Ontology Rules

Due to the sheer amount of stakeholders and information systems involved within the industry (Brewer, 2011), an integrated system has been demonstrated to be beneficial to railway operation by breaking the barrier while exchanging knowledge and data (Köpf, 2010; Morris et al., 2014; Roberts et al., 2011; Saa et al., 2012; Tutcher et al., 2017; Umiliacchi et al., 2009, 2011). However, it is necessary to lower the entry level for using and developing an ontology-based system, to gather more interest and recognition across the industry.

As mentioned previously, ontologies can capture human knowledge, encoding it in a way that machines understand. It is reasonable to assume if an ontology-based system allows users without a relevant professional background to formalise their knowledge in the form of ontology rules, plus ontologies' benefits, more attention can be drawn to ontology-based applications.

Indeed, while some rules can be coded into ontologies during their development, it is difficult to cover everything in the first place. Personnel such as

trackside engineers and decision makers who have abundant knowledge in their fields are not directly involved in the encoding process as their knowledge is extracted by ontology developers via methods such as interviews or questionnaires (Guarino, 1997). If they need to update or supplement what has been captured in ontologies, they still need to seek help from professionals.

While ontologies intended for the UK railway industry already exist (Tutcher et al., 2017), the question remains of how we can allow non-ontology professionals to work with them. An outstanding trait of ontologies is that they can infer new facts based on certain logic and rules (Horrocks et al., 2004; Munir and Sheraz Anjum, 2017; Sirin and Parsia, 2004). The challenge is that industry professionals seem unwilling to spend additional effort on learning SWRL rules, whereas existing tools require users to code to some extent.

Therefore, it is reasonable to assume that it is beneficial to allow non-ontology professionals to edit, assert and insert rules by a 'drag and drop' method, which enables them to use ontologies without fully understanding the technology.

The Semantic Web Rule Language (SWRL) is a semantic rule language that describes human logics, i.e., from antecedent to consequent, to help machines process and express inference with ontologies (Horrocks et al., 2004). It underlies the Semantic Web system, which is also an important part that underpins the capability of inference, and as a W3C's standard, it has been

under active development (World Wide Web Consortium, 2012). Many existing tools and reasoners can handle rules in SWRL to infer facts based on knowledge models as discussed in Chapter 2, so that it is necessary to allow those who cannot code ontology rules in SWRL to edit SWRL rules to generate more value from existing data with their corresponding expertise in the UK railway industry, as illustrated in Fig. 45. It targets to bridge the gap between domain experts' knowledge and digitalised formal knowledge representation. Therefore, to answer the survey candidates' concerns and to fill in the gap of lack of graphic ontology rule editor, a prototype graphic SWRL rule editor with a validator will be presented.

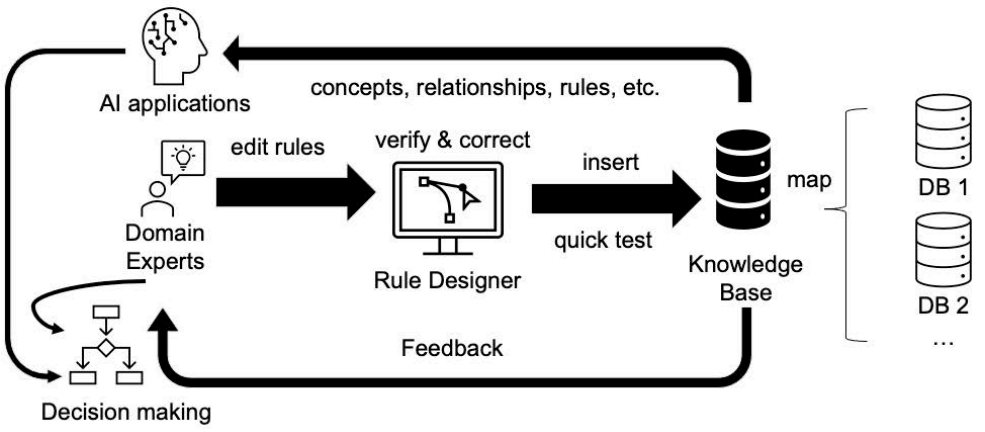


FIG. 45 DEMONSTRATION OF HOW THE RULE DESIGNER BRIDGES THE GAP BETWEEN NON-IT DOMAIN EXPERTS AND DIGITALISED KNOWLDEGE BASE

6.3 An SWRL Rule Design Kit

In this section, an SWRL rule editor kit will be presented. The kit consists of three parts: the navigator, designer and validator.

6.3.1 High-Level Architecture of SWRL Editor

The high-level architecture is depicted in Fig. 46; there are four major components: the graphic designer, parser, validator, and storage connector.

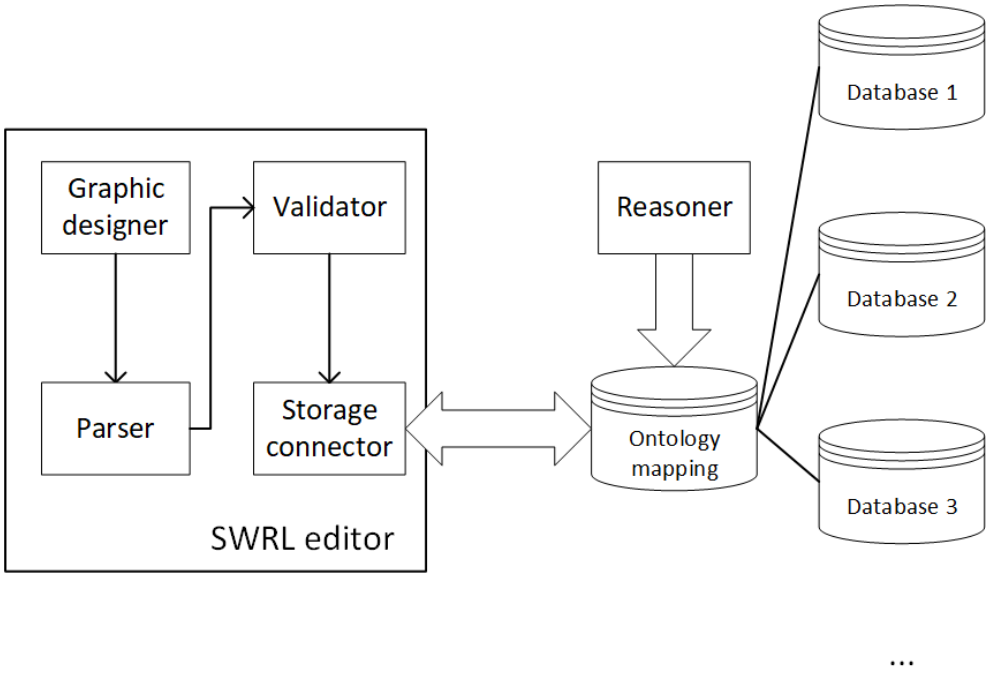


FIG. 46 HIGH-LEVEL ARCHITECTURE

As discussed in Chapter 2, the triple store, a kind of database that is optimised to store ontologies for semantic queries, is an integral part of using ontologies. It allows users to store ontologies; in Fig. 46, 'ontology mapping' denotes the triple store that stores the ontology. Stardog Community

Version has been adopted as the triple store in this project owing to its reasoning ability (Clark et al., 2011; Stardog Union, 2017).

The end-user interacts directly with the SWRL editor. It has been assumed that the user has access to data in the form of triples (i.e., subject-predicate-object) from the triple store, and also basically understands the class of data it belongs to as well as the data properties and object properties. The editor includes a validator which is designed to detect errors in the given diagram and rectify them.

6.3.2 FLOW

The high-level process is illustrated in Fig. 47; the flow is elaborated in Fig. 48.

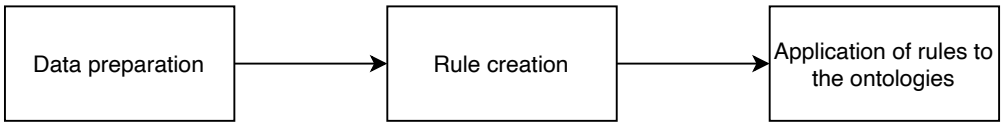


FIG. 47 HIGH-LEVEL ILLUSTRATION OF THE FLOW

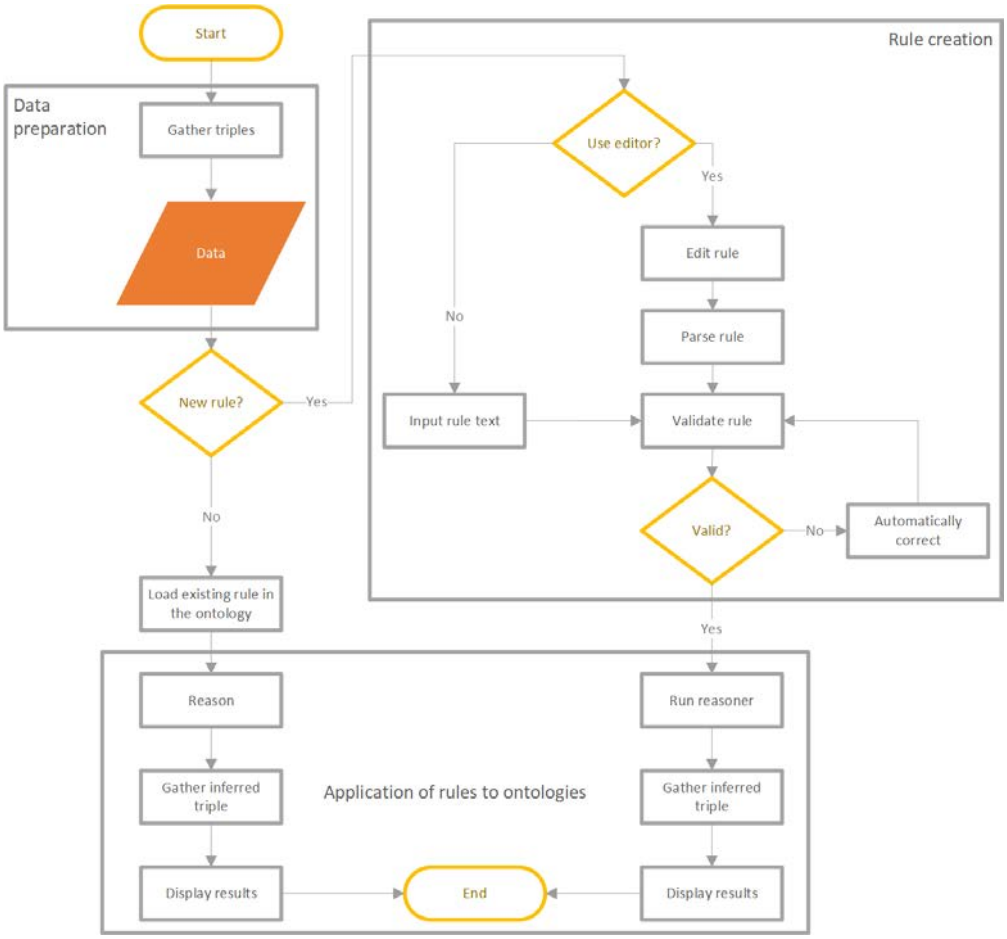


FIG. 48 HIGH-LEVEL FLOW

The process starts by gathering the data which is awaiting analysis. The user then has an option to use existing rules in the ontology or edit a new one instead. The processor accepts Protégé-like SWRL rules, enabling people who are familiar with SWRL rules to utilise the validator. As an example, suppose that a rule is '***Person(?p), hasSibling(?p, ?m), Man(?m) -> hasBrother(?p, ?m)***', as illustrated in Fig. 49.

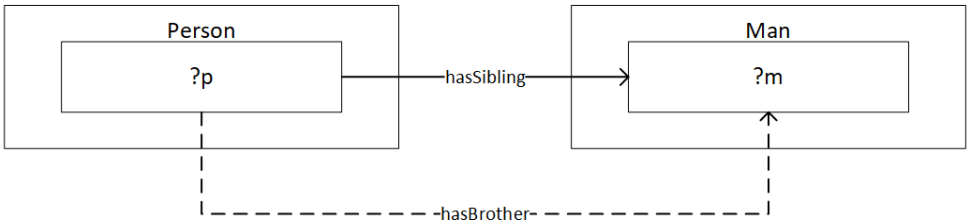


FIG. 49 GRAPHIC ILLUSTRATION OF A PROTÉGÉ-STYLE SWRL RULE

Once the editing is submitted to the validator, the validator checks the correctness of the rule and makes corrections if necessary. When this procedure is completed, a temporary database is created in the triple store and the reasoner is called to infer new facts according to the newly asserted rule. The results are formatted and shown to the user at the end.

6.3.3 A Graphic Rule Designer For SWRL

The interactive rule designer has two modules, a user interface renderer and a back-end processor, which are backed by an open-sourced JavaScript library, mxGraph[47] (JGraph Ltd, n.d.), and an open-sourced Python package, Owlready2[48] (Lamy, 2016), respectively, as illustrated in Fig. 50.

6.3.3.1 Renderer

The renderer was implemented with mxGraph in JavaScript, that directly shows the editing process. A drawn diagram is encoded in an XML string which is then passed to the back-end processor. The processor has three

[47] More details are available at https://github.com/jgraph/mxgraph; mxGraph is an open-sourced JavaScript library that can render a diagram in a secure and scalable manner.
[48] More details are available at https://owlready2.readthedocs.io; Owlready2 is designed to realise ontology-oriented programming in Python; it can also load OWL ontologies as Python objects for manipulations, run reasoning over the graph, initiate in-memory triple store, etc.

modules, the ontology loader, rule parser and validator, which are called in order. Once the encoded diagram is parsed by the rule parser, the validator validates it and correct errors according to the loaded ontology.

The renderer renders the user interface as demonstrated in Fig. 51, where there are three kinds of shape, box (rectangle), oval and diamond. The user interface allows users to drag and place the aforementioned shapes to represent the concepts captured in ontologies. The usage of these shapes is shown in Table 17.

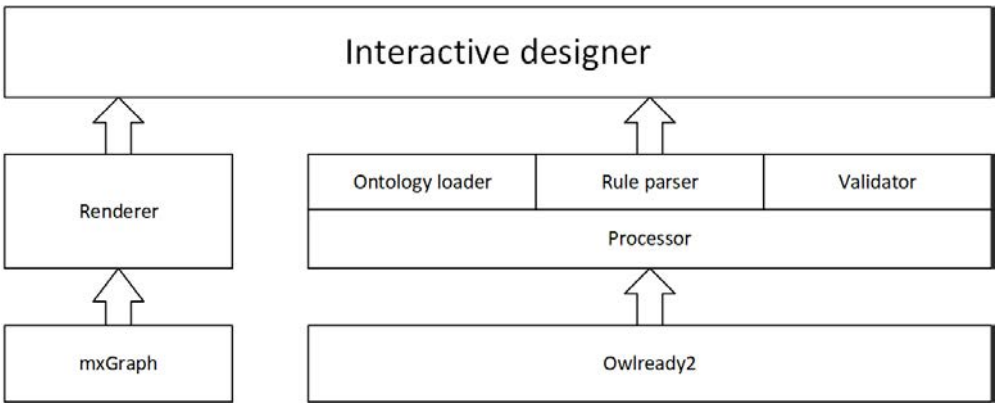


FIG. 50 MODULE ARCHITECTURE OF THE RULE DESIGNER

FIG. 51 SCREENSHOT OF THE USER INTERFACE

A connector is used to connect shapes, which can be also seen as an abstract connection between concepts. The only scenario in which a connector is labelled is when built-ins for SWRL are used.

The rule head and body are distinguished by groups, forming the structure ***Body -> Head*** (i.e., Protégé-style rule). Conceptually, the rule body represents prerequisites which have to be met in order to infer the concepts implied by the rule head. There must only be two groups in a graph, which are connected by a connector to realise the *Body -> Head* structure. Overall, when connecting groups, the connector connects from an antecedent group to a consequent group.

Supposing that there is an ontology that captures the concepts of a family, a rule can be inserted based on the fact that we know if a man's son (instance A) is older than another son (instance B), then A has a younger brother B. The graphic depiction of such a rule is illustrated in Fig. 52.

TABLE 17 USAGE OF SHAPES

Shape	*Usage*
Rectangular	A class in which instances belong
Oval	A property
Diamond	A value that can be a Boolean, a string or a number

This graph can be transformed into a rule:

Man(?man), hasSon(?man, ?son), hasBrother(?son, ?bro), hasAge(?son, ?age_s), hasAge(?bro, ?age_b), greater-Than(?age_s, ?age_b) -> hasYoungerBrother(?son, ?bro)

Under the circumstance where a specific value is involved, an example is demonstrated in Fig. 53, representing that if a person's age is greater than 18, they are an adult. The rule can be translated as:

Person(?p), hasAge(?p, ?age), greaterThan(?age, 18) -> isAdult(?p, true)

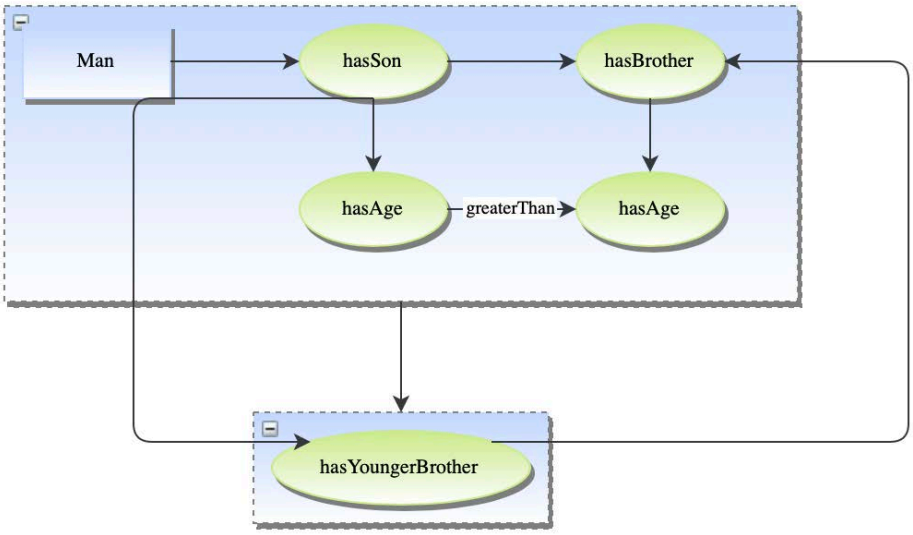


FIG. 52 EXAMPLE OF RULE GRAPH DRAWN IN THE DESIGNER

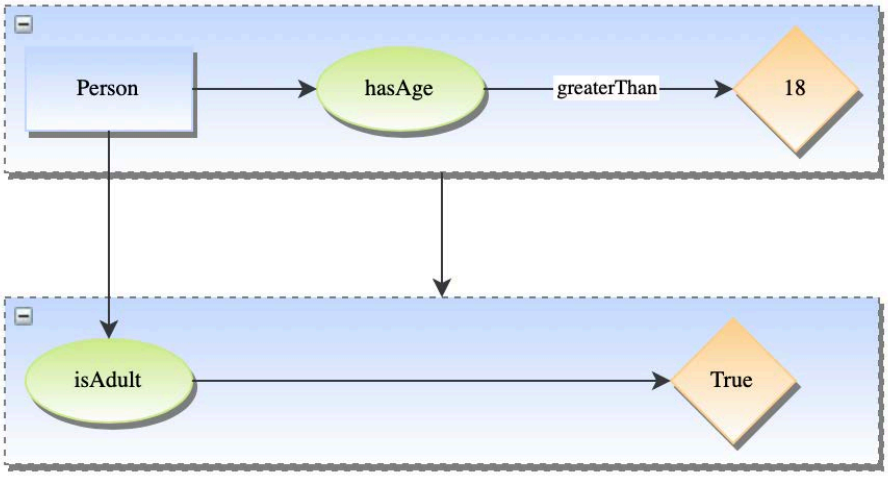


FIG. 53 EXAMPLE OF RULE GRAPH WHEN VALUES ARE INVOLVED

Please note that in both examples, variables are defined for readability. In practice, they are randomly generated without any duplication.

The graph is essentially an XML string from the machine's point of view. The graph is automatically encoded and passed to the processor once the editing is completed.

6.3.3.2 PROCESSOR

The processor, as depicted in Fig. 50, contains three sub-modules, the ontology loader, graph parser and validator. When the processor finishes validating the given graph, the rule can be submitted to a Web Ontology Language (OWL) reasoner which is capable of inferring consequent facts based on the rule in SWRL.

The ontology loader can load ontologies encoded in RDF/XML or OWL into the memory. It will store classes, properties and instances for the next steps. The loaded ontology is an instance of the class Ontology from Owlready2 (Lamy, 2016). Namespaces are read by the loader, too. It avoids the user spending effort on locating the correct namespaces.

Once the graph is submitted, the processor 'translates' the graph. The flow of the translation process is illustrated in Fig. 54.

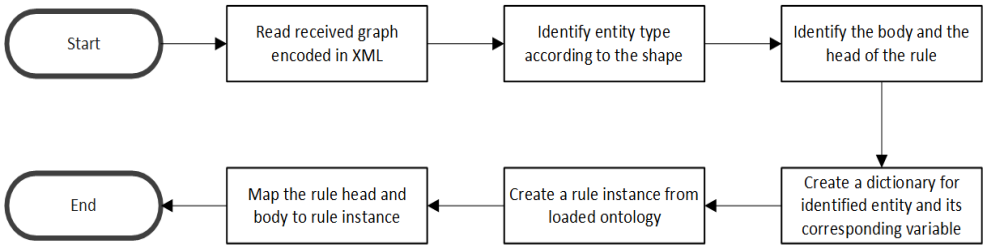


FIG. 54 FLOW OF THE TRANSLATION PROCESS

The parser needs to ensure that there are two and only two groups, the rule body and head, respectively, presented in the diagram. If the diagram meets this requirement, the parser extracts the shapes and connectors for each group separately. The parser converts the XML string into a Document Object Model (DOM) object in Python, gaining access to the value labelled to shapes. The labels put on shapes are stored as an attribute '*value*' of the tag of a shape. For example, Fig. 55 shows a snippet of XML string that represents a set of instances belonging to a class *Man*:

```
<Rect label="Class" href="" value="Man" id="2">
    <mxCell vertex="1" parent="1">
      <mxGeometry x="340" y="160" width="80" height="40" as="geometry"/>
    </mxCell>
</Rect>
```

FIG. 55 SNIPPET OF XML STRING DENOTING AN INSTANCE OF A CLASS '*MAN*'

In this example, the rectangle represents a class whose value is Man. The parser can translate this snippet to ***Man(?arg)***.

The parser also needs to identify the connection between entities. To realise this, the parser has to understand the source and the target of a connector. For instance, Fig. 56 shows a snippet of received XML string:

```
<Rect label="Class" href="" value="Man" id="2">
      <mxCell vertex="1" parent="1">
        <mxGeometry x="210" y="160" width="80" height="40" as="geometry"/>
      </mxCell>
    </Rect>
<Ellipse label="Property" href="" value="hasSon" id="3">
      <mxCell style="ellipse" vertex="1" shape="ellipse" parent="1">
        <mxGeometry x="400" y="160" width="80" height="40" as="geometry"/>
      </mxCell>
</Ellipse>
<Connector label="" href="" id="4">
      <mxCell style="exitX=1;exitY=0.5;exitDx=0;exitDy=0;entryX=0;entryY=0.5;entryDx=0;entryDy=0;"
      edge="1" parent="1" source="2" target="3">
        <mxGeometry relative="1" as="geometry"/>
      </mxCell>
</Connector>
```

FIG. 56 XML STRING GENERATED FROM USER DRAWING

There are two entities in this snippet, a class *Man* with id 2 and an object property *hasSon* with id 3. The ID of an entity is unique so that the parser can determine the direction of a connector based on attributes '*source*' and '*target*' within the tag '*mxCell*' under the tag '*Connector*'. From the example, it can be seen a connector connects the entity '*Man*' and '*hasSon*' in Fig. 52, implying that the instances aggregate '*?b*' must satisfy the condition *men (?a) who have a son (?b)* (i.e., *Man(?a), hasSon(?a, ?b)*). The only situation where the connector can be labelled is when using the SWRL built-in functions to compare or calculate. The spaces or tabs in the label will be removed by the parser to normalise the input strings.

When the connector connects two groups, the source group is the body (i.e., antecedent) of the rule, while the target group is the head (i.e., consequent) of the rule. The parser generates an object that eventually represents the SWRL rule from Owlready2. The object is readable by the validator.

The validator is the key to ensuring the correctness of the graph. A high-level architecture diagram of the validator is illustrated in Fig. 57. The validator takes entities loaded in the memory and given by the diagram as inputs, generating a suggestion as output. The flow of the validation process is shown in Fig. 58. The validator checks two things: if the given entity is presented in the loaded ontology and if the given hierarchy is the same as the ontology.

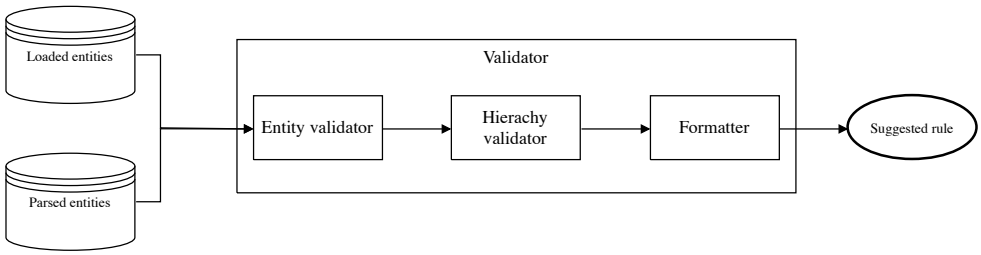


FIG. 57 ARCHITECTURE OF THE VALIDATOR

The first step is to ensure that the entities presented in the rule are legitimate for the ontology. If any are not, the validator can correct them according to similarities in spelling, using the Python package FuzzyWuzzy (seatgeek, n.d.) to conduct a fuzzy search and replace the wrong entities based on the highest score which implies the most similar entity in the ontology.

Second, to find the hierarchy defined in the ontology, some example instances have to be extracted. They are used as references when a hierarchical error is found by the validator. Instances can be extracted via SPARQL from the original ontology with given classes. Because the *Classes* presented in the rule have been already corrected in the first step, if the type of entity

is a ***ClassAtom***, the hierarchy verification of this entity is skipped. Otherwise, the validator has to determine whether the given entity is an ***IndividualPropertyAtom*** or ***DatavaluedPropertyAtom***. An ***IndividualPropertyAtom*** connects individuals of a class to another individual or a set of individuals (e.g., ***entity(?individual_a, ?individual_b)***), while a ***DatavaluedPropertyAtom*** connects individuals to values (e.g., ***entity(?a, true)***).

Once the missing intermediate entity is determined, it can be created according to its type according to the correct structure specified in the ontology. This process is repeated until the rule passes the validation.

The last step is to assign new arguments to missing entities, linking them to entities originally in the graph. For instance, supposing the original wrong structure is ***A(?a), C(?a, ?c)***, the correct structure could become ***A(?a), B(?a, ?b), C(?b, ?c)***, where property *B* has been inserted into the rule with its argument *?a*, *?b*.

6.3.4 BACK-END ONTOLOGY

RaCoOn has been employed as the knowledge model owing to its comprehensive description of rail assets; it has also been recognised and adopted for data architecture in the C4R project for infrastructure and operation managements (Capacity for Rail, 2017; Tutcher et al., 2017) as discussed in Chapter 2.

