LINKED DATA AND SEMANTIC WEB TECHNOLOGY FOR INTERCONNECTED MAINTENANCE INFORMATION

MASTER THESIS

MARC GROOTERS

Construction Management & Engineering
University of Twente

LINKED DATA AND SEMANTIC WEB TECHNOLOGY FOR INTERCONNECTED MAINTENANCE INFORMATION

DEVELOPING AND VALIDATING A MAINTENANCE ONTOLOGY FOR EFFICIENT INTERCONNECTION OF MAINTENANCE INFORMATION

Author

Name	Marc Grooters
Student numbers	s1758748
E-mail	m.grooters@student.utwente.nl
Date	08-03-2020
Date of final presentation	10-03-2020

University

Institute	University of Twente
Faculty	Faculty of Engineering Technology
Master Track	Construction Management & Engineering

Graduation committee

Chairman	Prof. dr. ir. A. Hartmann
Supervisor	Dr. ir. F. Vahdatikhaki
Supervisor	Dr. ir. L. L. Olde Scholtenhuis

External advisors: Ir. W. M. Lubbers (Semmtech, Semantic Web Consultancy)
Dr. ir. H. M. Böhms (TNO, Senior Data Scientist)

UNIVERSITY OF TWENTE.

Summary

The rational for the research, presented in this report, is the identified inefficiency that non-interoperable information sources cause with respect to collecting information from multiple sources, relevant for supporting maintenance related tasks. Currently maintenance managers need to consult and search through these information sources manually in order to collect the information that they consider to be relevant for supporting a certain maintenance related task.
In theory this inefficiency could be solved by storing all the necessary maintenance information as an integrated whole in an IFC-model. However, some maintenance related information is generated outside the domain of BIM, and therefore will take additional work (re-entering of information) to put it in an IFC-model. Therefore, instead of trying to put maintenance related information in IFC-models, this research proposes to solve the interoperability issue with the use of Semantic Web and Linked Data (SWLD) Technology. The core utility of SWLD-technology that will be exploited enables storing information in different sources, while at the same time allowing the information in these sources to be queried as an interconnected whole. This means that cross-source relations become identifiable with the use of automated queries. And thus maintenance managers do not have to search through these sources manually anymore.

In order to enable interconnecting and querying of distributed information efficiently, with the use of SWLD-technology, a main requirement is that the information in the sources is structured properly in a standard manner. In SWLD-technology, this is realized with the use of an ontology. An ontology describes how information should be structured by defining the concepts of interest, the attributes of those concepts and the relations between concepts in an explicit manner. Currently no ontology exists that is capable of expressing maintenance related information efficiently. As a result the hypothetical usefulness of SWLD-technology has never been proven.
Therefore, this thesis aims to proof that SWLD-technology can enable automated querying of distributed, building-related maintenance information, by developing and validating an ontology on the basis of which distributed maintenance related information can be interconnect and queried as an integrated whole.
With respect to the limited time frame of six-months the validation process shall only take place with respect to one specific maintenance related task. This task was formulated in cooperation with organizations that experience the above described inefficiency caused by non-interoperable information in the maintenance phase. These organizations are the maintenance departments of TU-Delft (campus maintenance) and Eigen Haard (housing cooperation). The task that has been formulated is; to support the validation of the actual need for maintenance, planned on the basis of theoretical deterioration rates (formulated upfront of the commissioning of a new building), by interconnecting an querying all the information, residing in distributed sources, that the experts consider to be relevant with respect to this task.

Based on the information that the experts considered to be relevant for the described maintenance related task an ontology was developed. The validation of the designed ontology was conducted on the basis of how well the relevant information could be retrieved with the use of SPARQL-queries.
Based on the results of the validation experiments the research proofed that the designed ontology is capable of interconnecting and querying information residing in distributed sources, with respect to supporting the specific maintenance related task that was at the center of this research. Fundamental reasons for this success are inherent to the use of SWLD-technology. Being;

- The fact that, with SWLD-technology, resources (e.g. building elements or deficiencies) are represented by an Uniform Resource Identifier (URI) enables an information management architecture where multiple information sources assign information to the same resource by referring to the URI that represents that resource.

- When information is made available in the form of RDF-data, the SPARQL query language (as part of SWLD-technology stack) can be used for querying information from multiple sources, provided that necessary information is properly stored and links are properly defined.

These two fundamentals imply that the strategy used to interconnect and query information in this research, with respect to the specific use case, is scalable and can be used to interconnect and query information, relevant for supporting maintenance related tasks, in general. This makes an information management system based on SWLD-technology a promising value proposition.
However, as a consequence of the limited time period, the validation scope of the research was limited to one specific task. In order to improve the validity of the ontology as a tool and SWLD as underlying technology for interconnecting and querying information, relevant for maintenance related tasks, in general. Future research is needed to validate the functionality of the ontology for more maintenance related tasks.

Table of Contents

Summary 2
Table of Contents 4
List of figures 7
List of tables 8

PART-I INTRODUCTION AND METHODOLOGY 9
1 Introduction....9
1.1 Problem description 9
1.2 Research Limitations 11
1.3 Research Goal 11
1.4 Research Question.... 11
1.5 Methodology 12
1.6 Thesis outline.... 14

PART-II BACKGROUND INFORMATION.... 15
2 Building Information Modeling (BIM) 15
2.1 Building Information Modeling 15
2.2 Industry Foundation Classes (IFC) 15
3 - Semantic Web and Linked Data technology (SWLD).... 18
3.1 Resource Description Framework (RDF) 18
3.2 RDF Schema (RDFS) and Web Ontology Language (OWL).... 19
3.3 SPARQL Protocol and RDF Query Language (SPARQL) 20
4 SWLD-technology and BIM.... 21
4.1 IfcOWL ontology.... 21
4.2 Building Topology Ontology (BOT) 21
5 Related Research.... 22

PART-III INVESTIGATION.... 24
6 Literature review 24
6.1 Information categories relevant for supporting maintenance related tasks.... 24
6.2 Information structure and format.... 28
6.3 Summary.... 29
7. Interviews.... 31
7.1 Relevant information flows with respect to maintenance.... 32
7.2 Maintenance categorization.... 32
7.3 Organizational structure with respect to maintenance 33
7.4 Protocols, structures and formats for information generation and creation 33

7.5 Information use ... 35
7.6 Summary ... 36
7.7 Comparison ... 37
8 Case Description ... 38
8.1 Task description ... 38
8.2 Case description ... 39
8.2.1 Case 1; TU-Delft ... 39
8.2.2 Case 2; Eigen Haard ... 42
PART-IV DEVELOPMENT AND VALIDATION ... 45
9 Ontology Engineering ... 45
9.1 General design principles ... 45
9.1.1 Domain and scope ... 45
9.1.2 Reusing existing ontologies ... 46
9.1.3 General Modeling Techniques/Mechanisms ... 47
9.2 Classes, Properties and Relationships ... 50
9.2.1 maint:Deficiency ... 50
9.2.2 maint:DeficiencyDetectionReport ... 51
9.2.3 maint:MaintenanceTask ... 52
9.2.4 maint:BuildingProject ... 53
9.2.5 maint:Warranty ... 54
9.2.6 maint:DesiredConditionValue ... 54
9.2.7 maint:StrategicInformation ... 55
10 Linking ... 56
11 Ontology Validation ... 58
11.1 Experiment setup ... 58
11.1.1 Converting the sources into Linked Data ... 58
11.1.2 Executing the queries ... 59
11.2 Experiment results ... 60
11.2.2 Experiment results; Case 2; Eigen Haard ... 61
PART-V CONCLUSION ... 63
12 Conclusion ... 63
13 Discussion ... 64
14 Future work ... 65
References ... 66
Appendix ... 69
Appendix-A Interview Guide ... 69

Appendix-B Maintenance Ontology 70
Appendix-C Python script 78
Appendix-D SPARQL-Queries 83
D.1 Query 1 83
D.2 Query 2 84
D.3 Query 3 85
D.4 Query 4 86
D.5 Query 5 87
D.6 Query 6 88
D.7 Query 7 89
D.8 Query 8 90
D.9 Query 9 91
D.10 Query 10 92
D.11 Query 11 93
D.12 Query 12 95
D.13 Query 13 96
D.14 Query 14 97
D.15 Query 15 98

List of figures

Fig. 1 Ontology development method steps, describe by Noy, and McGuinness (2001) 12
Fig. 2 Engineering cycle steps, described by Wieringa (2014) 12
Fig. 3 Ontology Engineering Method integrated in the Design Cycle 12
Fig. 4 research design in relation to the research questions 14
Fig. 5 research design in relation to the ontology engineering method describe by Noy, and McGuinness (2001) and Design Cycle described by Wieringa (2014) 14
Fig. 6 IFC data schema architecture with conceptual layers 16
Fig. 7 Triple statement (Pauwels & Terkay, 2016) 18
Fig. 8 Example of a RDF data set 18
Fig. 9 RDF data set represented as a graph 18
Fig. 10 distributed RDF data 19
Fig. 11 Class instantiation 19
Fig. 12 Ontology and instance data connected as a graph 19
Fig. 13 Subclass declaration 20
Fig. 14 Mapping visualization from query to data set 20
Fig. 15 Data structure of nongeometric data requirements (Becerik-Gerber, et al. 2012) 24
Fig. 16 Relations between datatypes (Wang, et al. 2013) 26
Fig. 17 Information flows 33
Fig. 18 EWI-building; campus TU-Delft 41
Fig. 19 TNW-building; campus TU-Delft 41
Fig. 20 IFC-model of the Rivierenhuis building 44
Fig. 21 The Rivierenhuis building 44
Fig. 22 Ontology development method steps, describe by Noy, and McGuinness (2001) 45
Fig. 23 Maintenance Ontology 47
Fig. 24 'owl:Class' instantiation 48
Fig. 25 Declaring sub-classes with the 'rdfs:subClassOf' concept 48
Fig. 26 Objectified property modeling example 48
Fig. 27 Direct property modeling example 49
Fig. 28 typed literal as object value (left); plain literal as object value (right) 49
Fig. 29 Visualization of an 'maint:Deficiency' class instance 50
Fig. 30 Visualization of how metadata is indirectly assigned to 'maint:Deficiency' instances 51
Fig. 31 'maint:DeficiencyDetectionReport' visualization 51
Fig. 32 Instance examples of the 'maint:DeficiencyNotificationReport' and 'maint:ConditionMeasurementReport' instance 51
Fig. 33 Sub class visualization of the 'maint:MaintenanceTask' class 52
Fig. 34 'maint:MaintenanceTask' class instance example 53
Fig. 35 'maint:BuildingProject' class instance example 53
Fig. 36 'maint:Warranty' class instance example 54
Fig. 37 'maint:DesiredConditionValue' class instance example 54
Fig. 38 'maint:StrategicInformation' class instance example 55
Fig. 39 Maintenance Ontology with conceptual link set demarcated in purple 56
Fig. 40 'bs:isConditionOf' link between a 'maint:Deficiency' and 'BuildingElement' instance 57

List of tables

Table 1 SPARQL query example........ 20
Table 2 Requirements for building maintenance and scheduling according to Mayo & Issa (2015).... 26
Table 3 Information requirements for maintenance personnel according to Cavka, Staub-French & Poirier (2017)........ 27
Table 4 Information requirements according to Sadeghi, Mehany & Strong (2018)........ 27
Table 5 Taxonomy described by Farghaly, et al. (2018)........ 28
Table 6 Information categories described by 3 or more identified researches........ 30
Table 7 List of interviewed experts........ 31
Table 8 Specific case description per organization........ 38
Table 9 Information sources; case TU-Delft........ 40
Table 10 Information points; case TU-Delft........ 41
Table 11 Information sources; case Eigen Haard........ 42
Table 12 Information points; case Eigen Haard........ 43
Table 13 OWL restriction example........ 49
Table 14 Query 5 with query results........ 59
Table 15 Experiment results of Case 1 described by the maintenance department of TU-Delft......... 60
Table 16 Experiment results of Case 1 described by the maintenance department of Eigen Haard ... 61

PART-I INTRODUCTION AND METHODOLOGY

This part of the research introduces the reader to the research by elaborating the problem that is the focus of this research (sub-chapter 1.1), Research limitations (sub-chapter 1.2), the goal of the research (sub-chapter 1.3) and the research questions that need to be answered in order to achieve the research goal (sub-chapter1.4). Subsequently, in this part, the methodology (sub-chapter 1.5) and thesis outline (sub-chapter 1.6) are described.

1 Introduction

1.1 Problem description

Since the development of Building Information Modeling (BIM), easy accessibility and interoperability[1] of information have been outlined as major benefits of BIM. The usage of BIM-models, for easy interoperability of building information, from different parties during the design and engineering phase, has proven to be beneficial in comparison to incompatible[2] stored building information. A main benefit of interoperable information storage is that tasks which require information of multiple sources can be (semi) automated. For example, clash detection; before the introduction of BIM the identification of clashes between information of different stakeholders had to be done manually by construction engineers (Ashar, 2011).

In literature, researchers have suggested that using BIM-models, developed in the design and engineering phase, can realize similar benefits in the maintenance phase (Kim, et al., 2018; Thabet & Lucas, 2017). However, contrary to the advanced BIM-model based information storage processes, already deployed in the design and engineering phase, BIM-model based information storage is not used by maintenance managers in the maintenance phase (Patacas, Dawood, Vukovic, & Kassem, 2015; Khaja, McArthur, & Seo, 2016; Becker, et al., 2018; Thabet & Lucas, 2017). One of the main reasons for not reusing BIM-models as a means of information storage, during the maintenance phase, is the difficulty to make data in BIM-models interoperable with data in information sources, outside of the BIM-environment, that also contain relevant information for the maintenance phase. For example, warranty documentation and condition measurement reports (Bonduel, Vergauwen, Klein, Rasmussen, & Pauwels, 2018).

As a consequence, maintenance managers only use BIM-models (if at all they make use of it) in an incompatible relation to other information sources during the maintenance phase. This means that, when information from multiple information sources is needed, maintenance managers have to identify relevant cross-source information relations manually (Thabet & Lucas, 2017).

The search time that maintenance managers spend on this task could be saved by enabling automated identifications of these cross-source information relations. In theory, this can be achieved by storing all the necessary maintenance information in a BIM-model. This would make it unnecessary to make the BIM-model interoperable with information sources outside of the BIM-environment, that contain relevant information for the maintenance phase, because there would be no such information sources. However, some information necessary for maintenance is generated outside the domain of BIM, and therefore will take additional work (re-entering of information) to put it in a BIM-model. For example, Geographical related information (e.g. terrain) is stored in the Geographical Information Systems (GIS) domain (Deng, Cheng, & Anumba, 2016). Therefore, this research will focus on making multiple information sources interoperable, instead of trying to integrate all the information in one BIM-model.

[1]In this report the working definition of 'interoperable', in relation to data-sources, refers to the way in which software tools can conduct cross source computational tasks.

[2] In this report the working definition of 'incompatible', in relation to data-sources, refers to the inability for software tools to conduct cross source computational tasks.

Currently, the Industry Foundation Classes (IFC) is the established open standard for storing and exchanging BIM-models in the construction industry. The IFC-schema is described in EXPRESS and XSD. Instantiated data, based on the IFC-schema (the building models) can be represented in SPFF and XML format. In practice the functionality of IFC has been tested and proven, with respect to exchanging geometric information during the design and engineering phase (Kim, et al., 2018; Juan, 2013). However, Bonduel et al. (2018) also point out limitations of the IFC-standard inherent to the technology (EXPRESS, SPFF, XSD and XML) that IFC is based on;

- EXPRESS and XSD languages lack methods for defining formal semantics, making it difficult to apply generic reasoning and querying methods on IFC-models (also mentioned by Beetz & Krijnen (2018)).
- Developers can propose extensions for the IFC-schema, but the technology does not allow to extend the IFC-schema on-the-fly in a user-friendly way.
- Fine grained linking of building information stored in an IFC-file to related data (e.g. building product information) is not possible.

These limitations are part of the reason that make it hard to interconnect information in IFC-models with information in other sources. To allow for efficient/easy interconnecting of information in IFC-models with information in other sources, Researches have proposed the use of Semantic Web and Linked Data (SWLD) technology for providing an alternative to the EXPRESS/SPFF and XSD/XML based IFC-representations (Pauwels & Deursen, 2012; Koch, Vonthron, & Konig, 2016) (see chapter 3 for an extensive elaboration on SWLD-technology).
In 2016 the research of Pauwels and Terkay (2016) resulted in a Web Ontology Language[3] (OWL) representation of the IFC-schema, named ifcOWL[4]. The ifcOWL ontology[5] is a SWLD based representation of the IFC-schema (similar to the EXPRESS and XSD based IFC-schema representations). In addition to the ifcOWL ontology Orsakari (2016) developed the IFC-to-RDF converter. This tool converts a SPFF based IFC-model into a Resource Description Framework[6] (RDF) based IFC-model. An RDF based IFC-model is an SWLD based representation of an IFC-model (similar to the SPFF and XML based IFC-model representations).
With the developed ifcOWL-ontology and IFC-to-RDF converter it becomes possible to make use of the advantages that SWLD-technology provide to make IFC-models interoperable with other information sources (which are also made available as RDF based data). However, during the development of ifcOWL, one of the main criteria was to make the ontology backward compatible with the EXPRESS representation of the IFC-schema. Meaning that the ifcOWL-ontology mirrors the way in which connections between concepts are modeled with EXPRESS. This results in cumbersome connections in RDF based IFC-models. Because of this query writing becomes an unnecessarily difficult. Additionally, mirroring the EXPRESS based IFC-representation results in an ontology with 1313 classes and 1580 object properties, this makes finding the right concepts a complex task. These two characteristics have proven the ifcOWL ontology to be too complicated to work with, adoption in practice is close to zero (Pauwels, 2018). In response to this observation the W3C Linked Building Data Community Group[7]

[3] Within the SWLD technology stack OWL provides language constructs that allow for ontology modeling
[4] Currently ifcOWL has the status of 'candidate standard'. This means that ifcOWL is in the process of acquiring international consensus before being submitted to the standards committee for the final vote (buildingSMART, 2019).
[5] Within the SWLD domain the word ontology is interpreted a synonym for a RDF-based data model that aims at describing knowledge.
[6] RDF is the meta data-model on which all other SWLD-technologies are built on top of.
[7] This group brings together experts in the area of building information modelling (BIM) and Web of Data technologies to define existing and future use cases and requirements for linked data-based applications across the life cycle of buildings.

(W3C LBD CG) initiated the development of a set of compact and modular ontologies for the building industry to enable simplified methods for linking and querying information in multiple sources. One of these ontologies is already published, the Building Topology Ontology (BOT). The BOT ontology only covers the core concepts of a building (site, building, story, space, zone and building element). This simplified ontology makes it (1) possible to work with a limited set of core building classes, and (2) extend those as needed towards specific domain ontologies that are in the hands of business professionals or domain specific standardization bodies (Rasmussen, Pauwels, Hviid, & Karlshoj, 2017). However, the BOT ontology does not contain all the necessary concepts for describing maintenance specific information.

This creates the following problematic situation; in practice, SWLD-technologies are not being used for making maintenance information sources interoperable with each other. Although, there is an ontology (ifcOWL) available that contains concepts to do so. In practice this ontology has proven to be too large and complex in order to use efficiently.

To demonstrate the merit of SWLD-technology for interconnecting maintenance information, to enable automated cross source information querying, a standard ontology is needed that contain all the necessary expressivity, for describing maintenance specific information, while at the same time is not being too complex for effective use. In response to this need, the research presented in this report aims to design an ontology with exactly these characteristics.

1.2 Research limitations

Although the aim of this research is to develop a domain ontology that allows interconnecting and querying all the sorts of information that are considered to be relevant during the maintenance phase. To keep the size of the research project realistic with respect to the six-month time period, the ontology will only be validated on a specific task that is formulated in co-operation with external partners that experience the above described problem (see chapter 9 for the formulation of these tasks). This means that the resulting ontology should be interpreted as a proof of principle demonstration, for using SWLD-technology to interconnect and query maintenance information from multiple sources.

1.3 Research goal

The research goal of the research presented in this report is;

Enable a Semantic Web and Linked Data approach for interconnecting and querying, building-related, distributed maintenance information, by developing and validating a maintenance ontology.

1.4 Research question

To achieve the goal of this research, the following research questions have to be answered;

1) *Which information is necessary to support maintenance related tasks (formulated in co-operation with external organizations?)*
2) *How is information necessary to support maintenance related tasks (formulated in co-operation with external organizations) generated and stored?)*
    a. *Where is the information stored?*
    b. *In which format is the information generated/stored?*
    c. *In which structure is the information generated?*
3) *Which maintenance related concepts are necessary in order to enable interconnecting and querying the information in the information sources?*
4) *How should the necessary concepts and relations between these concepts be modelled in order to interconnect and query the sources?*

5) *How does the newly designed Maintenance Ontology perform, with respect to the distributed queries that it needs to facilitate?*

1.5 methodology

As described in chapter 1.1, the aim of this research is to develop an ontology that allows for efficient interconnecting and querying of maintenance information. Therefore, it is important that the research design suites ontology development. For this reason, the ontology development method, describe by Noy, and McGuinness (2001), containing seven steps (See Fig. 1), is used as starting point for the research design.

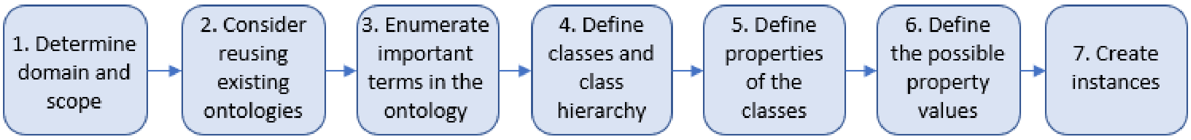

Fig. 1 Ontology development method steps, describe by Noy, and McGuinness (2001)

However, design research is not only designing, it also involves identifying a problem that justifies the initiation of a design and a strategy for validating the success of the design (Wieringa, 2014). To make sure that the research is structured properly, the Design Science Methodology described by Wieringa (2014) is used as framework for the research model. The methodology describes a Design Cycle existing out of three steps; Problem investigation, Treatment design and treatment validation.
The design cycle is a sub-set of the engineering cycle (see Fig. 2), the engineering cycle extends the research process by implementing a design in practice with the Treatment implementation step.