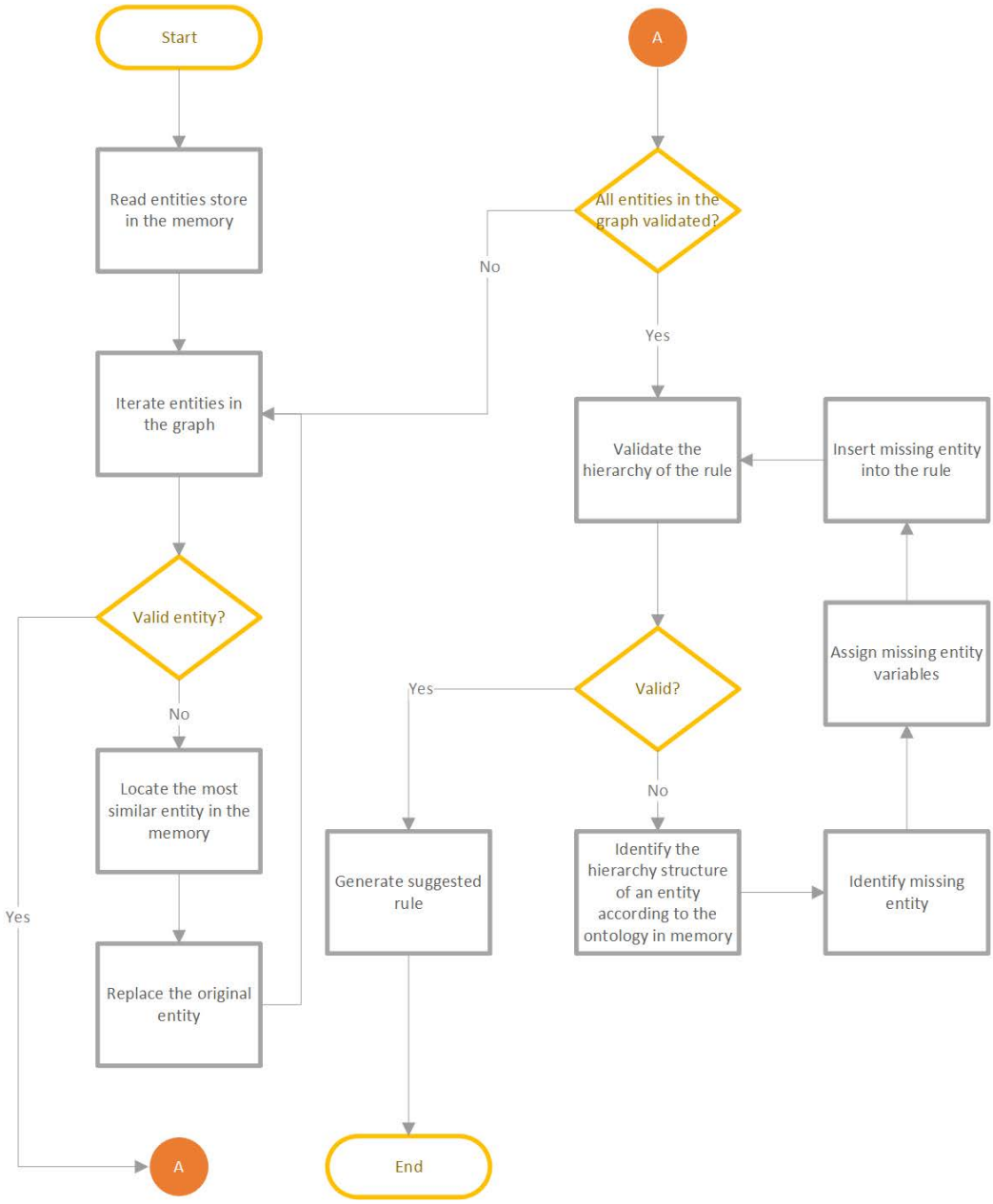


FIG. 58 FLOW OF THE VALIDATION PROCESS

6.4 CASE STUDY

As mentioned above, dedicated ontologies have been researched and developed for the railway system in Europe (Köpf, 2010; Tutcher et al., 2017; Verstichel et al., 2011). In the UK specifically, Network Rail (2017a) and RSSB (TSLG, 2012) have demonstrated their will to adopt ontologies to realise further data integration and better data availability. However, working with ontologies has not been properly recognised within the industry or third-party developers. A lack of understanding of ontologies and their usage has hindered the development of ontology applications in the UK rail industry. Despite proven benefits, there is still no clear evidence showing there is any development of ontology applications. According to the survey discussed in Chapter 4, that was carried out to collect opinions towards ontologies in the UK railway industry, 60% of volunteers have heard of ontologies and half of them have used ontologies. Professional developers and some researchers working in the rail industry tend to stick with relational databases as they are more mature and also faster to deploy. However, despite some participants having no plans to use ontologies in the future, they would be interested if there were tools available and no demand for them to master ontology modelling.

To address the lack of tools for ontologies, this case study will demonstrate how the proposed interactive editor could help when a decision maker

needs to have a general understanding of whether a site has a potential low adhesion hazard, as a response to objective 2).

6.4.1 Temperature and Low Adhesion Hazard

The rail low adhesion phenomenon is complex, relating to many factors such as weather conditions, track contamination, etc. For demonstration purposes, only temperature has been taken into account here. RSSB research projects (T1077 The effect of water on the transmission of forces between wheels and rails and T1042 Investigation into the effect of moisture on rail adhesion) have investigated the effect of water and moisture, i.e. the wet rail phenomenon, on wheel/rail interaction, concluding that low adhesion is more likely to occur during drizzle, when there is dew on the rail head, in misty conditions and in the momentary transition between wet and dry rail (RSSB, 2018a, 2018b; White et al., 2018). Therefore, when the temperature drops below the dew point where water in the air can form dew and adhere to the rail head surface, low adhesion can occur, hence requiring attention.

According to GE/GN8540 Guidance on Low Adhesion between the Wheel and the Rail – Managing the Risk[49], G2.1.4 specifies that infrastructure managers are provided with weather data and G2.2.1 clarifies that the infrastructure manager should identify sites where low adhesion may occur. When managing low adhesion, a decision maker might want to predict the low adhesion condition based on the weather forecast for a site in order to take

[49] The guidance is available at https://standards.globalspec.com/std/1665738/GE/GN8540

precautionary measures if necessary. To do so, they can refer to the current dew point and current ambient temperature.

A decision maker will not necessarily understand coding or perform data extraction from databases, but they will have profound experience and knowledge in determining the low adhesion hazard. It is not a simple task to predict the condition without sufficient programming and data analysis, especially if there are many different types of data which need to be taken into account. The decision maker might be capable of making a subjective prediction, but the most secure way would still be comprehensive data analysis.

However, RaCoOn lacks weather representation. Thus, RaCoOn has been extended with the capability to represent weather parameters such as temperature, humidity, weather condition, wind, etc. A Python script was also developed to request weather information from OpenWeather[50]. The received weather data is in JSON, which is thereafter converted to the form of triples by another script and finally captured by RaCoOn. The weather data is associated with location data which has been already captured by RaCoOn beforehand.

[50] More details are available at https://openweather.co.uk/about

6.4.2 DRAW A RULE

To simplify the rule, it can be seen that if the current temperature of a location is below the current dew point, this location has a potential low adhesion hazard. The rule can be drawn in the designer as illustrated in Fig. 59.

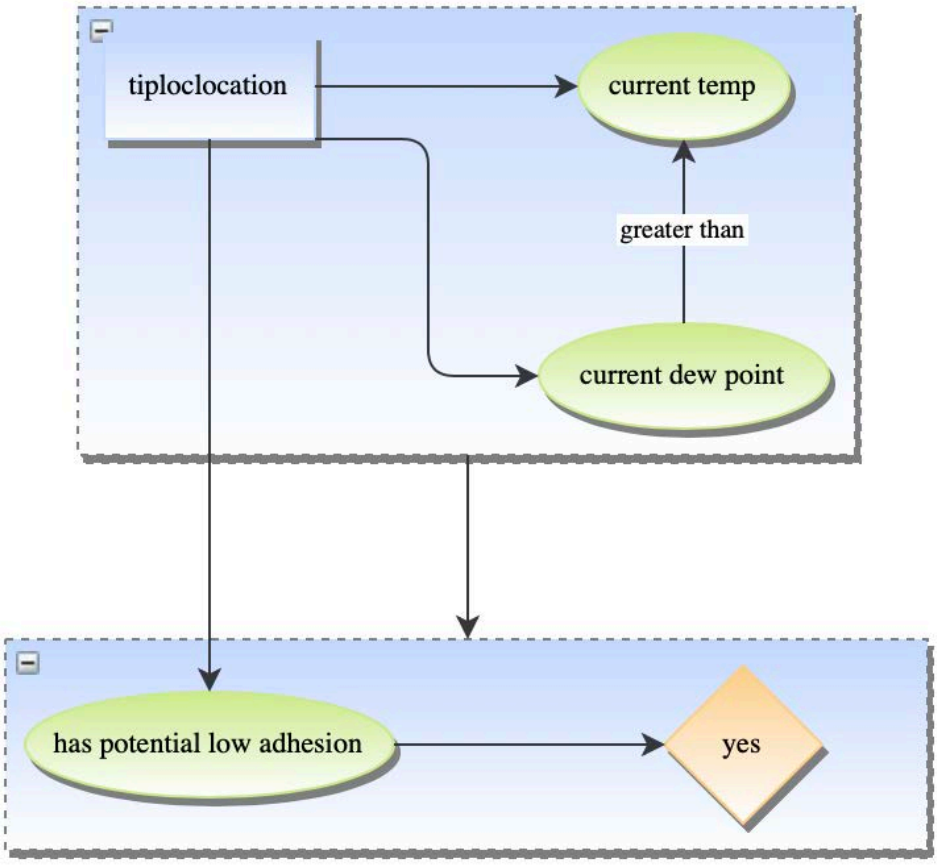
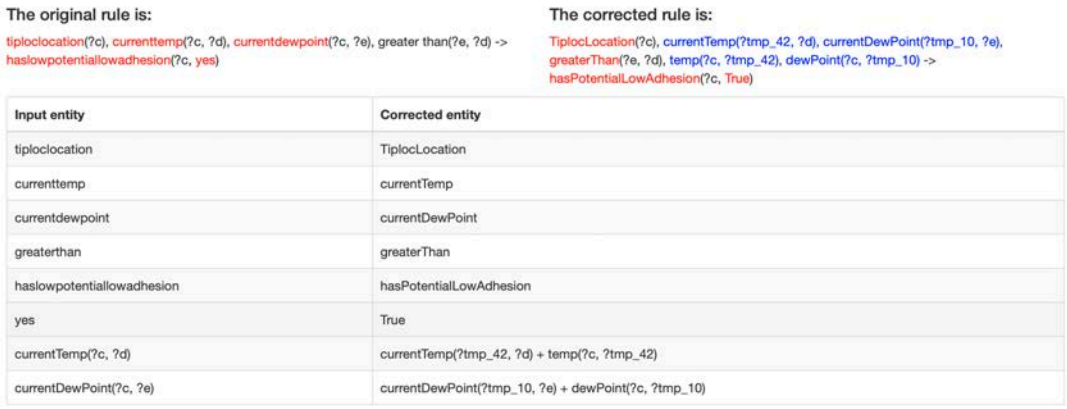


FIG. 59 RULE DETERMINING POTENTIAL LOW ADHESION

6.4.3 Validation and Correction of the Drawn Rule

After submission, the rule is corrected and listed next to the original rule, as shown in Fig. 60; in the user interface, wrong entities are marked with red while structure errors are coloured in blue.



The original rule is:

tiploclocation(?c), currenttemp(?c, ?d), currentdewpoint(?c, ?e), greater than(?e, ?d) -> haslowpotentiallowadhesion(?c, yes)

The corrected rule is:

TiplocLocation(?c), currentTemp(?tmp_42, ?d), currentDewPoint(?tmp_10, ?e), greaterThan(?e, ?d), temp(?c, ?tmp_42), dewPoint(?c, ?tmp_10) -> hasPotentialLowAdhesion(?c, True)

Input entity	Corrected entity
tiploclocation	TiplocLocation
currenttemp	currentTemp
currentdewpoint	currentDewPoint
greaterthan	greaterThan
haslowpotentiallowadhesion	hasPotentialLowAdhesion
yes	True
currentTemp(?c, ?d)	currentTemp(?tmp_42, ?d) + temp(?c, ?tmp_42)
currentDewPoint(?c, ?e)	currentDewPoint(?tmp_10, ?e) + dewPoint(?c, ?tmp_10)

FIG. 60 COMPARISON OF RULE BEFORE AND AFTER CORRECTION

In this example, the validator has 'guessed' what the user meant and rectified the rule. For example, the user labelled '*current temp*' which is not a valid entity in the ontology. The validator located the most similar entity '*currentTemp*' and replaced the original in the corrected rule by a series of lexical matching against RaCoOn. In the rule head, the user input '*yes*' cannot be recognised by the reasoner because it is not capable of relating '*yes*' to the Boolean value '*true*', despite this seeming very intuitive to a human. The validator ensures that the user input can be converted to machine-understandable format, transforming '*yes*' to '*True*' in this scenario.

Moreover, the hierarchy presented in the diagram has errors, too. An example hierarchy for temperature data defined in ontology is illustrated in Fig.

62A. Compared to the user input illustrated in Fig. 62B, it can be seen that *TiplocLocation has temp and temp has currentTemp,* whereas the user might not be aware of this and put *TiplocLocation has currentTemp* intuitively. The validator has addressed such issues and corrected them with replacement arguments, assuring the correct structure is passed to the reasoner. It can be seen that the user was only required to draw a rule that reflects the 'low adhesion hazard rule', while the rest was handled by the back-end processor.

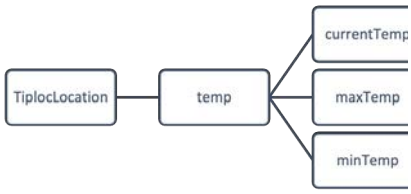


FIG. 62A DATA ARCHITECTURE FOR

FIG. 62B INCORRECT HIERARCHY GIVEN IN THE EXAMPLE

In the example illustrated in Fig. 63, it can be seen that an instance's property, ***hasPotentialLowAdhesion***, has been inferred as *True*, even though it was not explicitly defined in the ontology, proving that it is possible to use a reasoner to realise simple data processing with a drag-and-drop rule.

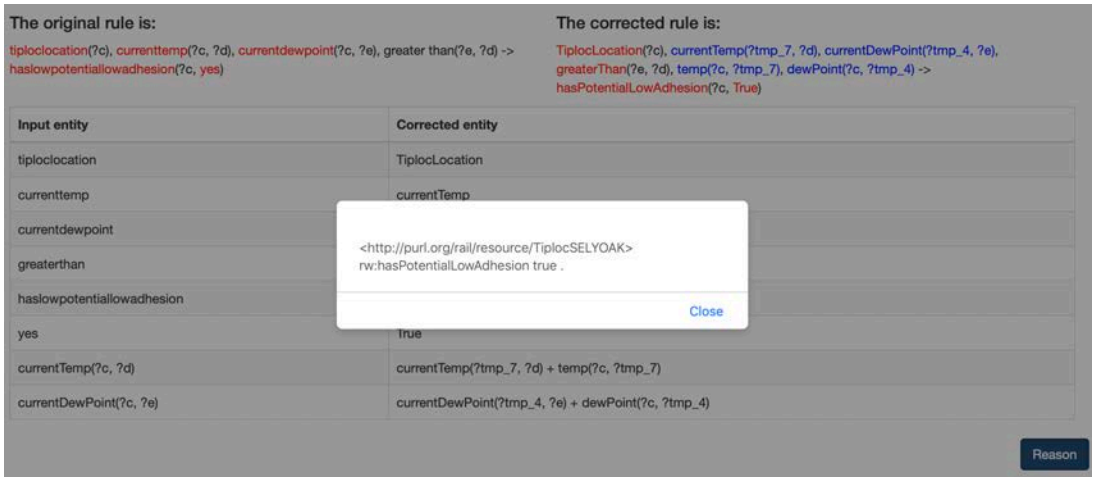


FIG. 63 RESULT OF REASONING

After correcting wrongly defined entities and wrong hierarchy, the processor passed the rule to the reasoner which provided the result that met the expectation. The extended RaCoOn captured weather data and location data, correspondingly linking these two kinds of data; based on the given rule, the reasoner inferred the fact (i.e., if a location has a potential low adhesion hazard) in conjunction with the knowledge model and captured data, ultimately presenting the result to the end-user. The whole process required no human intervention; the users was only requested to draw a rule that accords with their knowledge. This allows personnel who are not familiar with ontologies or even IT technologies, such as maintenance operators and in-field engineers, to interact with Linked Data and analyse data without mastering coding or ontology-related technologies.

6.5 User Acceptance Testing (UAT)

In order to ensure that the proposed rule designer can satisfy the end-user's requirements, UAT is necessary to determine whether or not a system will be accepted (Hambling and Van Goethem, 2013).

UAT can be performed in a structured way that is set up with a user case scenario (Hambling and Van Goethem, 2013). To prove that the proposed tool can achieve a similar result yet does not require the user to have strong SWRL knowledge, and assuming that the triple store and local editing environment have been appropriately set up, the UAT process consists of two parts:

- Replicate the same user case scenario described in section 6.4
- Perform the same rule editing process using Protégé-style SWRL code and run the reasoning function as the comparison

The acceptance criteria are:

- The candidate can perform the same operation as described in section 6.4 without prerequisite knowledge of SWRL coding rules
- The candidate agrees that using the drag-and-drop method is more intuitive and approachable than coding in Protégé

The flow of the validation process was developed with reference to a UAT guideline (Hambling and Van Goethem, 2013); Fig. 64 shows the process performed by the testers.

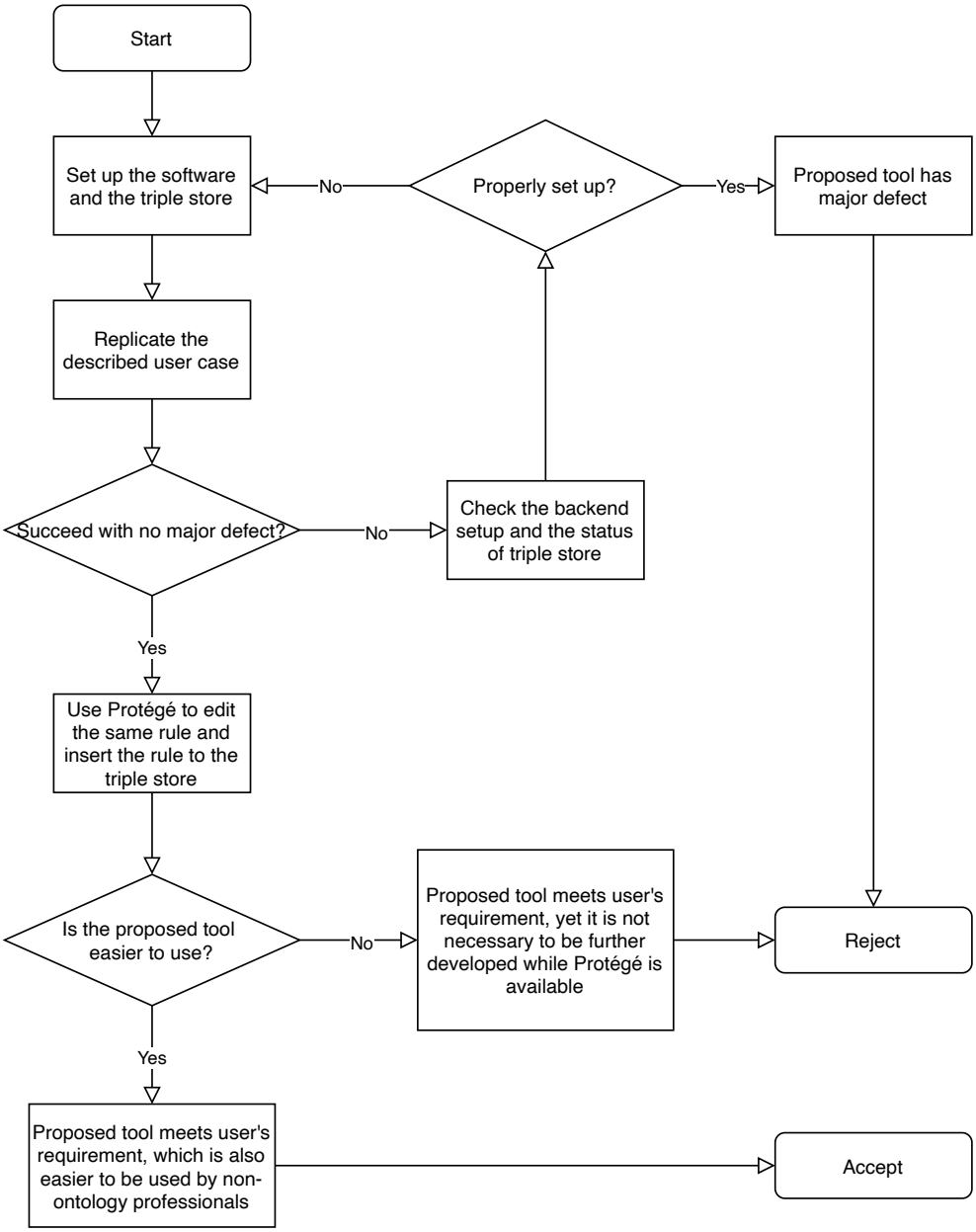


FIG. 64 VALIDATION PROCESS FOR THE RULE DESIGNER

The test was conducted on a remote server[51], and the testers were required to complete a questionnaire[52] after testing the proposed solution. There were 17 testers, and their responses are shown in Table 18. The Cronbach's alpha is 0.832, showing the questionnaire's high level of reliability, hence its result is acceptable.

TABLE 18 RESPONSES TO UAT

Question	Answer	Frequency	Percentage (%)
What level of using ontologies do you reckon you are at?	No experience	9	52.9
	Know the basis, but rarely use it	5	29.4
	Know well, but rarely use it	2	11.8
	Know well, and often use it	1	5.9
The designer can help you design the rule for the scenario outlined in the instruction.	Strongly disagree	1	5.9
	Disagree	1	5.9
	Neither agree nor disagree	1	5.9
	Agree	6	35.3
	Strongly agree	8	47.1
You did not encounter a major issue	Strongly disagree	1	5.9
	Disagree	0	0

[51] The application was deployed on a server running Ubuntu 18.04.5 with one core and 1024 MB RAM.

[52] Available at http://smartsurvey.co.uk/s/PKIKCD/; the full question list is also available in the Appendix.

while performing the test.	Neither agree nor disagree	1	5.9
	Agree	7	41.2
	Strongly agree	8	47.1
You can complete the scenario outlined in the instruction without knowing how to code SWRL rules.	Strongly disagree	1	5.9
	Disagree	0	0
	Neither agree nor disagree	0	0
	Agree	7	41.2
	Strongly agree	9	52.9
Using a graphical UI is more user-friendly than a coding-based tool (like Protégé).	Strongly disagree	0	0
	Disagree	1	5.9
	Neither agree nor disagree	0	0
	Agree	3	17.6
	Strongly agree	13	76.5
The proposed solution should be further investigated and refined to facilitate wider adoption of ontology-based applications.	Strongly disagree	0	0
	Disagree	0	0
	Neither agree nor disagree	0	0
	Agree	6	35.3
	Strongly agree	11	64.7
The proposed solution is useful so that it can be accepted.	Strongly disagree	0	0
	Disagree	0	0
	Neither agree nor disagree	0	0
	Agree	4	23.5
	Strongly agree	13	76.5

The testers' level of ontology knowledge is distributed as illustrated in Fig. 65; it can be clearly seen that over half the testers have no experience in ontologies. This justifies the selection of testers with reference to the target user group of the proposed solution, who are likely to know little about ontologies.

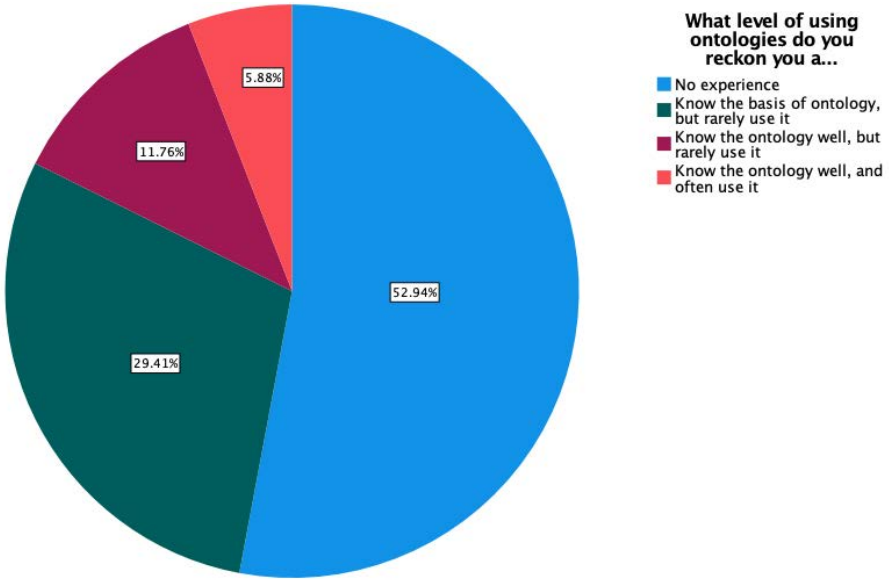


FIG. 65 TESTERS' LEVEL OF USING ONTOLOGIES

All testers agreed that the proposed solution can be accepted, 13 of them agreeing strongly; thus, it is safe to presume that the proposed solution has fulfilled its established objectives. It is worth noting that 10 out of 13 testers who strongly accepted the solution also strongly agreed that this solution should be further investigated in order to facilitate wider adaptation of ontology-based applications, and all other testers also agreed its value for additional investigation, as illustrated in Fig. 66.

However, despite general acceptance of a graphical drag-and-drop method for designing a rule, one tester, an ontology professional, prefers a traditional tool such as Protégé to gain a higher level of flexibility and customisation, because he failed to duplicate his rule that can be done in the form of SWRL code. The limitation of the proposed rule designer is mainly around the incompatibility of some built-in functions of SWRL plus a lack of support for URI, which are likely to be mastered by ontology professionals.

Another issue is the stability of the software. A few users encountered major or minor performance issues while testing. One tester reported that the application crashed, and he had to wait for the server to be restarted, gaining a bad user experience. An investigation was conducted into this issue. It arose from the limited RAM (1G in total) of the server on which the application was deployed. Due to insufficient work on optimising the application, the RAM limitation made the server kill the back-end thread that handles the validation and triple store process.

Overall, despite the existing issues, the proposed designer has gained testers' acceptance; it is a more user-friendly solution and can help users with little knowledge of ontology to work with SWRL rules. All testers reckoned it should be accepted, and further investigation and development in the relevant field are recommended to facilitate wider adoption of ontology-based applications.

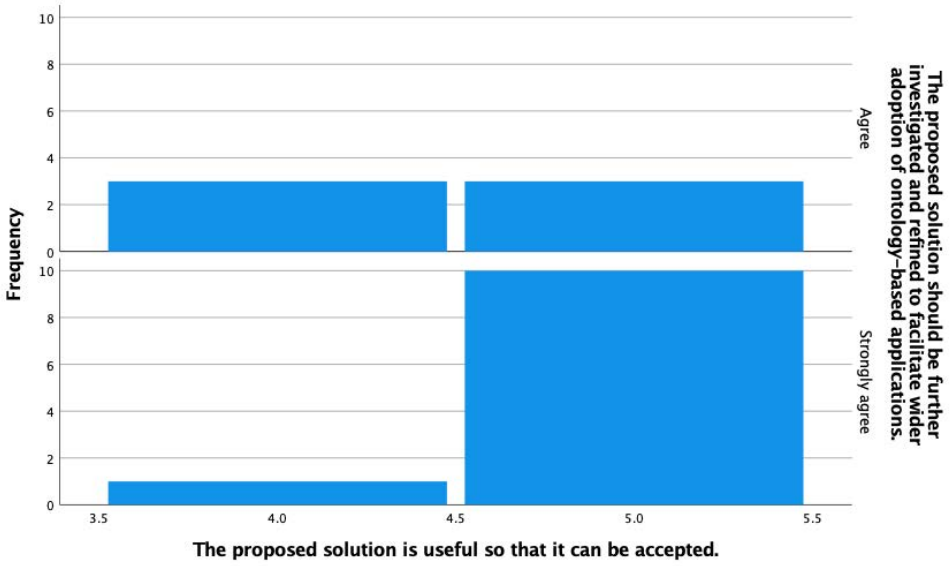


FIG. 66 COUNT OF RESPONSES TO WHETHER THE PROPOSED SOLUTION CAN BE ACCEPTED

6.6 Discussion about the Rule Designer

The key idea of this rule designer is to allow non-ontology professionals to edit and insert ontology rules. Ontology rules play an important role in ontology-based applications. Therefore, in order to facilitate the establishment of ontology-based applications in the railway industry, it is of importance to enable those who are not familiar with ontologies to edit and insert ontology rules to ontology models. This tool fulfils the proposed purposes. After correcting wrongly defined entities and wrong hierarchy, the processor passed the rule to the reasoner which provided the result that met the expectation. The extended RaCoOn captured weather data and location data, linking these two kinds of data correspondingly; based on the given rule, the reasoner inferred the fact (i.e., if a location has a potential low adhesion hazard) in conjunction with the knowledge model and captured data, presenting the results to the end-user. The whole process required no human intervention; users were only requested to draw a rule that accords with their knowledge. This allows personnel who are not familiar with ontologies or even IT technologies, such as maintenance operators and in-field engineers, to interact with Linked Data and analyse data without mastering coding or ontology-related technologies. According to the UAT discussed in section 6.5, the proposed solution can be accepted with value for further investigation and development.

This design can be also justified with the existing design of a graphic rule visualiser as part of a famous graph store, Neo4j. Fig. 67 illustrates how a rule is presented, where different elements were represented with different colours. However, the implementation by Neo4j only included a visualiser and lacked an editing function, so that in order to insert a rule to the knowledge model, users still had to code the rule. Interactive design, including 'drag and drop', is the key to increasing user acceptance and experience (Petzold, 2005), which has been adopted by some famous software such as Visual Studio. The design demonstrated in this chapter has avoided requesting users to code, which should be also referenced by other similar future development.