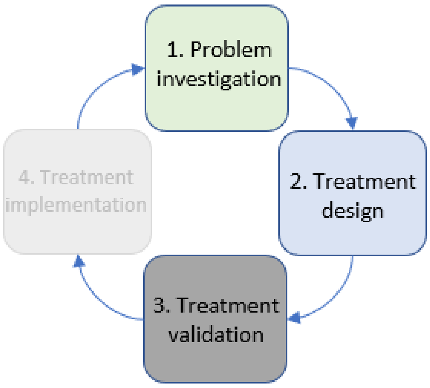

Fig. 2 Engineering cycle steps, described by Wieringa (2014)

Because the Design Science Methodology is specifically developed for research that involves the design of 'artifacts' (in the case of this research an ontology) in the information systems and software engineering domain (Wieringa, 2014), the ontology engineering method describe by Noy, and McGuinness (2001) fits perfectly within the Design Cycle. The ontology engineering method serves as process method for the design step (see Fig. 3).

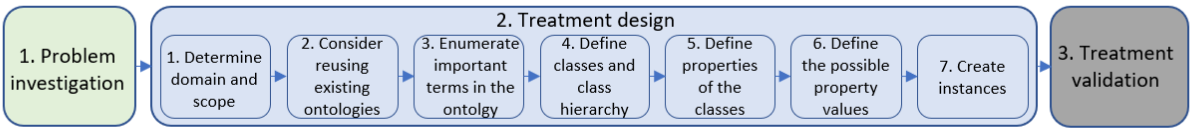

Fig. 3 Ontology Engineering Method integrated in the Design Cycle

The 'Problem investigation' step is already conducted for the research proposal that was written prior to this research, sub-chapter 1.1 is a summary of the 'Problem investigation'.

With respect to the 'Determine domain and scope' step of the ontology engineering process the 'Problem investigation' has also determined the domain of the, to be designed, ontology, being; information relevant for supporting maintenance related tasks. However, as described in sub-chapter 1.2 it is unrealistic to design and validate an ontology that covers all the information that is perceived to be relevant, with respect to supporting maintenance related tasks, within a 6-month time period. Therefore, the research only focuses on validating the ontology for one specific maintenance task, formulated in cooperation with external parties that experience the problem described in sub-chapter 1.1, in practice (The specific task is formulated in chapter 8). However, although the ontology is only validated with respect to this specific task. The aim is to design the ontology in a manner that allows the ontology to be used for interconnecting information relevant for supporting maintenance related tasks in general. Therefore, the research tries to answer the first two research questions (described in sub-chapter 1.4) from two perspectives; With respect to information relevant for supporting maintenance related tasks in general, and specifically with regard to supporting the specific maintenance related task. To do so; first a literature review will be conducted to find out which findings literature can provide with respect to these research questions from a general perspective and secondly experts in the field of information management, in relation to building maintenance, were interviewed. The goal of the interviews was to identify what information, people from practice, consider to be relevant for supporting maintenance related tasks from a general perspective and explicitly for one maintenance related task (see appendix-A for the interview guide). The input generated from the answers with respect to the general perspective will be used as input for the ontology design to ad validity to the aim of making the ontology usable for interconnecting information, relevant for supporting maintenance related task in general. However as mentioned the usability of the ontology is only validated with respect to the specific maintenance task formulated by the interviewed experts (chapter 8). The last step of the research, 'treatment validation' will done by validating if the ontology is capable of serving the competency questions defined in the scoping phase (explained above). This will be done by conducting single case mechanism experiments in which will be validated to which extent it is possible to automatically derive the information that the interviewed experts consider to be relevant, for the specific maintenance task that they defined for this research. See figure Fig. 4 for a visualization of the research design as a (matrix-)model.

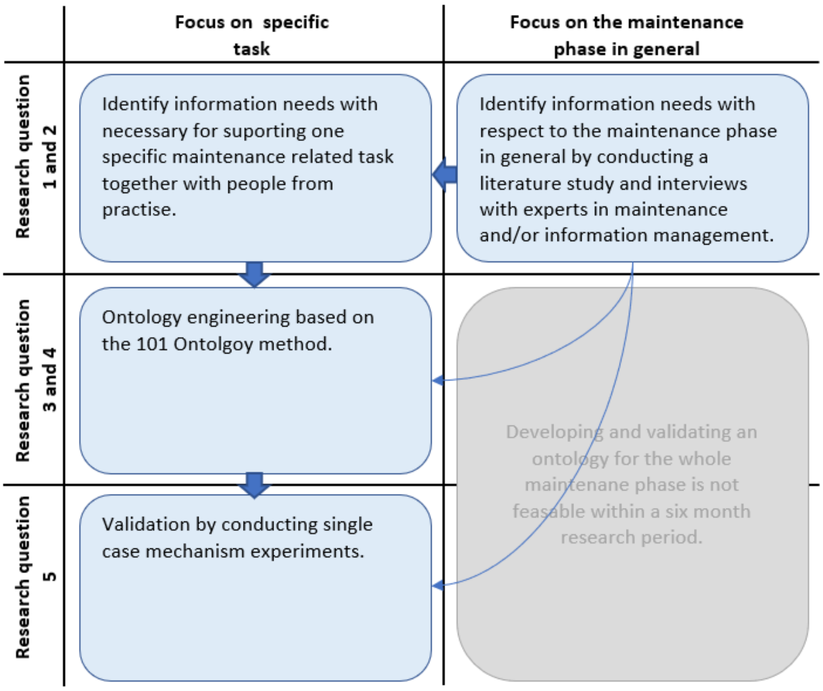

Fig. 4 research design in relation to the research questions

See Fig. 5 research design in relation to the ontology engineering method describe by Noy, and McGuinness (2001) and Design Cycle described by Wieringa (2014) for a visualization of how the research activities are related to the ontology engineering method describe by Noy, and McGuinness (2001) and Design Cycle described by Wieringa (2014).

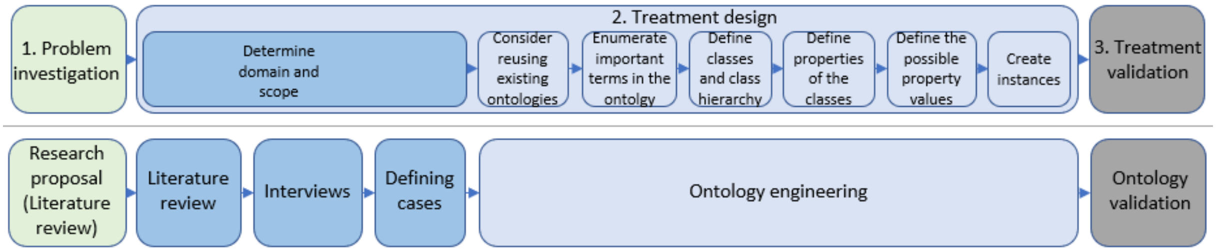

Fig. 5 research design in relation to the ontology engineering method describe by Noy, and McGuinness (2001) and Design Cycle described by Wieringa (2014)

1.6 Thesis outline

The thesis is structure in five parts, dedicated to a specific phase of the research, Being;

- PART I – Introduction and methodology
- PART II – Background information
- PART III – Investigation
- PART IV – Ontology Engineering and Validation
- PART V – Conclusion

The first part of the thesis introduces the reader to the research by elaborating the problem that the research is focusing on (sub-chapter 1.1), the goal of the research (sub-chapter 1.2) and the research questions that need to be answered in order to achieve the research goal (sub-chapter 1.3). Subsequently, in this part, the methodology (sub-chapter 1.5) and thesis outline (this sub-chapter) are presented.

The second part of the report gives an elaboration of the main themes related to the research. The goal of this part is to give the reader a deeper understanding of relevant subjects and the existing research gap that this research focuses on. The themes that will be discussed are; Building Information Modelling (chapter 2), Semantic Web and Linked Data technologies (chapter 3), the intersection of these two themes (chapter 4) and additionally, related research will be discussed (Chapter 5).

The third part of the report presents the findings of the literature study (see chapter 6) and conducted interviews (see chapter 7). As part of the interviews interviewees were also asked to define specific cases and provide input for these cases to use for the development and validation process, these cases are described in chapter 8. This part of the research report provides answers for research question one and two, described in sub-chapter 1.4.

This fourth part of the report presents the designed 'maintenance ontology' with rationale for made design decisions (see chapter 9). Additionally, it presents the validation experiments and the results derived from these experiments (see chapter 11). This part of the research report provides answers for research question three, four and five, described in sub-chapter 1.4.

The fifth part of the research presents the conclusions, discussions and future work suggestions based on the executed research, described in this report.

PART-II BACKGROUND INFORMATION

This part of the report gives an elaboration of the main themes related to the research. Being; Building Information Modeling (chapter 2), SWLD-technologies (chapter 3) and the intersection of these two themes (chapter 4). Additionally, the current state of the art in literature related to this research, in the construction industry in general, and specifically to maintenance will be presented (Chapter 5). The aim of this part is to give the reader a deeper understanding of relevant themes with respect to the research and the existing research gap that this research focuses on.

2 Building Information Modeling (BIM)

2.1 Building Information Modeling

When searching through literature, many different definitions of BIM can be found, thus one definition that conforms to everyone's view of BIM is hard to give. For the purpose of clarity and according to the interpretation of this research, this research will use the definition described in the ISO 29481-1:2016 norm which defines BIM as;

"the use of a shared digital representation of a built object (including buildings, bridges, roads, process plants, etc.) to facilitate design, construction and operation processes to form a reliable basis for decisions"

In practice these digital representations are created with the use of native modeling software, e.g. Revit or Allplan (Eastman, Teicholz, Sacks, & Liston, 2008).
A distinction can be made between an open- and closed BIM process. Closed BIM means that parties, working together for a project, all use the same native modeling software to guarantee that the produced building models can be used directly by all the parties involved. However, because most parties involved in building projects do not prefer the same native modeling tool, the native data file format (and data-model) in which they store building models is not directly usable/interactable with the different native modeling tools used by other parties.
In order to solve the above mentioned problem, all parties can agree on exchanging their building models in an open BIM format/data-model which their native software tools support (meaning they can upload and import the file format), this is called Open BIM (Juan, 2013). In this regard, according to Zhang et al. (2019), Kim, et al. (2018) and Juan (2013) the Industry Foundation Classes (IFC) can serve as an international open standard file format to ensure interoperability among different native modeling tools. The IFC schema is developed by BuildingSMART[8] as an integrated object data model to systematically and semantically manage a building's elements throughout its lifecycle.

2.2 Industry Foundation Classes (IFC)

The complete IFC schema is developed as a set of individual topic schemas. Each topic schema typically represents a consistent overall idea (e.g. structural analysis, HVAC, cost, etc.) (Wix & Liebich, 2008). The data schema architecture of IFC defines four conceptual layers which groups the individual topic schemas (Fig. 6), each individual topic schema is assigned to exactly one conceptual layer. The conceptual layers are; Resource layer, Core layer, Interoperability layer and Domain layer. The lowest layer is the resource layer, this layer contains all individual schemas containing resource definitions, these definitions do not include a globally unique identifier (GUID) and shall not be used independently of a definition declared at a higher level. The core layer contains the kernel schema, the core extensions

[8] BuildingSMART is an open and nonprofit community of industry participants with the vision to promote Open BIM within the global built environment sector. With one of the core commitments being the development and maintenance of the IFC standard (buildingSMART, 2019).

Control and Product and Process. These contain the most general entity definitions. All entities defined at the core layer, or above carry a GUID. The interoperability layer contains schemas that contain entity definitions that are specific to a general product, process resource specialization used across several disciplines. Those definitions are typically utilized for inter-domain exchange and sharing of construction information. The domain layer includes schemas containing entity definitions that are specializations of products, processes or resources specific for a certain discipline.

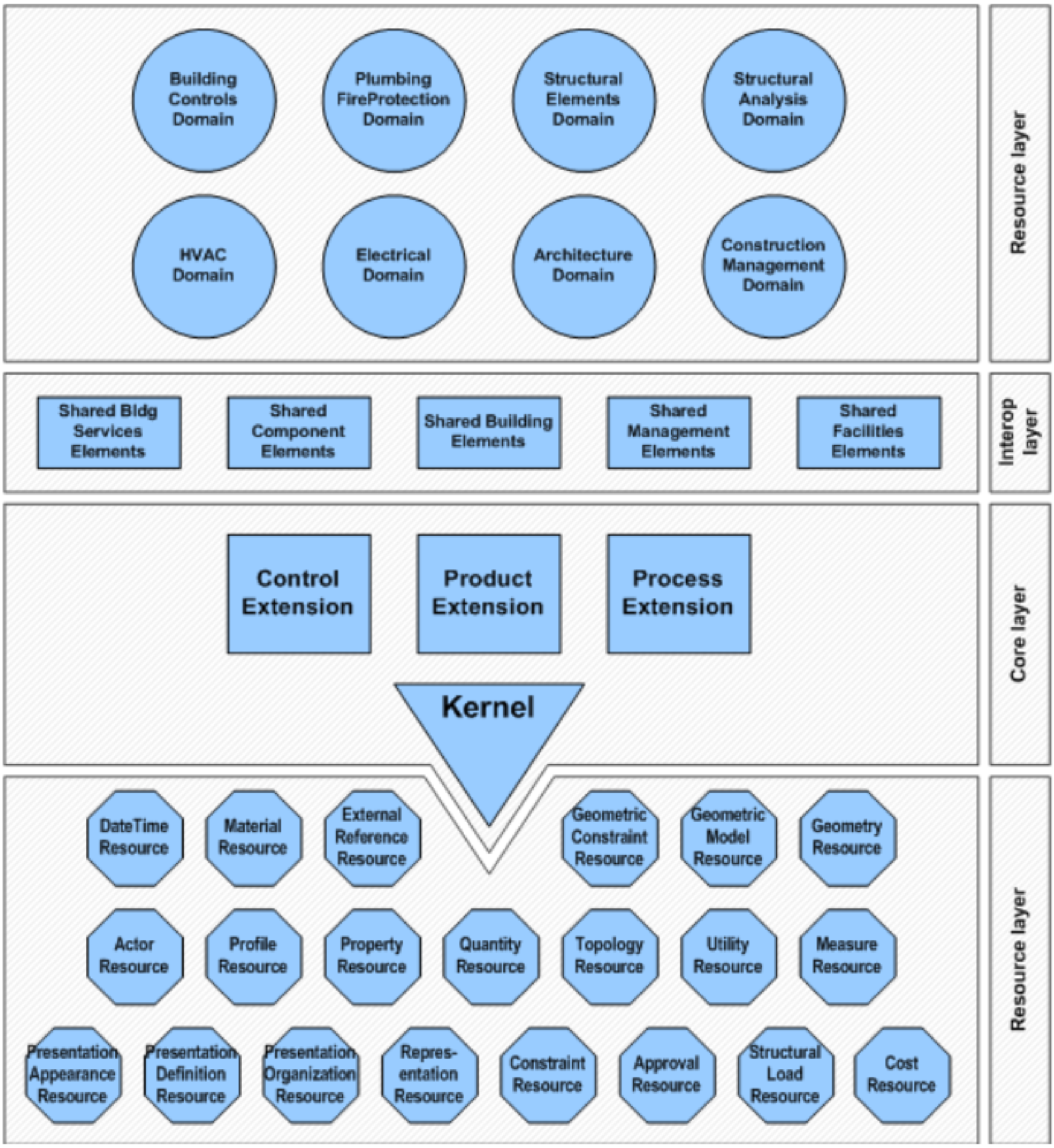

Fig. 6 IFC data schema architecture with conceptual layers

Although the IFC schema provides a standard description for building objects and their relevant properties for the design and engineering phase, the IFC schema does not provide a standard description for 'all' the information needs in the maintenance phase (Kim, et al., 2018). Information expression limitations of the IFC schema are also identified for the integration of data[9] collected by sensors in buildings (Theiler, Dragos, & Smarsly, 2018). To solve this problem Theiler, Dragos and Smarsly (2018) have proposed an extension of the IFC schema in order to reside sensor information within the IFC schema as well. Extending the IFC schema could theoretically be a solution for information that is specifically needed in the maintenance phase as well. However, this potential solution has some fundamental disadvantages that comes with it, Namely:

- some information necessary for maintenance is generated outside the scope of the BIM domain. For example, most of the Geographical related information is created and stored in the Geographical Information Systems (GIS) domain (Deng, Cheng, & Anumba, 2016).
- Additionally, it does not solve the problem of inefficient non-geometric based information management, identified by Mayo & Issa (2016) as one of the main reasons for the lack of BIM-model usage in the maintenance phase (Pauwels & Deursen, 2012).

In order to avoid these problems, Semantic Web and Linked Data technologies have been proposed in order to integrate IFC-models with the other sources in the maintenance phase (Pauwels & Deursen, 2012; Koch, Vonthron, & Konig, 2016). With the use of Semantic Web and Linked Data technology it becomes possible for machines (computers) to automatically identify connections about one single 'resource' (e.g. a space) from different information sources. This makes it possible to retrieve information from the different sources simultaneously by conducting federated queries. As opposed to integrating all the maintenance information in an IFC-model, this solution does not create the need for re-entering information and the information can still be managed in the original source. Therefore, this research will focus on using Semantic Web and Linked data technologies in order to solve the problems, explained in chapter 1.1. In order to get a better understanding of Semantic Web and Linked Data technology, a general explanation will be given in chapter 3.

[9] In the context of this research proposal the word 'data' is used to describe information that is explicitly available in a computer readable format.

3 Semantic Web and Linked Data technology (SWLD)

The web that we are accustomed to is made up of documents that are linked to one another. Any link between a document (web page) and the things it describes can only be identified by the person who reads the document. With SWLD-technologies these links become computer readable and interpretable, and thus querying for such links can be automated (Berners-Lee, Hendler, & Lassila, 2001). To realize this, a set of standardized technologies developed and maintained by the World Wide Web Consortium (W3C) can be used, the most fundamental technologies will be explained in this chapter

3.1 Resource Description Framework (RDF)

RDF refers to the framework according to which data needs to be formulated in order to enable it to become interconnected and computer readable. RDF serves as the foundation on which all other Sematic web and linked data specifications are built on.

In the semantic web ‘things’ are referred to as *resources* a resource can be anything that someone might want to talk about. These resources have to be identifiable with a so-called Uniform Resource Identifier (URI) this URI serves as the Globally Unique Identifier (GUI) for a resource. With URIs for resources it becomes computer readable when different information sources contain information about the same resource, because with RDF both resources refer to the same URI every time a statement is made about it.

In RDF, statements are represented in the form of so called *triples.* Triples consist of a subject, predicate and object. The subject is the *resource* that is being described in the triple. The predicate is a property (which also is a resource) that indicates what is being described about the subject. And the object as the value of that property, which can be a resource as well but does not have to be (Allemang & Hendler, 2011), see Fig. 7.

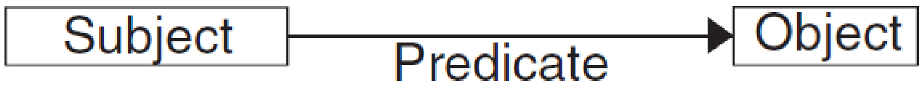

Fig. 7 Triple statement (Pauwels & Terkay, 2016)

In SWLD-Technology a data set is nothing more than a collection of triples, triples might also be related to each other. For instance the resource that is the Object in one triple might me the Subject in five other triples. For example, a wall is described as part of a building in on triple, and its properties are described in other triples, see Fig. 8.

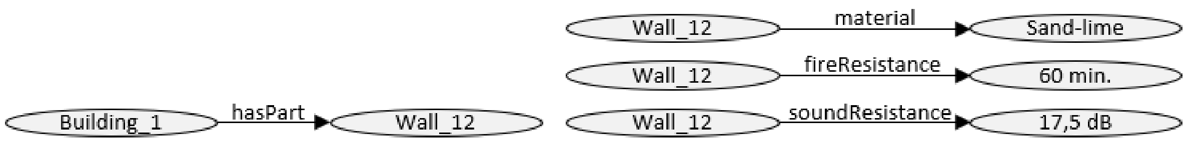

Fig. 8 Example of a RDF data set

Because, in all these triples ‘Wall_12’ is represented by the same URI, a computer is able to understand that the wall that is part of that building is of material Sand-lime, has a fire Resistance of 60 min. and a sound resistance of 17,5 dB. To better understand this, RDF data sets are usually represented and referred to as *graphs*, see Fig. 9.

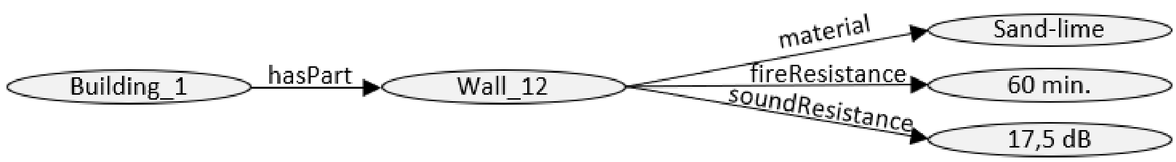

Fig. 9 RDF data set represented as a graph

Because of the uniform and unique representation of resources in RDF it does not even matter if the triples are grouped in the same data set / information source. Because with the use of another SWLD-standard, the SPARQL Protocol and RDF Query Language (SPARQL) (SPARQL will be discussed in more detail in sub-chapter 3.3), it is possible to query multiple distributed RDF data sets from one location. As long as resources that are in reality the same object are represented by the same URI. This enables the situation displayed in Fig. 10. Where all involved stakeholders in a project can described objects locally, but with the use of SPARQL the information is queryable as an integrated whole.