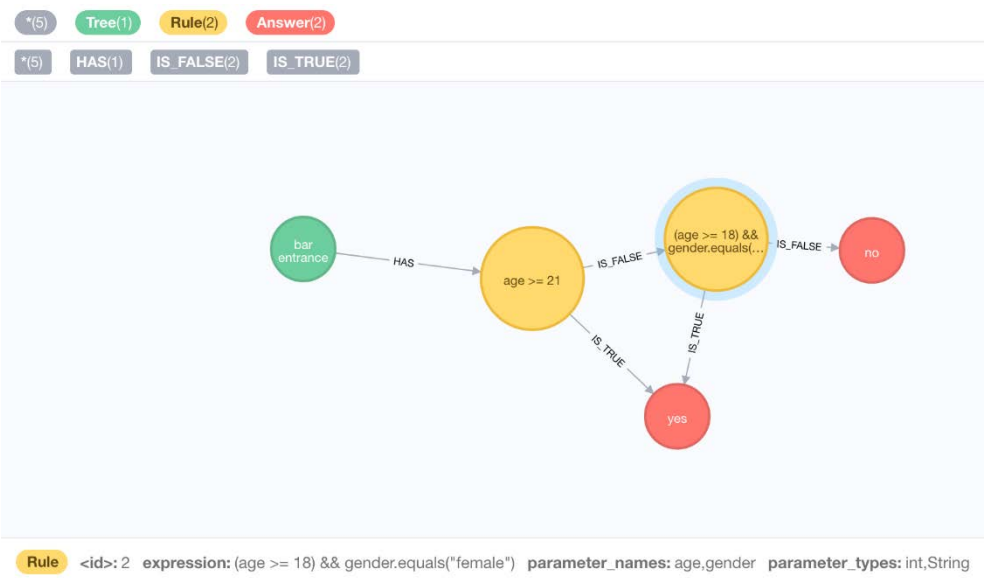


FIG. 67 A SCREENSHOT OF THE RULE VISUALISER OF NEO4J (MARZI, 2018)

Yet, there is still room for improvement:

- When users design a rule, auto-completion or auto-prompt would be helpful. For example, when the user inputs '*tiplocloca-tion*', the designer could automatically provide a suggestion '*Tip-locLocation*'.

- According to one tester who is proficient in using ontologies, the designer should accept a full URI or a '***namespace:suffix***' form of entity representation to enhance flexibility for professional users.

- The validator can only guess what entity the user suggests with reference to a given ontology (i.e., based on what already exists in the ontology); if the corrected entity is not the user's intention, the tool should allow the user to insert new concepts and knowledge (i.e., insert new triples when the ontology does not contain a concept or knowledge that the user seeks).

- The validator can be further generalised to work with other on-tologies.

- Some testers who know little about ontologies commented that despite being capable of completing the given user case scenar-ios, they had little clue of how to conceptualise their own rule without understanding how ontologies were modelled and they failed to figure out exact steps without out referring to the in-structions. They had trouble in identifying what they could do

with the rule designer in the first place, so the UI needs to be further refined to be more intuitive and enable instinctive interaction with the proposed designer. This issue requires further study; it has been discussed in the literature (Bobkowska, 2013; Lee et al., 2013; Villani et al., 2018), plus there is a patent about context intuitiveness (Levermore et al., 2020), a good starting point for future study.

6.7 CONCLUSION

There is evidence revealing how people use ontologies to efficiently manage knowledge in the industry to ease the complexity of data retrieval and heterogeneity of data varieties (Bodenreider, 2008; Ebrahimipour and Yacout, 2015; Munir and Sheraz Anjum, 2017). However, there is no sign that ontologies have been adopted in the actual railway operation in the UK. The conclusion drawn from the survey in Chapter 4 demonstrates that although many people are interested in ontologies, they are deterred because of the requirement to learn much additional knowledge to interact with ontologies and the existing tools thereof. Unlike XML and relational databases, existing tools for ontologies demand users to code; many of them tend to adhere to different coding rules, which means that most ontology models can be only manipulated by relevant professionals. Meanwhile, it is not always feasible to let non-IT professionals learn to utilise ontologies in their jobs where they might have already established a stable pattern to tackle problems. This has

created additional difficulties for ontology-based applications to prosper in the UK railway industry.

Thus, to answer the research question:

> *Many ontology models can only be manipulated by relevant professionals; how can we enable those who are not familiar with ontologies to use them?*

it is necessary to address the issue that the entry level for using ontologies is still high, thus objective 1). This chapter focuses on addressing this issue, providing an interactive solution that allows users to edit and insert ontology rules in SWRL. Despite the room for improvement, the proposed tool has fulfilled the purpose: it helps users to validate and correct the rule they are going to insert. The proposed tool can also help users to make use of existing railway ontologies to perform quick inferences without affecting their existing working habit. To assert a rule drawn in the designer, the validator can analyse what has been captured in the supplemented ontology (i.e., RaCoOn in this case) and rectify error(s), ensuring the rule to be inserted is valid and legitimate to the reasoner. The built-in validator can be also reused as a Python package by other developers if necessary. A response to objective 1) has been provided with the proposed tool, yet further validation with professionals from the industry should be executed. The case study delivered in section 6.4 has responded to objective 2); it is possible to use the proposed tool to edit an SWRL rule by drag-and-drop operations, and the reasoner can

provide a preliminary inference result of the rule being applied to the knowledge repository. The existing solution presented by Neo4j reflected the correct development direction made in this study, and the solution proposed by this study rectified the defect of the rule visualiser of Neo4j where users could not interactively edit a rule.

However, an issue with the proposed solution is that the flow and UI are not sufficiently user-friendly. Some testers without an ontology background reported that despite useful functionality, they were confused if no instruction was supplied at the beginning. It was difficult to identify the context while using the proposed tool. A similar issue has been addressed in the literature (Bobkowska, 2013; Lee et al., 2013; Villani et al., 2018), and a patent on context intuitiveness is also available (Levermore et al., 2020); the question of how to balance the user experience for both novice users and professionals awaits further investigation.

Since the industry has revealed great interest in ontology-based data models, as concluded in Chapter 4, there are reasons to believe that tools such as the proposed rule designer can attract more attention and usage from the UK railway industry. However, many issues are still awaiting a solution; for example, ontology modelling in railway asset management requires expertise in railway assets, condition monitoring, etc., but an ontology developer might not necessarily possess that expertise, hence the difficulty in developing an ontology model. Ontology application requires industry-wide

collaboration that needs effort and funding. Nevertheless, it is still worth assuming that when ontologies do not seem mysterious to developers and users, more advanced tools and more efficient methods for knowledge extraction will be available, consequently helping industries to facilitate more automation and faster and better decision-making.

7 USING ONTOLOGIES TO REPRODUCE EXISTING MANUAL PROCESSES

7.1 BACKGROUND

The need for railway traffic in the UK has grown significantly during recent decades; data published by the ORR show that the annual passenger train km (pkm) rose from 495 in 2010 to almost 530 in 2018 (Office of Rail and Road, 2020). Although the increasing need can bring business prosperity to the UK railway industry, it also brings challenges and presents issues such as increasing demand for capacity. This has resulted in increasing demand for infrastructure maintenance and renewal, which is often presented with infrastructure intervention (Armstrong and Preston, 2019).

To obtain the optimal intervention plan, there is a need to comprehensively analyse the costs. Therefore, to assess the relative costs, some projects have been initiated with the aim to develop cost–benefit analysis tools (Armstrong et al., 2019; Bartram et al., 2008; Ortega et al., 2018; Zhang et al., 2016). To do so, there must be a data model that can be reused to generalise and integrate data from disparate sources. A one-off nature hinders the reusability of the existing data set and it is obvious that it is more beneficial and cost-effective to develop a generalised and standardised assessment method to conduct assessments for multiple scenarios. However, despite useful outputs, the data is rather project-oriented, which means that

it is difficult to reuse the data from these projects. Researchers and developers are often required to collect and cleanse data from scratch, incurring additional time and cost. Based on the conclusion addressed in Chapter 4, it remains unknown whether professionals can use ontologies to reproduce the existing process in the context of business. Based on the discussion in Chapter 6, ontologies can minimise the existing manual process in theory. Therefore, there are reasons to assume that the application of ontologies to the existing manual process is achievable and beneficial.

As discussed in Chapter 2, ontologies have been proven to be capable of modelling domain knowledge in a generic way. Therefore, there are reasons to believe that it is possible to use ontologies to model the data required to assess possible interventions, facilitating a higher level of digitalisation and efficiency. In addition to that, as drawn from Chapter 4, many developers and researchers working in the UK rail industry are not familiar with using ontologies to reduce the manual process; therefore, a research question arose:

> *How can we reproduce some manual processes using ontologies to achieve more digitalised and more effective processes in the railway industry?*

To answer the question, this chapter demonstrates an ontology-based approach, using ground-borne data analysis in Track to the Future (T2F), to

reproduce the current manual data handling process to realise a generically generalised data framework for rail intervention. The following objectives will be addressed:

1) Deliver an ontology-based approach to replicate the process proposed in T2F
2) Demonstrate the approach's applicability to an existing process with a case study

7.2 One-Off Nature

Track 21[53] and T2F[54] are research programmes that develop tools and approaches to model the costs and advantages of various infrastructure interventions. Both projects have identified the impact brought about by rail infrastructure intervention on the environment.

Track 21, funded by the Engineering and Physical Sciences Research Council (EPSRC), aimed to gain insights into the behaviour of track systems and civil engineering infrastructure in order to address and face challenges including more intense usage of the railway, faster movement of trains and less time for maintenance. As part of Track 21, researchers investigated methods to improve ballasted track; they found that Under Sleeper Pads (USPs) can prolong the life of ballast and reduce vibration and ground-borne noise, being a

[53] http://track21.org.uk
[54] https://t2f.org.uk

great candidate to replace long-term (plastic) settlement of railway tracks, despite a small increase in air-borne noise (Track 21, 2015).

The successor to Track 21, T2F, has kept focusing on various rail interventions. As a result of more frequent services and higher train speed, the rail track is under more pressure so that the time left for maintenance has been compressed. Under the circumstances where legacy infrastructure exists in the UK rail system, a potential intervention needs to be carefully assessed before implementation. T2F Project B is exploring the potential benefits of USPs based on the conclusion drawn from the previous work executed in Track 21 that USPs can improve the stability of the sleeper–ballast interface and reduce contact stresses (Track to the Future, 2020). Ortega et al. (2018) used a Vehicle Track Interaction Strategic Model (VTISM) to generate cost values; using the London to Portsmouth line as the case study, they analysed the business value brought by the installation of USPs. It decreased ambient noise and the reduction of ground-borne noise could bring a net financial profit of £30 million to Network Rail, increasing travel quality and reducing maintenance cost as well (Ortega et al., 2018). In 2020, Young et al. developed a transferable method for estimating the economic impacts of track interventions, more specifically, discussing the extent to which the application of USPs can reduce ground-borne noise; the data model they delivered in conjunction with VTISM revealed a high level of transferability.

However, despite the usefulness of the model, it relies strongly on the location and data, and the assessment process is not generic. The lack of a standard data model means that it requires researchers to collect data from various sources every time they need to assess the intervention for a new location. It seems that the existing assessment approach focuses on a single location which is always repeated, and is time-consuming; plus, one-off analysis is often required when an alternative location is presented. Moreover, analysis of the economic impact tends to be separated and carried out by experts, which, to some extent, creates additional difficulties in the data collection process. Although Young et al. (2020) demonstrated the workflow in which they used QGIS to join data from the rail network model and the population grid, mapping the data from the noise table and forming a data package that can be used for further economic analysis to facilitate data reuse, it is difficult to reuse the outcome owing to the lack of a standard data description framework. It has been observed that it is difficult to filter results based on conditions such as the distance to a specific segment of track.

In order to increase the replicability of different locations and allow reuse of further outcomes for a similar study, it is necessary to develop a standardised framework for the assessment of interventions in different locations and operating conditions, and the economic impacts thereof.

7.3 Using Ontologies to Realise a Standardised Framework

To date, standardised input templates for different interventions in T2F have been designed (Armstrong et al., 2020):

1. *Why will this intervention improve rail track systems?*

2. *In assessing the engineering impacts of this intervention what are the key input and output variables?*

3. *What are the main one-off (capital expenditure) and recurrent (operating expenditure) financial costs of the intervention?*

   Sub-question: What is your judgement of the magnitude of these costs and can they be quantified? If so, how?

4. *What are the main operator benefits? Operator refers to both infrastructure (Network Rail and its suppliers) and train services (TOCs and their suppliers).*

   Sub-question: What is your judgement of the magnitude of these benefits, and can they be quantified? If so, how?

5. *What are the main user benefits? User refers to customers of passenger and freight train services.*

   Sub-question: What is your judgement of the magnitude of these benefits, and can they be quantified? If so, how?

6. *What are the main non-user benefits? Non-user refers to users of rival transport systems (air, road), residents or the wider community.*

   Sub-question: What is your judgement of the magnitude of these benefits, and can they be quantified? If so, how?

However, the lack of means to process intervention data prohibits further implementation of the template while most of the data is stored in separate silos. Owing to the benefits ontologies possess, an ontology-based approach to handling and processing data seems to be a solution.

There are several reasons that an ontology-based approach can fit the purpose. First, different types of data coming from disparate sources need to be taken into account; as a result of disparate data sources, data needs to be consolidated. It is common for data to be missing, and the ability to identify and process missing data is of vast importance during analysis. Second, the model has to be generalised enough to preserve and handle data from different data silos, retaining the context of the data. Ontologies have been proven to be capable of achieving this as discussed in Chapter 2; therefore, an ontology-based approach has been advocated to integrate and process data stored in disparate sources in the industry (Morris and Easton, 2018).

The integration works in a way that is similar to a comprehensively centralised database which can efficiently process data storage, update and retrieval; however, ontology-based integration can also capture semantics and contexts (e.g., quantity and units, source, timestamp, type, etc.) plus rules.

The flexibility of ontologies enables the incorporation of other data models so that it is possible to transform and translate data from different sources within an ontology model, which means that it is feasible to change the data from the ontology side of the system rather than across multiple databases. As many legacy systems and systems supported by multiple service providers exist in the UK rail industry as discussed in Chapter 2, the ontology-based integration approach has potential value.

In relation to T2F and following similar projects, the ontology-based integration solution will allow the incorporation of various data sources into a more holistic view, which is believed to be beneficial to broad analytical scenarios.

7.3.1 INITIAL DEVELOPMENT

There are six high-level aspects that are of key interest:

- Impact categories (general)
- Inputs
- Processes
- Outputs
- Outcomes
- Impacts

The proposed aspects should be structured into the ontology. In this case, RaCoOn has been selected as the RDO to provide rail-relevant terms and vocabularies as well as their properties (i.e., predicates); it provides a generic method to describe all the data required for ground-borne noise analysis.

To address the common challenge that different software is used in different silos so that it is difficult for them to communicate with each other, a knowledge model can form a semantic middle layer which enables consistent queries and changes across multiple integrated sources (Lenzerini, 2011). An ontology, in this case, can be seen as a hierarchical data structure consisting of serialised metadata providers, and the metadata they provide can be exchanged and interpreted by different applications.

7.3.2 DATA TO BE INTEGRATED

There are several different types of data to take into account, including railway track information, the noise table, population grid[55] and USP data, as shown in Table 19.

TABLE 19 DATA INTEGRATED BASED ON ONTOLOGY

Data type	Format (filename extension)
Network model	Shape file (.shp)
Network rail track database	Microsoft Access database (.accdb)
Noise table	CSV file (.csv)

It can be seen that each type of data is presented in a different form. Some of them can only be loaded by certain software (e.g., Microsoft Access database file), whereas some are text-based data without context and structure (e.g., CSV file). Yet, despite data being stored in different data silos, silos still have relationships with each other, as illustrated in Fig. 68.

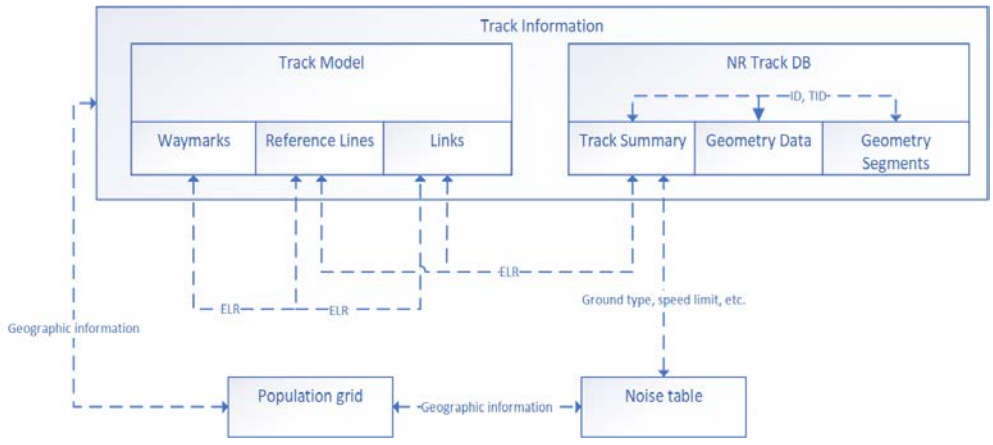


FIG. 68 EXAMPLE OF RELATIONSHIPS BETWEEN SILOS

[55] Sourced from OpenPopGrid Dataset (Murdock et al., 2015)

In this example, Track Model contains three modules, Waymarks, Reference Lines and Links, while Network Rail Track Database contains Track Summary, Track Geometry Data and Track Geometry segments. Sub-modules are also stored in separate silos, despite being sub-modules of one data object concept. These silos are interlinked by a key field, for example, ELR for Track Model and TID for Network Rail Track Database. Although the linkage was known by the researchers, it had to be specified during the data collection process. For instance, it required at least three queries against the database to consolidate Track Model data and present it in QGIS. On top of that, additional queries had to be performed for other objects to form a consolidated data table for further analysis, creating additional difficulty in managing and retrieving the desired data for a specific location, hence many previous works had a one-off nature.

RaCoOn was introduced to map these silos and consolidate data instances into a unified whole. With ontologies, the user only has to run one query to obtain all required data, greatly reducing the amount of work in the data collection process. Admittedly, relational databases can achieve a similar result; however, the dedicated effort to be invested in creating a data table plus future difficulty in performing maintenance make relational databases inferior. Meanwhile, ontologies can map the data on a semantic basis and provide a hierarchical data structure, hence being a decent candidate in this case.

7.3.3 MAPPING DATA FROM SILOS TO RACOON

The original development of RaCoOn did not include some concepts required for ground-borne noise analysis in T2F as described by Young el al. (2020), as shown in Table 20.

The concepts in Table 20 have been modelled in revised RaCoOn, and their ontology classifications are shown in Table 21. It is worth mentioning that imperial units have been adopted besides metric units in RaCoOn as the industry tends to use them. Owing to the flexibility of ontology models, this can be easily implemented.

TABLE 20 CONCEPTS MISSING FROM RACOON

Missing concept	*Brief introduction*
Reference Line	The abstract railway operation line that provides basic line illustration and guidance
Waymarks	Waymarks can be seen as the Mileposts concept that has been already modelled in RaCoOn
TRCODE	An internal code that represents line
TID/TRID	An internal code that represents track
OBJECTID	The internal object ID
ASSETID	The internal asset ID
M_POST_ID	The milepost ID that represents a specific milepost/way-mark
L_LINK_ID	The internal link ID that represents a line which could be a physical line or reference line
Length	The length of given line
Start yards (feet)	The start of a given physical or reference line in yards (or feet)
End yards (feet)	The end of a given physical or reference line in yards (or feet)

TABLE 21 CONCEPTS CREATED IN REVISED RACOON AND THEIR TYPE

Concepts	*Type*
Reference Line	Class
Waymarks	Class that is the same as *is:MilepostLocation**
TRCODE	Data type property, the data type is *xsd:integer*
TID/TRID	Data type property, the data type is *xsd:integer*
OBJECTID	Data type property, the data type is *xsd:long*
ASSETID	Data type property, the data type is *xsd:long*
M_POST_ID	Data type property, the data type is *xsd:long*
L_LINK_ID	Data type property, the data type is *xsd:long*
Length	Data type property, the data type is *xsd:float*
Start yards (feet)	Data type property, the data type is *unit:Yard (Foot)***
End yards (feet)	Data type property, the data type is *unit:Yard (Foot)*

**The name space of 'is' is http://purl.org/rail/is/*

***The namespace of 'unit' is http://qudt.org/vocab/unit#*

7.3.4 MODELLING THE GEOGRAPHICAL DATA

When RaCoOn was proposed (Tutcher et al., 2017), geospatial data modelling was not taken into account. To enable RaCoOn to work with QGIS, it is necessary to establish a standardised method to capture geospatial data.

Two candidates can help to capture geospatial information, WGS84 Geo Positioning (Brickley and Berners-Lee, 2003) and Open Geospatial Consortium (OGC)'s GeoSPARQL vocabularies (Battle and Kolas, 2012).

WGS84 Geo Positioning provides a set of vocabularies for describing latitude, longitude and altitude information with reference to the WGS84 geodetic reference system (Brickley and Berners-Lee, 2003). WGS84 Geo Positioning was published under the W3C namespace[56], yet it is not part of W3C recommendations. Despite this, many other ontologies[57] (Brickley and Berners-Lee, 2003) reuse it to represent latitude, longitude and altitude information, which, to some extent, has made it one of the de facto standards in the linked open data community. However, the downside of WGS84 Geo Positioning is obvious, too. Because it is fairly basic, it is mostly used to model geospatial points instead of lines; it seems to be incapable of modelling continuous data such as lines or shapes.

In comparison with WGS84 Geo Positioning, OGC GeoSPARQL provides more comprehensive support for representing complex geospatial data using Well-Known Text (WKT)[58] (Open Geospatial Consortium, 2019), including continuous data. Although GeoSPARQL is also a query standard that extends SPARQL (Perry and Herring, 2012), it also bundles with a small ontology derived from OGC standards, to establish a generalised method to capture geospatial data. GeoSPARQL vocabularies have two parts, GeoSPARQL [59]

[56] http://www.w3.org/2003/01/geo/wgs84_pos#

[57] According to the statistics published by Linked Open Vocabularies, 49 datasets reuse or extend WGS84 Geo Positioning, including another commonly seen vocabulary and a de factor standard, Friend Of A Friend (FOAF).

[58] WKT is an ISO standard (ISO/IEC 13249-3:2016), that is available at https://www.iso.org/standard/60343.html.

[59] The name space is http://www.opengis.net/ont/geosparql#; this name space often uses a prefix 'geo'.

vocabulary and GeoSPARQL Function[60]; the former is used to model geospatial data and the latter is used to query geospatial Linked Data. In order to enable the usage of GeoSPARQL queries, it is necessary to use GeoSPARQL vocabularies to model geospatial data. Besides that, using GeoSPARQL vocabularies enables the use of SPARQL to query geospatial data using GeoSPARQL Function vocabularies (Battle and Kolas, 2012), for example:

Suppose we want to query mileposts situated between University Station and Birmingham New Street Station, the SPARQL query string[61] and its query result are as demonstrated in Fig. 64:

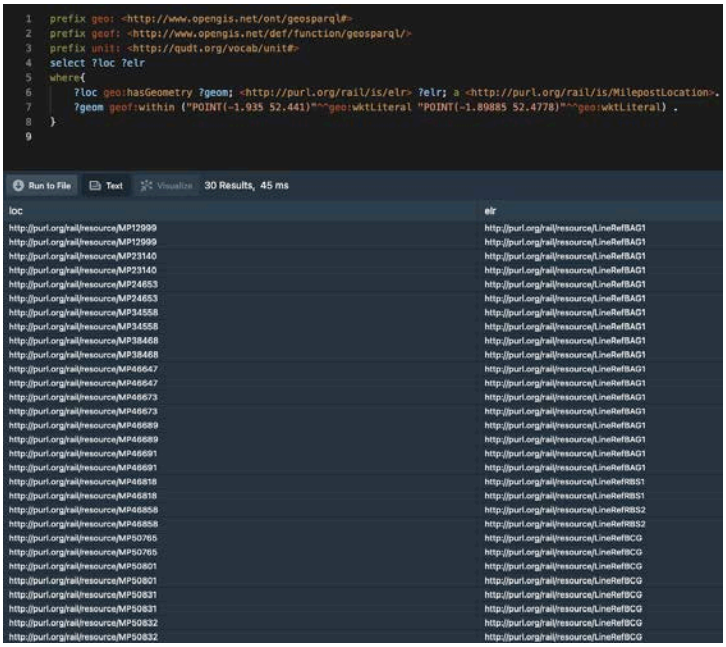


FIG. 69 EXAMPLE QUERY STRING FOR POINTS OF INTEREST SITUATED BETWEEN UNIVERSITY STATION AND BIRMINGHAM NEW STREET STATION (SUPPOSING COORDINATES ARE KNOWN)

[60] The name space is http://www.opengis.net/def/function/geosparql/; this name space is often represented with a prefix '***geof***'.
[61] Coordinates of University Station and Birmingham New Street Station were queried from DBPedia.

In this example, ***geof:within*** is used to obtain points of interest that are within a given shape. The model is illustrated in Fig. 70.

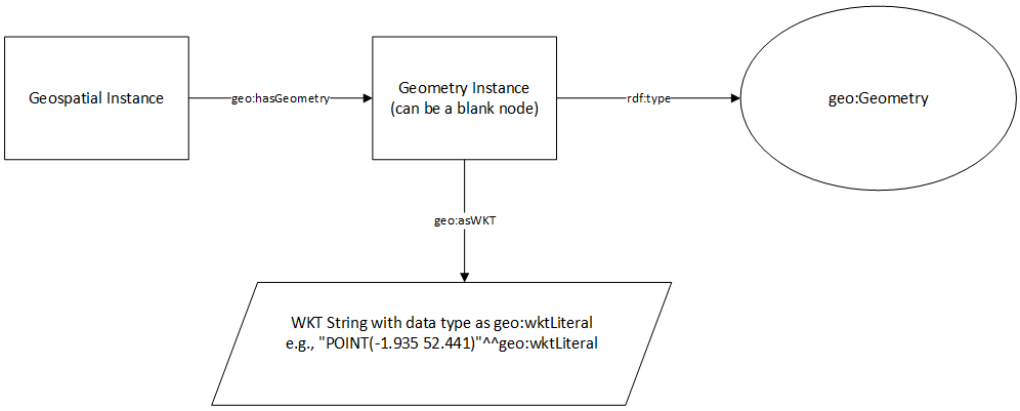


FIG. 70 ILLUSTRATION OF GEOSPATIAL INFORMATION MODEL[62]

It is also easy to use QGIS to load WKT to create vector points or lines, using the ***fromWkt()*** method from the ***QgsGeometry*** class. In conclusion, GeoSPARQL is suitable for modelling geospatial information for RaCoOn.

7.4 QGIS PLUGIN FOR SIMPLIFYING DATA PREPARATION AND CONSOLIDATION

This section presents a QGIS plugin, named RaCoOn Integration, that can help users to transform required data for T2F ground-borne data analysis based on RaCoOn accordingly and update the triple storage, forming layers and data tables that enable further analysis. The plugin can also enable users

[62] The namespace URIs of *geo* and *rdf* are http://www.opengis.net/ont/geosparql# and http://www.w3.org/1999/02/22-rdf-syntax-ns# respectively.

to insert new knowledge and create new data mapping in the existing ontologies.

7.4.1 Requirements Analysis

Before implementation of the plugin, it was vital to address the requirements. They can be divided into two parts, early requirements and late requirements (Castro et al., 2001) as shown in Table 22.

It can be seen that both early and late requirements involve data management with RaCoOn. They were derived because of a fundamental issue, that it is difficult to reuse the data. In their paper, Young et al. (2020) elaborated on their method to join the rail network model, demographical data and noise data; however, if the location is altered, the whole process might have to be completed again and the previous analysis result might be wasted. For example, suppose a user has completed the noise analysis process and obtained a result table; if the user wants to compare the data for a specific track segment after implementing different types of USP for the population that is within a certain distance of the link, it has to perform queries against the SQL database several times or with a complicated SQL query string that requires the user to have abundant knowledge of SQL. In this case, it might seem easier to change the parameters of the original model and run the whole analysis process again. However, with an ontology, first of all, the noise table can be modelled into RaCoOn so that the data joining process is

not essential; second, it is easier to query existing data as the internal relationships have already been modelled into RaCoOn.

TABLE 22 REQUIREMENTS ANALYSIS OF THE PLUGIN

	Requirement	*Solution*
Early requirements	Prepare required data for the ground-borne noise analysis in QGIS	Use Python modules bundled with QGIS to generate attribute tables and layers
	Integrate data to RaCoOn	Extend RaCoOn to enable it to describe required data concepts
	Interact with RaCoOn (e.g., query, update, etc.)	Select a mature solution to store RaCoOn and its data instances
Late requirements	Be intuitive, requiring the most minimal knowledge of ontology to use	Use widgets such as drop boxes and buttons as much as possible
	Be capable of creating new instances to RaCoOn based on a given file or QGIS layer	Design an editor that allows the user to create mappings between data origin and RaCoOn
	Form new data packages from existing data	Use RaCoOn to manage data including analysis result

On the other hand, the solution proposed by Young et al. (2020) requires users to have access to multiple data sources during the data preparation process, and data sources have to be available each time a user wants to run their model. Since RaCoOn can integrate this data, data sources no longer need to be provided; instead, everything can be mapped in one place, forming a unified whole so that users can simply query everything from the database with reference to a unified ontology structure.

Other requirements are mainly about establishing the means to interact with RaCoOn. The key principle of designing this part is for it to require as little coding as possible, i.e., using drop boxes and plain-field input as much as possible, creating an interactive machine–human interface with minimised learning (Nielsen, 1994).

7.4.2 Plugin Architecture and Workflow

The architecture of the proposed plugin is illustrated in Fig. 71. There are three main layers. The top layer is the interaction layer where the user can interact with the plugin, which is formed based on two modules, the definition of the main dialog and another UI that might be initialised by the main dialog. The second layer supports UI display and functions of the main dialog, also including the declaration of UI and actions of each UI element. The third layer contains the back-end libraries and utilities for interacting with RaCoOn remotely. The fourth layer includes all other required dependencies of the layers above.