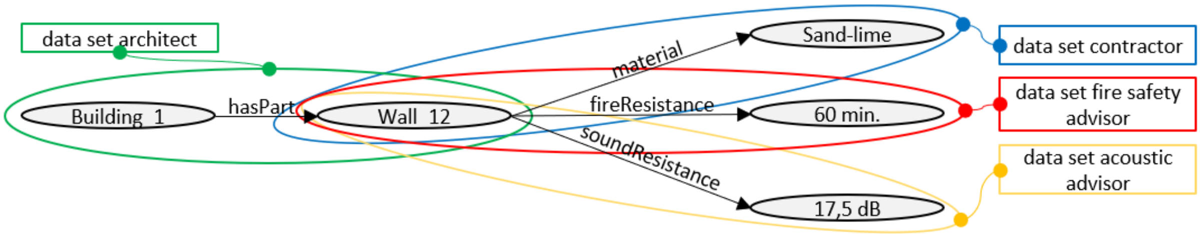

Fig. 10 distributed RDF data

3.2 RDF Schema (RDFS) and Web Ontology Language (OWL)

With the RDF-framework as the foundation that allows interconnecting data across multiple sources (see Fig. 10). RDFS and OWL are two standards that make it possible to provide more expressiveness to RDF data by creating *ontologies*. In SWLD-terms, ontologies are data schemas that provide standardized terminology for describing how data should be interpreted. For example, RDFS and OWL provide a standardized way for grouping resources in classes and subclasses. An unique feature of SWLD-technology is that ontologies (functioning as data schema's) are expressed in the same manner as instance data. Namely, in the form of triples. For example, the way in which a concept is declared as a class, is in the form of a triple, see Fig. 11.

Fig. 11 Class instantiation

Because instance data as well as ontology data are both described in triples means that they can become part of the same (distributed) graph, see Fig. 12.

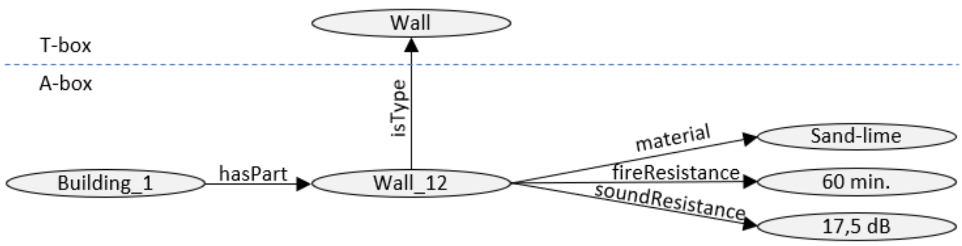

Fig. 12 Ontology and instance data connected as a graph

By adding more expressiveness to data with the use of an ontology, the ways in which information can be queried automatically[10] increases (Pauwels & Terkay, 2016). For instance in Fig. 12 it becomes possible to query for all resources that belong to the class 'Wall' with 'material' Sand-lime, 'Wall_12' will be the result of this query. Without describing instance data with the use of ontologies this is not possible. This might seem counter intuitive for instance in the example given above because the 'Wall_12' resource has Wall in its name. However, here it is important to distinguish the difference

[10] The term 'automatically' is used to indicate that the task of searching all the necessary information for conducting a maintenance task in multiple sources is done without the need of manual labor.

between human interpretation and computer interpretation. The fact that a human perceives that a resource represents a wall based on the meaning that it relates to the name, does not mean that a computer can do the same. In order for a computer to identify a resource as a wall with SWLD-technology, in needs to be described somewhere in a data schema. And in SWLD terms an ontology is the most expressive form of a data schema.
The use of the RDFS and OWL standard also enable the possibility for inference engines to infer data that is not explicitly stated in a data set (Pauwels & Terkay, 2016). For example, when an ontology describes that the Wall concept class is a sub-class of the BuildingElement class (see Fig. 13) it becomes possible to infer that the 'Wall_12' resource displayed in Fig. 12 is also a BuildingElement without explicitly declaring that the resource is of type 'BuildingElement' in the same way that the resource is declared to be of the type 'Wall'. The RDFS and OWL standards provide numerous of concepts accompanied with inference rules that apply to them, that can be used for developing ontologies. The goal of this report is not to describe all of these constructs, for more information on these standards the W3C website can be consulted.

Fig. 13 Subclass declaration

3.3 SPARQL Protocol and RDF Query Language (SPARQL)

Where the RDF-framework provides a way to represent distributed data, and RDFS and OWL provide standardized vocabulary for building ontologies that ad computer interpretable meaning to that data SPARQL is used to access the data. SPARQL is a query language developed for RDF based data. With SPARQL it is possible to query data from multiple sources (on the web).
The way in which SPARQL fundamentally works is by formulating triple patterns and describing the variables of interest in those triple patterns. A SPARQL-reasoner will filter all the triples in a data set (or multiple distributed data sets) that match the described triple pattern and will display the values in the data set that represent the variable of interest. For example, see the query in Table 1. When executing this query to the data set displayed in Fig. 12 it will search for all the variables that match the triple description formulated in the query; being the variable must be a part of 'Building_1' and be an instance of the 'Wall' class concept. In this example the resource Wall_12 matches this description and thus will be displayed as the result of the query.

Table 1 SPARQL query example

SELECT ?x WHERE{ Building_1 hasPart ?x . ?x isType Wall .}
Result: Wall_12

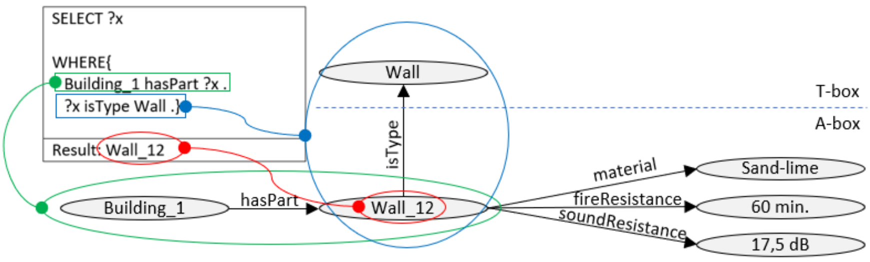

Fig. 14 Mapping visualization from query to data set

4 SWLD-technology and BIM

While SWLD-technology has already proven its practical usefulness in multiple disciplines such as library and information science, industry, biology and human science (Feigenbaum, Herman, Hongsermeier, Neumann, & Stephens, 2010). Just recently, SWLD-technology has attained the interest of academics in the BIM domain. This interest has resulted in a number of concrete developments that enable the exposure of information in BIM-models to the SWLD-principles described in chapter 3. These developments will be discussed in this chapter.

4.1 IfcOWL ontology

ifcOWL provides an OWL representation of the IFC schema, this representation of the IFC schema is being maintained by BuildingSMART. With the ifcOWL ontology it is possible to represent building data using Semantic Web and Linked Data technologies. Meaning that IFC data becomes available in directed labelled graphs (RDF). This graph model and the underlying web technology stack allows building data to be easily linked to material data, GIS data, product manufacturer data, sensor data, classification schemas, social data, and so forth. The result is a web of linked building data that brings major opportunities for data management and exchange in the construction industry and beyond. During the development of ifcOWL the main criterium was backward compatible with the main IFC EXPRESS schema. This however, resulted in a very complex ontology (Pauwels, 2018).

4.2 Building Topology Ontology (BOT)

In order to efficiently link multiple building related data sources together, the W3C Linked Building Data Community Group[11] is developing a set of minimalistic ontologies in order to efficiently connect multiple building data sources together. One of these ontologies has already been published, this is the Building Topology Ontology (BOT). BOT was developed with the acknowledgement that the ifcOWL ontology was too complex to use (containing 1313 classes and 1580 object properties). BOT only covers the core concepts of a building (site, building, story, space and building element), and three methods for extending this with domain specific ontologies. This approach makes it (1) possible to work with a limited set of core building classes, and (2) extend those as needed towards specific domain ontologies that are in hands of business professionals or domain specific standardization bodies (Rasmussen, Pauwels, Hviid, & Karlshoj, 2017).

Besides the BOT ontology the W3C Linked Building Data Community Group is also planning to publish a Building Product Ontology (PRODUCT), the Ontology for Property Management (OPM) and the Properties Ontology (PROPS). However, these ontologies are currently in the development phase.

A main design question to be answered during the research is if the BOT or ifcOWL (or concepts of both) ontology will be used in the data architecture.

[11] This group brings together experts in the area of building information modelling (BIM) and Web of Data technologies to define existing and future use cases and requirements for linked data based applications across the life cycle of buildings (W3C Linked Building Data Community Group, 2019).

5 Related Research

This chapter will summarize and elaborate previous conducted research, related to the research presented in this report.

Layer structure for a data architecture

Curry, et al. (2013) have proposed cross-domain AEC/OM data sharing and integration with the use of Semantic Web and Linked Data technology. The research proposes a fundamental layer structure for a data architecture. Consisting of a source layer which provides the information and makes it available in linked data format, a linked data layer which defines how the sources should be interconnected, a support layer which consists of queries that need to generate the necessary information for the Applications in the Application layer. However, besides defining the fundamental layer structure. The research does not provide specific domain ontology in the linked data layer on the basis of which the information in the multiple sources can be interconnected. Nor does it provide specific examples of how a domain ontology should be incorporated in the domain architecture. Corry et al. (2014) have proposed a similar layering concept as Corry et al. (2014) to access AEC/OM data across various stakeholders for the purpose of optimizing building performance. However, like the research of Corry et al. (2014) it does not provide specific examples of a domain ontology that can be used for linking multiple sources within the data architecture.

A specific linking example

Kim, et al. (2018) Propose a Semantic Web and Linked Data based method that interconnects BIM data to maintenance work records. This research provided a detailed example of how information in a BIM model can be connected with another information source with the use of Semantic Web and Linked Data technology. However, this research did not take into consideration the development of a domain ontology that can be used to interconnect all relevant information sources for the maintenance phase.

A domain ontology for the infra sector

In the Infra sector a research project has been conducted that had a similar goal in mind as this research. INTERLINK was a project that has focused on data integration, using linked data and semantic web technology in order to optimize information management for National Road Agencies (NRAs) (CEDR, 2019). The Project delivered a Basic European Object Type Library[12] (EUROTL). The EUROTL serves as the main ontology within the 'EUROTL-framework' that serves as a hub for linking concepts and data between different domain ontologies (e.g. ifcOWL ontology) (INTERLINK consortium, 2019).
However, the objects that this research focused on are infrastructure objects and only describes the interconnection of multiple domain ontologies not specific sources, such as work record. Thus, the ontology cannot be used for interconnecting all the information sources with respect to building maintenance.

In conclusion, literature does not provide a solution that enables reuse of BIM-models, in the maintenance phase. However, in literature two solution directions have been identified (1) the use of Semantic Web and linked Data technologies in order to interconnect multiple information sources (Corry, et al., 2014; Curry, et al., 2013; Lee, Chi, Wang, Wang, & Park, 2016; Kim, et al., 2018) and (2) to present an extension for the IFC schema so that it covers all the information necessary for the maintenance phase, similarly to what Theiler, Dragos and Smarsly (2018) have proposed for the sensor derived data. However, this last solution has some fundamental disadvantages that comes with it, Namely:

[12] An OTL is an ontology with standardized object-types names and properties or specifications. An object is described with its object-type data, geometry data and metadata.

- some information necessary for maintenance is generated outside the scope of the BIM domain. For example, most of the Geographical related information (e.g. Terrain) is created and stored in the Geographical Information Systems (GIS) domain (Deng, Cheng, & Anumba, 2016).
- Additionally, it does not solve the problem of inefficient non-geometric based information management, identified by Mayo & Issa (2016) as one of the main reasons for the lack of BIM-model usage in the maintenance phase (Pauwels & Deursen, 2012).

For this reason, this research will focus on the first solution direction in order to solve the problems that the current way of disconnected information storage in the maintenance phase fundamentally create (described in chapter 1.1).
While previous research has provided some general building blocks for using Semantic Web and Linked Data technology (e.g. domain ontologies like BOT and the IFC-to-LBD converter) in the AEC/OM industry. And some research has demonstrated the benefits of interconnecting multiple information sources with the use of Semantic Web and Linked Data technology for specific cases in the AEC/OM industry. No research has provided a domain ontology on the basis of which all information sources (including BIM-models), relevant for building maintenance, can be interconnected. The goal of this research is to fil in this gap in literature.

PART-III INVESTIGATION

When developing a domain ontology, usable for interconnecting and querying information from multiple information sources, it is important to have a good understanding of the structure and context of the information sources that need to be queried and linked. To get this understanding, a literature study is conducted and subsequently interviews with people from practice are conducted.
This part of the report presents the results and findings of the literature study (chapter 6) and conducted interviews (chapter 7). Interviewees were also asked to define specific cases and provide input for these cases to use for the development and validation process, see chapter 8 for the defined cases. This part of the research report contains the answers on research question one and two.

6 Literature review

The goal of the literature review was to identify useful insights that literature could provide with respect to the first two research questions. At the end of this chapter a summarization is given of the key findings that where derived from the literature study. In sub-chapter 7.7 the findings of this literature review are compared to the interview findings.

6.1 Information categories relevant for supporting maintenance related tasks

Becerik-Gerber, et al. (2012) state that BIM, holds undeveloped possibilities for providing and supporting practices in the Maintenance and Operations phase of buildings. As part of their study Becerik-Gerber, et al. (2012) research which information is necessary in order to support these practices. To identify which information is relevant for managing and conducting maintenance. Becerik-Gerber, et al. (2012) analyzed information captured and used in Facility Management (FM) practices and systems, and through expert interviews. Based on their research Becerik-Gerber, et al. (2012) categorized all the relevant non-geometric information needs in six categories, see Fig. 15.

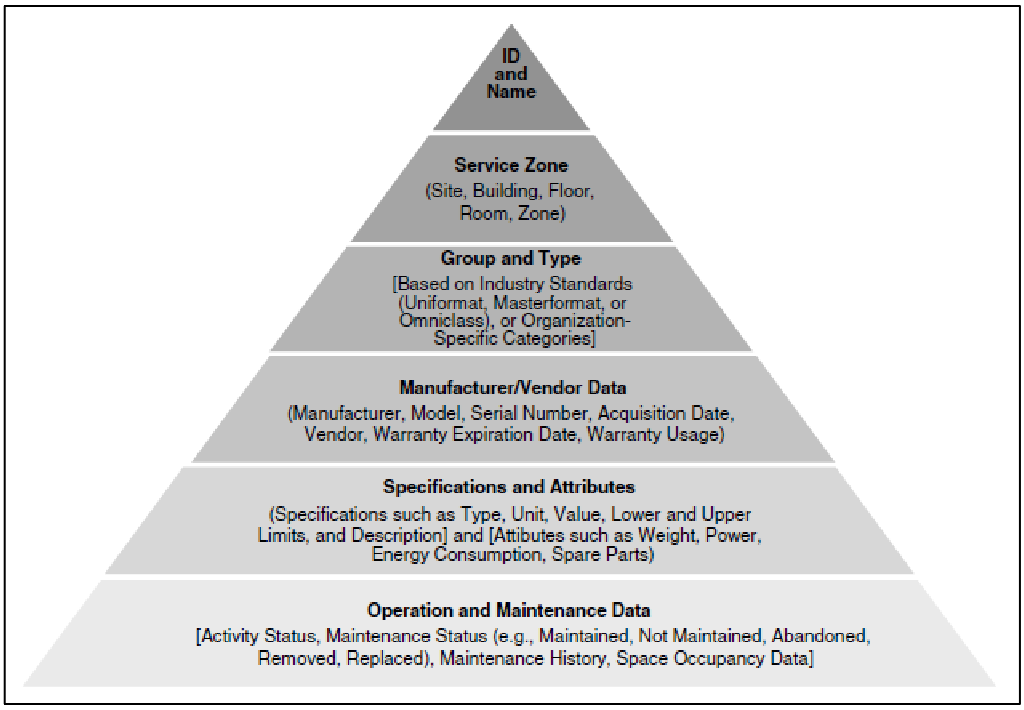

Fig. 15 Data structure of nongeometric data requirements (Becerik-Gerber, et al. 2012)

The pyramid figure illustrates the data classification on the basis of the sequence in which the data should be captured in the project life cycle. The volume of data increases substantially from top to bottom. This data should be identified and captured through a workflow at different stages of a project. In the pyramid figure Becerik-Gerber, et al. (2012) also describe explicit information points per category. However, nowhere in their study they state that this should be interpreted as an exhaustive summarization.
Although Becerik-Gerber, et al. (2012) state that; *'Naturally, up until now, most discussions have focused on the geometric information requirements for BIM-models to be implemented'*. They did not mention specific research papers which covered this topic. Nonetheless, Becerik-Gerber, et al. (2012) do state some of the geometric information requirements that came out of these discussions. However they explicitly stated; *'this are some of the geometric requirements'* and thus cannot be interpreted as an exhaustive list. The geometric information requirements Gerber, et al. (2012) state are;

- *Accurate as-built model of all building components including; architectural, structural, mechanical, electrical, plumbing, and fire protection systems and site plans including safety accesses,*
- *Accurate as-built model for main utility lines to the buildings,*
- *Accurate telecommunication representations, including proper placing and annotation of outlets,*
- *Labeled, annotated, and colored spaces according to FM guidelines, which should include standards for space type, description, space usage, and so on,*
- *Built in schedules in the model,*
- *Logical object tree organization to manage the various components within the model,*
- *Accurate clearance requirement for mechanical electrical, and plumbing (MEP) requirement to provide maintainability based on technical specifications.*

Becerik-Gerber, et al. (2012) also conclude that, from a data type perspective, the information needs for supporting building maintenance can be divided in three conceptual data types; Equipment and Systems, Data, and Documents. 'Equipment and Systems' refers to the geometrically modeled representation of assets. 'Data' refers to explicit information of those assets (e.g. vendor information, location information, description and attributes). 'Documents' refers to operation and maintenance manuals, manufacturer instructions etc.

Wang, et al. (2013) investigated the potential use of BIM in order to involve maintenance experts in the design stage, as part of this research they proposed which information should be collected.
In their research they distinguished the same conceptual types of data as Becerik-Gerber, et al. (2012). Being; Equipment and systems, data, and documents. In addition to describing the conceptual data types Wang, et al. (2013) also formulate how these conceptual data types relate to each other. They state that the 'Equipment and systems' category represents the actual individual entities of an building. And that these entities have two kinds of information that describes something about them; attributes-data (Data) and portfolios and documents (Documents), see figure Fig. 16 for a visualization of the relations that Wang, et al. (2013) describe in addition to Becerik-Gerber, et al. (2012)'s data type categorization for information in the maintenance phase. Fig. 16 also mentions some sub-categories with respect to the conceptual data types. However, Wang, et al. (2013) do not claim this to be an exhaustive categorization, it should be interpreted as examples.

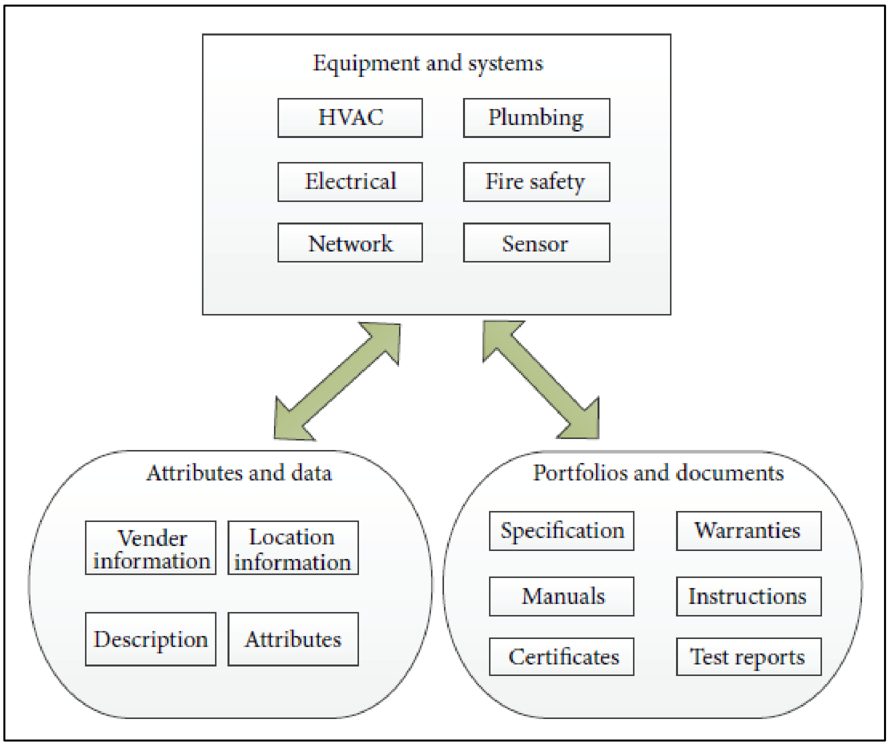

Fig. 16 Relations between datatypes (Wang, et al. 2013)

Mayo & Issa (2015) examined at a microlevel view the non-geometric information required for building owners to help them specify their closeout deliverables. Their research methodology included a Delphi panel of facility management personnel employed by universities in Eighteen states in the United States. The Delphi panel questions addressed the issue from the perspective of providing value as determined by the perceived need for product information in the operations and maintenance phase. The research resulted in a basic list of building information needs for FM, categorized by specific discipline, being; Building maintenance and scheduling, Building system analysis, Building asset management, Space management and Safety and disaster/emergency planning. With respect to the research in this report only the information requirements for building maintenance and scheduling that Mayo & Issa (2015) identified are relevant, see Table 2 for the identified data requirements.

Table 2 Requirements for building maintenance and scheduling according to Mayo & Issa (2015)

Requirements for building maintenance and scheduling according to Mayo & Issa (2015).		
- Manufacturer, model number, etc. - Maintenance instructions - Test reports - Certificates	- Manuals - Operation instructions - Installation guide - Cut (product) sheet/submittals	- Spare part information - Warranty documentation - Bar code information - Construction information (specifications, schedule, and RFIs)

Cavka, Staub-French & Poirier (2017) investigated two large owner organizations in Canada to better understand the process of developing and formulating BIM-requirements to support the lifecycle of their assets. As part of their research an array of requirements documentation were analyzed, interviews were performed with numerous facility management personnel, and BIM-models from four projects were analyzed. Based on their research, Cavka, Staub-French & Poirier (2017) proposed a formalization process for developing and formulating BIM requirements to support the lifecycle which exists out of 4 steps;

1. Identify the range of owner requirements through investigation of data sources such as organizational guidelines, technology infrastructure, personnel requirements, and owners BIM-requirements,
2. Investigate and classify the scale and scope of owner requirements in the context of project delivery and asset management,
3. Identify requirements that are computable and can be supported through BIM-based project delivery and asset management. Exemplify required model information through analysis of computable requirements,
4. Develop a conceptual framework for relating the project's digital and physical product with the owner requirements, and organizational context.