The plugin was created from a template generated by QGIS Plugin Builder (Sherman, 2013). The required Python dependencies are:

- RDFLib (RDFLib Team, 2002)
- SPARQLWrapper (Herman et al., 2020)
- pystardog (Stardog Union, 2020)[63]

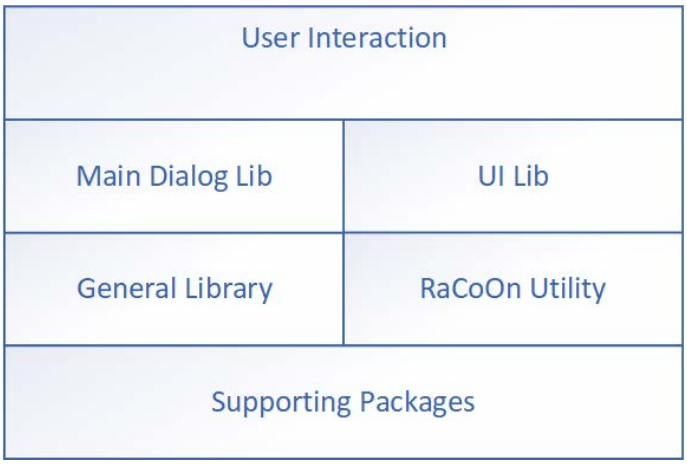


FIG. 71 ARCHITECTURE OF RACOON INTEGRATION

[63] According to the comparison made in Chapter 2, Stardog was selected as the triple store for this study.

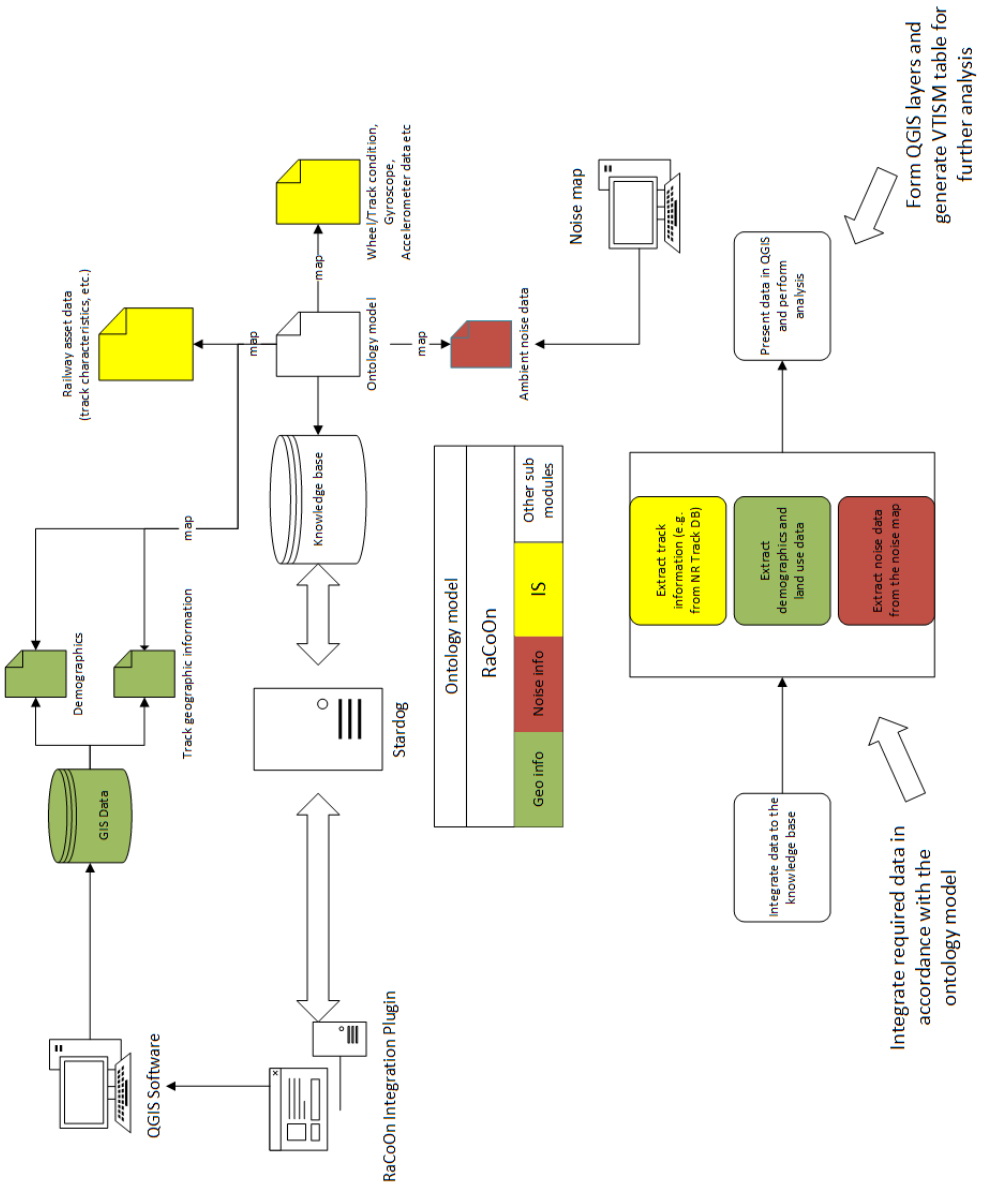


FIG. 72 WORKFLOW OF THE PLUGIN

The workflow of the plugin is illustrated in Fig. 72, where different data sources and their corresponding models are marked in corresponding colours, respectively. The workflow starts by using the plugin to map the required data to RaCoOn and store the integrated data in the data store. This operation ensures that the data is accessed solely from one place. Once the data is integrated to the store and the triple store is set up, the following

operations specified by Young et al. (2020) in QGIS can be performed. The ontology model provides a generalised method to describe the data used in the process proposed by Young et al. (2020), which is independent of the plugin and completely safe to modify[64].

7.4.3 FUNCTIONALITY

There are three major functions of the plugin, updating the database from the file based on pre-defined configuration and manually created mapping, QGIS data preparation and consolidating data for specific track segment(s), as discussed in section 7.4.1.

The triple store must be set up for first-time usage; the flow for this process is shown in Fig. 73. To update the database based on a pre-defined configuration, because no editing process is involved, the user can just provide the object type and the corresponding file and the rest can be automatically completed by the plugin, whereas to update with manual mappings, the user can create mappings between the data field and classes in RaCoOn. However, a personalised update requires the user to have a certain understanding of the ontology; this was designed to authorise more flexibility to ontology professionals. This responds to the UAT findings discussed in section 6 in Chapter 6, that professional users tend to seek a higher level of customisation and flexibility.

[64] Changes might have to be made to the plugin.

Meanwhile, because data exchange between the local machine and remote triple store might take a while, the update process is handled with multiple threads to increase the speed and prevent the process being judged unresponsive by the OS. Available child threads are illustrated in Fig. 74.

In terms of personalised updating, the user has to create mappings. Two drop boxes need to be selected, 'Data type' and 'Property'. The available data types are shown in Table 23. The design refers to W3C's introduction to ontologies (W3C OWL Working Group, 2012), covering all necessary special elements.

Objects are not taken into account in relational models. Although data is normally presented in the form of numbers or strings from the source, the data can refer to an object (instance) in ontologies. For example, ELR code is presented as string in a shape file, such as 'TBH2'; but in RaCoOn, ELR is a class where the model has been defined as illustrated in Fig. 75. Admittedly, ELR code can be modelled as a string that is connected by a ***owl:DatatypeProperty***, but the problem is that when ELR code has to be modified, every instance related to that ELR has to be updated; if the instance is pointing to an ELR object that captures the corresponding ELR code, the modification can take place just once, i.e., change the ELR code value for that sole ELR object.

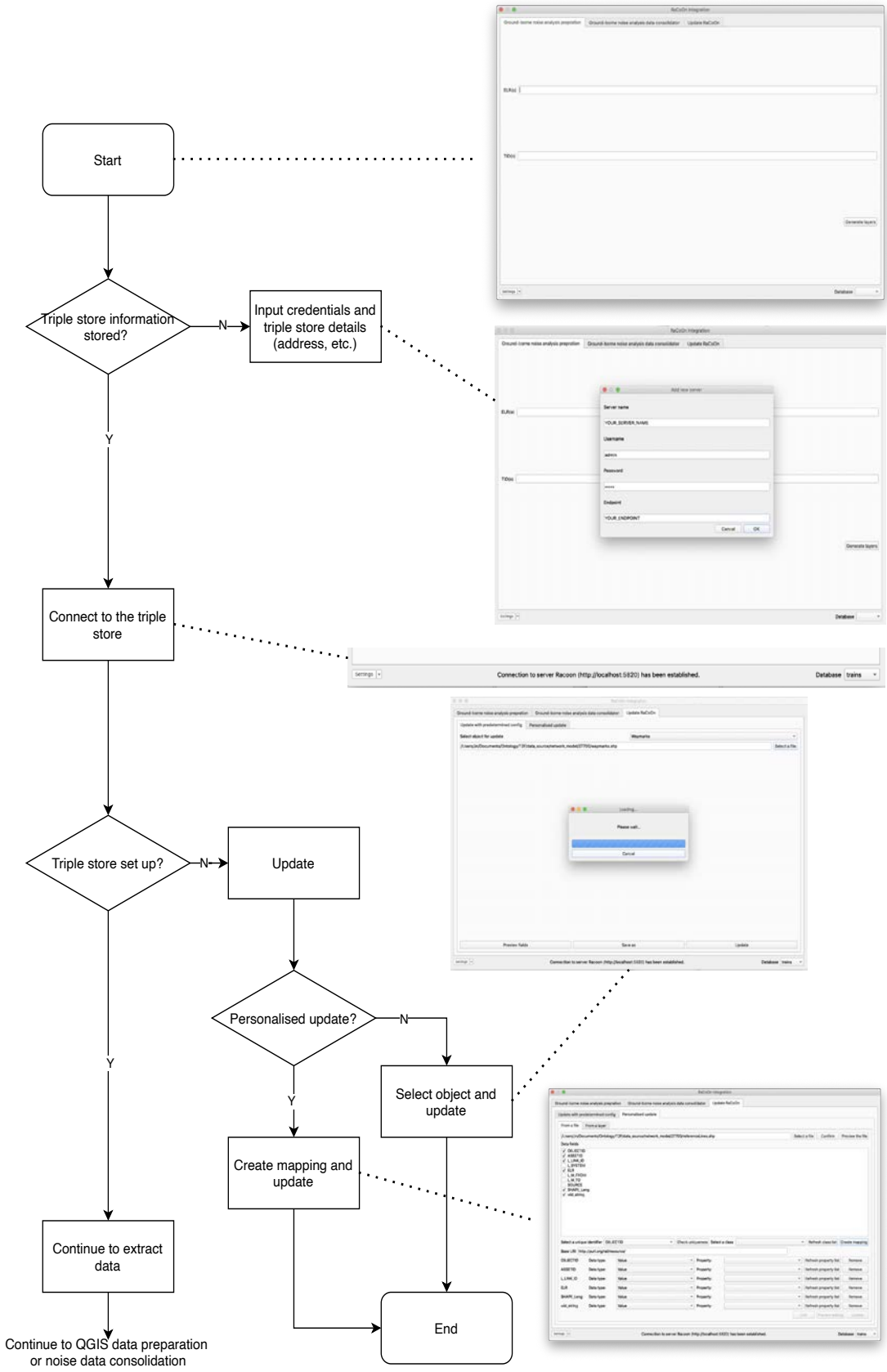


FIG. 73 FLOW FOR UPDATING TRIPLE STORE WITH ILLUSTRATION OF CORRESPONDING DIALOGS

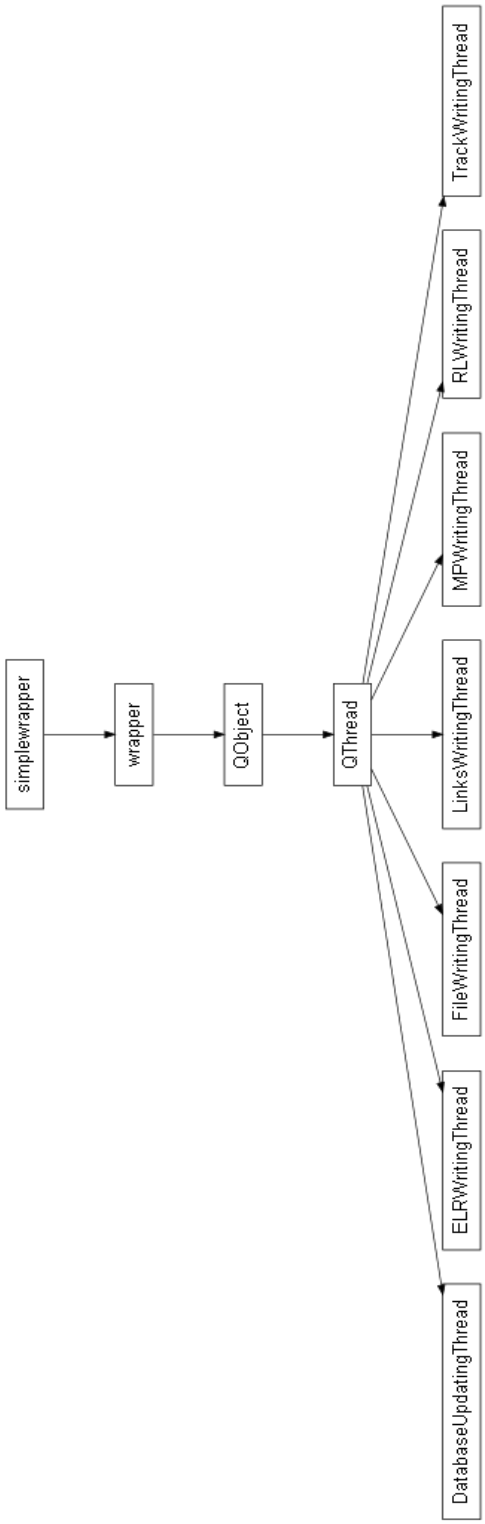


FIG. 74 AVAILABLE CHILD THREADS

TABLE 23 AVAILABLE DATA TYPES

Data type	*Ontology property type*
Value	*owl:DatatypeProperty*
Object	*owl:ObjectProperty* pointing to an instance
Blank node of a class	*owl:ObjectProperty* but pointing to a blank node

Using an object is recommended as it can better represent concepts (e.g., address, location, etc.), and when multiple instances relate to that concept, using objects instead of data values can be clearer and easier to maintain (Gangemi and Presutti, 2009). However, it is not necessary to explicitly declare concept objects being connected to save computational power and decrease the size of URI pool, hence using a blank node. A blank node is a special instance without a URI, indicating the existence of a thing instead of identifying a particular thing (W3C, 2014).

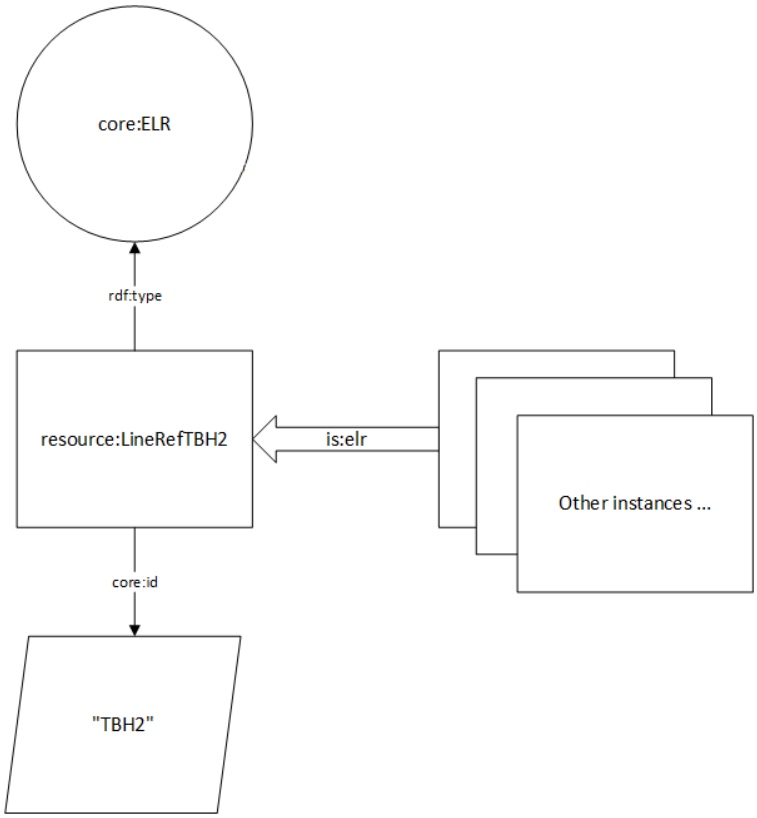


FIG. 75 ELR MODEL

Before proceeding to the final editing procedure, a unique identifier has to be selected to construct the unique URI for each new data instance under a selected class.

The identifier can be selected from one of the data fields presented by a drop box. However, in order to ensure the uniqueness of the selected identifier, a verification process is required because different instances are not permitted to share the same URI, otherwise they would be seen as the same instance.

Once the identifier is confirmed, the final mapping process that allows the user to specify the data type and targeting object can be carried out. An

example dialog is shown in Fig. 76, covering all data type selections. For '*Value*', the user can set the *owl:DataType*[65]. For '*Object*', the user can set classes that target instances belonging to and matching the property, i.e., the condition to judge if they are related based on whether the value is the same as one of the instances of the selected class with a given property. For example, a reference line with ELR code 'TBH2' relates to the ELR instance ***resource:LineRefTBH2*** because ***resource:LineRefTBH2*** holds the property ***core:id*** that has a value 'TBH2', as shown in Fig. 77.

If a user chooses to map to a blank node, they have to define the class to which the node belongs and select a data property that holds the value.

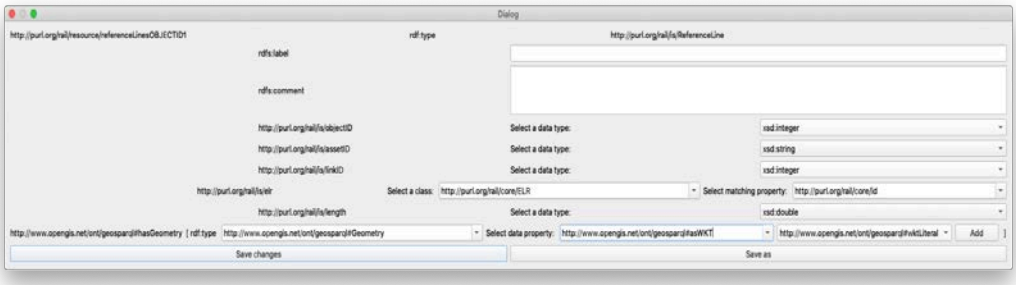


FIG. 76 SET DATA TYPES AND TARGETING OBJECT

[65] Please note that *owl:DataType* specifies what kind of data it is, e.g. string or int, whereas data type represents a high-level concept, i.e. if the data is just a set of values or pointing to an object.

Example data table

ID	AssetID	ELR	…
1	0000000	TBH2	…
2	1111111	BAG1	…
3	2222222	BAG2	…
…	…	…	…

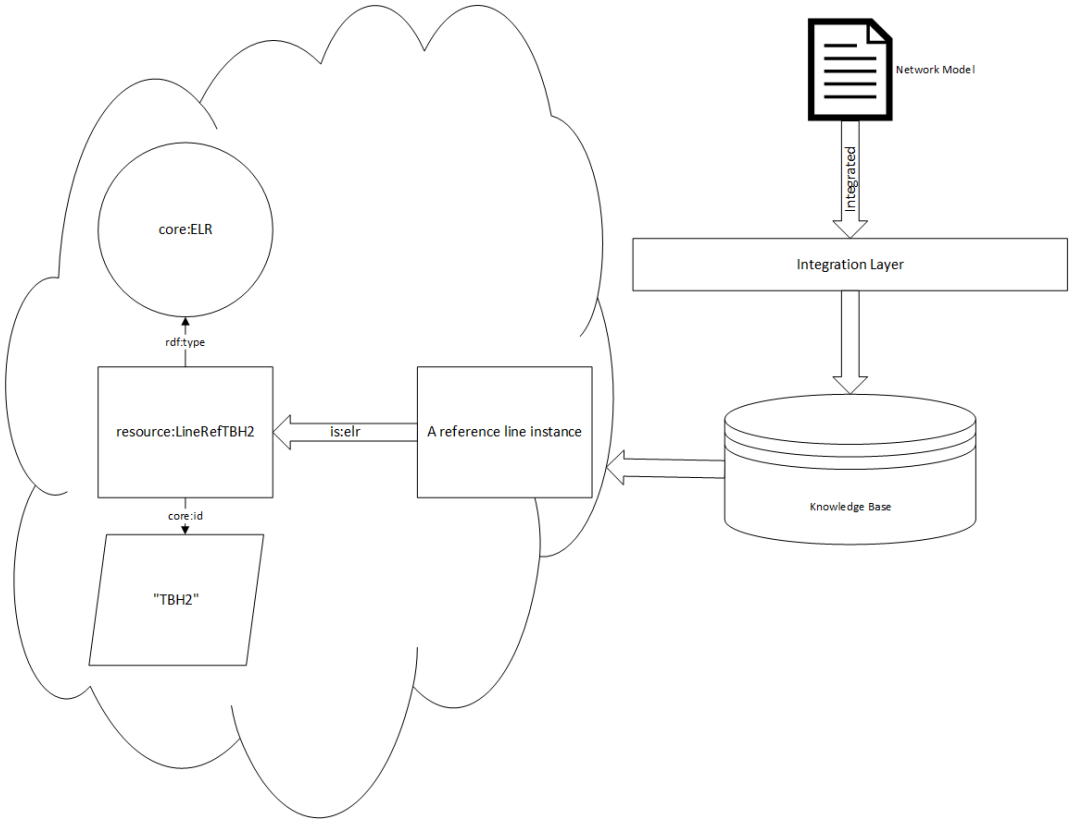


FIG. 77 MAPPING SOURCE TO ONTOLOGY

Whether the user chooses to map the data with a pre-determined configuration or manual method, the plugin can retrieve the track model(s) in conjunction with corresponding data and complete the ground-borne noise analysis data preparation when necessary. There will be several vector layers generated in the current QGIS project based on the given ELR code(s) and TID code(s); the flow is illustrated in Fig. 78, where lines with ELR code 'TBH2' and 'BLI1' with only 'up main' line (TID code 1100) are rendered in the current QGIS project as illustrated in the example screenshot at the bottom. In comparison with the approach of Young et al. (2020), the integration removes the steps for setting shape files, expediting the data preparation process.

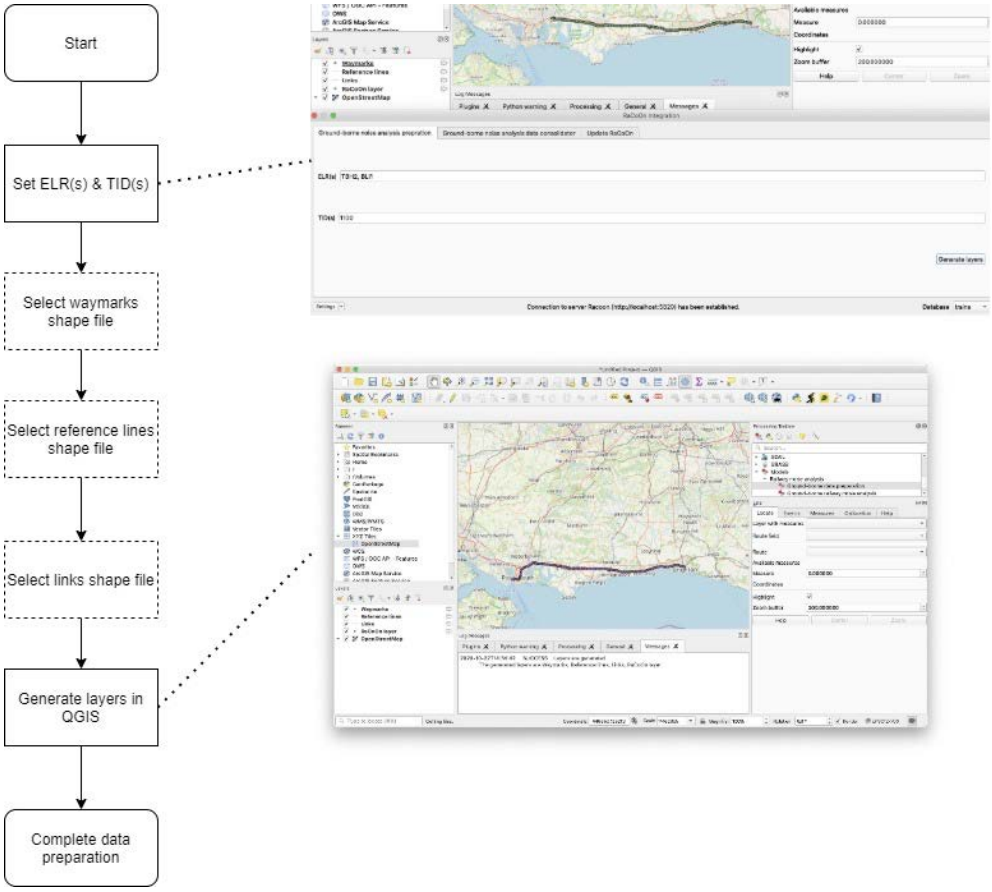


FIG. 78 FLOW FOR GROUND-BORNE NOISE ANALYSIS DATA PREPARATION (YOUNG ET AL.'S MANUAL PROCESS (2020) HAS BEEN MARKED WITH DASHED RECTANGULARS)

The high-level flow is illustrated in Fig. 79. The process flow for the case where manual mapping is chosen also applies to inserting a new instance from layers or source files.

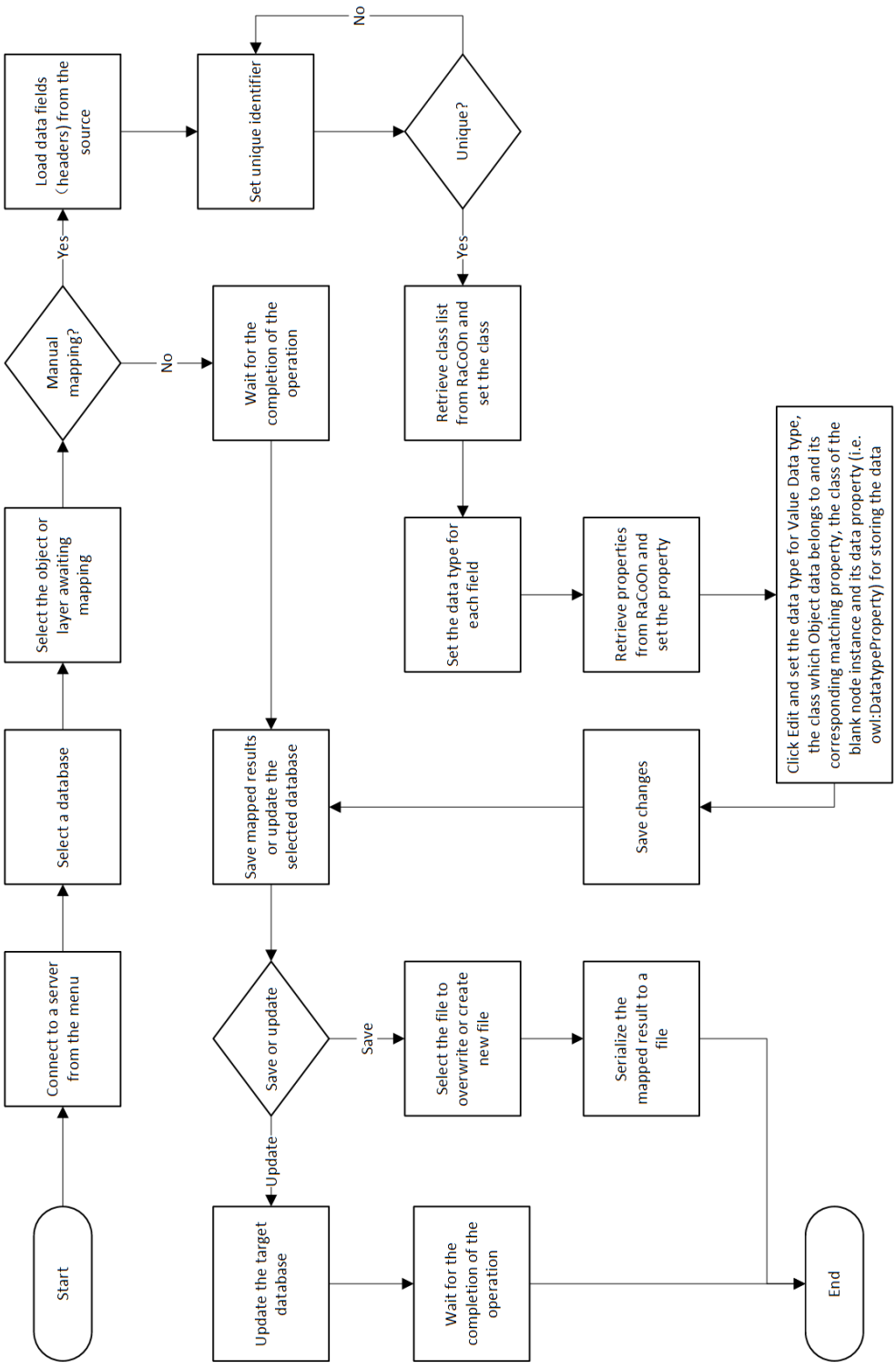


FIG. 79 UPDATING THE DATABASE WITH DEFAULT MAPPING

7.5 Example of Using the Plugin for a Particular Location

This section presents a case study of replicating the noise modelling process reported by Young et al. (2020), using the proposed ontology-based data integration approach and the QGIS plugin.

7.5.1 Original Approach

The discussion in this section is mainly with regard to the ground-borne noise modelling process in GIS analysis. According to the description presented by Young et al. (2020), they selected the route between Brighton and Portsmouth, known as the West Coastway Line, to combine the noise data while implementing different types of USP with demographic data and track data. The goal of implementing GIS analysis is to establish a transferable foundation to identify the resident population affected with wide reproducibility; the following steps are to be completed (Young et al., 2020):

- Represent railway model and corresponding information
- Identify population that is close to the track
- Calculate the probable level of ground-borne noise, including each intervention, and calculate the population affected

In order to represent the railway model in QGIS[66], Young et al. used the data listed in section 7.3.2. With given maximum train speeds and distances[67] to

[66] CRS is *EPSG:27700*
[67] Distances < 7.5 m are adjusted to 7.5 m and others are rounded to the nearest 10 m.

the track, a merged layer that contains the mapped ground-borne noise levels can be generated, and an SQL query that filters population points and joins them with the reference track and noise table can be run to instantiate an SQL database (assuming the ground type is 1). The whole process uses the high-level flowchart shown in Fig. 80.

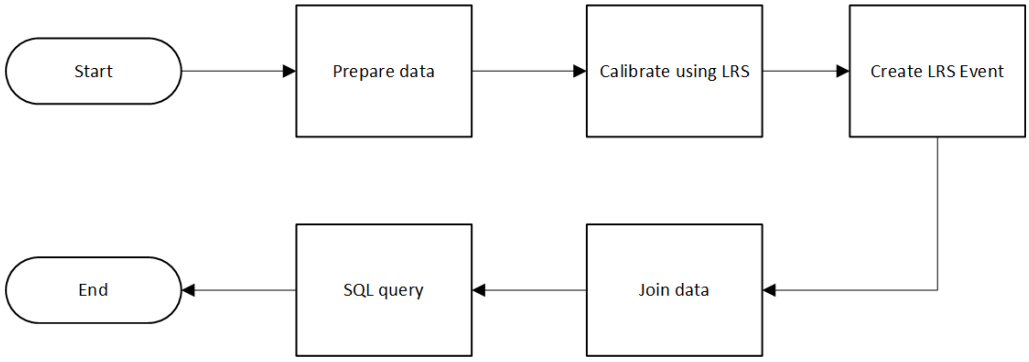


FIG. 80 GIS ANALYSIS FLOW PROPOSED BY YOUNG ET AL. (2020)

7.5.2 PROBLEM STATEMENT

Even though Young et al. (2020) emphasised the importance of transferability and reproducibility, network information and noise information were stored in different files (silos), which means there is no generalised framework to represent the data. This leads to the user having to provide three files, ESRI Shapefiles for Reference Lines, Waymarks and Links, besides accessing a database or CSV file that instantiates Network Rail's Track Database (originally stored in Microsoft Access format), as shown in Fig. 81; this imposes difficulties on utilising the network model and track characteristics in other applications, e.g., if the user wants to find out all track segments outside QGIS.

Another issue is that a relational model is not suitable for representing network models and joined results as it is not an optimal choice to reference objects (Halpin and Morgan, 2010). Yet, the data used in the proposed approach is rather object-oriented, e.g., each track segment can be seen as a specific instance of a network link.

In conclusion, the proposed approach contains a manual data feeding process owing to the lack of a means to represent both input and output data in a unified and standard way.

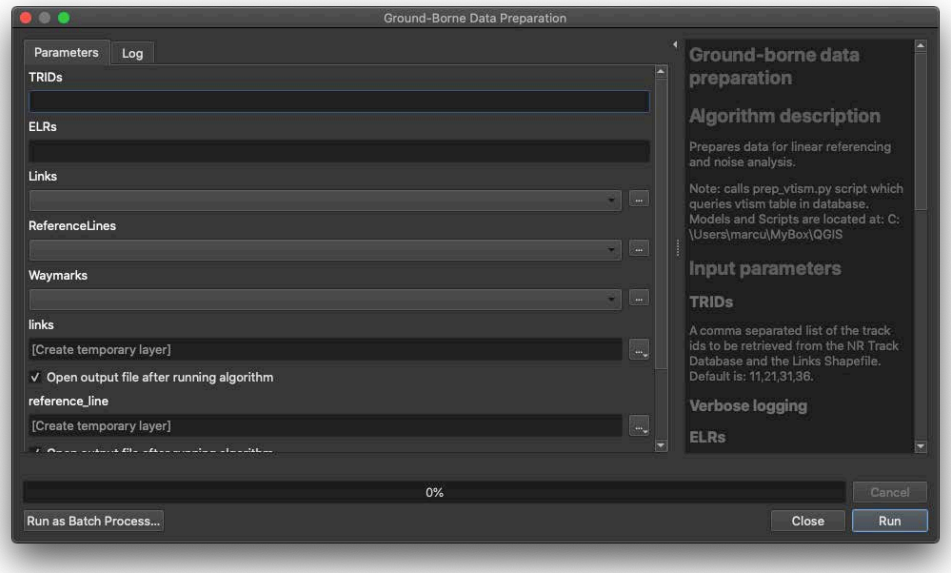


FIG. 81 SCREENSHOT OF ORIGINAL GROUND-ANALYSIS DATA PREPARATION DIALOG

7.5.3 USING THE ONTOLOGY-BASED APPROACH

To facilitate a high level of data integration to reduce manual processes, using ontologies to represent data is suitable, as discussed in section 7.3. The plugin will help to replicate the data preparation and result modelling

processes. Meanwhile, in order to address the issues revealed in Chapter 4, no coding process will be needed in this case.

The source and target of the integration are shown in Table 24, and the high-level integration system architecture is shown in Fig. 82.

TABLE 24 INTEGRATION SOURCES AND TARGETS

Data source	**Target class**	**Ontology module**	**Number of rows**
Waymark	is:MilepostLocation	Infrastructure	42432
Reference Line	is:ReferenceLine	Infrastructure	1576
Link	core:MainLine	Core	49761
Network Rail's Track Database	core:Track	Core	678850
Noise table	iv:NoiseData	Intervention	756
Namespaces: • iv: http://purl.org/rail/iv/ • is: http://purl.org/rail/is/ • core: http://purl.org/rail/core/			

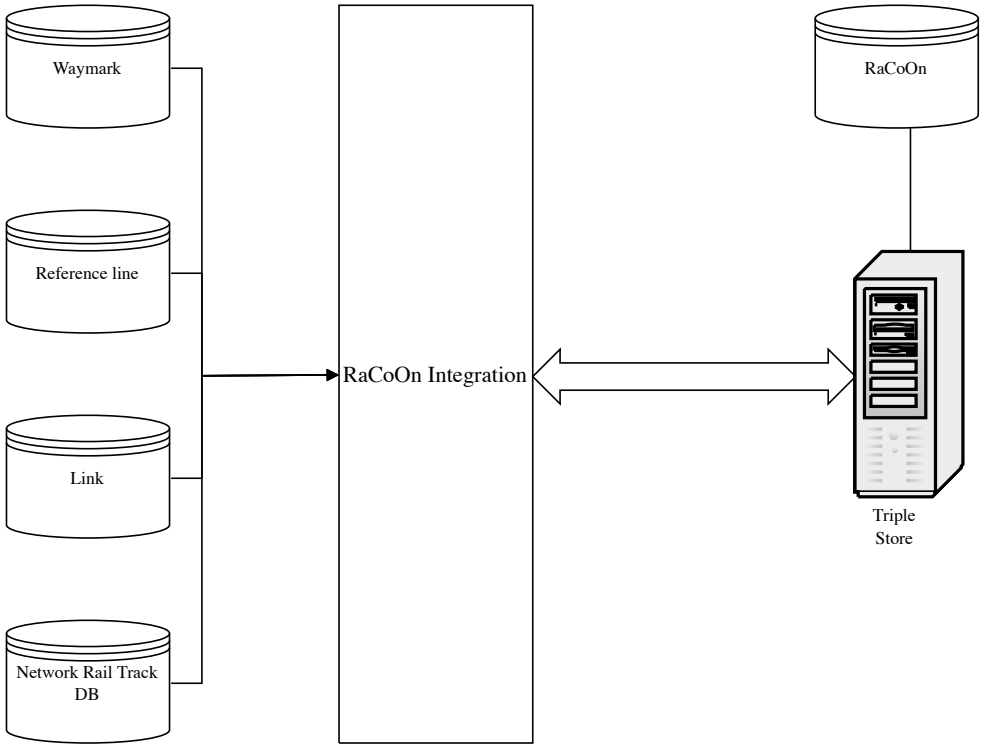


FIG. 82 INTEGRATION SYSTEM ARCHITECTURE

RaCoOn Integration, as a plugin for QGIS, can help to form an Integration Layer (IL) to facilitate data reuse in other applications, that is to allow data management in a sole storage site instead of multiple ones.

After using the plugin to map the data with a pre-defined configuration, all required data fields can be updated to the selected database. An example snippet of serialised waymark data in the form of a turtle is shown in Fig. 83. Once every object is mapped to RaCoOn, there is no longer a requirement to provide the source file, and the plugin can help the user to query required data from the triple store.

Three ELR codes are required to replicate the existing result, TBH2, BLI1 and WPH2, with TID 11, 21, 31 and 36, as illustrated in Fig. 84. Once the

operation is completed, the result layers are generated in the current QGIS project as shown in Fig. 85.

```
resource:MP1 a is:MilepostLocation ;
    rdfs:label "Milepost (Waymark) ID 1" ;
    core:id 1 ;
    is:assetID "9004000193" ;
    is:elr resource:LineRefACR ;
    is:mileage "421080"^^unit:Foot,
        "79.132"^^xsd:float ;
    is:mpID 1 ;
    is:objectID 25312 ;
    geo:hasGeometry [ a geo:Geometry ;
            geo:asWKT "Point (627866.86500000022351742 164441.11559999920427799)"^^geo:wktLiteral ;
            wgs:lat "164441.1156"^^xsd:float ;
            wgs:long "627866.865"^^xsd:float ] .
```

FIG. 83 SNIPPET OF MAPPED RESULT FOR WAYMARK

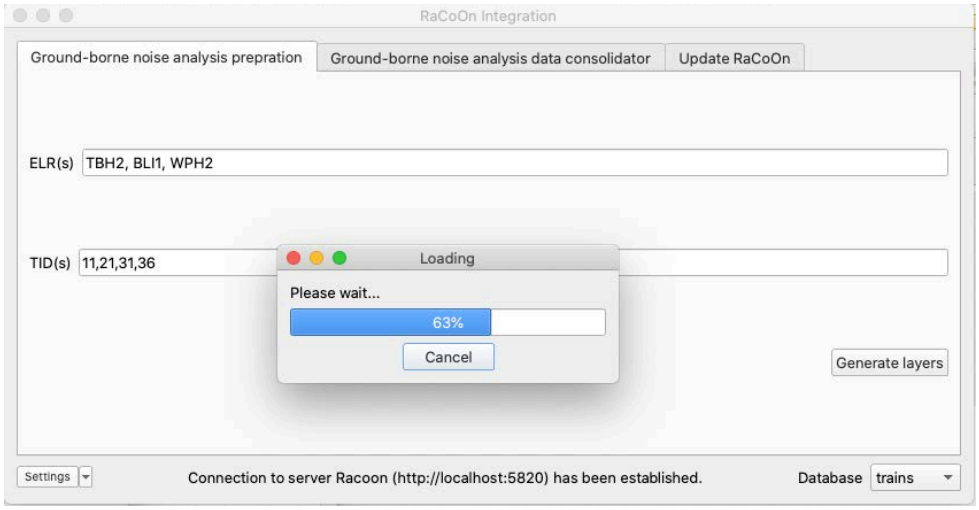


FIG. 84 INPUT REQUIRED ELR AND TID CODES AND GENERATE LAYERS

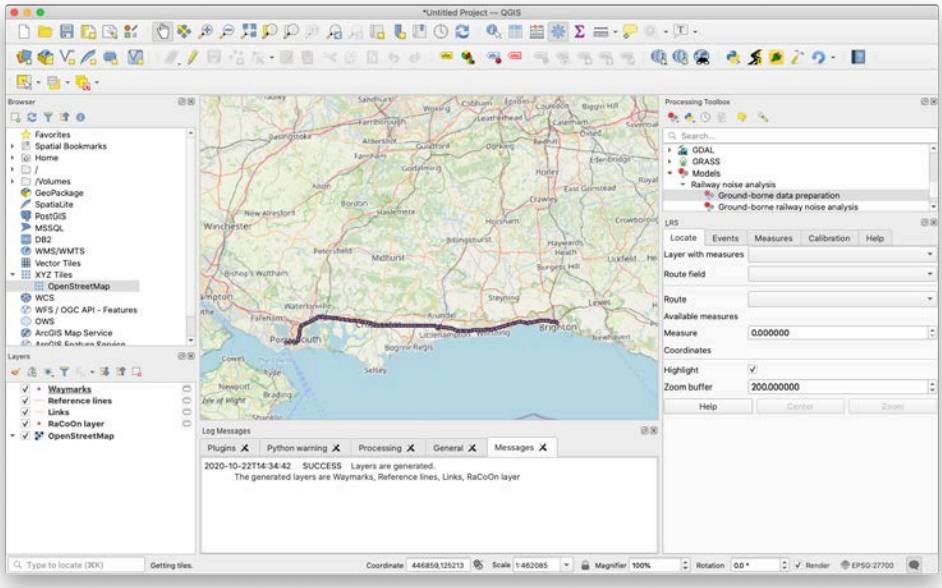


FIG. 85 FOUR LAYERS GENERATED IN CURRENT QGIS PROJECT

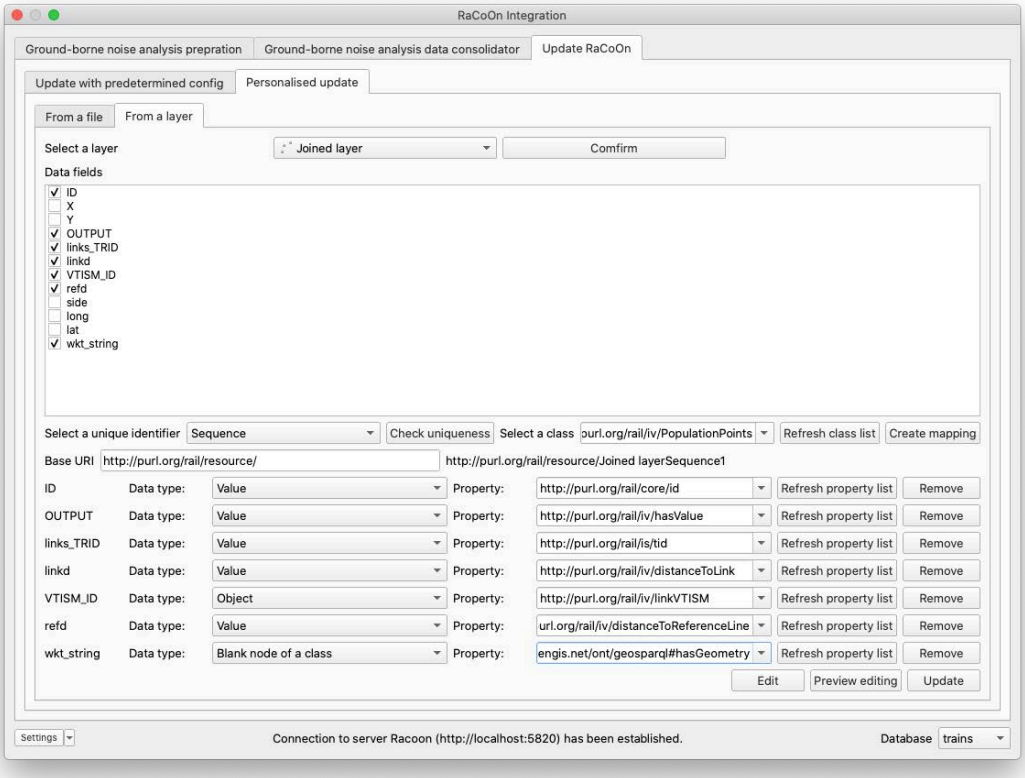


FIG. 86 CREATE A MAPPING FOR THE POINT LAYER

The rest of the GIS analysis steps remain the same until the generation of population points, which can be inserted into RaCoOn with a manually created mapping configuration. Some of the generated population points are shown in Fig. 87, where the deeper the colour of a point is, the denser the location. The configuration is shown in Fig. 86 and Fig. 88. An example of serialisation is shown in Fig. 89. According to the result, there were 112368 population points inserted into the database.

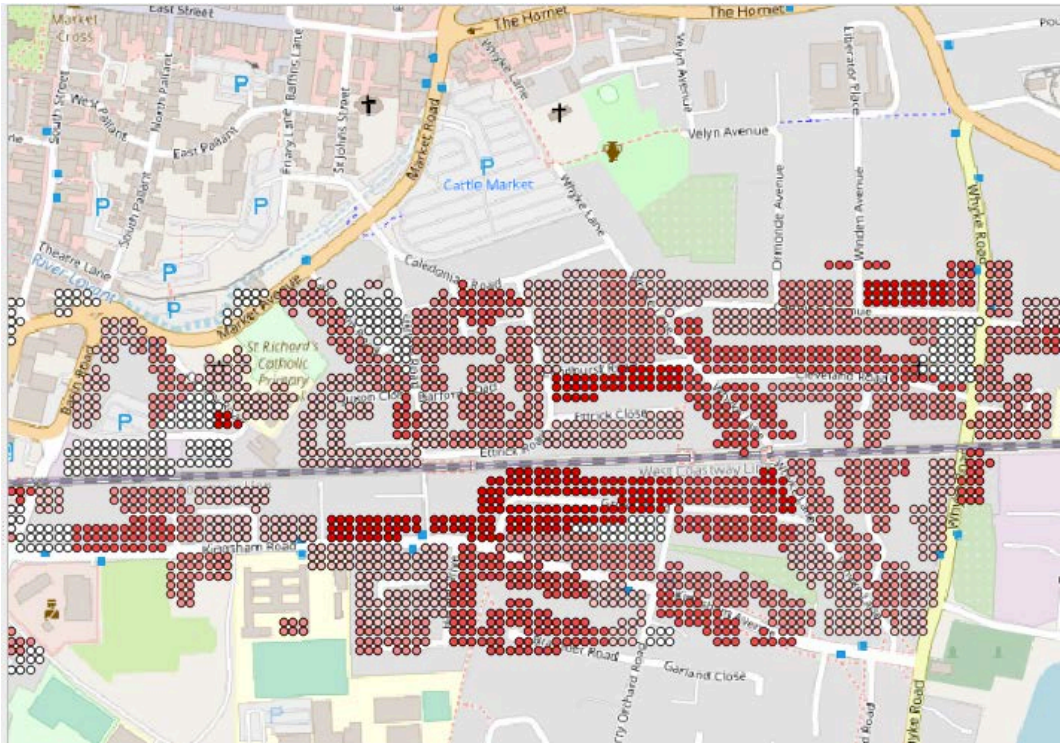


FIG. 87 EXAMPLE SNIPPET OF POPULATION POINTS[68]

[68] The example illustrates the population distribution within a given distance from a given track.

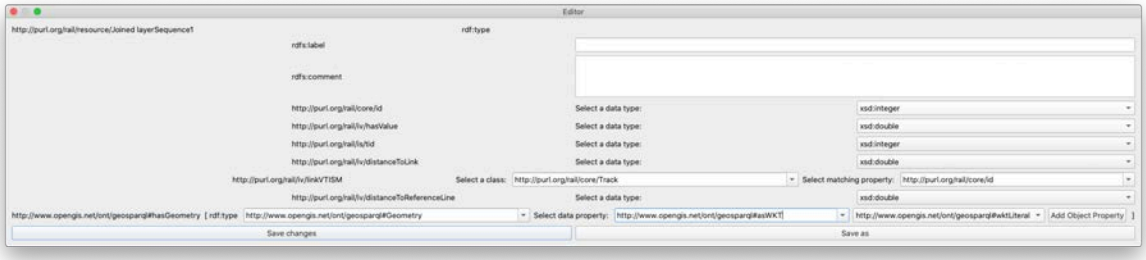


FIG. 88 SET TARGET DATA TYPE OR OBJECT MATCHING CONDITION

```
resource:PopulationPointsSequence1 a iv:PopulationPoints ;
    core:id 1 ;
    is:tid 2100 ;
    iv:distanceToLink 1.947896e+02 ;
    iv:distanceToReferenceLine 1.974527e+02 ;
    iv:hasValue 6.937e-01 ;
    iv:linkVTISM resource:Track_659150 ;
    geo:hasGeometry [ a geo:Geometry ;
            geo:asWKT "Point (463285 99795)"^^geo:wktLiteral ] .
```

FIG. 89 MAPPED RESULT SERIALISED IN TURTLE

In terms of creating linkage between the noise table and population points, unlike the approach of Young et al. (2020) in which the table has to be explicitly defined in a PostgreSQL database, using ontologies enables implicit definition between concepts as such an explicit definition can be completed while running a query. For example, if the user wants to obtain the aggregated population experiencing each expected noise level on route ‘TBH2’ while intervention with the implementation of ‘soft pad 120’ is in place, the SPARQL query can be formed by the plugin shown in Fig. 90; the result is shown in Fig. 91, which is verified as the same as that using the original approach by Young et al.

FIG. 90 EXAMPLE SPARQL QUERY STRING

```
SELECT DISTINCT
(sum(?pop_amount) as ?pop)
?db

where{
    ?pop_point a iv:PopulationPoints; iv:hasValue ?pop_amount; iv:distanceToLink ?d; iv:linkVTISM ?vt.
    BIND(IF(?d - 0.8 <= 7.5, 7.5, ROUND((?d - 0.8) / 10) * 10) as ?d_link)
    BIND('TBH2' as ?elr)
    ?vt core:id ?tr_id; is:elr/core:id ?elr; is:tid ?tid; is:maxSpeed ?_speed.
    BIND(ROUND(?_speed * 1.60934) as ?m_speed)
    BIND(iv:USP_softpads120 as ?iv_factor)
    BIND(iv:ground_type_1 as ?ground_type)
    ?noise_tb a iv:NoiseData;
              iv:distanceToLink ?d_link;
              iv:hasGroundType ?ground_type;
              is:maxSpeed ?_m_speed;
              iv:relatesToIntervention [
                  iv:interventionFactor ?iv_factor;
                  iv:hasValue ?db;
              ];
    # enforce type conversion to string in case of different data type (e.g. xsd:decimal and xsd:double), despite looking like the same value
    FILTER (STR(?m_speed) = STR(?_m_speed))
}
GROUP BY ?db
```

pop	db
0.9714	-19.8
9.687800000000001	-19.9
38.380799999999994	20.1
47.269299999999994	-20.3
32.673400000000015	-20.7
22.816600000000005	-21.1
637.7089999999992	21.3
655.4253000000007	-21.5
9.0957	-21.9
52.48390000000003	-22.3
40.428500000000035	-22.7
26.503300000000007	-23.1
658.5927000000016	-23.5
10.973400000000002	-23.9
1.9428	-24.0
55.07640000000002	-24.2
41.849800000000045	-24.7
21.50250000000001	-25.1
636.1751000000008	-25.5
3.3795	25.4
11.267399999999997	-25.8
616.4673000000022	-25.9
52.85780000000001	-26.2
684.4671999999996	-26.5
42.50619999999997	-26.7
650.4314000000003	-26.9
19.06180000000001	-27.0
42.37529999999999	-27.1
617.7903000000005	-27.4
48.15040000000003	-27.6

FIG. 91 QUERY RESULT OF FIG. 90 (AVERAGE NUMBER OF INHABITANTS PER 100 SQUARE METRE[69] AT EACH EXPECTED NOISE LEVEL)

Besides that, if the result for the new modelling falls within the previous result, the previous result can easily be reused. For example, it is possible to extract specific track segments with reference to parameters such as a given location, etc.

[69] The resolution of original population grid data was 10m.

To conclude, the ontology-based approach provides a standard framework to describe both input and output data for a GIS analysis process so that the user can always reuse the mapped data. The data preparation process is also easier than the solution proposed by Young et al. (2020).

7.5.4 Time-Consumption Comparison

To prove that the proposed ontology-based approach is more efficient than the original method, several rounds of timing were completed for both approaches. The proposed ontology-based approach was timed within the plugin's Python script using the Python *decorator with* ***time.perf_counter()*** method as ***time.perf_counter()*** achieves more accurate performance measurement (Python Software Foundation, 2015). Because manual data preparation was used in the approach of Young et al. (2020), the timing procedure could not be completed automatically, so that the timing for their approach was completed by two ArcGIS professional users following instructions supplied by Marcus Young[70].

The total time taken for each approach was then compared, and the result is shown in Table 25.

[70] The instruction video is available at http://screencasts.graspit.co.uk/watch/cY1jFXCiWc

TABLE 25 TOTAL TIME TAKEN FOR THE TWO APPROACHES (AVERAGE OF FIVE ROUNDS OF TESTS)

Step	*Ontology-based approach*	*Original approach*	
	Auto timing (s)	Tester 1 timing (s)	Tester 2 timing (s)
Data selection (file)	-	30.25	31.15
Layer generation	16.73	18.21	18.04
LRS calibration	2.04	1.55	1.64
Output	123.35	121.21	120.97
Total	142.12	171.22	171.8

Remarks:

- *The total elapsed time was an average of five rounds of tests performed on QGIS MacOS version 3.14 running on a machine with 16 GB RAM and a 2.3 GHz Quad-Core processor.*
- *The ontology-based approach accessed a triple store locally, and the SQL database required for the original approach was also set up and accessed locally.*
- *The required data was mapped to the ontology and stored in the triple store.*

It can be seen from Table 25 that the ontology-based data integration solution eliminates the data selection procedure, and only requires the user to provide the ELR codes and corresponding TID codes (i.e., ELR codes TBH2, BLI1 and WPH2, and TID codes 11, 21, 31 and 36, respectively, as shown in Fig. 84). The ontology-based integration saves an average of 16% of the total time taken compared to the original approach.

However, other than the layer generation process, the rest of the processes tend to take more time due to the extra annotation brought by the ontology. It provides rich semantics but also an additional workload to process the data. Despite the additional workload, the improvement in time owing to the reduction in manual involvement is obvious when the data is mapped to the ontologies.

7.6 CONCLUSION

In this chapter, the use of ontology to replace the current manual process is proposed and demonstrated. Using ontologies enables more possibilities for data analysis and management, especially under circumstances where the data tends to be stored in different silos and in different proprietary formats (Technische Universität Dresden et al., 2016). Such a condition brings a problem to rail intervention projects. It can be difficult to collect the required data and others rarely reuse the data once the project is completed, i.e., it has a one-off nature, owing to the lack of a generalised and standardised method to represent the data. Two objectives were proposed in section 7.1:

1) Deliver an ontology-based approach to replicate the process proposed in T2F
2) Demonstrate its applicability to the existing process using a case study

The response to objective 1) is an ontology (RaCoOn)-based approach to manage and represent rail intervention data with a supplementary QGIS plugin, to help the user to perform the GIS analysis process proposed by Young et al. (2020). The plugin enables those who are new to ontologies to work with RaCoOn, also addressing the issue identified in Chapter 4. In comparison to the original SQL-based approach, the proposed approach reduces manual data pre-processing and data insertion at a later stage. Using RaCoOn to manage the mapped data also enables other applications to

interact with the intervention data in the future if necessary. By replicating the same modelling process for the West Coastway Line, it has been proven that it is possible to use the proposed solution to replicate the existing process, providing a response to objective 2). The key contribution was to remove manual process from data preparation and consolidation, replacing them with straightforward and easy-to-use data framework. The proposed solution can not only integrate data in different proprietary formats, but also consolidate the data with assurance of the consistency and formal semantics. Fig. 92 illustrates the comparison between traditional solution and proposed solution, where it clearly shows how the data management during the analysis procedure has become easier.

However, there is room to improve the proposed approach. First, the plugin is dependent on QGIS, which raises the level of difficulty to make it work with other applications or software unless QGIS will open new APIs to allow data exchange between external software and QGIS. According to some feedbacks from testers, it is more common in the industry to use ArcGIS; the proposed solution has not yet been transplanted to ArcGIS. However, this is rather a software engineering issue with little impact on the concept of adopting ontologies for similar tasks.