Although the focus of the research is to identify specific BIM requirements, in their framework Cavka, Staub-French & Poirier (2017) do this by first identifying all information requirements in step 2 of their formalization process. Based on the analyzed data, they categorized the requirements in five groups; Codes & design standards, Organizational requirements, Project requirements, Personnel requirements and BIM-requirements. In the context of the research presented in this report only the personnel requirements are relevant because Cavka, Staub-French & Poirier (2017) divine these as the requirements which O&M personnel need in order to perform their tasks, see Table 3.

Table 3 Information requirements for maintenance personnel according to Cavka, Staub-French & Poirier (2017)

Information requirements for maintenance personnel according to Cavka, Staub-French & Poirier (2017).		
- Design criteria - Commissioning information - Replacement part information - Vendor information - Serial number - Location	- Warranty information - Cost (to replace, maintain etc.) - System visualization - System performance information	- Location of panels and valves that control equipment - Sequence of operation (start-up-shut down information - Maintenance history

Sadeghi, Mehany & Strong (2018) researched the FM information requirements for 'Higher Educational Institutions' asset portfolios. Based on a literature review and pilot study they classified the information requirements into four main categories; Attributes of the physical facility, operation and maintenance (O&M) data, institutional factors, and decision alternative, see Table 4 for an overview of al the descriptions.

Table 4 Information requirements according to Sadeghi, Mehany & Strong (2018)

Information requirements according to Sadeghi, Mehany & Strong (2018)		
- ID/Name - Location - Specifications - Manufacturer data - Maintenance strategy - Work order history	- Maintenance status - Condition assessment - Expected life span - External factors (laws and regulations, state funds, donors)	- Internal factors (Mission criticality annual budget, business or historical criticality) - Total cost of ownership (TCO) - Time - Owner's other priorities

Farghaly & Wood (2017) proposed a taxonomy for the non-geometric data required for assets that consume energy. For their research Farghaly & Wood (2017) conducted different research methods such as literature-review, semi-structured interviews and focus groups. The research resulted in a taxonomy that contains 40 parameters. The taxonomy adopts a hierarchical structure with two levels.

The top level is classified into five main categories namely; space/location, BIM capex (capital expenditure), specifications, warranty, and asset's capex and opex (operating expenses). At the second level, the 40 parameters are presenting the required BIM data/attributes for asset management during handover stage. Later Farghaly, et al. (2018) conducted an additional study which resulted in a revision of the initial taxonomy. Now it contains six top-level classes because the 'Asset's Capex and Opex' class was divided into the 'Asset Capex' class and 'Maintenance' class. Furthermore at the top-level, the 'BIM Capex' class was renamed to the 'Classification' class. At the second level some subclasses were added which results in a total of 60 subclasses which represent the required BIM data/parameters, see Table 5 for an diagrammatic representation of the taxonomy, proposed by Farghaly, et al. (2018). Although the taxonomy was specifically developed for energy consuming building assets, all the data requirements except for some in the specifications class can be applied to all building elements in general.

Table 5 Taxonomy described by Farghaly, et al. (2018)

Classes	**Sub-classes (Attributes)**
Classification	Revit classification, ACE-IM Classification, Uniclass2, NRM3, SFG20, Revit ID, Type Name, Unique Type ID, Asset Type, Control Panel Revit ID
Space/Location	Facility type, Building name, Building number, Level name, Zone name, Department name, Room name
Asset Capex	Asset ID, Barcode ID, Control panel ID, Cost, Purchase order no., Purchase document
Specifications	Manufacturer, Supplier, Model name, Serial number, Color, Insulation class, Voltage, Phase, Power (kW), Current, Water, Gas, Heat generated, Specifications, Documentations, Code compliance, Spare Parts info
Warranty	Installation date, Installation guide, Test reports, Certificates, Certificates description, Lifecycle phase, Warranty start date, Warranty duration, Warranty description
Maintenance	Maintenance documents, Maintenance scope, Maintenance status, Maintenance history, Maintenance annual cost, Maintenance frequency, Maintenance instructions, Maintenance accessibility

6.2 Information structure and format

Mayo & Issa (2015) identify the use of the OmniClass as the object-oriented standard for classifying and structuring BIM-information. OmniClass is specifically designed for the construction industry as a means of organizing and retrieving information, Becerik-Gerber, et al. (2012) also briefly state the use of OmniClass for classifying building objects in a BIM environment.

OmniClass is designed to provide a standardized basis for classifying information throughout the full facility life cycle from conception to demolition or reuse. OmniClass consists of 15 tables, each of which represents a different facet of construction information. Each table can be used independently to classify a particular type of information, or entries on it can be combined with entries on other tables to classify more complex subjects (CIC 2013). According to Mayo & Issa (2015) table 21, 22 and 23 are especially useful as information structure for capturing facility information. Table 21 provides an formalized list of building elements and sub-categorization going four levels deep, table 22 provides an formalized list of tasks related to the design, construction and life-cycle of the building with an sub-categorization going four tiles deep, table 23 provides an list building products with an subcategorization going seven tiles deep.

Farghaly, et al. (2018) state the use of IFC (and COBie as a subset of IFC) as the most used data structure for exchanging information in BIM-models, relevant for maintenance, to the entities in the lead during the maintenance and operations phase. They state; *'IFC-models have been used as the standard file format for importing BIM models into Computer Aided Facilities Management (CAFM) platforms to overcome the lack of interoperability between existing CAFM tools and the growing number of commercially available BIM packages'* (See sub-chapter 2.2 for an extensive elaboration on IFC).

Besides IFC, COBie is noted as a standard information structure (Becerik-Gerber, et al. 2012; Mayo & Issa, 2015; Farghaly, et al. 2018; Sadeghi, Mehany & Strong, 2018). COBie is a Model View Definition (MVD) which means that it is a subset of the IFC-data model for a specific use. In the case of COBie, the use is information handover as part of the delivery phase of construction projects.

6.3 Summary

The main findings derived from the literature review are;

- The identified literature mainly describe information needs, relevant for practices in the maintenance and operations phase, in terms of general categories (e.g. warranty information), not in terms of explicit information points (e.g. warranty end-date). The only research that did provide an exhaustive list of specific information points, focused exclusively on information needs related to energy consuming elements of buildings. With respect to information needs in the maintenance phase multiple studies describe the same categories to be relevant. Table 6 lists the information categories that where mentioned by three or more researches identified in the literature review.
- Information, relevant for the maintenance phase, can be categorized in three conceptual data types/formats; Systems and Equipment, Data, and, Documents. 'Equipment and Systems' refers to the geometrically modeled representation of physical assets. 'Data' refers to explicit information of those assets (e.g. vendor information, location information, description and attributes). 'Documents' refers to operation and maintenance manuals, manufacturer instructions etc. (Becerik-Gerber, Jazizadeh, Li, & Calis, 2012). In addition to describing the conceptual data types Wang, et al. (2013) also formulate how these conceptual data types relate to each other. They state that the 'Equipment and systems' category represents the actual individual entities of an building. And that these entities have two kinds of information that describes something about them; attributes-data (Data) and portfolios and documents (Documents)
- With respect to structuring information three open-standards, where mentioned. Being; Omniclass, IFC, and COBie.
    - OmniClass has been mentioned by Mayo & Issa (2015) and Gerber, et al. (2012) As an standard for categorizing and structuring building objects. According to Mayo & Issa (2015) it is 'the' established standard in BIM, with respect to classifying objects.
    - IFC and COBie have been mentioned as open standards useful for handing over information in BIM-models in a structured manner.
- Most of the identified literature only describes information needs for the whole maintenance phase in general, not for specific processes/tasks within the maintenance place (e.g. planning maintenance or replacing building parts).

Table 6 Information categories described by 3 or more identified researches

	Categories	**Identified in literature**
1	Naming and ID	(Becerik-Gerber, Jazizadeh, Li & Calis, 2012; Farghaly, Abanda, Vidalakis, & Wood, 2018; Sadeghi, Mehany, & Strong, 2018)
2	Attribute/Product information	(Becerik-Gerber, Jazizadeh, Li & Calis, 2012; Wang, Wang, Wang, Yung, & Jun,2013; Mayo & Issa, 2015; Farghaly, Abanda, Vidalakis, & Wood, 2018; Sadeghi, Mehany, & Strong, 2018)
3	Location information	(Becerik-Gerber, Jazizadeh, Li & Calis, 2012; Wang, Wang, Wang, Yung, & Jun,2013; Mayo & Issa, 2015; Farghaly, Abanda, Vidalakis, & Wood, 2018; Sadeghi, Mehany, & Strong, 2018)
4	Geometric As-built representation / Visualization	(Becerik-Gerber, Jazizadeh, Li & Calis, 2012; Wang, Wang, Wang, Yung, & Jun,2013; Mayo & Issa, 2015; Farghaly, Abanda, Vidalakis, & Wood, 2018; Sadeghi, Mehany, & Strong, 2018)
5	Manufacturer and Vendor information	(Becerik-Gerber, Jazizadeh, Li & Calis, 2012; Wang, Wang, Wang, Yung, & Jun,2013; Mayo & Issa, 2015; Farghaly, Abanda, Vidalakis, & Wood, 2018; Sadeghi, Mehany, & Strong, 2018)
6	Warranty information	(Becerik-Gerber, Jazizadeh, Li & Calis, 2012; Wang, Wang, Wang, Yung, & Jun,2013; Farghaly, Abanda, Vidalakis, & Wood, 2018)
7	Maintenance information	(Becerik-Gerber, Jazizadeh, Li & Calis, 2012; Mayo & Issa, 2015; Farghaly, Abanda, Vidalakis, & Wood, 2018; Sadeghi, Mehany, & Strong, 2018)
8	Spare part / Replacement information	(Becerik-Gerber, Jazizadeh, Li & Calis, 2012; Mayo & Issa, 2015; Cavka, Staub-French, & Poirier, 2017)
9	Certificates	(Wang, Wang, Wang, Yung, & Jun,2013; Mayo & Issa, 2015; Farghaly, Abanda, Vidalakis, & Wood, 2018)

7 Interviews

The goal of the interviews was to identify what information, people from practice, consider to be relevant for supporting maintenance related tasks (research question 1), the form and structure of this information and how the information is used (research question 2). To do so, seven experts in the field of information management, in relation to building maintenance, were interviewed. Table 7 describes the background of the interviewed experts.
The interviews consisted of two parts. In the first part experts where asked to answer question (see Appendix-A) from a general perspective, considering relevant information with respect to maintenance related tasks in general. The goal of this part of the interview was to generate insights that could be incorporated in the design of the maintenance ontology to make sure that the ontology can be considered as a tool that can be used to structure information relevant for supporting maintenance related tasks in general. In the second part experts where asked the same questions but now with respect to information relevant for supporting a specific maintenance related task. The goal of this part was to get a list of specific information points, in order to use for validating the maintenance ontology. The findings derived of the first part of the interviews are presented in this chapter. The findings of the second part of the interviews are presented in chapter 8.

The interviews are conducted in the form of semi-structured interviews. See Appendix-A for the interview guide, used as bases for the semi-structured interviews. Because of the semi-structured nature of the interviews, insights that had no direct relation to the research questions were also identified. Although these findings have no direct relation with the goal of this research, they are also added in the report because they give a better understanding of some general obstacles that future research should address in order for asset owners to attain and use information more effectively.
At the end of this chapter a summarization is given of the findings that where derived from the interviews (sub-chapter 7.6). The findings are also compared with the findings from the literature study (sub-chapter 7.7).

Table 7 List of interviewed experts

Function-Title	Organization	Context	Additional
Process and information manager	Erasmus Medical Center (EMC), Department maintenance and Project management	Asset owner	
Process and information manager	Dura Vermeer, Dutch Contractor	Project delivery	Chairman of the BIM basis-ILS
Information and Maintenance manager	Rijksvastgoedbedrijf (RVB), responsible for maintenance and real estate development for the Dutch government	Asset owner	In the committee of the NEN-2660, Administrator BOEI-manual
Information/BIM-Manager	Eigen Haard, Dutch Housing Cooperation	Asset owner	Involved with the development of a standard ILS for maintenance
BIM-Coordinator	Besix Nederland, Dutch Contractor	Project delivery	
Project manager	Technical University (TU) Delft, Department maintenance and Project management	Asset owner	Chairman of UVIP and involved with the development of the NL-SfB
Information/BIM-Consultant	Arcadis, Engineering and Consulting	Consultancy	

7.1 Relevant information flows with respect to maintenance

In all interviews four main flows of information where mentioned to be relevant for planning maintenance. These are; handover information from the project delivery phase, the information that results from condition monitoring, the log information when maintenance is being conducted (a.k.a. maintenance history), and strategic information described by the asset management department. Fundamentally, this are the requirements and goals that the asset should meet.

With respect to handover information, two interviewees mention the categorization of three types of information; geometric data, non-geometric data and documentation. they refer to the ISO-19650 norm which describes this data categorization. Geometric data refers to geometrically modeled objects in a BIM-model. Non-geometric data refers to all the information that is not geometric data that can be represented in a structured manner that allows it to be computer interpretable. 'Documents' refers to all the information that is not represent in a manner that makes it computer readable. This information is only interpretable by humans. With respect to the last two information types ('non-geometric data' and 'documents') the interviewees state that, currently, there is too much information represented in the form of 'documents' while the nature of the information is perfectly suitable for documenting as 'non-geometric data' which exposes the information to the potential of automated interpretation. Technical information and requirements documentation where given as examples.

7.2 Maintenance categorization

All the interviewees, involved with managing maintenance related processes, defined the same three categories of maintenance tasks that (can) take place during the maintenance and operations phase of buildings. These are; Planned maintenance, Corrective maintenance and user or owner desired changes. Planned maintenance is maintenance, planned on the basis of pre-defined maintenance cycles based on theoretical degradation curves. For instance; window frame painting or roof topping replacement are tasks that have a predefined maintenance cycle to them. Corrective maintenance is maintenance that directly needs to take place in reaction to a failure of some kind of element, for example; an identified roof leakage, a defect boiler or a broken window. The last category, 'user or owner desired' refers to all actions that take place to improve or change a building in reaction to a user or owner desired plan, for example; upgrading the energy efficiency performance, or a client desired refurbishment project in order to change lecture rooms into laboratories because of changing capacity needs of an organization. The interviewees state they treat this last category similarly to new construction projects. See figure Fig. 17 Information flows for a schematization of the four defined information flows and from where they come.

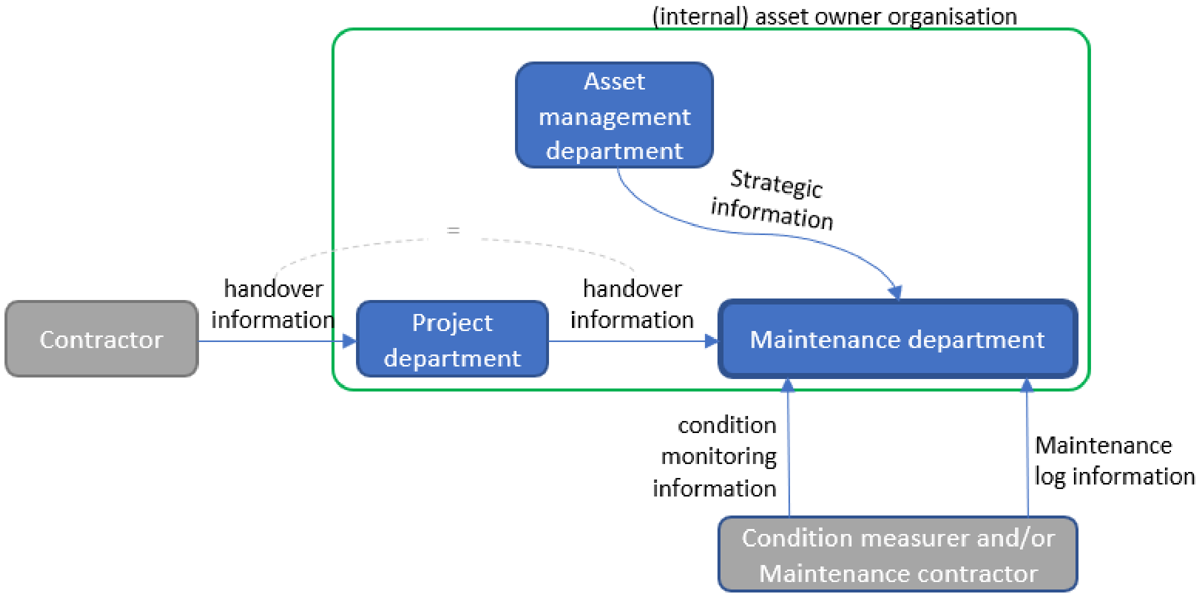

Fig. 17 Information flows

7.3 Organizational structure with respect to maintenance

All the interviewees, operating as client, describe that their organization has separate departments for dealing with the realization phase (new construction) and the maintenance phase. The consequence of this separation is that the information handover process is taken care of by the project department while the information is used by the maintenance department.

All interviewees, operative in maintenance departments, point out that they are not getting all the information they consider to be relevant for supporting maintenance related tasks, during the information handover, nor do they get it in the structure or form they want. As a result maintenance departments need to invest a lot of time in gathering, manipulating and cleaning the handover information in order to become useable for supporting their tasks. As a reason for not receiving the handover information they want, the interviewees point out that project departments are not willing to listen to their information requirements and thus do not take them into consideration when communicating the handover requirements to the contractor.

On the other hand, interviewees operative as contractor, note that clients are not able to / do not define (explicitly) what information they need for their maintenance processes. The interviewees mention that they do not know if, the maintenance departments of a client is involved in the information handover process. Considering the statements from the interviewees, operative at maintenance departments, the problem is that contractors only communicate with the project department of a client, while the maintenance department has the answer for this question. The goal of this research is not to find out what exactly is going on here. However, in order to optimize information management processes, asset owners will have to address this issue. Because, if they are not able to get the information they need in the form and structure they need it is impossible for the maintenance department to optimize their information management processes no matter what technology they intend to use.

7.4 Protocols, structures and formats for information generation and creation

In this sub-chapter the information structures and formats in which maintenance departments get and use information will be discusses. Also the way in which they (try) to make sure that they get the information in the way they want are discussed. It was observed that protocols, structures and formats where closely related to one of the four information flows, defined in sub-chapter 7.1, therefore they will be discussed per information flow.

Handover information

The handover information compromises all the information that describes the as-build situation of the delivered building. This information is created by the parties involved in the design and engineering phase of the building.

With respect to information delivery, all interviewees describe two sorts of handover protocols. The main protocol exists of checklist that states which documents have to be handed over. For some documents a specific document format (PDF, Excel, Word, DWG and IFC) is prescribed. For geometric information, additional protocols are used that describe how handover deliverables should look like in more detail. For all other information no specific structure or explicit information points that the documents should contain are prescribed.

All interviewed asset owners state that they have the ambition to use information in BIM-models as input for their maintenance planning processes (two asset owner already contractually prescribes the delivery of BIM-models as part of the handover information). In order to realize this ambition, asset owners realize that they need to prescribe all the information that should be contained in the BIM-model and how this information should be structured. To do so, asset owners have created/are creating new protocols to make sure that handed over BIM-models contain all the information they need in the structure and form they want.

All the interviewees describe the use of two types of protocols; ‘BIM-manual’ and an ‘Information Delivery Specification’ (IDS) (Dutch: Informatie Leverings Specificatie (ILS)). The BIM-manual describes the process and rules according to which the BIM-model has to be created. The purpose of the IDS is to prescribe exactly what elements should be modeled and what properties should be added per element type.

All the interviewees mention that they prescribe the IFC-scheme as the information structure according to which BIM-models need to be handed over. Within the IFC-models all the interviewees also want the 4-digit NL-SfB table-1 coding added to the IFC-elements as a parameter. The NL-SfB is a Dutch standard that describes classification categories for building elements. When asked if mapping the NL-SfB element structure to the IFC-element structure created problems, all the interviewees stated that this actually was quite easy to do and did not caused problems, all the asset owners defined this mapping in their IDS.

Condition monitoring

All the interviewed asset owners state that, when a building is taken in use, they periodically monitor the actual degradation status of the building by conducting ‘condition measurements’ according to the method described in the Dutch norm NEN-2767-1[13]. The method prescribes that condition measurement should be done on the building element level. Next to a methodology the NEN-2767 also defines an specific element structure and a list with predefined defects and deficiencies (NEN-2767-2). This structure shows great similarity with the NL-SfB element structure. However, one of the interviewees states it is not exactly the same. The predefined defects and deficiencies are related to a specific building element.

In most cases, the interviewees state that, the maintenance department outsources condition monitoring to a specialized company or lay the responsibility with the contractor as part of the

[13] This norm contains 4 parts of which only the first two are relevant for buildings. The first part (NEN 2767-1) describes the method and the second part (NEN 2767-2) defines a element structure.

maintenance contract. Although, in both cases the maintenance department wants to get the results of the condition measurements. The results are handed over to them in a PDF document as a report. One of the asset owners indicates that they sporadically preform condition measurements on small samples of element in order to compare with the results of the condition measurements conducted by contractors.

Maintenance log

All the interviewed asset owners state that they document maintenance actions, performed by the contractors[14]. To do so, asset owners require the contractors to deliver documentation of the maintenance that they conducted. However, although most asset owners also have a maintenance-manual that intends to describe process and organization with respect to conducting maintenance on a general level. None of the asset owners has a protocol in place that explicitly describes which information should be handed over and how it must be handed over with respect to documenting conducted maintenance. As a result the quality of this documentation varies enormously. For example, one interviewee states that sometimes they get maintenance logs in the form of an Excel table that was printed out and filled in by hand. Many of the asset owners state that more often than not the maintenance history was not archived properly, resulting in incomplete and un usable maintenance history.