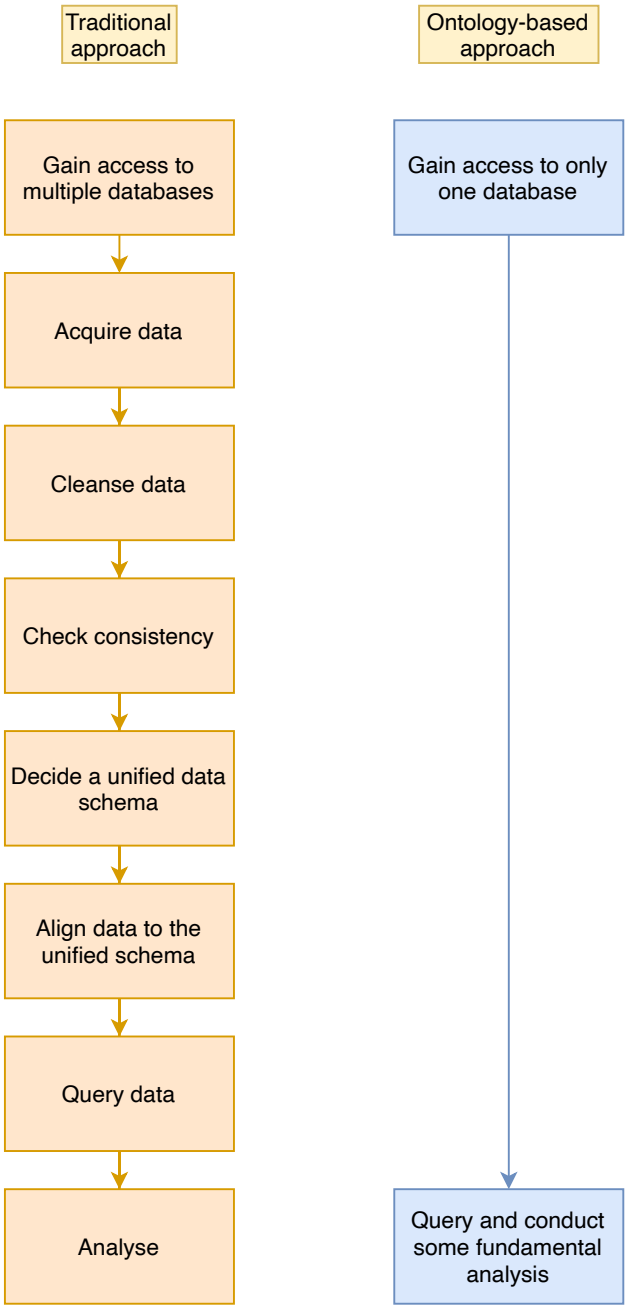


FIG. 92 COMPARISON BETWEEN THE TRADITIONAL AND NEW APPRAOCH

Second, although the plugin allows the user to create new concepts and data instances in RaCoOn, there is no local mechanism to assert the consistency of the inserted concepts with reference to existing models in RaCoOn. Currently, this task is completed remotely by Stardog, which would be better

done locally to ease pressure on the data storage server while dealing with Big Data. A consistency checking algorithm should be designed in a future study.

Third, although using RaCoOn and the proposed plugin can reduce the manual process, the manual process still exists, and cannot be removed from the process chain. The main reason is that the data flow between different plugins is not accessible by the developer. This is because QGIS does not allow background data exchange between plugins, and more importantly, different plugin developers use different methods to represent data internally, which cannot be resolved presently unless all involved plugins use RaCoOn to represent their internal data, too. This can be solved by developing a generalised analysis architecture from scratch and appealing to project participants to follow the standard of representing their data with RaCoOn.

Fourth, although the data representation framework is established, further investigation of applications and validation of the proposed approach is required because the proposed plugin is still under active development. For instance, the additional workload brought by richer annotation of data could also create an issue of scalability when dealing with a larger volume of data. This has little influence when the size of data is small, but it could be computationally expensive to handle industry-level data.

In conclusion, on top of the benefits of using ontologies discussed in Chapter 2, it is possible to use ontologies to facilitate the existing process, too. Rail

intervention projects often require systematic analysis beforehand, thus the need for a flexible and standard data description framework. With such a framework, an IL can be established, which enables fewer manual data processes and more reuse of data, decreasing the overall time for the whole process.

8 CONCLUSION AND FUTURE WORK

The work presented in this thesis identified issues with adoption of ontology-based approaches in the UK railway industry. Based on concluded factors, examples for using ontologies in the railway industry have been demonstrated, aiming to provide insights and inspiration for those who are seeking further development of ontologies in the UK rail industry. This chapter summarises a series of conclusions and envisages future directions of work.

8.1 CONTRIBUTION

The key idea of using ontologies in the railway industry is to establish a highly interoperable data integration mechanism to manage and reuse existing data, and such a mechanism can be based on ontologies in compliance with the discussion in Chapter 2. The key contribution of this thesis can be described as the following:

- Investigating factors which hinder further development and adoption of railway ontologies in the UK, despite known and proven benefits of using ontologies.
- Demonstrating a method for using ontologies to manage unstructured data, proving that rail ontologies can not only integrate

operational data such as sensor readings but also unstructured data[71]. Ontology can enhance existing classification algorithms with its rich semantics and strict hierarchy. Triples can be good references for classification tasks.

- Demonstrating a tool that incorporates an SWRL rule validator to validate and correct a rule drawn in a graphic rule designer, and initiating inference based on the given rule with the reasoner. Existing solutions are coding-based while the proposed tool aims to lower the entry level for designing an SWRL rule with a drag-and-drop method.
- Proposing an ontology-based approach to represent the data required for the ground-borne noise analysis process in T2F, plus a supplementary QGIS plugin, named RaCoOn Integration, to interact with RaCoOn, presenting a novel solution for using ontologies to replicate the existing manual process and avoid one-off usage of collected data. The case study proves that an ontology-based solution can help the user to better prepare the required data, and the overall analysis time was decreased in the case study.

This thesis has investigated some applications for applying ontologies to the UK railway industry to answer the following research questions:

[71] Only textual data was considered in this study.

- What work has been completed to demonstrate the usefulness of ontologies in large complex systems overall?
- Given the fact that both the rail industry and the research community are interested in ontology-based applications, why is there no sign that an ontology-based system has been implemented within the rail industry with an appropriate system architecture?
- Given the fact that ontologies can integrate data, how can we use ontologies to manage unstructured data in the railway industry?
- Many ontology models can only be manipulated by relevant professionals; how can we enable those who are not familiar with ontologies to use them?
- How can we reproduce some manual processes using ontologies to achieve more digitalised and effective processes in the railway industry?

The above-mentioned questions are discussed and answered in Chapters 2, 4, 5, 6 and 7, respectively.

8.1.1 What Work has Been Completed to Demonstrate the Usefulness of Ontologies in Large Complex Systems Overall?

As discussed in Chapter 2, data generated in large complex systems is often kept in separate silos in proprietary formats based on the system that generated it. The works discussed in Chapter 2 have demonstrated that ontologies can help to semantically integrate data across multiple systems

regardless of the supplier or existing structure. Such a trait could be particularly beneficial to the UK railway industry where many legacy data systems still exist and are impossible to replace in the near future. Many researchers have addressed the issue of the UK's non-integrated railway data management, suggesting that it would be more cost- and time-efficient if there were a way to integrate data as a unified whole. The resulting model is often presented in the form of a graph which enables users to clearly identify the implicit relationships between nodes, facilitating more efficient data modelling and easier model maintenance.

In addition to benefits in terms of data management, ontologies are also applied in applications such as data analysis, knowledge management, etc., in large complex systems. As such, the UK railway industry has shown interest in using ontologies to manage data.

Although ontologies have been applied in a wide range of applications in many domains, as shown in Chapter 2, researchers are still investigating more possibilities of using them.

8.1.2 Given the Fact That Both the Rail Industry and Research Community are Interested in Ontology-Based Applications, Why is There No Sign That an Ontology-Based System Has Been Implemented Within the Rail Industry With an Appropriate System Architecture?

The attention on using ontologies has been continuously increasing in many industries, including the rail industry; however, there is no solid proof

revealing that actual implementation of an ontology-based system is being completed in the UK rail industry. Since Network Rail and some other major railway operators have shown their interest, there should have been projects investigating the possibilities of applying ontologies in the industry, but there is no evidence of any. Therefore, an investigation was initiated to determine the factors that discourage the development and adoption of ontologies in the industry, which is presented in Chapter 4.

The investigation was conducted by distributing a questionnaire amongst users of Open Rail Data, most of whom are developers and researchers either working for the industry or having a great interest in facilitating better usage of railway data for the UK railway system. After categorising and validating their responses, three factors should be addressed:

- Despite having been deployed in other industries, little has been completed to demonstrate how ontologies could be used to structure data in the railway industry.
- The lack of supplementary tools for using ontologies repels potential users.
- There has been little investigation into replicating current manual processes with ontologies in a commercial environment in the railway industry in the UK. Therefore, professionals working in the industry tend to stick to their existing working routines with little will to try to use ontologies to manage data.

8.1.3 Given the Fact that Ontologies Can Integrate Data, How Can We Use Ontologies to Manage Unstructured Data in the Railway Industry?

It has been proven that ontologies can facilitate data integration in the previous context. A problem which still remains to be investigated is that although it is possible to integrate and manage operational data such as asset data, condition data, etc., is it possible to use ontologies to manage unstructured data, such as files, in the industry?

In Chapter 5, this question was answered by proposing an ontology-based document classification framework. The approach uses ontologies with appropriate machine learning models, extending RaCoOn to support complex event modelling. The proposed solution provides a practical demonstration of enhanced model training and information retrieval with ontologies. The result of the case study elaborated in Chapter 5 proves that the proposed approach facilitates text processing and slightly improves the performance of existing document classification algorithms owing to ontologies' rich semantics.

Yet, limitations exist. The study only focused on textual data; other kinds of unstructured data, such as images, were not included. This study, as a proof of concept, has not taken other types of unstructured data into account; it requires further investigation. Meanwhile, owing to limited access to the document repository, the total amount of samples is rather small. A larger

dataset should be tested with the proposed framework as the scalability of the proposed solution remains unknown.

It is also worth mentioning that knowledge model extraction from unstructured data was not involved in this study. Because the proposed study was designed to prove that ontologies can benefit the management of unstructured data in the industry as the proof of concept, knowledge model extraction, i.e., ontology learning, from existing textual data was not attempted. A well-designed knowledge extraction algorithm could greatly enhance the performance of the learning models as a result, which should be addressed in the future study of ontology alignment and ontology extraction.

In addition, despite the improvement of existing classification algorithms, the extent to which ontologies can improve the performance of learning models remains unknown. Combining knowledge management and modelling by using ontologies to model textual data with AI technologies might be beneficial for an automatic or even autonomous decision-making process.

In conclusion, despite the limitations of this study, applying ontologies to manage unstructured data is feasible and beneficial to the industry, and ontologies can facilitate the automated classification process to simplify document management.

8.1.4 Many Ontology Models Can Only be Manipulated by Relevant Professionals; How Can We Enable Those Who Are Not Familiar With Ontologies to Use Them?

In conformity with the factors revealed in Chapter 4, many candidates reported that the learning cost for using ontologies is relatively high, hence there is little will to invest time and effort in learning how to use them, despite the interest. They pointed out that supporting tools should be made available, to allow those knowing little about ontologies to interact with them. Most currently available tools are professional-oriented. Meanwhile, SWRL, as part of the rule mechanism of ontologies, plays an important role in semantic technologies. To enable non-ontology professionals to use SWRL, a graphic rule designer and validator are presented in Chapter 7, which aim to lower the entry level for using SWRL with existing ontologies (RaCoOn in this particular study).

With a little drag-and-drop editing, the user can design an SWRL rule with reference to RaCoOn. The designed rule can be further validated by a validator which can be also reused as a Python module, ensuring that the rule insertion is valid and legitimate to the reasoner.

The case study simulated a scenario in which a user-edited low adhesion rule based on real-world policy to manage low adhesion hazard was created with the proposed rule designer. The validator verifies the rule and corrects it, then the reasoner can conduct inference based on the corrected rule. The

whole process requires no human intervention; users are only requested to draw a rule that accords with their knowledge, allowing personnel unfamiliar with ontologies or even IT technologies, such as maintenance operators and in-field engineers, to design a simple rule to complete the inference using an ontology.

The study elaborated in Chapter 6 was initiated to provide the inspiration for the design of future tools for ontology-related application, so that the entry level for using ontologies can be lowered. More graphics-based tools should be introduced. There are reasons to believe that more attention could be gained from the UK rail industry if more tools that allow non-ontology professionals to interact with ontologies are available. Although such a task might require collaboration across the industry, the effort would be transformed into business value based on the discussion in Chapter 2.

According to the UAT result, it is also worth noting that the participants reported that the UI should be made more intuitive to allow those who know little about ontologies to conceptualise their knowledge and make it conform with a given ontology. Besides that, more flexibility should be given to ontology professionals while properly considering the requirement for making the tool friendly to novices. More research should be conducted to achieve this goal.

8.1.5 How Can We Reproduce Some Manual Processes Using Ontologies to Achieve More Digitalised and More Effective Processes in the Railway Industry?

It has been proven that ontologies can enable more efficient data management and provide more possibilities in terms of data analysis. Yet, there is a lack of discussion about replacing existing manual process with ontologies.

It has been identified that the data collected for many rail intervention research projects is of a one-off nature, that it is hardly reused, and the result data is solely used as part of proof of concept for a specific study. In T2F, because of the lack of a generalised and standardised data description framework, it is difficult to manage existing rail intervention data and process data from different sources and of differing provenance, and, where necessary, of variable quality and criticality. Although some researchers have proposed a transferable approach that addresses this issue, they have not considered data representation. In Chapter 7, an ontology-based approach is proposed with reference to transferability and interoperability, to replicate an existing approach. The proposed ontology-based solution, including a supplementary QGIS plugin, simplifies the data management process, providing a generic method to represent a network model, track characteristics, intervention factors and geospatial data. A case study was conducted to demonstrate how to use the proposed approach to model the data and reproduce the existing manual data preparation and conclusion process. Integrated data storage was established, making it possible to

combine newly input data and previous output data as a new data package. The GIS analysis process was made further transferable, providing proper reference to other similar works. This addresses the importance of establishing a generic data representation framework within a domain.

The overall time taken was also compressed owing to data integration. The data collection process was completed with single queries from the triple store instead of multiple shape files. The time-consumption test result revealed that the whole process became 16% faster once the required data has been properly mapped to RaCoOn.

However, it seems impossible to fully eliminate the manual process because of the barrier between different plugins and software. This requires collaboration between software developers. Should they use RaCoOn as the means to represent the data, this issue could be resolved. In addition, richer annotation of data has a slight impact on performance. Despite this slight impact, analysing the industry-level data volume could take a long time, which requires a further test of scalability.

8.2 CONCLUSION

According to the thorough review presented in Chapter 2, ontology models have benefited many industries, such as the oil and gas industry, in a range of business and research activities by integrating heterogeneous and fragmented data silos. Notably, the complexity and heterogeneity in the oil and gas industry, and the UK railway industry are similar, so the successful establishment of ontology-based data integration and management solutions has set a great example for the UK railway industry. Meanwhile, the research into the adoption of ontology-based data models and solutions thereof gained interest from both the research community and the railway industry in the UK. However, no evidence was available to prove the adoption of ontologies was undertaken, despite being researched for years with proven benefits and necessity.

To address this issue, a mixed research methodology was taken to identify the factors that deterred railway industry professionals to use ontologies and example solutions to resolve accordingly. Chapter 4 elaborated on a survey to UK railway professionals and researchers, and through concluding survey candidates' responses and comments, plus supporting literature, three factors were summarised, that are:

a) Despite having been deployed in other industries, little has been done to demonstrate how to use ontologies to structure data in the railway industry.

b) A lack of supplementary tools for using ontologies repels potential users.

c) There has been little investigation into replicating the current manual process with ontologies in a commercial environment in the railway industry in the UK.

Example solutions were identified based on some existing literature using the combination of action and empirical research methodology; they were thereby demonstrated in Chapter 5, 6 and 7 respectively.

In Chapter 5, the role and the benefits of the ontology to manage the unstructured data in the UK railway industry were discussed, and an example prototype to use an ontology alongside existing machine learning algorithm to enhance the learning performance and realise automated and integrated unstructured documents was demonstrated. This chapter set an example of how ontologies could help the professionals to complete their existing work with less effort and enable more possible interaction with other technologies.

In Chapter 6, it aimed to provide a direction of development that how we can enrich choices of tools for ontologies. It has been found that most tools for editing a rule in SWRL which is an important part of the Semantic Web and ontology inference system require sufficient prerequisite IT knowledge, whereas rail domain experts are necessarily proficient in IT, but they would benefit from ontology rules to conduct preliminary tests on the data they

have. To enable non-ontology professionals to edit a rule in SWRL, a graphic rule designer was designed. The design was proven to be valuable to be further investigated and refined via a UAT. It also set up a development direction for other similar supplementary tools for ontologies.

Chapter 7 demonstrated how the ontology could practically replace manual processes through a project where fragmented and heterogenous data was involved. The proposed solution saved time and labour by removing or shortening the manual processes during the analysis.

Through the comprehensive reviews and proposing example solutions, the thesis intended to provide other researchers or professionals working in the UK railway industry with more insights into the benefits of ontologies by elaborating on why and how to adopt them in the practice. It aimed to fill in the gap between theories of existing ontology-based techniques and practical applications in the UK railway industry, inspiring similar study in the future.

8.3 FUTURE WORK

According to the results from Chapter 4, there is still a need for more dedication to populating ontologies in the UK rail industry. Achieving this requires a joint effort across the industry and perhaps, a government mandate, too, because it is often seen that stakeholders tend to keep their own data silos. Meanwhile, the entry level for using ontologies should be further lowered. Tools with an intuitive operation flow that allow non-ontology experts

to interact with ontologies should be made more and more available. Programming frameworks that can help users to make use of existing ontologies should be considered, forming middleware to facilitate an easier integration process. According to the result concluded from the investigation, more people could find it appealing to attempt to use ontologies once there are more tools and programming frameworks available. The UK rail industry is still craving for better data management and greater data accessibility (Rail Delivery Group, 2020); and, as this thesis suggests, ontologies could contribute to this goal. The full potential of ontologies, as a valuable technology for data management, has not yet been exercised, so more studies into implementing a domain-wide system using either RaCoOn or other suitable candidates should be considered.

Additionally, despite the demonstration in this thesis of some possible applications with an RDO, named RaCoOn, these applications have not been optimised because the emphasis of the thesis was to supply proof of concept. Thus, the tools and methods proposed in this thesis can be further refined, and the following points can be considered.

First, the proposed SWRL rule designer cannot support some user-friendly functions such as URI auto-completion or automatic assertion of new concepts, as there are no supporting tools for working ontologies available in JavaScript. Again, this is because of the lack of supporting tools, which can be addressed in future work.

Second, the learning framework can only exploit RaCoOn as a reference for model training. In fact, the research on using learning techniques to extract knowledge models is getting increasing attention, and could be greatly beneficial to tasks performed in the rail industry, such as risk assessment, seasonal risk prevention, predictive maintenance, digitalisation document management, etc. However, although ontology learning, ontology matching and alignment techniques have a long history, their application in rail-relevant topics has not been discussed. Some example works can be referenced as the start (Euzenat and Shvaiko, 2007; Euzenat et al., 2013; Haendel et al., 2018; Ivanova, 2011; Jain et al., 2010; Kacfah Emani et al., 2015). Knowledge model extraction, i.e. ontology learning, could be incorporated into other AI-based tasks to achieve more automated or even autonomous operation (Chungoora, 2019). Another issue with the work presented in Chapter 5 is that the proposed framework only applies to textual data. Other types of unstructured data should be mapped and tested in the future. On the other hand, the training and testing samples used for the case study are relatively small. The performance with mass unstructured data requires investigation.

Third, an ontology-based intervention data representation method and GIS analysis approach is presented in Chapter 7. Because of limited access to data, additional economic modelling and analysis could not be completed. This reflects the issue in the UK rail industry that related data is often kept by different participants; it is not always a simple task to be granted access to all required data, plus it is also difficult to ensure the data obtained from

different parties is consistent. In order to resolve this issue, a global IL that is built on top of existing data silos should be considered, consisting of a licensing system to grant access to the corresponding data package with satisfactory data consistency. Meanwhile, the proposed plugin and approach presented in Chapter 7 are still under active development. The plugin is not yet optimised, which, in particular, could impact its performance. Because it is a software engineering issue, it was not a focus of this proof of concept. However, addressing this issue might be beneficial to the production environment because it has been discussed that ontology-based computing tends to take more time due to richer data annotation.

Another issue with the proposed plugin is that it can only work with RaCoOn; thus, a solution to make it accept other ontologies should be considered to achieve a higher level of generalisation. It would also be of interest to enable the plugin to work with other available Linked Data stores open on the Web, e.g., DBPedia, but this might require the design of an additional ontology-related algorithm (such as ontology learning, ontology alignment), etc. Besides, ideally, a similar solution that could be in the form of stand-alone software that can communicate with QGIS would be better because this would provide more flexibility for future extensions. Porting the plugin to be stand-alone software was not possible at the point the study was conducted, because relative API(s) to exchange data with external software were not supplied by QGIS.

Third, it was also identified that the manual process cannot be fully removed from the existing process because data flow between different plugins within QGIS is not always permitted so that there still might be a manual data setup with reference to the result generated from previous steps. Additional manual involvement could be removed with the following attempts:

- Establish a generalised analysis architecture with recommendations of data flow for software/plugin development
- Use RaCoOn to represent the required data with consensus from major rail intervention operators to facilitate the generalisation of software development and deployment

Overall, further investigation is needed to improve the proposed extension of RaCoOn as well as the performance of the plugin. The plugin should be set with additional extensibility to enable users to customise their own approach to work with not only ground-borne noise data modelling but also other tasks in QGIS.

Last but not the least, as the major trend of development of AI has revealed the need for human knowledge representation (Gruber, 2017), research into ontology-based AI for railway operation would be valuable. This conforms to the developing trend of AI so that it is necessary to invent a mechanism to facilitate a higher level of rail system antonymy for the railway operation to keep increasing the safety level and operation efficiency with ontology-based AI agents. Due to the complexity of the UK railway industry, plus data

being proprietary, it is not always a simple task to coordinate every stake-holder to attempt to use new technology. Applying industry-wide ontologies might require a government mandate, which also requires additional effort and costs to facilitate. Thus, relevant policy and plans of fundings should be also investigated.

LIST OF REFERENCES

Abanda, H., Ng'Ombe, A., Tah, J.H.M., et al. (2011) An ontology-driven decision support system for land delivery in Zambia. *Expert Systems with Applications*, 38 (9): 10896–10905. doi:10.1016/j.eswa.2011.02.130.

Abdi, H. and Williams, L.J. (2010) Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2: 433–459. doi:10.1002/wics.101.

Ackoff, R.L. (1989) From data to wisdom. *Journal of Applied Systems Analysis*, 16 (1): 3–9.

Alatrish, E.S. (2012) Comparison of ontology editors. *e-RAF Journal on Computing*, 4 (2): 23–38.

Alfaries, A., 2010. Ontology learning for semantic web services (Doctoral dissertation, Brunel University, School of Information Systems, Computing and Mathematics Theses).

Armstrong, J., Ortega, A., Blainey, S., et al. (2019) Noise reduction for ballasted track: A comparative socio-economic assessment. *International Journal of Transport Development and Integration*, 3 (1): 15–29. doi:10.2495/TDI-V3-N1-15-29.

Armstrong, J. and Preston, J. (2019) Balancing railway network availability

and engineering access. *Proceedings of the Institution of Civil Engineers - Transport*, 173 (4): 209–217. doi:10.1680/jtran.19.00045.

Armstrong, J., Rempelos, G., Wei, J., et al. (2020) “Developing a generalised assessment framework for railway interventions.” In *Computers in Railways XVII*. 2020. pp. 127–138. doi:10.2495/cr200121.

Ashburner, M., Ball, C.A., Blake, J., et al. (2000) Gene Ontology: Tool for the unification of biology. *Nature Genetics*, 25 (1): 25–29. doi:10.1038/75556.

Avison, D.E., Lau, F., Myers, M.D. and Nielsen, P.A., 1999. Action research. Communications of the ACM, 42(1), pp.94-97.

Barry, J.A., Mollan, S., Burdon, M.A., et al. (2017) Development and validation of a questionnaire assessing the quality of life impact of Colour Blindness (CBQoL). *BMC Ophthalmology*, 17: 179. doi:10.1186/s12886-017-0579-z.

Bartram, D., Burrow, M. and Yao, X. (2008) A computational intelligence approach to railway track intervention planning. In Yu, T., Davis, L., Baydar C. and Roy R. (eds.) *Evolutionary Computation in Practice (Studies in Computational Intelligence)*. Springer. pp. 163–198. doi:10.1007/978-3-540-75771-9_8.

Bastinos, A.S and Krisper, M. (2013) Multi-criteria decision making in ontologies. *Information Sciences*, 222: 593–610.

doi:10.1016/j.ins.2012.07.055.

Battle, R. and Kolas, D. (2012) GeoSPARQL: Enabling a geospatial semantic web. *Semantic Web Journal*, 3 (4): 355–370. doi:10.3233/SW-2012-0065.

BBC (n.d.) *Ontologies*. Available at: https://www.bbc.co.uk/ontologies (Accessed: 7 May 2020).

Beel, J., Gipp, B., Langer, S., et al. (2016) Research-paper recommender systems: A literature survey. *International Journal on Digital Libraries*, 17: 305–338. doi:10.1007/s00799-015-0156-0.

Benbasat, I. and Zmud, R.W., 1999. Empirical research in information systems: The practice of relevance. MIS quarterly, pp.3-16.

Bergman, M. (2009) The fundamental importance of keeping an ABox and TBox split. *Ai3*, pp. 1–5. Available at: http://www.mkbergman.com/489/ontology-best-practices-for-data-driven-applications-part-2/ (Accessed: 20 November 2017).

Berners-Lee, T. (2006) Linked Data - Design issues. *Design Issues*. Available at: https://www.w3.org/DesignIssues/LinkedData.html.

Berners-Lee, T. (2009) *The Next Web*. Available at: https://www.ted.com/talks/tim_berners_lee_on_the_next_web.

Berners-Lee, T., Hendler, J., Lassila, O. (2001) The Semantic Web. *Scientific*

American, 284 (5): 1–9. doi:10.1038/scientificamerican0501-34.

Bird, S., Klein, E. and Loper, E. (2009) *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, Inc.

Bishr, Y. (1998) Overcoming the semantic and other barriers to GIS interoperability. *International Journal of Geographical Information Science*, 12 (4): 299–314. doi:10.1080/136588198241806.

Bizer, C., Heath, T. and Berners-Lee, T. (2009) Linked Data - The story so far. *International Journal on Semantic Web and Information Systems*, 5 (3): 1–22. doi:10.4018/jswis.2009081901.

Bizer, C., Heath, T., Idehen, K., et al. (2008) "Linked data on the web (LDOW2008)." In *Proceedings of the 17th International Conference on World Wide Web 2008, WWW'08*. 2008. pp. 1265–1266. doi:10.1145/1367497.1367760.

Bjørnsen, H.N., Eilertsen, M.E.B., Ringdal, R., et al. (2017) Positive mental health literacy: Development and validation of a measure among Norwegian adolescents. *BMC Public Health*, 17: 717. doi:10.1186/s12889-017-4733-6.

Blomqvist, E. (2014) The use of Semantic Web technologies for decision support – A survey. *Semantic Web*, 5 (3): 177–201. doi:10.3233/SW-2012-

0084.

Bobkowska, A. (2013) “On explaining intuitiveness of software engineering techniques with user experience concepts.” In *ACM International Conference Proceeding Series*. 2013. pp. 1–8. doi:10.1145/2500342.2500348.

Bodenreider, O. (2008) Biomedical ontologies in action: Role in knowledge management, data integration and decision support. *Yearbook of Medical Informatics*, 3841: 67–79. doi:me08010067 [pii].

Bodenreider, O. and Stevens, R. (2006) Bio-ontologies: Current trends and future directions. *Briefings in Bioinformatics*, 7 (3): 256–274. doi:10.1093/bib/bbl027.

Borst, W. (1997) *Construction of Engineering Ontologies for Knowledge Sharing and Reuse*. PhD Thesis, University of Twente, Enschede, Netherlands.

Braun, S., Schmidt, A. and Walter, A. (2007) Ontology maturing: A Collaborative Web 2.0 approach to ontology engineering. *World Wide Web Internet And Web Information Systems*, 273 (WWW ’07).

Brewer, J. (2011) *Information Systems Architecture for the Rail Industry - an Initial Overview T962*. Available at: https://www.rssb.co.uk/library/research-development-and-

innovation/research-brief-T962.pdf (Accessed: 6 December 2017).

Brickley, D. and Berners-Lee, T. (2003) *WGS84 Geo Positioning (geo)*. Available at: https://lov.linkeddata.es/dataset/lov/vocabs/geo (Accessed: 14 October 2020).

Brochhausen, M., Spear, A.D., Cocos, C., et al. (2011) The ACGT Master Ontology and its applications - Towards an ontology-driven cancer research and management system. *Journal of Biomedical Informatics*, 44 (1): 8–25. doi:10.1016/j.jbi.2010.04.008.

Cai, H., Zheng, V.W. and Chang, K.C.C. (2018) A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Transactions on Knowledge and Data Engineering*, 30 (9): 1616–1637. doi:10.1109/TKDE.2018.2807452.

Calvanese, D., De Giacomo, G., Lembo, D., et al. (2011) The MASTRO system for ontology-based data access. *Semantic Web*, 2 (1): 43–53. doi:10.3233/SW-2011-0029.

Camous, F., Blott, S. and Smeaton, A.F. (2007) "Ontology-based MEDLINE document classification." In *1st International Conference on Bioinformatics Research and Development*. 2007. pp. 439–452. doi:10.1007/978-3-540-71233-6_34.

Capacity for Rail (2017) *Collaborative Project SCP3-GA-2013-60560*

Increased Capacity 4 Rail Networks Through Enhanced Infrastructure and Optimised Operations Deliverable D3.4.2 Data Architecture. Available at: http://www.capacity4rail.eu/IMG/pdf/c4r_d3.4.2_verified_data_architecture.pdf.

Carmen Suárez-Figueroa, M., Gómez-Pérez, A., Fernández-López, M., et al. (2012) "The NeOn Methodology for Ontology Engineering." In Suárez-Figueroa, M., Gómez-Pérez, A., Motta, E. and Gangemi, A. (eds.) *Ontology Engineering in a Networked World*. Springer. pp. 9–34. doi:10.1007/978-3-642-24794-1_2.

Castro, J., Kolp, M. and Mylopoulos, J. (2001) "A requirements-driven development methodology." In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2001. pp. 265–280. doi:10.1007/978-3-642-36926-1_21.

Chandrasekaran, B., Josephson, J.R. and Benjamins, V.R. (1999) What are ontologies, and why do we need them? *IEEE Intelligent Systems and Their Applications*, 14 (1): 20–26. doi:10.1109/5254.747902.

Chang, X. (2008) *Ontology Development and Utilization in Product Design*. PhD Thesis, Virginia Tech, Blacksburg, VA.

Che, H. (2006) A Semantic Web primer. *Journal of the American Society for Information Science and Technology*, 57 (8): 1132. doi:10.1002/asi.20368.

Chen, D., Müller, H.M. and Sternberg, P.W. (2006) Automatic document classification of biological literature. *BMC Bioinformatics*, 7: 370. doi:10.1186/1471-2105-7-370.

Cheng, C., Pan, X. and Kurfess, F. (2004) "Ontology-based semantic classification of unstructured documents." In Nürnberger, A. and Detyniecki, M. (eds.) *Adaptive Multimedia Retrieval*. Springer. pp. 120–131. doi:10.1007/978-3-540-25981-7_8.

Choi, Y. (2014) "From siloed data to linked data: Developing a social metadata repository." In *Proceedings of the ASIST Annual Meeting*. 2014. pp. 1–4 doi:10.1002/meet.2014.14505101144.

Chollet, F. (2015) *Keras*. doi:10.1016/j.it.2007.05.003.

Chungoora, T. (2019) *Practical Knowledge Modelling Knowledge & its Lifecycle*. Available at: https://www.udemy.com/course/practical-knowledge-modelling/ (Accessed: 1 December 2019).

Cimiano, P., Haase, P., Herold, M., et al. (2007) "LexOnto: A model for ontology lexicons for ontology-based NLP." In *International Semantic Web Conference*. 2007.

Clark, K., Parsia, B., Grove, M., et al. (2011) *Pellet: Owl 2 Reasoner for Java*. Available at: http://clarkparsia.com/pellet.

Codd, E.F. (1970) A relational model of data for large shared data banks.

Communications of the ACM, 13 (6): 377–387. doi:10.1145/362384.362685.

Collet, C., Huhns, M.N. and Shen, W.M. (1991) Resource integration using a large knowledge base in Carnot. *Computer*, 24 (12): 55–62. doi:10.1109/2.116889.

Competition Commision (2007) *Rolling Stock Leasing Market Investigation: Industry Background Working Paper*. Available at: https://webarchive.nationalarchives.gov.uk/20080108225350/http://www.competition-commission.org.uk//inquiries/ref2007/roscos/pdf/working_paper_industry_background.pdf.

Compton, M., Barnaghi, P., Bermudez, L., et al. (2012) The SSN ontology of the W3C semantic sensor network incubator group. *Journal of Web Semantics*, 17: 25–32. doi:10.1016/j.websem.2012.05.003.

Cormode, G. and Krishnamurthy, B. (2008) Key differences between Web 1.0 and Web 2.0. *First Monday*, 13 (6). doi:10.5210/fm.v13i6.2125.

Cortes, C. and Vapnik, V. (1995) Support-vector networks. *Machine Learning*, 20: 273–297. doi:10.1023/A:1022627411411.

Cortina, J.M. (1993) What is coefficient alpha? An examination of theory and applications. *Journal of Applied Psychology*, 78 (1): 98–104.

doi:10.1037/0021-9010.78.1.98.

Crawford, J.R., Stewart, L.E., Cochrane, R.H.B., et al. (1989) Construct validity of the National Adult Reading Test: A factor analytic study. *Personality and Individual Differences*, 10 (5): 585–587. doi:10.1016/0191-8869(89)90043-3.

Cronbach, L.J. (1951) Coefficient alpha and the internal structure of tests. *Psychometrika*, 16: 297–334. doi:10.1007/BF02310555.

Cronbach, L.J. and Meehl, P.E. (2017) "Construct validity in psychological tests." In Bynner, J. and Stribley, K.M. (eds.) *Research Design: The Logic of Social Inquiry*. New York: Routledge. pp. 225–238. doi:10.4324/9781315128498.

Cruz, I.F. and Xiao, H. (2005) The role of ontologies in data integration. *Engineering Intelligent Systems for Electrical Engineering and Communications*, 13 (4): 245.

Davis, F.D., 1993. User acceptance of information technology: system characteristics, user perceptions and behavioral impacts. International journal of man-machine studies, 38(3), pp.475-487.

Davies, S., Daniels, J., Li, T., et al. (2019) *Can the Railway Industry Utilize Data to its Full Potential?* Available at: https://www.wsp.com/-/media/Insights/Canada/Documents/2019/Whitepaper_Big_Data_And_Rai

l_EN.pdf.

Davis, D. (2019) *AI Unleashes the Power of Unstructured Data*. *CIO*. Available at: https://www.cio.com/article/3406806/ai-unleashes-the-power-of-unstructured-data.html (Accessed: 4 November 2020).

Department for Transport (2011a) *Rail Technical Specification Interoperability (TSI) - gov.uk*. Available at: https://www.gov.uk/government/publications/rail-interoperability-tsi-faq/rail-technical-specification-interoperability-tsi (Accessed: 29 March 2020).

Department for Transport (2011b) *Realising the Potential of Rail in Great Britain*. Available at: https://www.gov.uk/government/publications/realising-the-potential-of-gb-rail (Accessed: 3 April 2020).

Department for Transport (2017) *Connecting People: A Strategic Vision for Rail*. Available at: https://www.gov.uk/government/publications/a-strategic-vision-for-rail/connecting-people-a-strategic-vision-for-rail (Accessed: 7 December 2017).

Department for Transport (2018) *Joint Rail Data Action Plan: Addressing Barriers to Make Better Use of Rail Data*. Available at: https://www.gov.uk/government/publications/joint-rail-data-action-plan/joint-rail-data-action-plan-addressing-barriers-to-make-better-use-of-

rail-data (Accessed: 8 May 2020).

Dill, J. (2019) “Big Data.” In *Advanced Information and Knowledge Processing*. Cham: Springer. pp. 11–31. doi:10.1007/978-3-030-24367-8_2.

Dillon, T., Chang, E., Hadzic, M., et al. (2008) “Differentiating conceptual modelling from data modelling, knowledge modelling and ontology modelling and a notation for ontology modelling.” In *Conferences in Research and Practice in Information Technology Series*. 2008.

Dong, X.L. and Srivastava, D. (2013) Big data integration. *Proceedings of the VLDB Endowment*. doi:10.14778/2536222.2536253.

Durazo-Cardenas, I., Amaraegbeni, C. and Starr, A. (2016) Fusion of railway network data streams for asset usage in intelligent maintenance systems. *Euromaintenance 2016*, pp. 59–64.

Durk, J. (2013) *Customer Information Strategy Update*. London: ATOC.

Easton, J.M., Davies, J.R. and Roberts, C. (2010) “Railway modelling - The case for ontologies in the rail industry.” In *KEOD 2010 - Proceedings of the International Conference on Knowledge Engineering and Ontology Development*. 2010. pp. 257–262. doi:10.5220/0003092102570262.

Ebrahimipour, V. and Yacout, S. (2015) “ISO 15926.” In Ebrahimipour, V. and Yacout, S. (eds.) *Ontology Modeling in Physical Asset Integrity Management*. Springer. pp. 1–16. doi:10.1007/978-3-319-15326-1_1.

Ehrlinger, L. and Wöß, W. (2016) “Towards a definition of knowledge graphs.” In *CEUR Workshop Proceedings*. 2016.

Elkjaer, B. and Simpson, B., 2011. Pragmatism: A lived and living philosophy. What can it offer to contemporary organization theory?. In Philosophy and organization theory. Emerald Group Publishing Limited.

Ericson, G. and Rohm, W.A. (2017) *How to Choose Machine Learning Algorithms*. Microsoft Docs. Microsoft. Available at: https://docs.microsoft.com/en-us/azure/machine-learning/studio/algorithm-choice.

ERTMS Solutions (n.d.) *Ontologies For Railways: IT Integration Using Ontologies*. Available at: www.ertmssolutions.com/wp-content/uploads/2018/01/Ontologies.pdf (Accessed: 19 May 2020).

Escórcio, A.L.N. and Cardoso, J. (2007) “Editing tools for ontology creation.” In Cardoso, J. (ed.) *Semantic Web Services: Theory, Tools and Applications*. IGI Global. pp. 71–95. doi:10.4018/978-1-59904-045-5.ch004.

Euzenat, J. and Shvaiko, P. (2007) *Ontology Matching*. Springer. doi:10.1007/978-3-540-49612-0.

Euzenat, J., Shvaiko, P., Euzenat, J., et al. (2013) “Classifications of ontology matching techniques.” In *Ontology Matching*. Springer. pp. 61–72. doi:10.1007/978-3-642-38721-0_4.

Fader, A., Soderland, S. and Etzioni, O. (2011) “Identifying relations for open information extraction.” In *Proceedings of the Conference of Empirical Methods in Natural Language Processing (EMNLP ’11)*. Edinburgh, Scotland, UK, July 2011.

Fang, J., Guo, L., Wang, X., et al. (2007) Ontology-based automatic classification and ranking for web documents. *Proceedings - Fourth International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2007*, 3 (1): 627–631. doi:10.1109/FSKD.2007.432.

Farooq, K., Hussain, A., Leslie, S., et al. (2011) “Ontology-driven cardiovascular decision support system.” In *2011 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops*. 2011. IEEE. pp. 283–286. doi:10.4108/icst.pervasivehealth.2011.246092.

Fensel, D. (2002) Ontology-based knowledge management. *Computer*, 35 (11): 56–59. doi:10.1109/MC.2002.1046975.

Fitzner, K. (2007) Reliability and validity: A quick review. *Diabetes Educator*, 33 (5): 775–776. doi:10.1177/0145721707308172.

Fredsall, A. (2015) *What is data silo ? - Definition from WhatIs.com*. Available at: https://searchdatamanagement.techtarget.com/definition/data-silo (Accessed: 2 November 2019).

Gabrilovich, E. and Markovitch, S. (2005) “Feature generation for text categorization using world knowledge.” In *IJCAI International Joint Conference on Artificial Intelligence*. 2005.

Gaitanou, P. (2009) "Ontologies and ontology-based applications." In Sicilia, M.-A. and Lytras, M.D. (eds.) *Metadata and Semantics*. Springer US, Boston, MA. pp. 289–298.

Gangemi, A. and Presutti, V. (2009) “Ontology design patterns.” In Staab, S. and Studer, R. (eds.) *Handbook on Ontologies*. Berlin, Heidelberg: Springer. pp. 241–243. doi:10.1007/978-3-540-92673-3_10.

Gardner, S.S.P. (2005) Ontologies and semantic data integration. *Drug Discovery Today*, 10 (60761002): 2440–2443. doi:10.1016/S1359-6446(05)03504-X.

George, D. and Mallery, P. (2003) *SPSS for Windows Step by Step*. Boston: Allyn and Bacon.

Gibbins, N. and Shadbolt, N. (2011) “Resource description framework (RDF).” In Bates, M.J. (ed.) *Understanding Information Retrieval Systems: Management, Types, and Standards*. Auberback Publications. Chapter 45. doi:10.1201/b11499-53.

Glen, S. (2016) *Kaiser-Meyer-Olkin (KMO) test for Sampling Adequacy*. StatisticsHowTo.com. Available at:

https://www.statisticshowto.com/kaiser-meyer-olkin/.

Goforth, C. (2015) *Using and Interpreting Cronbach's Alpha*. Available at: https://data.library.virginia.edu/using-and-interpreting-cronbachs-alpha/.

Gogos, S. and Letellier, X. (2016) IT2Rail: Information technologies for Shift to Rail. *Transportation Research Procedia*, 14: 3218–3227. doi:10.1016/j.trpro.2016.05.265.

Golafshani, N. (2003) Understanding reliability and validity in qualitative research. *The Qualitative Report*, 8 (4): 597–606. doi:10.46743/2160-3715/2003.1870.

Gómez-Pérez, A., Fernández-López, M., Corcho, O., et al. (2010) *Ontological Engineering with Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web*. Springer Verlag.

Google (2018) *Cloud Natural Language. Cloud Natural Language API*. Google Cloud. Available at: https://cloud.google.com/natural-language/ (Accessed: 14 November 2018).

Goyal, P. and Ferrara, E. (2018) Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems*, 151: 78–94. doi:10.1016/j.knosys.2018.03.022.

Gray, J.P. (1997) CASE tool construction for a parallel software development methodology. *Information and Software Technology*, 39 (4):

235–252. doi:10.1016/S0950-5849(96)01145-7.

Groß, A., Pruski, C. and Rahm, E. (2016) Evolution of biomedical ontologies and mappings: Overview of recent approaches. *Computational and Structural Biotechnology Journal*, 14: 333–340. doi:10.1016/j.csbj.2016.08.002.

Gruber, T. (2009) "Ontology (computer science)." In Liu, L. and Özsu, M.T. (eds.) *Encyclopedia of Database Systems*. doi:10.1007/978-0-387-39940-9_1318.

Gruber, T.R. (1993) A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5 (2): 199–220. doi:10.1006/knac.1993.1008.

Gruber, T. (2017). How AI can enhance our memory, work and social lives [Video]. TED Conferences. https://www.ted.com/talks/tom_gruber_how_ai_can_enhance_our_memory_work_and_social_lives?language=en

Guarino, N. (1995) Formal ontology, conceptual analysis and knowledge representation. *International Journal of Human-Computer Studies*, 43 (5–6): 625–640. doi:10.1006/ijhc.1995.1066.

Guarino, N. (1997) Understanding, building and using ontologies. *International Journal of Human-Computer Studies*, 46 (2–3): 293–310.

doi:10.1006/ijhc.1996.0091.

Guarino, N. (1998) Formal ontology and information systems. *Proceedings of the First International Conference on Formal Ontology in Information Systems*, Trento, Italy, June 1998. pp. 3–15. doi:10.1.1.29.1776.

Guarino, N., Oberle, D., Staab, S., et al. (2009) “What is an ontology?” In Staab, S. and Studer, R. (eds.) *Handbook on Ontologies*. Springer. pp. 1–17. doi:10.1007/978-3-540-92673-3_0.

Gyorodi, C., Gyorodi, R., Pecherle, G., et al. (2015) “A comparative study: MongoDB vs. MySQL.” In *2015 13th International Conference on Engineering of Modern Electric Systems, EMES 2015*. 2015. doi:10.1109/EMES.2015.7158433.

Haendel, M.A., Chute, C.G. and Robinson, P.N. (2018) Classification, ontology, and precision medicine. *New England Journal of Medicine*, 379: 1452–1462. doi:10.1056/NEJMra1615014.

Hales, S.D. and Johnson, T.A. (2003) Endurantism, perdurantism and special relativity. *Philosophical Quarterly,* 53 (213): 524–539. doi:10.1111/1467-9213.00329.

Haller, A., Janowicz, K., Cox, S., et al. (2017) *Semantic Sensor Network Ontology*. Available at: https://www.w3.org/TR/vocab-ssn/#intro (Accessed: 9 January 2018).

Halpin, T. and Morgan, T. (2010) *Information Modeling and Relational Databases*. Morgan Kaufmann. Available at: http://books.google.it/books?id=puO_VlbR_x4C&printsec=frontcover&dq=intitle:Information+Modeling+and+Relational+Databases+Second+Edition+The+Morgan+Kaufmann+Series+in+Data+Management+Systems&hl=&cd=1&source=gbs_api.

Hambling, B. and Van Goethem, P. (2013) *User Acceptance Testing: A Step-by-Step Guide*. Swindon: BCS Learning & Development.

Hammarberg, K., Kirkman, M. and de Lacey, S., 2016. Qualitative research methods: when to use them and how to judge them. Human reproduction, 31(3), pp.498-501.

Heath, T. and Bizer, C. (2011) Linked Data: Evolving the Web into a global data space. *Synthesis Lectures on the Semantic Web: Theory and Technology*. doi:10.2200/S00334ED1V01Y201102WBE001.

Hendler, J. (2009) Web 3.0 emerging. *Computer*, 42 (1). doi:10.1109/MC.2009.30.

Heo, M., Kim, N. and Faith, M.S. (2015) Statistical power as a function of Cronbach alpha of instrument questionnaire items. *BMC Medical Research Methodology*,15: 86. doi:10.1186/s12874-015-0070-6.

Herman, I., Fernández, S., Alonso, C.T., et al. (2020) *GitHub* -

RDFLib/SPARQLWrapper: A Wrapper for a Remote SPARQL Endpoint. Available at: https://github.com/RDFLib/sparqlwrapper (Accessed: 15 October 2020).

Hogan, T.P., Benjamin, A. and Brezinski, K.L. (2000) Reliability methods: A note on the frequency of use of various types. *Educational and Psychological Measurement*, 60 (4): 523–531. doi:10.1177/00131640021970691.

Horrocks, I., Patel-Schneider, P.F., Boley, H., et al. (2004) *SWRL : A Semantic Web Rule Language Combining OWL and RuleML*. W3C Member submission 21. Available at: http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/.

IBM (2019) *KMO and Bartlett's Test*. Available at: https://www.ibm.com/support/knowledgecenter/en/SSLVMB_subs/statistics_casestudies_project_ddita/spss/tutorials/fac_telco_kmo_01.html.

IIP Steering Group (2005) *PPT - Integrated Information Platform for Reservoir and Subsea Production Systems PowerPoint Presentation - ID:4375722*. Available at: https://www.slideserve.com/liko/integrated-information-platform-for-reservoir-and-subsea-production-systems (Accessed: 2 November 2020).

IIP Steering Group (2008) *IIP Project Report*. Høvik. Available at: https://www.posccaesar.org/raw-

attachment/wiki/IIP/IIP_sluttrapport_2008_Public.pdf].

Ikeda, M., Seta, K., Kakusho, O., et al. (1998) Task ontology: Ontology for building conceptual problem solving models. *Proceedings of ECAI98*, pp. 126–133.

Ivanova, T. (2011) Ontology alignment. *International Journal of Knowledge and Systems Science*, 1 (4). doi:10.4018/jkss.2010100102.

Jacquette, D. (2014) *Ontology*. London: Routledge. doi:10.4324/9781315710655.

Jain, P., Hitzler, P., Sheth, A.P., et al. (2010) "Ontology alignment for linked open data." In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2010. doi:10.1007/978-3-642-17746-0_26.

JGraph Ltd (n.d.) *GitHub - jGraph/mxGraph: mxGraph is a fully client side JavaScript diagramming library*. Available at: https://github.com/jgraph/mxgraph (Accessed: 25 August 2019).

Kacfah Emani, C., Cullot, N. and Nicolle, C. (2015) Understandable Big Data: A survey. *Computer Science Review*, 17:70–81. doi:10.1016/j.cosrev.2015.05.002.

Kaiser, M.O. (1974) Kaiser-Meyer-Olkin measure for identity correlation matrix. *Journal of the Royal Statistical Society* 52: 296–298.

Kalibatiene, D. and Vasilecas, O. (2011) “Survey on ontology languages.” In Grabis, J. and Kirikova, M. (eds.) *Lecture Notes in Business Information Processing*. Berlin, Heidelberg: Springer. pp. 124–141. doi:10.1007/978-3-642-24511-4_10.

Kambles, T., Roma, P., Mittal, N., et al. (2017) *Analyzing Dark Data for Hidden Opportunities*. Deloitte Insights. Available at: https://www2.deloitte.com/us/en/insights/focus/tech-trends/2017/dark-data-analyzing-unstructured-data.html (Accessed: 11 June 2020).

Knowledge Hub (2017) *What are Ontologies and What are the Benefits of Using Ontologies?* Available at: https://www.ontotext.com/knowledgehub/fundamentals/what-are-ontologies/ (Accessed: 22 November 2019).

Köpf, H. (2010) InteGRail - publishable final activity report. *InteGrail—Intelligent Integration of Railway Systems—Project No. FP6—012526, Tech. Rep. IGR-PDAP-156*, 7 (September): 64.

Kothari, C.R., 2004. Research methodology: Methods and techniques. New Age International.

Kotis, K. and Vouros, G.A. (2006) Human-centered ontology engineering: The HCOME methodology. *Knowledge and Information Systems*, 10 (1):

109–131. doi:10.1007/s10115-005-0227-4.

Lamy, J.B. (2016) “Ontology-oriented programming for biomedical informatics.” In *Studies in Health Technology and Informatics*. 2016. doi:10.3233/978-1-61499-633-0-64.

Leal, D. (2005) ISO 15926. *Oil & Gas Science and Technology*, 60 (4): 629–637.

Lee, Y.Y., Chen, N. and Johnson, R.E. (2013) “Drag-and-drop refactoring: Intuitive and efficient program transformation.” In *Proceedings - International Conference on Software Engineering*. 2013. doi:10.1109/ICSE.2013.6606548.

Legris, P., Ingham, J. and Collerette, P., 2003. Why do people use information technology? A critical review of the technology acceptance model. Information & management, 40(3), pp.191-204.

Lehmann, J., Isele, R., Jakob, M., et al. (2015) DBpedia - A large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web*, 6 (2): 167–195. doi:10.3233/SW-140134.

Lenzerini, M. (2011) “Ontology-based data management.” In *Proceedings of the International Conference on Information and Knowledge Management*. 2011. doi:10.1145/2064227.2064251.

Levermore, D.M., Gardner, J.M. and Blaylock, J. (2020) *Systems and*

Methods for Creating Intuitive Context for Analysis Data. Available at: https://patentimages.storage.googleapis.com/a5/92/31/fdf4f88bdd8ddd/US10748092.pdf.

Lewis, R. (2015) *A Semantic Approach to Railway Data Integration and Decision Support*. PhD Thesis, University of Birmingham, Birmingham, UK. Available at: https://etheses.bham.ac.uk/id/eprint/5959/1/Lewis15PhD.pdf (Accessed: 10 November 2017).

Lewis, R., Fuchs, F., Pirker, M., et al. (2006) "Using ontology to integrate railway condition monitoring data." In *IET International Conference on Railway Condition Monitoring*. 2006. IEE. pp. 149–155. doi:10.1049/ic:20060060.

Lieberman, J., Singh, R. and Goad, C. (2007) *W3C Geospatial Ontologies*. Available at: https://www.w3.org/2005/Incubator/geo/XGR-geo-ont-20071023/ (Accessed: 27 November 2019).

Liu, R.X., Kuang, J., Gong, Q., et al. (2003) Principal component regression analysis with SPSS. *Computer Methods and Programs in Biomedicine*, 71 (2): 141–147. doi:10.1016/S0169-2607(02)00058-5.

Liu, X., Du, H. and Liu, N. (2011) "Research on high-speed railway ontology integration method based on semantic relationships." In *ITME 2011 - Proceedings: 2011 IEEE International Symposium on IT in Medicine and*

Education. 2011. pp. 295–298. doi:10.1109/ITiME.2011.6130837.

Lodemann, M. and Luttenberger, N. (2010) “Ontology-based railway infrastructure verification, planning benefits.” In *International Conference on Knowledge Management and Information Sharing - KMIS 2010*. 2010.

Lu, J., Roberts, C., Lang, K.R., et al. (2006) “The application of semantic web technologies for railway decision support.” In *Intelligent Decision-making Support Systems: Foundations, Applications and Challenges*. London: Springer London. pp. 321–337. doi:10.1007/1-84628-231-4_17.

Mahesh, K. and Nirenburg, S. (1995) A situated ontology for practical NLP. In *Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, International Joint Conference on Artificial Intelligence (IJCAI-95)*.

Malone, J., Parkinson, H., Malone, J., et al. (2010) Reference and application ontologies. *Ontogenesis*. Available at: http://ontogenesis.knowledgeblog.org/295 (Accessed: 13 August 2018).

Malone, R. (2007) *Structuring Unstructured Data*. Available at: https://www.forbes.com/2007/04/04/teradata-solution-software-biz-logistics-cx_rm_0405data.html?sh=19f5ed3c7a42 (Accessed: 5 November 2020).

Manning, C., Surdeanu, M., Bauer, J., et al. (2014) “The Stanford CoreNLP natural language processing toolkit.” In *Proceedings of 52nd Annual*

Meeting of the Association for Computational Linguistics: System Demonstrations. 2014. pp. 55–60.

Marr, B. (2019) *What Is Unstructured Data and Why is it so Important to Businesses? An Easy Explanation for Anyone*. Available at: https://www.forbes.com/sites/bernardmarr/2019/10/16/what-is-unstructured-data-and-why-is-it-so-important-to-businesses-an-easy-explanation-for-anyone/?sh=19892b4015f6 (Accessed: 4 November 2020).

Martinez-Cruz, C., Blanco, I.J. and Vila, M.A. (2012) Ontologies versus relational databases: Are they so different? A comparison. *Artificial Intelligence Review*, 38: 271–290. doi:10.1007/s10462-011-9251-9.

Marzi, M. (2018) *Dynamic Rule-Based Decision Trees in Neo4j.* Availble at: https://dzone.com/articles/dynamic-rule-based-decision-trees-in-neo4j (Accessed: 20 Feburary 2018)

McCormick, B.K., Salcedo, J. and Poh, A. (2018) *Creating and Using a Multiple-Response Set in SPSS*. Available at: https://www.dummies.com/education/math/statistics/creating-and-using-a-multiple-response-set-in-spss/ (Accessed: 8 December 2020).

Medina-Oliva, G., Voisin, A., Monnin, M., et al. (2014) Predictive diagnosis based on a fleet-wide ontology approach. *Knowledge-Based Systems*, 68: 40–57. doi:10.1016/j.knosys.2013.12.020.

Mihalcea, R. and Tarau, P. (2004) TextRank: Bringing order into texts. *Proceedings of EMNLP*. pp. 404–411. doi:10.3115/1219044.1219064.

Mizoguchi, R. (2003) Tutorial on ontological engineering. Part 1: Introduction to ontological engineering. *New Generation Computing*, 21 (4): 365–384.

Mohan, R. and Arumugam, G. (2005) Constructing railway ontology using web ontology language and semantic web rule language. *International Journal of Computer Technology and Applications*, 2 (2): 314–321. Available at: citeulike-article-id:11551183.

Morris, C. (2017) *Data Integration in the Rail Domain*. PhD Thesis, University of Birmingham, Birmingham, UK.

Morris, C. and Easton, J. (2018) "Use of ontology for data integration in a degraded mode signalling system." In *WIT Transactions on the Built Environment*. 2018. pp. 215–223. doi:10.2495/CR180191.

Morris, C., Easton, J. and Roberts, C. (2014) "Applications of linked data in the rail domain." In *Proceedings - 2014 IEEE International Conference on Big Data, IEEE Big Data 2014*. 2014. IEEE. pp. 35–41. doi:10.1109/BigData.2014.7004429.

Morris, C., Easton, J.M. and Roberts, C. (2015) "Position paper: Ontology in the rail domain." In *KEOD*. 2015. pp. 285–290.

Munir, K. and Sheraz Anjum, M. (2017) The use of ontologies for effective knowledge modelling and information retrieval. *Applied Computing and Informatics*, 14 (2): 116–126. doi:10.1016/j.aci.2017.07.003.

Murdock, A.P., Harfoot, A.J.P., Martin, D., et al. (2015) *OpenPopGrid: An Open Gridded Population Dataset for England and Wales*. GeoData, University of Southampton.

Musen, M.A. (2015) The Protégé project: A look back and a look forward. *AI Matters*, 1 (4). doi:10.1145/2757001.2757003.

Nash, A., Huerlimann, D., Schuette, J., et al. (2010) railML - A standard data interface for railroad applications. *Timetable Planning and Information Quality*, 74: 3–10. doi:10.2495/978-1-84564-500-7/01.

Nath, K., Dhar, S. and Basishtha, S. (2014) "Web 1.0 to Web 3.0 - Evolution of the Web and its various challenges." In *ICROIT 2014 - Proceedings of the 2014 International Conference on Reliability, Optimization and Information Technology*. 2014. doi:10.1109/ICROIT.2014.6798297.

National Rail (2015) *National Rail Enquiries - WiFi Facilities*. Available at: http://www.nationalrail.co.uk/stations_destinations/44866.aspx#NT (Accessed: 16 November 2017).

National Rail Enquiries. (n.d.). *Darwin Data Feeds*. Available at: https://www.nationalrail.co.uk/100296.aspx (Accessed: 9 April 2020).

Neches, R., Fikes, R., Finin, T., et al. (1991) Enabling technology for knowledge sharing. *AI Magazine*, 12 (3): 36.

Neo4J (2018) *5 Sure Signs it's Time to Give Up Your Relational Database - Neo4j Graph Database Platform*. Available at: https://neo4j.com/blog/five-signs-to-give-up-relational-database/ (Accessed: 13 October 2020).

Network Rail (2013) *Technical Strategy (June 2013)*. Available at: https://cdn.networkrail.co.uk/wp-content/uploads/2017/02/Network-Rail-Technical-Strategy-2013.pdf (Accessed: 14 December 2017).