Strategic information

Strategic information refers to all decisions that are being made with respect to real estate usage that potentially delivers economic benefit when taken into consideration in the maintenance planning process. For example, one information managers states that when roof topping needs to be replace, but a refurbishment project is planned one year later, for which the roof topping needs to be cut open at main places (e.g. because bigger ventilation units have to be placed in order to meet a higher ventilation capacity which is needed for the new laboratories). It is from an economical perspective more efficient to delay the replacement of the roof topping with 1 year and combine it with placing the new ventilation units on the roof. Another example is when an organization has the ambition to upgrade all its buildings to a certain energy-label (an ambition that all the interviewed asset owners have). It is interesting to combine task that have to take place for this goal with maintenance tasks. For instance when considering roof topping replacement again, money will be saved when this is combined with the upgrading the roof insulation value.
The way in which this sort information is being communicated towards the maintenance department is on an ad-hoc basis, no standard protocols where identified. One interviewee states that it is not always clear what ambitions and plans are made with respect to a specific building, when this is the case a maintenance planer has to check with the responsible department in person in order to ask if any plans are made which should be taken into consideration.

7.5 Information use

Handover information

All the interviewees use computerized maintenance management systems (CMMS) to keep track of their buildings on an element level. The initial input for these systems is a list of the building elements derived from the handover information. The process of deriving the building elements from the handover information and putting them in the CMMS-system is a manual process. All interviewees

[14] For clarification, in this case the contractor is the one actual conducting the maintenance, this is not by definition the same as the contractor that delivered the project.

state that they are or want to investigating the potential of automatically filling their CMMS-system with data from IFC-models by directly importing it or with one step in between; first exporting IFC-data to a Excel sheet and then importing it (depending on what their CMMS-system allows them to import). However, none of them have been able to do this yet. In practice this means that currently CMMS-systems are filled manually most of the time and IFC-models are only used occasionally for visualizing purposes if at all they are used[15].

Condition monitoring and maintenance log information

Condition monitoring results and maintenance history could give the maintenance planner a good impression of the actual maintenance needs for a building. However, because of the quality of the sources that (should) contain these information. This validation process is an frustrating process. One interviewee states that in many cases maintenance planers even prefer going to a building on-site to spot the actual situation instead of consulting the inspection reports, maintenance history and technical information.

Strategic information

As described in sub-chapter 7.5 strategic information is described and stored on an ad-hoc basis, each time that strategic information is created, the author decides how to document it and where to store it based on his own preference. As a result strategic information is scattered around in different places and documented in different ways. Because of this, maintenance planners have no clear view of all the strategic information that is available or where they can find it. One interviewee states that it is not always clear what the exact plans, when this is the case maintenance planers check with the responsible department in person in order to ask if any plans are made which he/she should know of.

7.6 Summary

This sub-chapter summarizes the main findings that where derived from the interviews;

- Information, relevant for supporting maintenance related tasks, origins from four different information flows. Being; project delivery (handover information), status monitoring, maintenance logs and strategic information.
- With respect to handover information, asset owners, have three types of protocols in place; a general protocol prescribing all the handover requirements in general, a BIM-manual describing the process and rules according to which the BIM-model has to be created and a Delivery Specification’ (IDS) (Dutch: Informatie Leverings Specificatie (ILS)) that prescribe exactly what elements should be modeled and what properties should be added per element
- the IFC-schema is mentioned by all interviewees as the established structure for BIM-models, additionally all interviewees mention the use of the NL-SfB table-1 (4-digits) coding as building element classification system.
- All asset owners mention the NEN-2767-1 condition measurement method for monitoring building element states, no specific structure or form is prescribed for the resulting inspection reports.
- All asset owners document the maintenance history of buildings. However, no specific structure, form or content is prescribed.
- Strategic information is being communicated to the maintenance department on an ad hoc basis, no standard structure or format is present.

[15] This is in line with the research problem described in sub-chapter 1.1.

7.7 Comparison

In this sub-chapter, the findings are compared with the findings that where derived from the literature study in order to validate the findings, the conclusions from this comparison are summarized down below;

- With respect to standard information structures both the findings to the literature review as well as the interviews, the IFC-schema is mentioned as the established information structure according to which information in BIM-models is structured. The literature study also mentions COBie as a specific MVD of the IFC-schema specifically developed to facilitate the information handover process. However, during the interviews COBie was not mentioned at al.
- with respect to building element categorization the literature study only mentions table 21 of the OmniClass classification system, which is an United States standard. Whereas the interview findings mention the NL-SfB table-1 (4-digits) coding as building element classification system, the NL-SfB is a Dutch standard.
- The conceptual information categorization identified in the literature review (Systems and Equipment, Data, and, Documents) shows great similarity with the categorization mentioned by two interviewees (Geometric data, Non-geometric data and Documents). In this comparison the Systems and Equipment category is synonym for Geometric data, data is synonym for non-geometric data and documents with documents.

8 Case description

As part of the interviews, the experts were asked to define specific tasks, that could serve as cases for the validation process. For these tasks the experts were asked to describe which information they considered to be relevant, in which sources this information resides and finally to make these sources available so that they could be used for the validation process. Two experts (from Eigen Haard and TU Delft) were able to support cases with information sources. All the cases that the experts formulated relate to one specific task type; 'maintenance planning validation'.

This chapter first describes the task, that the experts formulated, from a general perspective (sub chapter 8.1). Secondly, The individual cases will be described in more detail The sources that the experts consider to be relevant will be described. And the specific information points in these sources that they would like to see queried and represented in a single place to support the task they described. The list of information points that the experts considered to be relevant will serve as the competency question, used in the single case mechanism experiments (see sub-chapter 11.1) to validate if the designed maintenance ontology is able to support the necessary queries.

8.1 Task description

As mentioned in chapter 7, the interviewed experts, active at organizations, with as core business providing buildings to facilitate a certain need, (e.g. campus facilities or social housing) do not have inhouse capacity for conducting maintenance tasks. Instead, they outsource maintenance. For this reason, the experts were not able to formulate a case with a maintenance tasks as the subject[16]. experts from Eigen Haard and TU-Delft were able to formulate a similar case; The validation of the 'multiple-year maintenance planning' (MYMP). From a technical perspective the task is similar at both organizations. However, the need for validating the MYMP originate from different context, these are explained in Table 8 down below;

Table 8 Specific case description per organization

Eigen Haard At Eigen Haard, drafting the MYMP is an inhouse activity. Based on the MYMP, Eigen Haard outsources maintenance tasks to contractors. The MYMP is based on theoretical deterioration rates for specific building elements. However, in reality deterioration rates of building elements can (based on a variety of variables) deviate from the theoretical deterioration rate used in the MYMP. Because of this, the reality is that many maintenance activities that are planned for execution, based on the MYMP, are in reality not necessary, based on the actual deterioration rate of the building object of interest. Eigen Haard want to make sure that they only outsource maintenance tasks if they are really necessary. To guarantee this, Eigen Haard cannot solely rely on a (MYMP). Luckily, Eigen Haard does generate a lot of information about their buildings which is useful input for validating the MYMP. However, this information is scattered over multiple information sources. Additionally, many of the information sources have no- or only very limited filter functionality, which prohibits easy and fast searching for specific information points. For these reasons, maintenance planners at Eigen Haard experience this as a tedious and complex task. As a consequence maintenance planners rather choose to go to the building on site to determine if maintenance is necessary. The result is that the validation process costs much more resources than necessary. See sub-chapter 8.2 for a detailed description of all the sources and specific information point in those sources that Eigen Haard considers to be relevant for the case.

[16] In the research proposal, this type of tasks was envisioned to serve as cases for the validation process.

TU-Delft CRE
At TU-Delft, Maintenance is outsourced on the basis of functional requirements for a fixed period of time (e.g. 10 years). In this time period the contractor has the freedom to plan maintenance in the way that he considers to be the most cost efficient in order to meet the functional requirements. However, TU-Delft has experienced that giving the contractor total freedom without control, opens the possibility for undesirable situations. For example, the contractor can decide to communicate to TU-Delft that way more maintenance is necessary than actually is the case, and subsequently billing this unnecessary maintenance to TU-Delft. To prevent this, TU-Delft demands that contractors hand in a MYMP so that they can validate if MYMP is valid. From here, the problem that TU-Delft encounters is identical to that of Eigen Haard, the organisation does generate useful information for this validation. However, this information is scattered over multiple information sources. Additionally, many of the information sources have no- or only very limited filter functionality, which prohibits easy and fast searching for specific information point. For these reasons, maintenance validators experience this as a tedious and complex task. As a consequence maintenance validators rather choose to go to the building on site to determine if maintenance is necessary or even worse validation does not take place at all. The result is that the validation process costs much more resources than necessary or in the case that validation does not take place at all, TU-Delft is still exposed to the risk that contractors mis use the contract. See sub-chapter 8.2 for a detailed description of all the sources and specific information point in those sources that Eigen Haard considers to be relevant for the case.

8.2 Case description

In the case, the validation of planned 'roof topping replacement' will be the focus. Maintenance planners need to validate if, based on the actual status of the roof topping, replacement is necessary. For this case TU-Delft and Eigen Haard both have provided information sources and a description of the information points, in these documents, that they consider necessary for validation (the competency question).
Sub-chapter 8.2.1 Describes the sources and information points in those sources that TU-Delft considers to be relevant for this task, sub-chapter 8.2.2 describes the sources and information points in those sources that Eigen Haard considers to be relevant for this task.

8.2.1 Case 1; TU-Delft

The information sources that TU-Delft considers to be relevant are listed in Table 9. The table also contains a description of the origin of the source and the form in which TU-Delft has provided the source.
Table 10 contains a summarization off all the information points that TU-Delft considers to be useful for validating planned roof topping replacement tasks. Additionally the table describes in which information source these information points reside, the format of the source and the structure within the source. The summarization off information points function as the competency question for the single case mechanism experiments described in chapter 11.

Not all the sources describe the same building. The reason for this is that TU-Delft only has IFC-models of recently delivered buildings. Because these buildings are recently delivered no substantial maintenance history of these buildings is generated yet. However, maintenance history from an older building is provided. For the purpose of this study, the sources are assumed to be related to the same building. The BIM-model is a representation of the TNW-building (see Fig. 19), all other sources describe the EWI-building (see Fig. 18). The use of information sources related to different buildings does not diminishes the results of this research, because the goal of this research is to proof a technical point with respect to the nature of the information sources.

Table 9 Information sources; case TU-Delft

Information source	**Description**
BIM-Model	An IFC-model containing a geometric representation of the building with additional non-geometric parameter information; The BIM-model is created in the design and engineering phase and delivered to TU-Delft as part of the handover information.
Warranty list	An excel document in which TU-Delft keeps track of warranty information with respect to building elements of a building.
Condition monitoring report	An PDF report that describes deficiencies according to the NEN 2767-1 Condition monitoring method and condition score's derived from these deficiency descriptions; As part of the maintenance contract, the contractor is obliged to monitor the status of all the building object ones in the three years And share the results of these reports with TU-Delft. TU-Delft also monitors samples of building parts to compare with the monitoring results of the contractor.
Maintenance history	TU-Delft registers maintenance history in database software called 'TOPdesk FMIS'. TU-delft has made an excel export of this register available for the research.
Projects list	TU-Delft keeps track of their planned projects in Microsoft Project, a software program used for making schedules. TU-delft has provided an .ppm file containing all the information with respect to planned documents.
Aspect Priority Schema	In the Aspect Priority Schema, TU-Delft defines the level of priority that should be given to a building deficiency according to the type (aspect) of deficiency. The Aspect Priority Schema is documented in the form of a PDF-report.
Desired Condition Level Table	TU-Delft has formulated the minimum required condition level that is desires building elements to have according to the NEN 2767-1 Condition monitoring method. This is documented in a PDF-document.

Table 10 Information points; case TU-Delft

Sources	Format	Structure	Information needs
BIM-model (Drawing)	IFC-SPF	IFC 4	1. Roof surface Area (m^2) 2. Roof edges (m^1) 3. Roof finishing type
Warranty list	Excel	Document specific columns	4. Warranty status of an building element 5. Warranty end date on an building element 6. Warranty provider
Condition monitoring report.	PDF	NL-SfB	7. Deficiency type 8. Deficiency description 9. Intensity of a deficiency 10. Scale of a deficiency 11. Condition score based on a deficiency 12. Aspect type of a deficiency 13. Date of identification of a deficiency 14. Name inspector 15. Name Organization responsible for the report
Maintenance history logbook.	Excel	Document specific columns	16. Date defect (and repair) 17. Deficiency type 18. Name identifier 19. Type of repair
Projects list	Excel	Document specific columns	20. Planned client/owner desired projects on the building 21. Start date of the project 22. End date of the project 23. Type of project
A/P schema	PDF	No structure	24. Priority related to aspect of an deficiency
Desired Condition Level Schema	Excel		25. Desired condition level of the building object

Fig. 18 EWI-building; campus TU-Delft

Fig. 19 TNW-building; campus TU-Delft

8.2.2 Case 2; Eigen Haard

The information sources that the expert, interviewed at Eigen Haard, considers to be relevant with respect to the described task are listed in Table 11. The table also describes the origin of the source and the form in which Eigen Haard has made the source available for use with respect to the research. Table 12 presents all the explicit information points that the expert, interviewed at Eigen Haard, considers to be necessary in order to validate a planned maintenance task with respect to roof topping. Additionally, Table 12 describes in which information source these information points reside, the format of the source and the structure in the source. The information points function as competency questions for the single case mechanism experiments (described in chapter 11).

The BIM-model that Eigen Haard provided (see Fig. 20) represents a housing block located in the Rivierenbuurt neighborhood in Amsterdam (see Fig. 21). Except for the BIM-model, Eigen Haard did not provide the information sources. Instead, Eigen Haard provided 'screenshots' of the sources. From these screenshots, the structure (e.g. table columns in a database are excel document) and nature of the information points (e.g. cell values in a table or database) in the sources was derived. Based on the derived structure and nature of the sources, imitation sources were created in order to use for the single case mechanism experiment described in sub-chapter 11. The fact that the research uses imitations sources does not diminishes the results of this study because the research focusses on interconnecting information points. This means that the values of these specific information points are not relevant as long as the structure according to which the information points reside in the original sources is identical in the imitated sources.

Table 11 Information sources; case Eigen Haard

Information source	**Description**
BIM-Model	An IFC-model containing a geometric representation of the building with additional non-geometric parameter information; The BIM-model is created in the design and engineering phase and delivered to Eigen Haard as part of the handover information.
Condition Status documentation Deficiency notification history, Maintenance history	Eigen Haard stores condition status of roof elements in the database of an asset management IT-tool, named Dakota. Just like at TU-Delft, the condition scores are determined on the basis of the NEN 2767-1 Condition monitoring method. However, as opposed to TU-Delft, only the condition score is stored, the parameters that lead to the condition score (Intensity, Scale, stadium) and descriptive information are not stored. Notified deficiencies and accompanying counter measures are also stored in Dakota.
Strategic information	Eigen Haard keeps track of strategic information with respect to its buildings in a software tool called 'real-estate information model' (Dutch: 'vastgoed informatie model (VIM)). In VIM Eigen Haard keeps track of strategic information such as; - Is Eigen Haard planning to sell the building,

	- Are there plans with respect to sustainability goals with respect to the building, - Are there housing units in the complex that are owner-occupied.
Planned maintenance	Eigen Haard documents its planned roof maintenance activities in an Excel document.

Table 12 Information points; case Eigen Haard

Sources	**Format**	**Structure**	**Information needs**
IFC-model	IFC-SPF Aedes-ILS 1.0	IFC 2x3	1. Surface Area (m^2) 2. Roof edges (m^1) 3. Roof outlets (nr.) 4. Roof height (m^1) 5. Finishing type (material) 6. NL-SfB code
Deficiency detection, maintenance history and warranty information	Dakota (roof specific CMMS)	- NL-SfB - Database specific columns	7. Condition score 8. Desired condition score 9. Inspector 10. Responsible organization 11. Notified deficiencies 12. Type of notified deficiency 13. Date of notified deficiency 14. Planned counter measure for notified deficiency 15. Status of the counter measure 16. Warranty status 17. Warranty start date 18. Warranty end date 19. Warranty provider
Planned projects list	Excel document	Document specific columns	20. Project name 21. Project type 22. Project description 23. Project status 24. Project start date
Strategic information	VIM	Database specific columns	25. Selling plans 26. Plans with respect to sustainability 27. Amount of owner occupied housing units in the complex

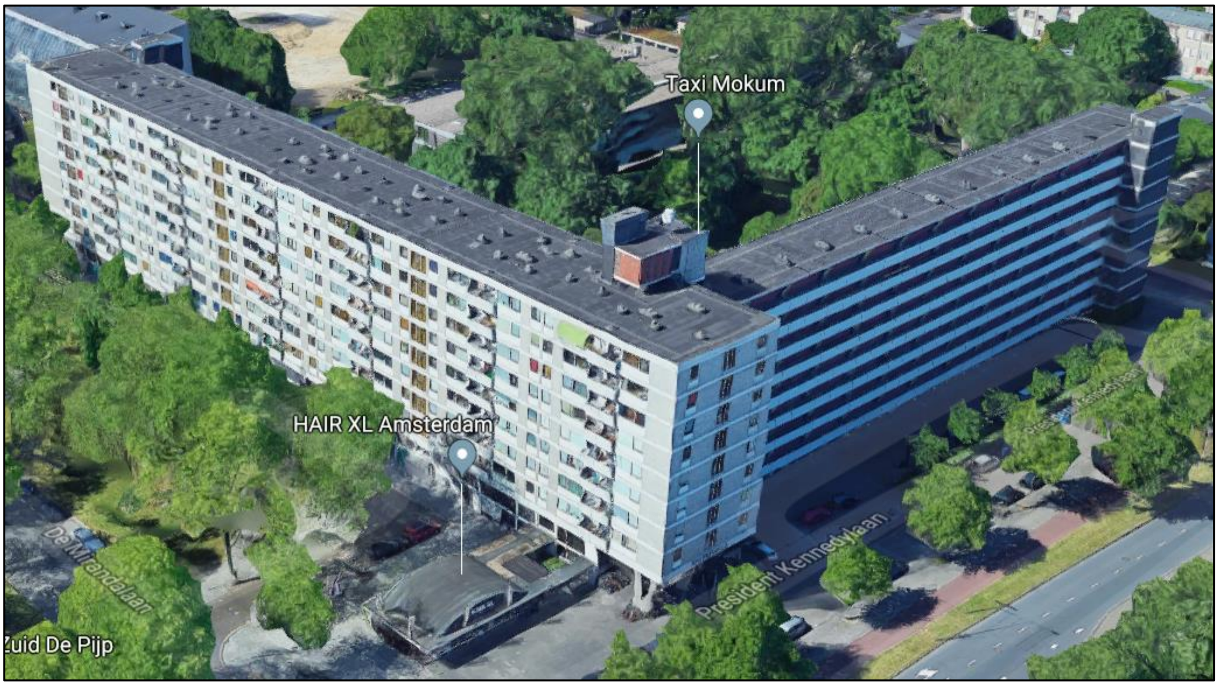

Fig. 21 The Rivierenhuis building

Fig. 20 IFC-model of the Rivierenhuis building

PART-IV DEVELOPMENT AND VALIDATION

The aim of this research is to develop an ontology, based on which more efficient query writing and linking of information sources becomes possible, compared to using the ifcOWL ontology, so that information from multiple sources can be efficiently queried to be represented in one location.
This part of the report presents the designed 'Maintenance Ontology' with rationale for made design decisions (see chapter 9 and 10). Additionally, it presents the validation experiments and the results derived from these experiments (see chapter 11). This part of the research report provides answers for research question three, four and five.

9 Ontology Engineering

As described in the research methodology (chapter 2) the ontology engineering process is based on the ontology development method, described by Noy, and McGuinness (2001), existing out of seven steps (see Fig. 22).

Fig. 22 Ontology development method steps, describe by Noy, and McGuinness (2001)

The results of the first two steps ('determine domain and scope', and 'consider reusing existing ontologies') are presented in sub chapter 9.1.1 and 9.1.2 The results of the last five steps are represented as an integrated body in sub chapter 9.2 this includes the rationale for all the designed classes, and relations and properties assigned to these classes.

Although the ontology development method, described by Noy, and McGuinness (2001), does provide a method for deciding 'which' classes, relations and properties should be modeled. It does not cover how the classes, properties and relations should be modeled, design decisions with respect to this matter will be covered in sub chapter 9.1.3. In appendix B the complete Maintenance Ontology is presented in turtle syntax.

9.1 General design principles

9.1.1 Domain and scope

As described as the goal of this research the ontology will be designed in order to "*interconnect and query, building related, maintenance information from multiple sources*". The literature study and interviews gave a high level view of what exactly contributes to "*maintenance information*".
The results of literature study and interviews have been taken into consideration during the ontology development process. However, as described in the methodology, it is not realistic to validate an ontology that covers the whole "*building related, maintenance information*" domain. Therefore, the validated scope of the designed ontology limits itself to the maintenance information that was considered to be important in the cases, described in chapter 8, these cases form the basis for the validation of the ontology described in chapter 11. The cases describe the information that experts from the maintenance department of Eigen Haard and TU-Delft consider to be relevant for supporting the validation if maintenance tasks based on the MYMP are really necessary. In the cases, roof topping is the building element that will be focused on.

9.1.2 Reusing existing ontologies

Reusing ontologies has proven to be a good practice in ontology modeling. It saves time to reuse concepts of existing ontologies and/or vocabularies instead of reinventing the wheel. And, it makes data more easily understood and processed by applications and ontology architects (Pattuelli, Provo, & Thorsen, 2015).

When reusing ontologies it is important to make a conceptual distinction between two types of ontologies; Modeling level ontologies (which are actually ontology modeling languages) and Domain level ontologies (Allemang & Hendler, 2011). The first type of ontologies can provide general modeling concepts (e.g. class, sub class, property, comment, etc.), the second type of ontologies are domain ontologies that contain domain specific concepts (e.g. building, maintenance task, roof topping or sensor), that might be useful for the new Maintenance Ontology. Reuse of both ontology types will be discussed in a separate sub-chapter (sub-chapter 9.1.2.1 and 9.1.2.2).

9.1.2.1 Reuse of modeling level ontologies

With respect to modeling level ontologies, the best practice is to reuse concepts described in (open) linked data standards, published by W3C. W3C is widely accepted as the authority on publishing (open) linked data standards, which means that most people, operative in semantic web and linked data technology, are familiar with their concepts and know how to work with them (Allemang & Hendler, 2011).

RDF (next to representing the fundamental meta data model for describing linked data in general, as described in chapter 3) also contains some of the most fundamental generic vocabulary concepts, for describing RDF-data, one of these is the ‘rdf:type’ concept. This concept is used to describe that an instance is part of a specific class concept. In the maintenance ontology the ‘rdf:type’ concept will be reused for this purpose.

As described in chapter 3, RDFS and OWL both are W3C specifications that provide fundamental vocabulary concepts for constructing ontologies. For the maintenance ontology, concepts of both standards will be used in order to design the ontology. See sub-chapter 9.2 for a detailed description of how RDFS and OWL concepts are used in the maintenance ontology.

For annotating purpose SKOS concepts are used. SKOS is also a W3C standard. ‘skos:definition’ will be used to describe the definition of the maintenance ontology concepts, this provides in giving users an option to consult the exact meaning of an concept. By doing so, the chance of misinterpretation is reduced. Additionally, ‘skos:prefLabel’ and ‘skos:altLabel are used in the ontology in order to provide an human readable interpretation of ontology concept for users of the ontology (because the representation that fundamentally embodies the concept is the URI which main goal is to be computer interpretable).

9.1.2.2 Reuse of domain ontology level ontologies

Almost all the competency questions (described in chapter 8) indicate that maintenance information needs to be connected/related to a building or one of the building elements that the building consists of. Ontologies that describe the ‘Building’ and ‘BuildingElement’ concept, already exist. For example, the BOT and ifcOWL ontology. Therefore, the maintenance ontology will not redefine concepts for describing buildings and their elements. Instead, the maintenance ontology will function as a module that can be used in conjunction with an ontology that already describes the ‘Building’ and ‘Building Element’ concept. As visualized in Fig. 23 the maintenance ontology can be connected as a module to a building topology ontology on two levels; The ‘building’ concept level and the ‘building element’ concept level, these are the connections that are necessary in order to query all the necessary maintenance related information of a building or building element, as specified in the competency questions (see chapter 8). Fig. 23 also visualizes the relations with which the main class concepts are linked to each other. For these relations, concepts are reused from the 'Top Level Conceptual Model' (CM) described in the NTA 8035[17]. This ontology is planned to become the standard for information modeling within the Dutch construction industry, thus it is expected that linked data architects, active in the construction industry will be familiar with these concepts. For the same reason, all the classes in the maintenance ontology will have a 'rdfs:subClassOf' mapping with a more generic concept of the CM.

The NTA 8035 will also be entered into the international standardization process, CEN TC442/WG4/TG3, the intention is to create an international variant of the NTA 8035 (NEN NC 381184 Working group, 2019).

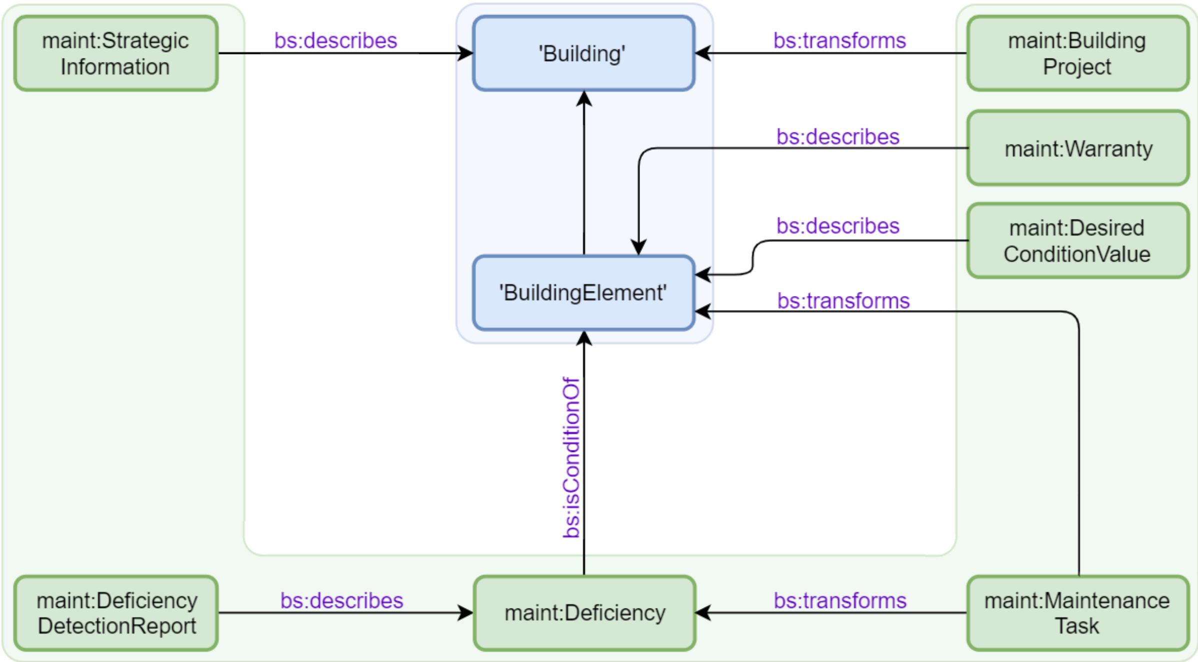

Fig. 23 Maintenance Ontology

9.1.3 General Modeling Techniques/Mechanisms

In this sub-chapter the, non-concept specific, design decisions are elaborated. Determining which modeling techniques to use is not an explicit part of the ontology development method describe by Noy, and McGuinness (2001). However, these design choices have an huge effect on the usability of the maintenance ontology. For Example, it influences the level of expressivity that the ontology allows to describe and the complexity of the queries that are necessary to query data, represented according to the ontology. Therefore, it is important to take these decisions in (explicit) consideration.

9.1.3.1 Modelling classes

Classes are the core of the ontology, without classes there is no ontology (Noy & McGuinness, 2001). For describing that a concept is a class, the W3C standards provide two options, being; 'rdfs:Class' and 'owl:Class'. The difference of these two concepts is that using the 'owl:Class' concepts allows for adding more expressivity to the class (e.g. restrictions) on the basis of which OWL reasoners can infer knowledge. Because the Maintenance Ontology makes use of restrictions (which will be discussed in

[17] The NTA-8035 use the bs: prefix as abbreviation for the namespace that is assigned to the Top Level Concept Model. Being; 'https://w3id.org/def/basicsemantics-owl#'.

more detail further on in this sub-chapter) the 'owl:Class' concept will be used. Fig. 24 visualizes how concepts are declared as a class.

Fig. 24 'owl:Class' instantiation

However, when modeling the more general domain classes (e.g. maint:MaintenanceTask), these classes often represent a group of more specific concepts which the ontology must allow to express. This can be handled in two ways; 1) sub-classes are created to represent the specializations of the super-class, or 2) a property is assigned to the class concepts which allows to say something specific per instance of the general concept. For the maintenance ontology the first modeling mechanism is chosen this makes querying for a specialization of the super-class less complex because less triple patterns have to be matched. The concept for describing the subclassification of classes will be done with the RDFS property concept 'rdfs:subClassOf' see Fig. 25 for a visualization of this mechanism.

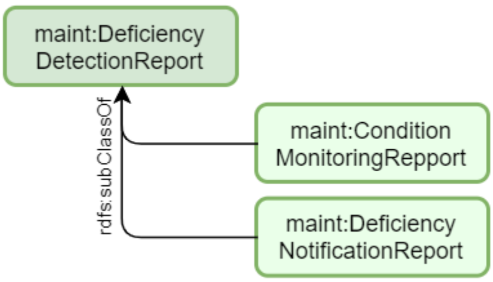

Fig. 25 Declaring sub-classes with the 'rdfs:subClassOf' concept

9.1.5.2 Modelling attributes

From an ontology modelling perspective, in linked data, properties are subdivided into two types of properties; properties that assign values to attributes of objects, and properties that describe relations between objects. This sub-chapter solely focusses on the first type of properties; the properties that assign values to attributes of objects. The second type of properties are discussed in sub-chapter 9.1.5.3.

With respect to property modeling there are two main modeling methods than can be used; objectified property use and direct property use (Rasmussen, Lefrancois, Pauwels, & Hviid, 2019).

Objectified properties are properties that are modeled as an instance of a class, see Fig. 26, her is visualized that the property "maint:taskDescription" is represented by a node instead of an arc. Because the property is objectified it becomes possible to link metadata to the property (e.g. who filled in the value of the property and on which date). Direct property modeling, as shown in Fig. 27, creates a direct link between the object and the value that the property describes.

Opposed to objectified properties, direct properties do not allow for linking metadata to the property. However, the added expressivity that objectified linking enables, comes with the prize of added complexity. Thus choosing for the objectified property modeling method is only justified if the added expressivity is necessary for a specific goal.

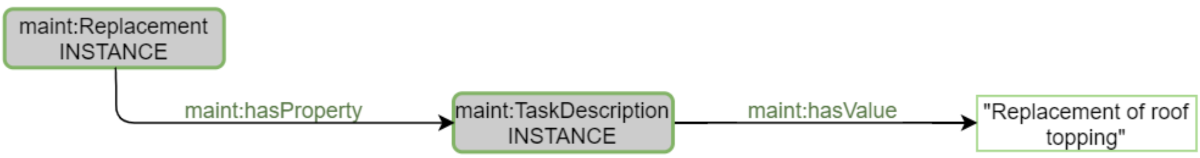

Fig. 26 Objectified property modeling example

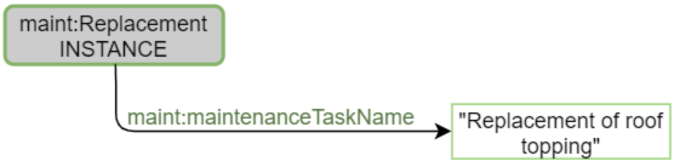

Fig. 27 Direct property modeling example

With respect to the maintenance ontology, property objectification does not serve a particular use. Therefore, properties in the maintenance ontology are modeled according to the direct property modeling method, so that unnecessary complexity is not added to the ontology.

9.1.5.3 Modelling Relations

As described in sub-chapter 9.1.5.2, relations are modeled as properties with class concepts (or instantiations of class concepts) as the objects as well as the subjects of a triple.
Just like datatype properties, Relations can be modeled as direct properties or objectified properties. In order to meet the competency questions described in chapter 8, objectified properties are not necessary. Therefore, relations are modeled as direct properties so that unnecessary complexity is not added to the ontology. As described in sub-chapter 9.1.2.2, all the relations in the maintenance ontology are reused from the Top Level Conceptual Model, described in the NTA 8035.
Because the relation concepts are reused from an existing ontology standard, with its own scope and use in mind, it is not possible to ad 'rdfs:domain' and 'rdfs:range' properties to these reused concepts in order to make them fit the Maintenance Ontology. Instead, the use of OWL restrictions, prescribed by W3C will be used. These restrictions will be assigned as a subclass ('rdfs:subClassOf') to the classes that makes use of NTA 8035 relation properties (this are the classes of which instantiations will be in the subject position of a triple of which the NTA 8035 relation is the predicate). See Table 13 for an example of the OWL restriction mechanism (in turtle syntax).

Table 13 OWL restriction example

Subject	Predicate	object
maint:Deficiency	rdf:type	owl: Class
	rdfs:subClassOf	[a owl:Restriction ; owl:onProperty maint:isConditionOf ; owl:allValuesFrom BuildingElement] .

9.1.5.4 Datatypes

The *plain literal* values that datatype properties relate to objects should be described as a specific data type (e.g. Boolean, string or decimal) (Noy & McGuinness, 2001). in SWLD terms typed values are called *typed literals* (W3C, 2004). Assigning data types allows tools to interpret the values correctly. Without explicitly assigning datatypes to values. Programs that use these values (e.g. for calculations) may interpret '800' as a string and not as an integer, this would make performing automated calculations, with these values as input, impossible.
With SWLD-technology there are multiple methods for assigning data types to literal values. The use of XML data types is incorporated in the RDF-standard, it allows for assigning XML datatypes to literals within the object part of a literal, see Fig. 28 for a visualization of this mechanism.

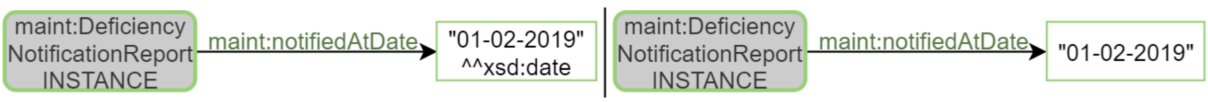

Fig. 28 typed literal as object value (left); plain literal as object value (right)

The only difference is the '^^xsd:date' annotation that is added to the object value at the right, with this annotation correct interpretation of literal values is enabled. Although, the interviewed experts did not mention the need for correct interpretation of machines of the data type. XML datatypes will be used as a good practice and for potential future use cases.

9.2 Classes, Properties and Relationships

9.2.1 maint:Deficiency

The 'maint:Deficiency' concept, represents an undesired change of a building elements condition for the worse. As concluded in chapter 8, the condition of an building element is determined on the basis of deficiencies that are identified on that building element (this is how condition scores are assigned according to the NEN 2767-1 method). The condition score, derived from the deficiency, is attached as a property to the deficiency (being; 'maint:hasConditonScore'). An instantiation of the 'maint:Deficiency' concept is linked to the building object instance, on which it is identified, with the 'bs:isConditionOf' relationship (object property). This relationship makes it possible to query for deficiencies related to a specific building element and thus also the condition score, which is attached as a property to the deficiency.

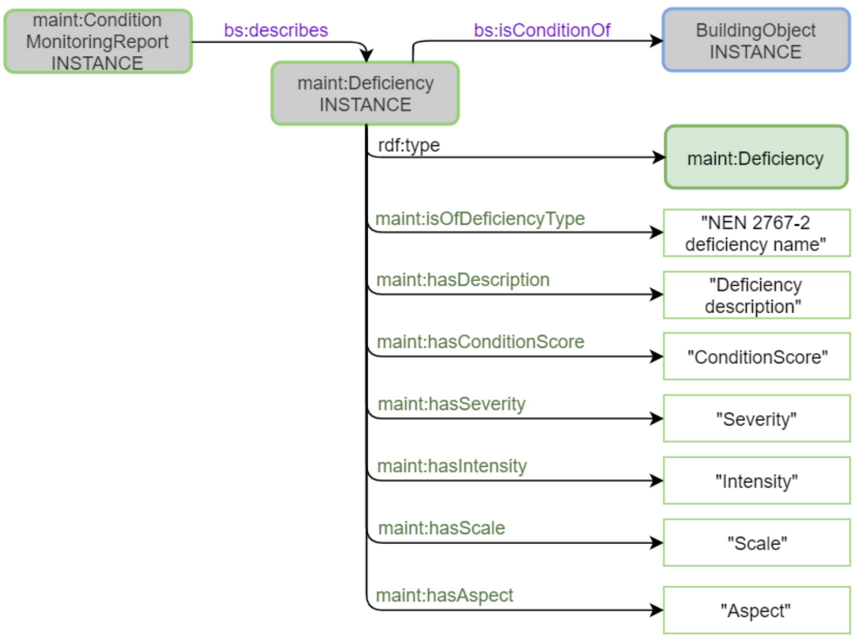

Fig. 29 Visualization of an 'maint:Deficiency' class instance

To allow for querying more context information of a deficiency (which the interviewed experts described as relevant, with respect to the formulated task), this information is also connected as a parameter to the 'maint:Deficiency' concept. See Fig. 29 for a full visualization of a 'maint:Deficiency' instance.

Meta data such as; inspector, responsible company, and inspection date (which the competency questions also describe as relevant information points) are not assigned as parameters to the 'maint:Deficiency' concept. Instead, these parameters will be assigned to the 'maint:Deficiency-DetectionReport' concept (this concept is elaborated in sub chapter 9.2.2). A 'maint:Deficiency' instance will have a 'bs:isDescribedBy' relation with a 'maint:DeficiencyDetectionReport' instance. This relation indirectly connects the meta data to the deficiencies that have a 'bs:isDescribedBy' relation with the 'maint:DeficiencyDetectionReport'. And thus querying for this information is possible (in subchapter 9.2.2 the rationale for this design decision will be elaborated).
Although, during the interviews, multiple experts described sub categorizations of deficiency types, these subcategorizations deviated from each other, and sometimes even conflicted, therefore no specializations (sub classes) of the 'maint:Deficiency' concept are incorporated in the maintenance ontology.

9.2.2 maint:DeficiencyDetectionReport

where the 'maint:Deficiency' concept represents a deficiency itself, the 'maint:DeficiencyDetection-Report' concept represents (documentation of) the inspection (or notification) which led to the identification of a deficiency. The choice for including this concept in the ontology is because it allows for assigning meta data, that is the same for all the detected deficiencies during an inspection, only one time to the inspection, instead of assigning this information, separately, to all the deficiencies. This strategy is derived from the way in which the INTERLINK project (described in chapter 5) used to enable adding meta data that is by its origin the same for all the deficiencies an document describes.
A 'maint:Deficiency' instance will be linked to an inspection instance with the 'bs:contains' relation property (See Fig. 30 for an instance level visualization of this mechanism).

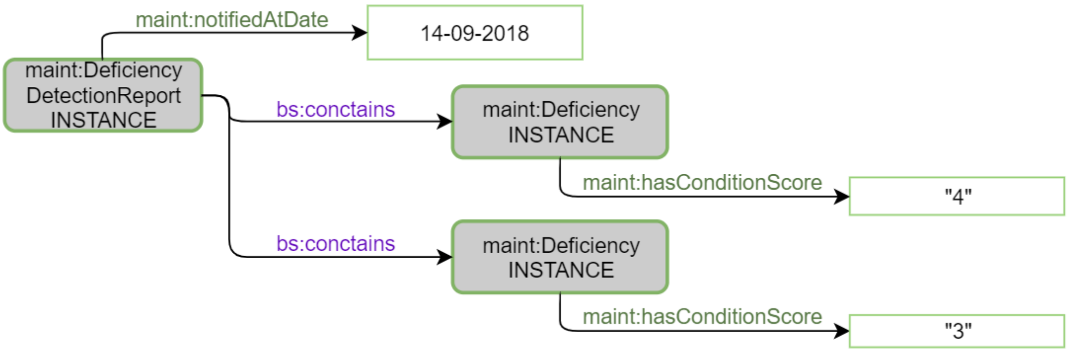

Fig. 30 Visualization of how metadata is indirectly assigned to 'maint:Deficiency' instances

As concluded in chapter 7 deficiency detection can occur in two ways; 1) during a planned inspection, conducted by a specialized inspector, and 2) when an user of the building stumbles on a deficiency and notifies this to the maintenance department. Because, these two types of deficiency detection have a different context and thus have different properties assigned to them, the 'maint:DeficiencyDetection-Report' concept is the superclass of two specializations (sub classes). Being, 'maint:Condition-MonitoringInspectionReport' for deficiencies that are detected during a planned inspection and 'maint:DeficiencyNotificationReport' for deficiencies, notified by an user of the building, see Fig. 32 for a visualization of both super and sub-classes. As shown in the figures, the properties that are the same for all the sub-classes are described at the super-class, these properties will automatically be inherited by its sub-classes, therefore they are not explicitly defined at the sub-class level.

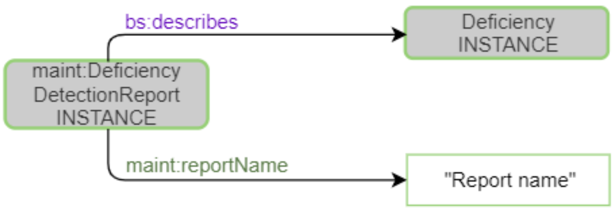

Fig. 31 'maint:DeficiencyDetectionReport' visualization

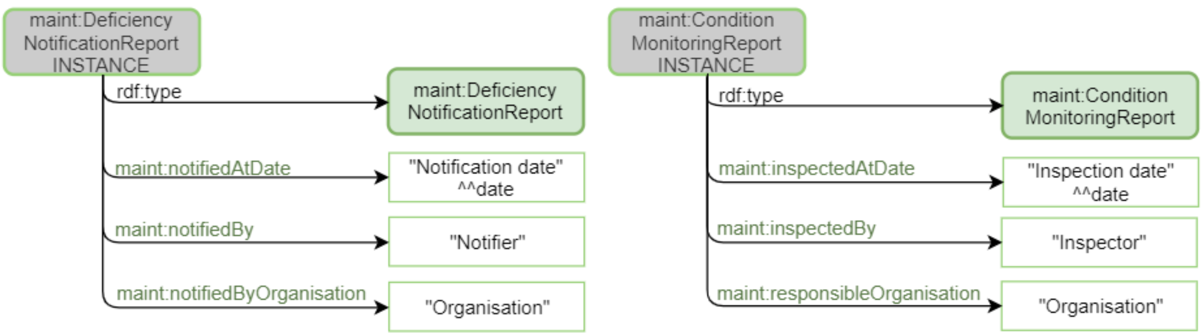

Fig. 32 Instance examples of the 'maint:DeficiencyNotificationReport' and 'maint:ConditionMeasurementReport' instance

9.2.3 maint:MaintenanceTask

The ‘maint:MaintenanceTask’ concepts represents a task that is carried out, during the maintenance and operations phase, in order to get a building object (back) in its intended condition and/or extend the duration in which it will preserve this condition.

One issue, with respect to this definition of ‘maintenance task’, is whether the replacement of a building element should be considered as a maintenance task. Although, the goal of this action is to make sure ‘there is a building element’ with the intended condition, technically on can argue that replacing an old element with a new element is something different from maintenance. During the interviews, experts at different organizations gave conflicting views with regard to this issue. However, from a practical perspective, it does not matter whether or not someone considers ‘replacement’ of building elements as maintenance. It is an activity of which the existence is necessary to be aware of, when a maintenance planner wants to validate if action on a building element is necessary. Therefore, The ‘maint:MaintenanceTask’ concept also covers replacement tasks.

Different types of maintenance indicate different things of the building status of a building element. For instance, recent replacement of a roof topping layer or newly placed layer over the old layer (over layering), indicates that the roof topping can be regarded to be in good condition (as good as new), While local repairs of roof cracks might indicate that the roof topping is in a critical condition and that more cracks can be expected soon. To enable an maintenance planning validator to quickly see which of these types of maintenance is conducted the ‘maint:MaintenanceTask’ is specialized in tree subclasses being; ‘maint:Replacement’, ‘maint:Overlayering’ and ‘maint:Repair’. This allows to query for a specific task type by specifying the subclass in the query, see Fig. 33 for a visualization of the specialization tree.