Network Rail Limited. (2017a) *Digital Railway*. Available at: https://www.networkrail.co.uk/our-railway-upgrade-plan/digital-railway/ (Accessed: 11 November 2017).

Network Rail Limited. (2017b) *Railway Upgrade Plan - Update 2017*. Available at: https://www.networkrail.co.uk/our-railway-upgrade-plan/ (Accessed: 12 November 2017).

Nielsen, J. (1994) "Usability inspection methods." In *Conference on Human Factors in Computing Systems - Proceedings*. 1994. doi:10.1145/259963.260531.

NLTK (2015) *Natural Language Toolkit*. NLTK. doi:10.1103/PhysRevB.62.4273.

Noble, W.S. (2006) What is a support vector machine? *Nature*

Biotechnology, 24: 1565–1567. doi:10.1038/nbt1206-1565.

Noy, N.F. and McGuinness, D.L. (2001) *Ontology Development 101: A Guide to Creating your First Ontology*. Stanford, CA: Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880. doi:10.1016/j.artmed.2004.01.014.

Nyberg, K., Raiko, T., Hyvönen, E., et al. (2010) "Document classification utilising ontologies and relations between documents." In *Proceedings of the 8th Workshop on Mining and Learning with Graphs, MLG'10*. 2010. doi:10.1145/1830252.1830264.

O'Reilly, T. (2006) Web 2.0 compact definition: Trying again. *Radar*. doi:10.2307/2640276.

Office of Rail and Road (2020) *Passenger Train km by Operator*. Available at: https://dataportal.orr.gov.uk/media/1486/passenger-train-kilometres-by-operator-table-1213.xlsx.

ON-TIME (2013) *Functional and Technical Requirements Specification for Large Scale Perturbation Management*. Available at: http://www.ontime-project.eu/download.aspx?id=2cf2b944-4e1b-4d60-86b7-3c35bb88cc7e.

Open Geospatial Consortium (2019) *Well-Known Text Representation of Coordinate Reference Systems*. pp. 1–113. Available at: https://www.ogc.org/standards/wkt-crs (Accessed: 15 October 2020).

Open Rail Data Wiki (n.d.). *TRUST vs Darwin*. Available at: https://wiki.openraildata.com/index.php/TRUST_vs_Darwin (Accessed: 9 April 2020).

Ortega, A., Blainey, S., Preston, J., et al. (2018) “Noise reduction for ballasted tracks: A socio-economic assessment.” In *WIT Transactions on the Built Environment*. 2018. pp. 461–472. doi:10.2495/CR180411.

Page, L., Brin, S., Motwani, R., et al. (1998) The PageRank citation ranking: Bringing order to the web. *World Wide Web Internet And Web Information Systems*, 54: 1–17. doi:10.1.1.31.1768.

Paredaens, J., Bra, P., Gyssens, M., et al. (1989) “Relational database model.” In *The Structure of the Relational Database Model*. Berlin Heidelberg: Springer. pp. 1–18. doi:10.1007/978-3-642-69956-6_1.

Parent, C. and Spaccapietra, S. (2000) Database integration: The key to data interoperability. In Papazoglou, M.P., Spaccapietra, S. and Tari, Z. (eds.) *Advances in Object-Oriented Data Modeling*. MIT Press. pp. 219–253.

Pasin, M. (2019) *Ontospy. Python Library and Command-Line Interface for Inspecting and Visualizing RDF Models.* Available at: http://lambdamusic.github.io/Ontospy/ (Accessed: 12 May 2020).

Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011) Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12

(Oct): 2825–2830. Available at: http://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html (Accessed: 12 November 2018).

Pérez, A., Baonza, M.D.F. and Villazón, B. (2008) NeOn methodology for building ontology networks: Ontology specification. In *NeOn Deliverable D5.4.1. NeOn Methodology for Building Contextualized Ontology Networks. NeOn Project*. http://www.neon-project.org. pp. 1–18.

Pérez, J., Arenas, M. and Gutierrez, C. (2009) Semantics and complexity of SPARQL. *ACM Transactions on Database Systems*, 34 (3): 16. doi:10.1145/1567274.1567278.

Perry, M. and Herring, J. (2012) *OGC GeoSPARQL - A Geographic Query Language for RDF Data*. Open Geospatial Consortium.

Peterson, L. (2009) K-nearest neighbor. *Scholarpedia*, 4 (2); 1883. doi:10.4249/scholarpedia.1883.

Petzold, C. (2005) *Does Visual Studio Rot the Mind? Ruminations on Psychology and Aesthetics of Coding*. Available at: http://www.charlespetzold.com/etc/DoesVisualStudioRotTheMind.html (Accessed: 5 January 2018).

Philosophy Terms (n.d.). *Ontology: Examples and Definition.* Available at: http://philosophyterms.com/ontology/ (Accessed: 7 December 2017).

Pinto, H.S. and Martins, J.P. (2004) Ontologies: How can they be built? *Knowledge and Information Systems*, 6 (4): 441–464. doi:10.1007/s10115-003-0138-1.

Purpura, A., Masiero, C., Silvello, G., et al. (2019) “Feature selection for emotion classification.” In *CEUR Workshop Proceedings*. 2019.

Python Software Foundation (2015) *16.3. Time — Time Access and Conversions — Python 3.4.4 documentation*. Available at: https://docs.python.org/3.6/library/time.html (Accessed: 26 February 2021).

Rail Delivery Group (2020) *Customer Information - An Industry Response to the Office of Rail and Road*. Available at: https://www.orr.gov.uk/media/21544/download.

Rajaraman, A. and Ullman, J.D. (2011) *Mining of Massive Datasets*. Cambridge University Press. doi:10.1017/CBO9781139058452.

Ramos, J. (2003) Using TF-IDF to determine word relevance in document queries. *Proceedings of the First Instructional Conference on Machine Learning*.

RDF Working Group (2014) *RDF - Semantic Web Standards*. W3C Recommendation February 2014.

RDFLib Team (2002) *rdflib 5.0.0 — rdflib 5.0.0 Documentation*. Available at:

https://pypi.org/project/pystardog/ (Accessed: 13 October 2020).

Reinsel, D., Gantz, J. and Rydning, J. (2018) *The Digitization of the World - From Edge to Core*. IDC White Paper.

Roberts, C., Easton, J., Davies, R., et al. (2011) *Rail Research UK Feasibility Account: The Specification of a System-Wide Data Framework for the Railway Industry–Final Report*. Available at: http://www.insufficientdata.co.uk/John_Easton_Homepage/Publications_files/DataFrameworkFeasibilityStudyFinalReport.pdf (Accessed: 7 December 2017).

Rouse, M. (n.d.) *What is Data Silo ? - Definition from WhatIs.com*. Available at: https://searchdatamanagement.techtarget.com/definition/data-silo (Accessed: 2 November 2019).

Rowley, J. (2007) The wisdom hierarchy: Representations of the DIKW hierarchy. *Journal of Information Science*, 33 (2): 163–180. doi:10.1177/0165551506070706.

RSSB (2018a) *Investigation into the Effect of Moisture on Rail Adhesion (T1042)*. Available at: https://catalogues.rssb.co.uk/research-development-and-innovation/research-project-catalogue/T1042 (Accessed: 30 August 2019).

RSSB (2018b) *The Effect of Water on the Transmission of Forces Between*

Wheels and Rails (T1077). Available at: https://catalogues.rssb.co.uk/research-development-and-innovation/research-project-catalogue/T1077 (Accessed: 30 August 2019).

Saa, R., Garcia, A., Gomez, C., et al. (2012) An ontology-driven decision support system for high-performance and cost-optimized design of complex railway portal frames. *Expert Systems with Applications*, 39 (10): 8784–8792. doi:10.1016/j.eswa.2012.02.002.

Safety Central (n.d.). *Trust*. Available at: https://safety.networkrail.co.uk/jargon-buster/trust/ (Accessed: 9 April 2020).

Santos, A. and Reynaldo, J. (2013) Cronbach's alpha: A tool for assessing the reliability of scales. *Journal of Extension*, 37: 1–5.

Saunders, M., Lewis, P. and Thornhill, A., 2009. Understanding research philosophies and approaches. *Research methods for business students*, *4*(1), pp.106-135.

Schmidt, M.-T., Hutchison, B., Lambros, P., et al. (2010) The Enterprise Service Bus: Making service-oriented architecture real. *IBM Systems Journal*, 44 (4): 781–797. doi:10.1147/sj.444.0781.

Scott, W. (2018) *Council Post: Why Data Silos are Bad for Business*. Available at: https://www.forbes.com/sites/forbestechcouncil/2018/11/19/why-data-silos-are-bad-for-business/#69f49ee75faf (Accessed: 2 November 2019).

seatgeek (n.d.) *GitHub - seatgeek/fuzzywuzzy: Fuzzy String Matching in Python*. Available at: https://github.com/seatgeek/fuzzywuzzy (Accessed: 28 August 2019).

Settouti, N., Bechar, M.E.A. and Chikh, M.A. (2016) Statistical comparisons of the top 10 algorithms in data mining for classification task. *International Journal of Interactive Multimedia and Artificial Intelligence*, 4: 46–51. doi:10.9781/ijimai.2016.419.

Shadbolt, N., Hall, W. and Berners-Lee, T. (2006) The Semantic Web revisited. *IEEE Intelligent Systems*, 21 (3): 96. doi:10.1109/MIS.2006.62.

Shannon, V. (2006) A “more revolutionary” Web. *The New York Times* (May

23). Available at: https://www.nytimes.com/2006/05/23/technology/23iht-web.html (Accessed: 11 November 2019).

Shearer, R., Motik, B. and Horrocks, I. (2009) “HermiT: A highly-efficient OWL reasoner.” In *CEUR Workshop Proceedings*. 2009.

Sherman, G. (2013) *QGIS Plugin Builder — QGIS Plugin Builder 2.0 Documentation*. Available at: https://g-sherman.github.io/Qgis-Plugin-Builder/#running-plugin-builder (Accessed: 13 October 2020).

Siegel, M. and Madnick, S.E. (1991) Context interchange: Sharing the meaning of data. *ACM SIGMOD Record*, 20 (4): 77–78.

Singhal, A. (2012) Introducing the Knowledge Graph: things, not strings. *Official Google Blog*.

Sirin, E. and Parsia, B. (2004) “Pellet: An OWL DL reasoner.” In *CEUR Workshop Proceedings*. 2004.

Smith, B. and Kumar, A. (2004) Controlled vocabularies in bioinformatics: A case study in the gene ontology. *Drug Discovery Today: BIOSILICO*, 2 (6): 246–252. doi:10.1016/S1741-8364(04)02424-2.

Spanos, D.E., Stavrou, P. and Mitrou, N. (2012) Bringing relational databases into the semantic web: A survey. *Semantic Web*, 3 (2): 169–209. doi:10.3233/SW-2011-0055.

Speiser, S. and Harth, A. (2010) “Taking the LIDS off data silos.” In *ACM International Conference Proceeding Series*. 2010. doi:10.1145/1839707.1839761.

Stanford Center for Biomedical Informatics Research (BMIR) (2017) *Protégé*. Available at: https://protege.stanford.edu/products.php (Accessed: 16 November 2017).

Stardog Union (2017) *Stardog—The Knowledge Graph Platform for the Enterprise*. Available at: http://www.stardog.com/ (Accessed: 20 November 2017).

Stardog Union (2020) *pystardog · PyPI*. Available at: https://pypi.org/project/pystardog/ (Accessed: 13 October 2020).

Studer, R., Benjamins, V.R. and Fensel, D. (1998) Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 25 (1–2): 161–197. doi:10.1016/S0169-023X(97)00056-6.

Taber, K.S. (2018) The use of Cronbach’s alpha when developing and reporting research instruments in science education. *Research in Science Education*, 48: 1273–1296. doi:10.1007/s11165-016-9602-2.

Taghva, K., Borsack, J., Coombs, J., et al. (2003) “Ontology-based classification of email.” In *Proceedings ITCC 2003, International Conference on Information Technology: Computers and Communications*. 2003.

doi:10.1109/ITCC.2003.1197525.

Taherdoost, H. (2018) Validity and reliability of the research instrument; how to test the validation of a questionnaire/survey in a research. *SSRN Electronic Journal*, 5 (3): 28–36. doi:10.2139/ssrn.3205040.

Tavakol, M. and Dennick, R. (2011) Making sense of Cronbach's alpha. *International Journal of Medical Education*, 2: 53–55. doi:10.5116/ijme.4dfb.8dfd.

Taylor, C. (2018) *Structured vs. Unstructured Data*. Available at: https://www.datamation.com/big-data/structured-vs-unstructured-data.html (Accessed: 10 June 2020).

Taylor, R.N., Belz, F.C., Clarke, L.A., et al. (1989) Foundations for the Arcadia environment architecture. *ACM SIGPLAN Notices*, 24 (2). doi:10.1145/64140.65004.

Technische Universität Dresden, University of Birmingham, Network Rail, et al. (2016) *Data Notation and Modeling - Public Deliverable D 3.4.1*. Available at: http://www.capacity4rail.eu/IMG/pdf/c4r_-_d341_-_data_notation_and_modelling_-_public.pdf.

The Gene Ontology Consortium (2001) Creating the Gene Ontology resource: Design and Implementation. *Genome Research*, 11 (8): 1425–1433. doi:10.1101/gr.180801.

The Institution of Engineering and Technology (n.d.) *ORBIS — Network Rail's Offering Rail Better Information Services*. Available at: http://www.theiet.org/sectors/information-communications/topics/connected-data/files/case-study-1.cfm?type=pdf. (Accessed: 3 December 2017).

The Technical Strategy Leadership Group (2017) *The Rail Industry's Data and Risk Strategy*. Available at: https://www.rssb.co.uk/Library/risk-analysis-and-safety-reporting/2017-03-strategy-data-and-risk.pdf (Accessed: 6 December 2017).

Thiyagu, R. and Sendhilkumar, S. (2011) Ontology based sentiment classification. *Proceedings of the 5th Indian International Conference on Artificial Intelligence, IICAI 2011*, pp. 1158–1169. Available at: http://www.scopus.com/inward/record.url?eid=2-s2.0-84872179161&partnerID=40&md5=e10f7ac039ed2d435cb53375b2b67be5 .

Thore, L. (2010) *Integrated Operations Reduce Risks and Improve Productivity in Offshore Oil Executive Briefing*. Available at: https://www.posccaesar.org/browser/pub/PCA/ MemberMeeting/201010/PresentationsDay2/ 20101021 Alan T Johnston.pdf.

Ting, S.L., Ip, W.H. and Tsang, A.H.C. (2011) Is Naïve Bayes a good classifier

for document classification? *International Journal of Software Engineering and its Applications*, 5 (3): 37–46.

Track 21 (2015) *Track 21 Key Findings*. Available at: http://track21.org.uk/wp-content/blogs.dir/sites/4/2015/08/150722-NR-The-Quadrant-22-July-2015-final.pdf.

Track to the Future (2020) *Projects. Track to the Future*. Available at: https://t2f.org.uk/projects/ (Accessed: 29 July 2020).

Tsarkov, D. and Horrocks, I. (2006) "FaCT++ description logic reasoner: System description." In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2006.

TSLG (2012) *The Rail Technical Strategy 2012*. Available at: https://www.rssb.co.uk/Library/Future Railway/innovation-in-rail-rail-technical-strategy-2012.pdf (Accessed: 16 November 2017).

Tutcher, J. (2015a) *Development of Semantic Data Models to Support Data Interoperability in the Rail Industry*. PhD Thesis, University of Birmingham, Birmingham, UK. Available at: http://etheses.bham.ac.uk/6774/1/Tutcher16PhD.pdf (Accessed: 15 April 2020).

Tutcher, J. (2015b) "Ontology-driven data integration for railway asset

monitoring applications." In *Proceedings - 2014 IEEE International Conference on Big Data, IEEE Big Data 2014*. 2015. pp. 85–95. doi:10.1109/BigData.2014.7004436.

Tutcher, J., Easton, J., Roberts, C., et al. (2014) *Ontology-Based Data Management for the GB Rail Industry: Feasibility Study (T952 Report)*. Available at: https://spark-uat.rssb.co.uk/Lists/Records/DispForm.aspx?ID=19661.

Tutcher, J., Easton, J.M. and Roberts, C. (2017) Enabling data integration in the rail industry using RDF and OWL: The RaCoOn ontology. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, 3 (2): F4015001. doi:10.1061/AJRUA6.0000859.

Udrea, O., Getoor, L. and Miller, R.J. (2007) "Leveraging data and structure in ontology integration." In *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data - SIGMOD '07*. 2007. doi:10.1145/1247480.1247531.

Umiliacchi, P. and Henning, U. (2008) "Standardized data interchange between railway systems: An integrated railway information system." In *WCRR 2008*. 2008. Available at: http://www.uic.org/cdrom/2008/11_wcrr2008/pdf/PN.1.13.pdf.

Umiliacchi, P., Lane, D. and Romano, F. (2011) "Predictive maintenance of railway subsystems using an ontology based modelling approach." In

Proceedings of 9th World Conference on Railway Research. 2011. pp. 1–10.

Umiliacchi, P., Van Den Abeele, D., Dings, P., et al. (2009) “Turning railways into an intelligent transportation system by better integration, management and exchange of information: The European integrated project integrail delivered high impact results for railways.” In *16th ITS World Congress*. 2009. pp. 1–8.

Uschold, M., King, M., Moralee, S., et al. (1998) The enterprise ontology. *Knowledge Engineering Review*, 13 (1): 31–89. doi:10.1017/S0269888998001088.

Verdonck, M., Gailly, F. and Poels, G. (2014) “3D vs. 4D ontologies in enterprise modeling.” In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2014. doi:10.1007/978-3-319-12256-4_2.

Verhelst, F. (2012) *Integrated Operations in the High North Joint Industry Project Final Report*. Available at: https://www.posccaesar.org/raw-attachment/wiki/IOHN/IOHN-final-report-public-web.pdf.

Verstichel, S., Ongenae, F., Loeve, L., et al. (2011) Efficient data integration in the railway domain through an ontology-based methodology. *Transportation Research Part C: Emerging Technologies*, 19 (4): 617–643. doi:10.1016/j.trc.2010.10.003.

Villani, V., Pini, F., Leali, F., et al. (2018) Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications. *Mechatronics* 55: 248–266. doi:10.1016/j.mechatronics.2018.02.009.

W3C (2014) *RDF 1.1 Semantics*. W3C Recommendation 25 February 2014. W3C.

W3C (n.d.) *Linked Data*. Available at: https://www.w3.org/standards/semanticweb/data (Accessed: 1 November 2019).

W3C OWL Working Group (2012) *OWL 2 Web Ontology Language Document Overview*. W3C.

Wang, H., Liu, S. and Chia, L.-T. (2006) "Does ontology help in image retrieval?" In *Proceedings of the 14th Annual ACM International Conference on Multimedia - MULTIMEDIA '06*. 2006. p. 109. doi:10.1145/1180639.1180672.

Wei, J. (2018) *Scalability of an Ontology-Based Data Processing System*. In *8th International Conference on Railway Engineering (ICRE 2018)*. 2018. pp. 1–5. doi:10.1049/cp.2018.0052.

Weston, P., Roberts, C., Yeo, G., et al. (2015) Perspectives on railway track geometry condition monitoring from in-service railway vehicles. *Vehicle*

System Dynamics, 53 (7): 1063–1091. doi:10.1080/00423114.2015.1034730.

White, B.T., Nilsson, R., Olofsson, U., et al. (2018) Effect of the presence of moisture at the wheel–rail interface during dew and damp conditions. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, 232 (4): 979–989. doi:10.1177/0954409717706251.

Wijewickrema, C.M. (2014) Impact of an ontology for automatic text classification. *Annals of Library and Information Studies*, 61 (4): 263–272.

Williams Rail Review (2019) *Rail in the Future Transport System*. Available at: www.gov.uk/dft (Accessed: 13 April 2020).

Wolstencroft, K., Lord, P., Tabernero, L., et al. (2006) Protein classification using ontology classification. *Bioinformatics*, 22 (14): e530–e538. doi:10.1093/bioinformatics/btl208.

World Wide Web Consortium (2012) *Semantic Web*. W3C. Available at: https://www.w3.org/standards/semanticweb/

World Wide Web Consortium (2014) *The Organization Ontology*. Available at: https://www.w3.org/TR/vocab-org/ (Accessed: 27 November 2019).

World Wide Web Foundation (2008) *History of the Web*. World Wide Web Foundation.

Xiaomeng, C., Lyu, Z. and Terpenny, J. (2015) “Ontology development and optimization for data integration and decision-making in product design and obsolescence management.” In Ebrahimipour, V. and Yacout, S. (eds.) *Ontology Modeling in Physical Asset Integrity Management*. Springer International Publishing. pp. 87–132. doi:10.1007/978-3-319-15326-1_4.

Xu, J., Wang, H. and Trimbach, H. (2016) “An OWL ontology representation for machine-learned functions using linked data.” In *Proceedings - 2016 IEEE International Congress on Big Data, BigData Congress 2016*. 2016. pp. 319–322. doi:10.1109/BigDataCongress.2016.48.

Yan, H. (2015) *Ontology Model and Semantic Web Knowledege Discovery*. Beijing: Tsinghua University Press.

Yang, Z., Yang, D., Dyer, C., et al. (2016) “Hierarchical attention networks for document classification.” In *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016 - Proceedings of the Conference*. 2016. doi:10.18653/v1/n16-1174.

Yong, A.G. and Pearce, S. (2013) A beginner’s guide to factor analysis: Focusing on exploratory factor Analysis. *Tutorials in Quantitative Methods for Psychology*, 9 (2): 79–94. doi:10.20982/tqmp.09.2.p079.

Young, M., Rempelos, G., Ntotsios, E., et al. (2020) A transferable method for estimating the economic impacts of track interventions: Application to

ground-borne noise reduction measures for whole sections of route. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, . doi:10.1177/0954409720953730.

Zhang, J., Hu, F., Wang, S., et al. (2016) Structural vulnerability and intervention of high speed railway networks. *Physica A: Statistical Mechanics and its Applications*, 462: 743–751. doi:10.1016/j.physa.2016.06.132.

Zhang, X., Hu, B., Ma, X., et al. (2014) Ontology driven decision support for the diagnosis of mild cognitive impairment. *Computer Methods and Programs in Biomedicine*, 113 (3): 781–791. doi:10.1016/j.cmpb.2013.12.023.

Zheng, A. (2015) *Evaluating Machine Learning Models*. O'Reilly Media, Inc.

Zhou, A., Yu, D. and Zhang, W. (2015) A research on intelligent fault diagnosis of wind turbines based on ontology and FMECA. *Advanced Engineering Informatics*, 29 (1): 115–125. doi:10.1016/j.aei.2014.10.001.

APPENDIX

A. SURVEY RESPONSES (CHAPTER 4)

Respondent	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Comment	Other answer for Q6
1	Student	No	N/A	No	N/A	N/A	N/A	N/A	N/A		
2	Developer	No	N/A	No	N/A	N/A	N/A	N/A	N/A		
3	Manager	No	N/A	No	N/A	N/A	N/A	N/A	N/A		
4	Student	No	N/A	No	N/A	N/A	N/A	N/A	N/A		
5	Developer	No	N/A	No	No	2,4	Yes	N/A	N/A		
6	Manager	Yes	Yes	N/A	Yes	N/A	N/A	N/A	N/A		
7	Developer	No	N/A	Yes	No	1,2,3,4,5	Yes	N/A	N/A		
8	Developer	No	N/A	No	N/A	N/A	N/A	Yes	Yes		
9	Developer	Yes	Yes	N/A	No	1,2,5	Yes	N/A	N/A		
10	Developer	Yes	No	Yes	N/A	N/A	N/A	Yes	Yes		
11	Researcher	Yes	Yes	N/A	Yes	6	Yes	N/A	N/A	The existing toolset for relational DBs is more comprehensive than that for ontologies - if you want to use a relational DB there is a range of frameworks, databases and management	Time: I'll deploy a relational DB quicker than an ontology right now

										tools to suit every possible deployment and budget. The eco-system for ontologies is at this moment not as developed, though it is improving fast	
12	Developer	Yes	No	N/A	No	3,4,5,6	Yes	N/A	N/A	It's not a term that comes up often in the rail data we work with	
13	Developer	Yes	No	Yes	No	N/A	N/A	Yes	Yes		
14	Developer	Yes	No	N/A	Yes	1,2	Yes	N/A	N/A	We can achieve the same result without using ontologies	
15	Developer	No	N/A	Yes	No	1	Yes	Yes	Yes	It's not a term I have come across before so wasn't consider-ing using it	
16	Developer	Yes	No	Yes	No	2	Yes	Yes	Yes		
17	Researcher	Yes	Yes	N/A	Yes	1,2,3,4,5,6	Yes	N/A	N/A	Although they are useful, they are obscure and difficult to model. Most of us have not received professional training and we lack tools to work with ontologies	We have professional tools and frameworks for relational DB
18	Developer	Yes	No	Yes	N/A	N/A	N/A	Yes	Yes		
19	Researcher	Yes	Yes	N/A	Yes	1,2,3,4,5,6	Yes	N/A	N/A	It is not a must in development as the same result can be achieved by relational DB. There are not many tools available, either	There is no point to use ontologies when there is no tool availa-ble
20	Researcher	Yes	Yes	N/A	No	1,4,5,6	Yes	N/A	N/A		The selection of tools is very limited. Most of time we have to de-velop tools from the very beginning
21	Developer	No	N/A	No	N/A	N/A	N/A	N/A	N/A		
22	Developer	No	N/A	No	N/A	N/A	N/A	N/A	N/A		

23	Manager	No	N/A	No	N/A	N/A	N/A	N/A	N/A		
24	Student	No	N/A	No	N/A	N/A	N/A	N/A	N/A		
25	Developer	No	N/A	No	No	2,4	Yes	N/A	N/A		
26	Student	Yes	Yes	N/A	Yes	N/A	N/A	N/A	N/A		
27	Developer	No	N/A	Yes	No	1,2,3,4,5	Yes	N/A	N/A		
28	Developer	No	N/A	No	N/A	N/A	N/A	Yes	Yes		
29	Developer	Yes	Yes	N/A	No	1,2,5	Yes	N/A	N/A		
30	Developer	Yes	No	Yes	N/A	N/A	N/A	Yes	Yes		
31	Researcher	Yes	Yes	N/A	Yes	6	Yes	N/A	N/A	Existing tools support comprehensive manipulation with relational DB, besides it is still difficult to create ontologies	Time: I'll deploy a relational DB quicker than an ontology right now
32	Developer	Yes	No	N/A	No	3,4,5,6	Yes	N/A	N/A	The term ontology is not common to see in the rail industry	
33	Developer	Yes	No	Yes	No	N/A	N/A	Yes	Yes		
34	Developer	Yes	No	N/A	Yes	1,2	Yes	N/A	N/A	A similar result can be achieved in other ways	
35	Developer	No	N/A	Yes	No	1	Yes	Yes	Yes	It is not something often heard about	
36	Developer	Yes	No	Yes	No	2	Yes	Yes	Yes		
37	Researcher	Yes	Yes	N/A	Yes	1,4,5,6	Yes	N/A	N/A	Lack of training and supportive tools put me off	Tools and frameworks work well at the moment with relational DB and other XML-based model

38	Developer	Yes	No	Yes	N/A	N/A	N/A	Yes	Yes		
39	Researcher	Yes	Yes	N/A	Yes	1,2,3,4,5,6	Yes	N/A	N/A	The same result can be achieved in other ways	Deploying something with relational DB is quicker
40	Researcher	Yes	Yes	N/A	No	1,2,3,4,5,6	Yes	N/A	N/A		Tools are very limited

Question number	***Question***
1	What is your role (e.g., developer, manager)?
2	Have you heard of ontologies?
3	If so, have you or your team members used them?
4	If not, would you be interested in learning about ontologies and the benefits thereof?
5	Do you or your team members have any plan to use ontologies in the future?
6	What factors have put you off continuing to use ontologies?
7	If there were tools available that allow people who do not know much about ontologies to work with them, would you continue using ontologies?
8	Would you be interested in using ontologies in the future after knowing their benefits?
9	Would you be interested in using tools that allow people who are not familiar with ontologies to use them?

Q6 choices	Choice statement
1	We can achieve the same result without using ontologies
2	We have to learn more to use ontologies
3	It is difficult to learn
4	We lack professionals who can use ontologies well
5	Ontology applications require industry-wide collaboration
6	Other

B. UAT Survey Response

Respondent	Q1	Q2	Q3	Q4	Q5	Q6	Q7
1	1	5	5	5	5	5	5
2	2	5	4	4	5	4	4
3	3	5	5	5	5	5	5
4	4	2	1	1	2	4	5
5	1	4	4	5	5	4	4
6	2	1	5	5	4	4	5
7	1	4	4	4	4	4	4
8	2	5	4	4	5	4	5
9	1	4	4	4	5	5	5
10	1	5	5	5	5	5	5
11	1	3	3	4	4	5	4
12	1	4	4	4	5	5	5
13	1	5	5	5	5	5	5
14	3	5	5	5	5	5	5
15	1	4	4	5	5	5	5
16	2	4	5	5	5	5	5
17	2	5	5	4	5	5	5

Question	Likert scale	Corresponding selection
1	1	No experience
	2	Know the basis of ontology, but rarely use it
	3	Know the ontology well, but rarely use it
	4	Know the ontology well, and often use it
2–7	1	Strongly disagree
	2	Disagree
	3	Neither agree nor disagree
	4	Agree
	5	Strongly agree