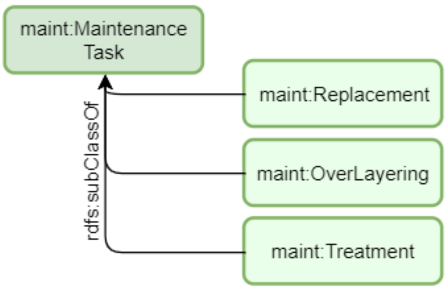

Fig. 33 Sub class visualization of the 'maint:MaintenanceTask' class

The ‘maint:Replacement’ concepts is for representing the replacement (as described above) of a building element, the ‘maint:Overlayering’ concept represents tasks that overlayer an existing element, the ‘maint:Treatment’ concept represents all the maintenance tasks that ‘treat’ the actual building element (this are all the activities that are not a replacement or over layering activity), for instance repairing a roof topping leakage.

In order to query the maintenance history of a specific building element, a ‘maint:MaintenanceTask’ instance has a ‘bs:transforms’ relationship with the building element that the task is executed on. It is also possible that a maintenance task is carried out in the direct response of a identified deficiency (for instance a leakage) in this case a ‘maint:MaintenanceTask’ instance also has a ‘bs:transforms’ relation with a ‘maint:Deficiency’ instance. This allows a maintenance planning validator to see which deficiencies are still not resolved. See Fig. 34 for a visualization of a ‘maint:MaintenanceTask’ instance.

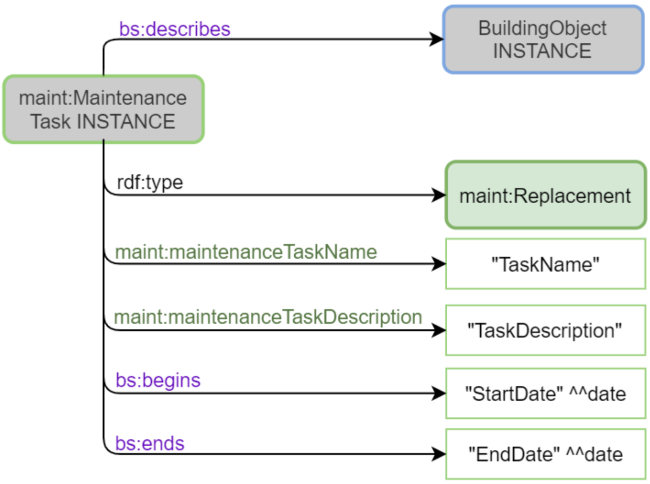

Fig. 34 'maint:MaintenanceTask' class instance example

9.2.4 maint:BuildingProject

The ‘maint:BuildingProject’ concept represents a project that creates, demolishes, changes or maintains a building. This concept is necessary because a maintenance planning validator needs to know what projects are planned related to the building, for which they are validating the maintenance planning. For instance, when a building is planned to be demolished within two years, it is not wise to conduct major maintenance tasks. Instead some minor repairs will do the job to bridge the two years (for much less costs).

A building project is related to the building it relates to with the ‘bs:transforms’ relationship concept. The most important information of a project, from a maintenance planning validation perspective, are the start and end date of the project. This information is assigned to the building project with the ‘bs:begins’ and ‘bs:ends’ properties. Additionally, properties for adding context information are assigned to the ‘maint:BuildingProject’ concept. See Fig. 35 for an instance example of the ‘maint:BuildingProject’ class concept.

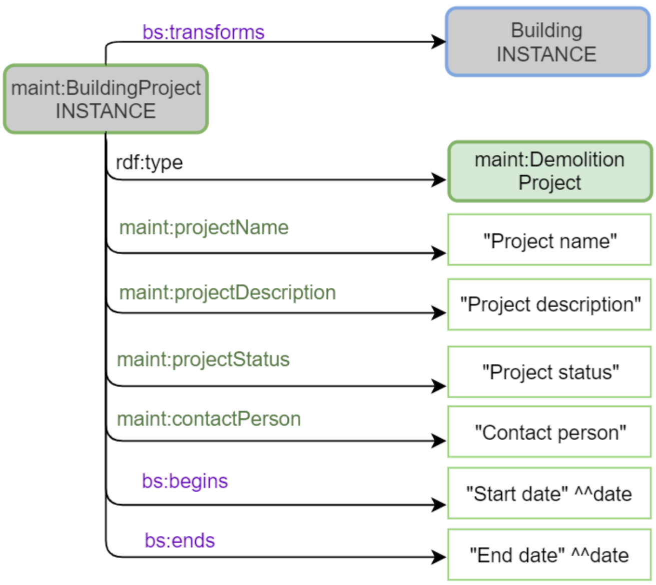

Fig. 35 'maint:BuildingProject' class instance example

9.2.5 maint:Warranty

The 'maint:Warranty' concept represents the warranty that has been given by the contractor or supplier to a building element. During the case formulation, experts indicated that the presence of warranty on a building element is relevant for validating if a maintenance task should be conducted. For instance when a building element is clearly nearing the end of its live expectancy (e.g. one or two years), but is still fulfilling the function it is designed to do. It might be wise to conduct preventive maintenance. However, in the case that warranty is still applicable on the building element for more than five years. Asset owners might consider to wait with replacing or over layering the building element, until the warranty is expired.
Warranty is related to the building element that it applies to with the 'NTA8035:describes' object property. See Fig. 36 for an instance representation of the 'maint:Warranty' concept.

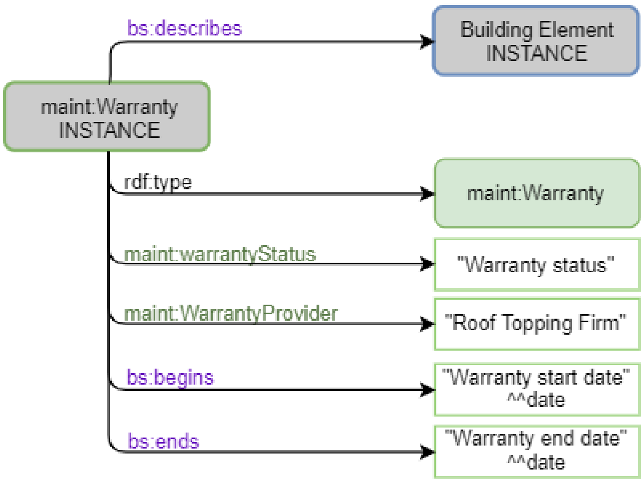

Fig. 36 'maint:Warranty' class instance example

9.2.6 maint:DesiredConditionValue

The 'maint:DesiredConditionValue' concept allows for describing the desired condition value that (an asset management department of) an organization decides to be applicable to an building element. The desired condition value is attached to the building element that it applies to with the use of the 'bs:describes' object property. This allows to query for the desired condition value of an building element and compare it with the actual condition value of that building element (according to the last inspection). See Fig. 37 for an instantiation of the 'maint:DesiredConditionValue' concept.

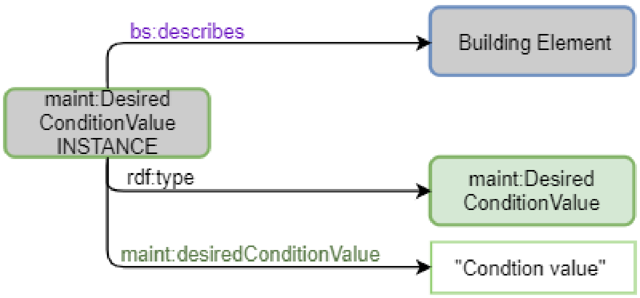

Fig. 37 'maint:DesiredConditionValue' class instance example

9.2.7 maint:StrategicInformation

The ‘maint:StrategicInformation’ concepts allows for strategic information to be expressed and related to a building instance. Examples of strategic information are selling plans or a sustainability strategy that an organization has with respect to its buildings. By allowing interconnecting this strategic information with a building instance in an IFC-model it is possible to query directly which strategic information is relevant wen validating the need for an certain maintenance task. The connection is made with the use of the bs:describes relation. See Fig. 38 for an instantiation of the ‘maint:StrategicInformation’ concept.

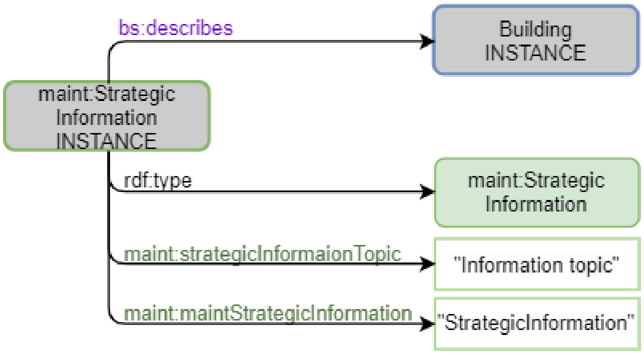

Fig. 38 'maint:StrategicInformation' class instance example

10 Linking

The main goal of the ontology is to support efficient linking of information points that reside in different information sources. Linking of information can take place on two levels;

1. The model/ontology level; this means linking concepts of different ontologies with each other (linking data models with each other).
2. The instance level; This means linking instance information from different information sources with each other (linking data sets with each other).

Linking of information can take place in two ways (NEN NC 381184 Working group, 2019);

1. Query linking; this is done by describing a link within a SPARQL query.
2. Link set linking; this is done by modeling the links as triples where the subject and object are represented by the concepts that need to be linked, the relation between the concepts is described by an object property, the collection of triples which link different ontologies/data sets with each other is called a 'link set'.

The use of the second linking method can actually be identified in the maintenance ontology, the link set in this case is the collection of all the triples that connect the maintenance ontology to 'a' building ontology, see Fig. 39 where the link set is demarcated in the purple block. However, it must be said that the way in which the maintenance ontology is technically linked to a building ontology is not exactly the same as visualized in Fig. 39. For example, the 'maint:Deficiency' class-concept is not actually is the condition of a 'BuildingElement' concept. Fig. 39 visualizes how instances of these class-concepts would be linked. For instance 'a' Deficiency actually is the condition of 'a' Building element.

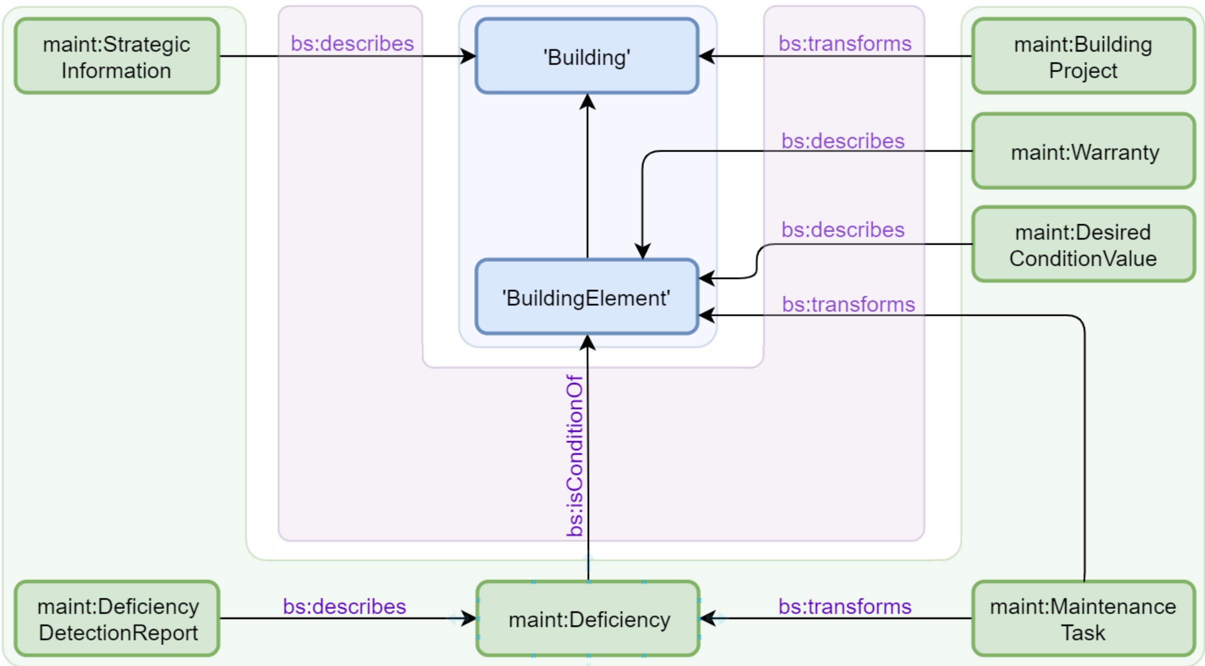

Fig. 39 Maintenance Ontology with conceptual link set demarcated in purple

In order to link ontology level concepts the use of OWL restrictions, prescribed by the W3C, is used. These restrictions will be assigned as a subclass (rdfs:subClassOf) to the classes that makes use of NTA-8035 relation properties. See sub chapter 9.1.5.3 for full explanation of this mechanism (see Table 13

for an example of this mechanism, visualized in turtle syntax). See appendix-B for the collection of all these triples which together form the link set between the Maintenance Ontology and a building ontology (more precise the 'Building' and 'BuildingElement' concept).
The link set that links the maintenance ontology to a building ontology is an example of ontology level linking. When concepts from different ontologies are linked (e.g. 'maint:Deficiency' concept with 'building element' concept', see Table 13), it becomes possible to query for these relations. Thus, when multiple ontologies say something about the BuildingElement concept, all these relations can be queried. This, dynamic, gives potential users of, data that is created according to these ontologies, an efficient and fast method for understanding the data and knowing what is possible with the data.

However, the fact that concepts are linked on an ontology level, does not automatically link instantiations of these concepts to each other (in other words, the fact that 'the' concept 'maint:Deficiency and 'the' concept 'BuildingElement' are linked, with the use of an OWL-restriction, does not mean that a computer automatically understands that 'this' deficiency relates to 'that' Building Element'). In order to enable querying for information in different sources that describe the same instance, action that makes the relation explicit on an instance level needs to be taken as well.

In the validation experiments (described in the next chapter) linking, of instance level information, is done by using the same URI for an instance, in different information sources, that say something about that instance. From a process perspective this means that the URI, generated when an instance is described(/created) for the first time, needs to be reused in all the information sources, that also describe something about that instance, created after the first information source.
In the case of the validation experiment this means that URI's created during the conversion of the IFC-model into RDF-data need to be reused in the maintenance information sources that also want to describe something about these building elements. From a process perspective, this linking strategy is feasible (also when taking maintenance information flows in general into account) because the IFC-model by default is the first information source that describes building elements in the maintenance phase since it is part of the information handover delivery, during the commissioning of a new building. With respect to the two linking methods, described above, the used instance level linking strategy can be interpreted as a variant of the link set method whereby the link set is integrated in the data source. See Fig. 40 for a graph visualization of a link between 'a' deficiency instance and 'a' BuildingElement instance.

Fig. 40 'bs:isConditionOf' link between a 'maint:Deficiency' and 'BuildingElement' instance

11 Ontology Validation

This chapter presents the setup and results of the validation experiments that where conducted to validate the designed Maintenance Ontology. The validation experiments where executed in the form of single-case mechanism experiments. The goal of the experiments was to validate if it was possible to query all the information points (from multiple information sources) that the experts considered to be relevant for supporting the maintenance related task that they formulated.

A single-case mechanism experiment is a test of a *single case* in which the researcher applies *stimuli* to the case and explains the *responses* in terms of mechanisms internal to the case (Wieringa, 2014). In this experiment the *case* is the linking of multiple information sources with the use of the designed maintenance ontology and subsequently proofing that, with the use of SPARQL queries, the information points of interest within these sources (described in Table 10 and Table 12) can be extracted from multiple sources and represented in one place. The *stimuli* in this experiment are the SPARQL queries and the Linked Data conversions of the information sources in line with the designed maintenance ontology, described in chapter 9. The *responses* in this experiment are the results that the SPARQL queries generate when being executed on the linked data conversions of the source documents that Eigen Haard and TU-Delft provided.

Sub-chapter 11.1 describes the setup of the single case mechanism experiments and sub chapter 11.2 presents the results of the single case mechanism experiments, per case (described in chapter 8).

11.1 Experiment setup

11.1.1 Converting the sources into Linked Data

From a data perspective there are three source types (data formats) that need to be converted; the IFC-models, Excel files and PDF files. For converting IFC-models into linked data, two converters have been identified; 1) the IFC-to-RDF converter[18], developed by Pauwels (2011) and, 2) the IFC-to-LBD converter[19] developed by Bonduel et al (2018). The difference between these two converters is that the IFC-to-RDF converter generates a turtle file[20] based on the ifcOWL ontology and the IFC-to-LBD converter generates a turtle file based on the BOT ontology. In theory the IFC-to-LBD converter would be a better choice for exposing IFC files as Linked Data because it generates a linked data conversion of a IFC-file that allows for easier query writing in comparison to the IFC-to-RDF converter. However, when experimenting with both converters it was concluded that a lot of the custom property sets in the IFC-models would get lost when using the IFC-to-LBD converter. For this reason the IFC-to-RDF converter was used in the experiments.

For exposing the data in the Excel files Python scripts are made which enabled automatic conversion of the Excel documents. Appendix-C presents one of these scripts, the structure of all other scripts uses the same libraries and has the same structure. The PDF documents, that were given by Eigen Haard and TU-Delft do not contain a, computer interpretable, data structure therefore the PDF sources where manually transformed into Excel documents, from here Python scripts where used the generate RDF data. RDF data can be serialized in multiple syntactical forms. In this research Turtle is chosen because it is the form that presents the data in a form, that is easiest interpretable for humans.

[18] The IFC-to-RDF converter is available here; https://github.com/pipauwel/IFCtoRDF.
[19] The IFC-to-LBD converter is available here; https://github.com/w3c-lbd-cg/IFCtoLBD.
[20] Turtle (.ttl) is one of the syntaxes in which RDF data can be represented.

11.1.2 Executing the queries

In order to execute the queries the turtle files were stored in a triple store (database for RDF-data). As triple store the opensource triple store 'Apache Jena Fuseki Server'[21] (version 3.9.0) was used. All the files where uploaded as named graphs in the database, this enables to specify (certain parts of) queries only to focus (process) on the triples in specific named graphs. To query the data in the triple store the interface that the Apache Jena Fuseki Server provides was used. The interface also shows the results of a query in a clear tabular format, see Table 14 for an representation of one of the queries with subsequent query results (See appendix-D all the queries and the query results).

Table 14 Query 5 with query results

Query

```
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX maint: <https://maintenanceontology.org/maint#>
PREFIX nta: <https://NTA8035.org/nta#>
PREFIX instifc: <http://linkedbuildingdata.net/ifc/resources/20191018_114250/>
PREFIX instmaint: <http://EWI/ConditieMeting.tudelft/>

SELECT  ?BuildingObjectInstance ?Deficiency ?Intensity ?Scale ?ConditionScore
WHERE {
 ?BuildingObjectInstance ?b ?Type .
 ?BuildingObjectInstance ?b ifcowl:IfcRoof .

 GRAPH <http://localhost:3030/IFC-TUdelft/data/ConditieMeting_11>{
 ?Deficiency nta:isConditionOf ?BuildingObjectInstance ;
         maint:hasIntensity ?Intensity ;
         maint:hasScale ?Scale ;
         maint:hasConditionScore ?ConditionScore .}
 }
```

Query results

BuildingObjectInstance	Deficiency	Intensity	Scale	Cond
instifc:IfcRoof_340	instmaint:Gebrek_23	"Gevorderdstadium"	"Plaatselijk"	"4.0"
instifc:IfcRoof_340	instmaint:Gebrek_24	"Eindstadium"	"Aanzienlijk"	"4.0"
instifc:IfcRoof_340	instmaint:Gebrek_25	"Eindstadium"	"Plaatselijk"	"3.0"
instifc:IfcRoof_635	instmaint:Gebrek_12	"Gevorderdstadium"	"Plaatselijk"	"2.0"
instifc:IfcRoof_635	instmaint:Gebrek_13	"No intensity has been specified"	"No size has been specified"	"2.0"
instifc:IfcRoof_13946	instmaint:Gebrek_6	"Gevorderdstadium"	"Regelmatig"	"1.0"
instifc:IfcRoof_22636	instmaint:Gebrek_19	"Gevorderdstadium"	"Plaatselijk"	"4.0"
instifc:IfcRoof_22636	instmaint:Gebrek_20	"Eindstadium"	"Regelmatig"	"4.0"
instifc:IfcRoof_22636	instmaint:Gebrek_21	"Eindstadium"	"Incidenteel"	"1.0"
instifc:IfcRoof_22636	instmaint:Gebrek_22	"No intensity has been specified"	"No size has been specified"	"1.0"

[21] The Apache Jena Fuseki server is available here: https://jena.apache.org/download/index.cgi

11.2 Experiment results

Down below the results are presented per experiment. Per case a table shows the results of the queries. In these tables each information point, described in Table 10 and Table 12, serves as a competency question. For each competency question the table describes if the query was successful, how many triples where necessary to retrieve the desired information and a reference to the query in the appendix is given. If a competency question could not be retrieved the table also shows an explanation for why this was the case.

11.2.1 Experiment results; Case 1; TU-Delft

Table 15 shows the results of the Case described by TU-Delft;

Table 15 Experiment results of Case 1 described by the maintenance department of TU-Delft

Competency Question	Query successful	Ontology	Appendix reference	Failure explanation
1. Roof surface Area (m^2)	Yes	ifcOWL	Query 1	n/a
2. Roof edges (m^1)	No	ifcOWL	n/a	The information was not queryable, because it was not properly added as a quantity property in the IFC-model. Theoretically it is possible to query it directly form the geometry representation. However, this is very complex and therefore not done in this study.
3. Roof finishing type	Yes/No	ifcOWL	Query 2	There was a property, present in the IFC-model, that described something with regard to the roof finishing type, However it was not a proper description.
4. Warranty (yes/no)	Yes	Maint	Query 3	n/a
5. Warranty end date	Yes	Maint	Query 3	n/a
6. Warranty provider	Yes	Maint	Query 3	n/a
7. Deficiency type	Yes	Maint	Query 4	n/a
8. Deficiency description	Yes	Maint	Query 4	n/a
9. Intensity of deficiency	Yes	Maint	Query 5	n/a
10. Scale of deficiency	Yes	Maint	Query 5	n/a
11. Condition score of deficiency	Yes	Maint	Query 5	n/a
12. Aspect type of deficiency	Yes	Maint	Query 6	n/a
13. Date of identification	Yes	Maint	Query 7, Query 8	n/a
14. Name inspector	Yes	Maint	Query 7, Query 8	n/a

15. Name Organization responsible for the report	Yes	Maint	Query 7	n/a
16. Date defect (and repair)	Yes	Maint	Query 8	n/a
17. Deficiency type	Yes	Maint	Query 8	n/a
18. Name notifier	Yes	Maint	Query 8	n/a
19. Type of repair	Yes	Maint	No	The column assigned for this information point was not filled in, so no useful information could be queried.
20. Planned client/owner desired projects on the building	Yes	Maint	Query 9	n/a
21. Start date of the project	Yes	Maint	Query 9	n/a
22. End date of the project	Yes	Maint	Query 9	n/a
23. Type of project	Yes	Maint	Query 9	n/a
24. Priority related to aspect of an deficiency	Yes	Maint	Query 6	n/a
25. Desired condition level of the building object	Yes	Maint	Query 10	n/a

11.2.2 Experiment results; Case 2; Eigen Haard

Table 16 shows the results of the Case described by Eigen Haard;

Table 16 Experiment results of Case 1 described by the maintenance department of Eigen Haard

Competency Question	Query successful	Ontology	Appendix reference	Failure explanation
26. Roof surface Area (m^2)	Yes	ifcOWL	Query 11	n/a
27. Roof edges (m^1)	No	ifcOWL	n/a	The information was not queryable, because it was not properly added as a quantity property in the IFC-model. Theoretically it is possible to query it directly form the geometry representation. However, this is very complex and therefore not done in this study.
28. Roof finishing type	Yes	ifcOWL	Query 11	n/a
29. Roof outlets (nr.)	No	ifcOWL	n/a	The information was not queryable, because there was no explicit relation between the roof and the roof outlets. The reason for

				this is that the roof outlets where only modeled on-top of the roof topping instead of growing through the roof topping, which would create roof opening relation.
30. Roof height	No	ifcOWL	n/a	The information was not queryable, because it was not properly added as a quantity property in the IFC-model. Theoretically it is possible to query it directly form the geometry representation. However, this is very complex and therefore not done in this study.
31. NL-SfB code	Yes	ifcOWL	Query 11	n/a
32. Condition score	Yes	Maint	Query 12	n/a
33. Inspector	Yes	Maint	Query 12	n/a
34. Responsible organization	Yes	Maint	Query 12	n/a
35. Desired condition score	Yes	Maint	Query 12	n/a
36. Notified deficiencies	Yes	Maint	Query 13	n/a
37. Type of notified deficiency	Yes	Maint	Query 13	n/a
38. Date of notified deficiency	Yes	Maint	Query 13	n/a
39. Planned counter measure for notified deficiency	Yes	Maint	Query 13	n/a
40. Status of the counter measure	Yes	Maint	Query 13	n/a
41. Warranty status	Yes	Maint	Query 11	n/a
42. Warranty provider	Yes	Maint	Query 11	n/a
43. Project name	Yes	Maint	Query 14	n/a
44. Project type	Yes	Maint	Query 14	n/a
45. Project description	Yes	Maint	Query 14	n/a
46. Project status	Yes	Maint	Query 14	n/a
47. Project start date	Yes	Maint	Query 14	n/a
48. Selling plans	Yes	Maint	Query 15	n/a
49. Sustainability plans	Yes	Maint	Query 15	n/a
50. Owner occupation in complex	Yes	Maint	Query 15	n/a

PART-V CONCLUSION

In this part of the report the conclusion, discussion and future work directions derived from this research are presented.

12 Conclusion

Main conclusion;
The goal of this research was to:

Improve Semantic Web and Linked Data approaches for interconnecting and querying, building-related, distributed maintenance data, by developing and validating a maintenance ontology.

Based on the research results it can be concluded that the designed Maintenance Ontology proved to be effective in interconnecting and querying the information (residing in multiple sources) that experts, from the maintenance departments of Eigen Haard and TU-Delft, considered to be relevant for supporting the maintenance related task that they formulated for the validation experiments. Being; validating the need for planned maintenance tasks, based on the MYMP.
Based on the results of the single case mechanism experiments (presented in sub-chapter 11.2), it can be concluded that the Maintenance Ontology is able to express all the information, that the experts mentioned to be relevant for supporting the task described in chapter 8, in a way that allows it to be queried automatically from multiple information sources.
Although, some competency questions could not be queried successfully. The reason was not related to the expressivity capabilities of the Maintenance Ontology;

- Competency question 3 was not successful because the desired information was not present in the IFC-model.
- Competency question 19 was not successful because the desired information was not present in the information source that should contain it.
- Competency question 2, 27, 29 and 30 were not successful because the information was only present in the ifcOWL-models, in the form of a 3D-geometric description of the roof topping element. The way in which this geometry is described in ifcOWL proofed to be too difficult to write automated queries for (although in theory it is possible).

Main design decisions;
The design decision that has the biggest influence on the design of the Maintenance Ontology is to design it with the intend to be used as a module that can be attached to a Building Ontology, or to be more precise, as a module that can be attached to an ontology that contains a class concept for 'Building' and 'Building Element'. Translated to instance information, expressed according to these ontologies, it means that all the information that can be expressed with the Maintenance Ontology always relates to specific 'Building' and / or 'Building Element' instances. The reason for this decision is that based on the interviews this is exactly wat is the case; maintenance related information always relates to a specific Building Element (e.g. condition monitoring information, maintenance history, warranty information) or to a building as a whole (e.g. strategic information, building plans).
The dependence of the Maintenance Ontology with respect to the 'Building' and 'Building Element' concept is also crucial for the linking strategy that is used in order to interconnect the information in

the different information sources. Because, all the links that need to be made explicit, is from maintenance related information to the building or building element that they describe.
Therefore the URI's, generated when a building or building element is generated for the first time needs to be reused in all the information sources, that also describe something about that instance, created after the first information source. From a process perspective, this linking strategy is feasible (also when taking maintenance information flows in general into account) because the IFC-model by default is the first information source that describes building elements in the maintenance phase since it is part of the information handover, during the commissioning of a new building.

13 Discussion

Barriers;
The goal of this research was to interconnect information from multiple sources in order to enable automated querying of that information, relevant for supporting maintenance related tasks. Upfront of this research the assumption was that all this information would already be collected properly and thus be present. However, an unintended conclusion, that can be derived from this research, is that information is not always collected (properly) and therefore also not always present. Based on the interviews it can be concluded that the reason for not having the desired information available has two main reasons; 1) Maintenance departments have no protocol in place that explicitly formulates how and what information should be collected/generated. As a result information is being generated/collected on an ad-hoc basis, whereby the creator of the information decides on structure and syntax of the information on the fly. As a result information is not present in a standardized way which makes it hard for maintenance planners to find and/or understand the information.
2) From an organizational perspective it was also identified that workflows with respect to data handover, are not structured optimal in order to facilitate the goals of the organization. The department that is assigned with the responsibility of collecting the handover information (the project department) is not the same ash the organization that will use this information (maintenance department). The problem here is that the project department is unaware of the exact information desires of the maintenance department (or reluctant to listen to these desires). As a result the maintenance department is not receiving all the information it desires from the handover process.
The implication of this conclusion with regard to the research is that information that is not present can also not be linked or queried no matter what technology is used.

As mentioned earlier the Maintenance Ontology is designed to be used as a model in relation with an ontology containing a concept for representing 'Building' and 'Building Element' instances. In this research the ifcOWL ontology was used for this purpose. With respect to building elements, this means that their geometry was also represented according to the ifcOWL ontology. Some competency questions can theoretically be queried based on the geometry. However, during the research it was concluded that this was too complex based on the way that the geometry-data is modeled in ifcOWL. Although, these barriers prevented some competency questions to be queried unsuccessful. The cause for this is not related to the design of the Maintenance Ontology.

Opportunities;
This research proofed that the Maintenance Ontology is capable of interconnecting and querying information residing in multiple sources, with respect to supporting the specific maintenance related task that was at the center of this research. Fundamental reasons for this success are inherent to the use of SWLD-technology. Being;

- The fact that, with SWLD-technology, resources (e.g. building elements or deficiencies) are identified by a URI ('Unique' Resource 'Identifier') enables an information management architecture where multiple information sources assign information to the same resource by referring to the URI that represents that resource.
- When information is made available in the form of RDF-data, the SPARQL query language (as part of SWLD-technology stack) can be used for querying information from multiple sources, provided that necessary information is properly stored and links are properly defined.

These two fundamentals imply that the strategy used to interconnect and query information in this research, with respect to the specific use case, is scalable and can be used to interconnect and query information, relevant for supporting maintenance related tasks, in general. This makes an information management system based on SWLD-technology a promising value proposition.

Next to the fact that SWLD-principles proofed to be effective in interconnecting maintenance related information, the ontologies that play a major part in the interconnecting process (in the case of this research the Maintenance Ontology) can also serve as a tool that communicates how and what information should be collected. This can fill the gap that is currently present at maintenance departments with respect to the missing protocols that explicitly describe how and what information needs to be collected with respect to information needs within maintenance departments.
Additionally this overcomes the need of custom data conversion tools that was the case in this research; being, a specific Python script for each document.

14 Future work

The aim of this research was to proof the use of a SWLD-technology based ontology for interconnecting and querying distributed information, relevant for maintenance related tasks. The results imply that SWLD-technology can remedy the problem that maintenance departments currently encounter with respect to disconnected information sources (see sub-chapter 1.1 for a full description of the problem). However, to keep the size of the research realistic with respect the six-month time period, the validation scope of the research was limited to one specific task. In order to improve the validity of the ontology as a tool for interconnecting and querying information, relevant for maintenance related tasks, in general. Future research needs to validate the functionality of the ontology for more maintenance related tasks.

References

Allemang, D., & Hendler, J. (2011). Semantic Web for the Working Ontologist. *Elsevier*.

Ashar, S. (2011). Building information modeling (BIM):Trends, benefits, risks and challenges for the AEC industry. *Leadership and management in engineering*.

Becerik-Gerber, B., Jazizadeh, F., Li, N., & Calis, G. (2012). Application Areas and Data Requirements for BIM-Enabled Facilities Management. *Journal of Construction Engineering and Management*, 431-442.

Becker, R., Falk, V., Hoenen, S., Loges, S., Stumm, S., Blankenbach, J., . . . Vallee, D. (2018). BIM - Towards the entire lifecycle. *Int. J. Sus. Dev. Plann. Vol. 13, No.1*, 84-95.

Beetz, J., & Krijnen, T. (2018). A SPARQL query engine for binary-formatted IFC building models. *Automation in Construction*, 46-63.

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The Semantic Web. *Scientific American*, 676-680.

Bonduel, M., Oraskari, J., Pauwels, P., Vergauwen, M., & Klein, R. (2018). The IFC to Linked Building Data Converter - Current Status*. *Proceedings of the 6th Linked DATA in Architecture and Construction Workshop.* London.

Bonduel, M., Vergauwen, M., Klein, R., Rasmussen, M., & Pauwels, P. (2018). A novel workflow to combine BIM and Linked Data for existing buildings.

buildingSMART. (2019, 11 21). *Standards Library*. Retrieved from buildingsmart.org: https://www.buildingsmart.org/standards/bsi-standards/standards-library/

Cavka, H., Staub-French, S., & Poirier, E. (2017). Developing owner information requirements for BIM-enabled project delivery and asset management. *Automation in Construction*, 169-183.

CEDR. (2019). *INTERLINK*. Retrieved from cedr: https://www.cedr.eu/strategic-plan-tasks/research/cedr-call-2015/call-2015-asset-information-using-bim/interlink/

CIC (Computer Integrated Construction). (2013). *BIM Planning Guide for Facility Owners.* University Park, PA, USA: The Pennsylvania State University.

Deng, Y., Cheng, J., & Anumba, C. (2016). Mapping between BIM and 3D GIS in different levels of detail using schema mediation and instance comparison. *Automation in Construction*, 1-21.

Eastman, C., Teicholz, P., Sacks, R., & Liston, K. (2008). BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors. *Wiley*.

Farghaly, K., & Wood, G. (2017). BIM for Asset Management: A taxonomy of non-geometric BIM data for asset management. *24th EG-ICE International Workshop on Intelligent Computing in Engineering 2017* (pp. 96-105). Notthingham; United Kingdom: University of Nottingham.

Farghaly, K., Abanda, F., Vidalakis, C., & Wood, G. (2018). Taxonomy for BIM and Asset Management Semantic. *Journal of Management in Engineering*.

Feigenbaum, L., Herman, I., Hongsermeier, T., Neumann, E., & Stephens, S. (2010). The Semantic Web in Action. *Scientific American*.

INTERLINK consortium. (2019). *Home*. Retrieved from roadotl: https://www.roadotl.eu/

Juan, D. (2013). The research to open BIM-based building information interoperability framework. *2013 2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation (IMSNA).* Toronto: IEEE.

Khaja, A., McArthur, J., & Seo, J. (2016). Optimizing BIM metadata manipulation using parametric tools. *Procedia Engineering*, 259-266.

Kim, K., Kim, H., Kim, W., Kim, C., Kim, J., & Yu, J. (2018). Integration of ifc objects and facility management work information using semantic web. *Automation in construction 87*, 173-187.

Koch, C., Vonthron, A., & Konig, M. (2016). A tunnel information modelling framework to support management, simulations and visualisations in mechanised tunneling projects. *Automation in Construction 83*, 78-90.

Mayo, G., & Issa, R. (2015). Nongeometric Building Informaiton Needs Assessment for Facilities Managements. *Journal in <anagement in Engineering*.

NTA-8035 NEN NC 381184 Working group. (2019). *NTA Semantische Gegevensmodellering en - Integratie in de Gebouwde Omgeving, concept versie 7 Augustes 2019.*

Noy, N., & McGuinness, D. (2001). Ontology Development 101: A Guide to Creating Your First Ontology.

Patacas, J., Dawood, N., Vukovic, V., & Kassem, M. (2015). BIM for facilities management: evaluating BIM standards in asset register creation and service life planning. *ITcon*.

Pattuelli, M., Provo, A., & Thorsen, H. (2015). Ontology Building for Linked Open Data: A Pragmatic Perspective. *Journal of Library Metadata*, 265-294.

Pauwels, P. (2018). *ifcOwl*. Retrieved from openbimstandards.org: http://openbimstandards.org/standards/ifcowl/

Pauwels, P., & Deursen, D. (2012). IFC/RDF: Adaptation, Aggregation and Enrichment. *LDAC 2012*, 1-3.

Pauwels, P., & Terkay, W. (2016). EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology. *Automation in Construction 63*, 100-133.

Pauwels, P., De Meyer, R., & Van Campenhout, J. (2011). Interoperability for the design and construction industry through semantic web technology. *Semantic Multimedia*, 143-158.

Rasmussen, M., Lefrancois, M., Pauwels, P., & Hviid, C. (2019). Managing interrelated project information in AEC Knowledge Graphs. *Automation in Construction 108*, 1-16.

Rasmussen, M., Pauwels, P., Hviid, C., & Karlshoj, J. (2017). Proposing a Central AEC Ontology that allows for domain specific extensions. *Proceedings of the Joint Conference on Computing in Construction* (pp. 237-244). Heraklion: DOI.

Sadeghi, M., Mehany, M., & Strong, K. (2018). Integrating Building Information Models and Building Operation Information Exchange Systems in a Decision Support Framework for Facilities Management. *Construction Research Congress 2018: Infrastructure and Facility Management* (pp. 770-779). New Orleans; United States: American Society of Civil Engineers (ASCE).

Schwabe, K., Dichtl, M., Konig, M., & Koch, C. (2018). COBie: A Specification for the Construction Operations Building Information Exchange. In A. Borrmann, M. Konig, C. Koch, & j. Beetz, *Building Information Modeling* (pp. 167-180). Springer.

Thabet, W., & Lucas, J. (2017). A6-step systematic process for model-based facility data delivery. *ITcon*.

Thabet, W., & Lucas, J. (2017). Asset data handover for a large educational institution: case-study approach. *American society of civil engineers*.

Theiler, M., Dragos, K., & Smarsly, K. (2018). Semantic description of structural health monitoring algorithms using building information modeling. *LNCS*, 150-170.

W3C. (2004, February 10). *Resourc Description Framework (RDF): Concepts and Abstract Syntax.* Retrieved from W3C: https://www.w3.org/TR/rdf-concepts/

Wang, Y., Wang, X., wang, J., Yung, P., & Jun, G. (2013). Engagement of Facilities Management in Design Stage through BIM: Framework and a Case Study. *Advances in Civil Engineering*, 1-8.

Wieringa, R. (2014). *Design Science Methodology for Information Systems and Software Engineering.* Enschede: Springer.

Wix, J., & Liebich, T. (2008). Using constraints to validate and check building information models. In A. Zarli, & R. Scherer, *eWork and eBusiness in Architecture, Engineering and Construction* (pp. 467-475). Sophia Antipolis, France: Taylor & Francis Group.

Zhang, J., Chen, Y., Liu, R., & Debs, L. (2019). Cartesian Points Visualization in Game Simulation for Analyzing Geometric representations of AEC Objects. *36th International Symposium on*

Automation and Robotics in Construction (pp. 144-151). Banff, Alberta, Canada: The International Association for Automation and Robotics in Construction.

Appendix

Appendix-A Interview Guide

Background questions
Q0.1-EN; Can you describe the organization that u work at? *Q0.1-NL; Kunt u de organisatie omschrijven waarbinnen u werkzaam bent?*
Q0.2-EN; Can you describe your function within the organization? *Q0.2-NL; Kunt u uw functie binnen de organisatie omschrijven?*
Q0.3-EN; Can you describe the tasks/responsibilities that your organization has with respect to maintenance? *Q0.3-NL; Kunt u uitleggen welke taken/verantwoordelijkheden uw organisatie heeft met betrekking tot onderhoud?*
Q0.4-EN; Can you describe the organizational structure, relevant with respect to maintenance? *Q0.4-NL; Kunt u de organisatiestructuur omschrijven die relevant is met betrekking tot onderhoud?*
Substantive questions
Q1.1-EN; Do you have certain (standardized) processes or protocols in place with respect to managing and conducting maintenance tasks, if so which? *Q1.1-NL; Maakt u gebruik van (gestandaardiseerde) processen of protocollen met betrekking tot het genereren van informatie benodigd voor het plannen, voorbereiden en uitvoeren van onderhoud, zo ja welke?*
Q1.2-EN; Wo creates/generates the information, relevant for maintenance related tasks and in which phase is the information created. *Q1.2-NL; Wie creëert/genereerd de informatie, noodzakelijk voor het plannen, voorbereiden en uitvoeren van onderhoud. En in welke fase wordt de informatie gegenereerd?*
Q1.3-EN; Do you use certain (standardized) data models or standards with respect to information capture, if so which? *Q1.3-NL; Maakt u gebruik van (gestandaardiseerde) datamodellen of schema’s voor de representatie en opslag van informatie, zo ja welke?*
Q1.4-EN; How is the information created? *Q1.4-NL; Hoe is de informatie gecreëerd?*
Q1.5-EN; How is the information stored and used? *Q1.5-NL; Hoe wordt de informatie opgeslagen en gebruikt?*

Appendix-B Maintenance Ontology

Prefixes

	Namespace	*URI*
@prefix	rdf:	<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@prefix	rdfs:	<http://www.w3.org/2000/01/rdf-schema#>
@prefix	xsd:	<http://www.w3.org/2001/XMLSchema#>
@prefix	owl:	<http://www.w3.org/2002/07/owl#>
@prefix	dcterms:	<http://purl.org/dc/terms/>
@prefix	vann:	<http://purl.org/vocab/vann/>
@prefix	bs:	The NTA Top Level Conceptual Model, not published online, the following imaginary URI will be used; <https://NTA8035.org/nta#>
@prefix	maint:	The designed Maintenance Ontology is not officially published online, the following URI will be used; <https://maintenanceontology.org/maint#>

Metadata

Metadata
<https:w3id.org/maint#> rdf:type owl:Ontology ;
dcterms:modified "2019-11-30T12:00:00"^^xsd:dateTime ;
owl:versionInfo "1.0" ;
dcterms:title "Maintenance Ontology "@en ;
dcterms:creator "Marc Grooters" ;
vann:preferredNamespacePrefix "maint" ;
vann:preferredNamespaceUri <https://w3id.org/maint#> .

Classes

Subject	*Predicate*	*Object*
maint:Deficiency	rdf:type	owl:Class ;
	rdfs:subClassOf	bs:State ;
	skos:definition	"a change in a building object status for the worse"@en ;
	skos:prefLabel	"Deficiency"@en ;
	skos:altLabel	"Gebrek"@nl .

Subject	*Predicate*	*Object*
maint:Deficiency-DetectionReport	rdf:type	owl:Class ;
	rdfs:subClassOf	bs:InformationObject , [a owl:Restriction ; owl:onProperty nta:contains ; owl:allValuesFrom maint:Deficiency] ;
	skos:definition	"Document that describes identified deficiencies"@en ;
	skos:prefLabel	"Deficiency detection report"@en ;
	skos:altLabel	"Conditie meting rapport"@nl .

Subject	*Predicate*	*Object*
maint:Condition-MeasurementReport	rdf:type	owl:Class ;
	rdfs:subClassOf	bs:InformationObject , maint:DeficiencyDetectionReport ;
	skos:definition	"Document that describes deficiencies that are detected during a planned inspection"@en ;
	skos:prefLabel	"Deficiency detection report"@en ;
	skos:altLabel	"Gebrek identificatie rapport"@nl .

Subject	*Predicate*	*Object*
maint:Deficiency-NotificationReport	rdf:type	owl:Class ;
	rdfs:subClassOf	bs:InformationObject , maint:DeficiencyDetectionReport ;
	skos:definition	"Document that describes deficiencies notified by a user of the building"@en ;
	skos:prefLabel	"Deficiency notification report"@en ;
	skos:altLabel	"Gebrek notificatie rapport"@nl .

Subject	*Predicate*	*Object*
maint:Maintenance-Task	rdf:type	owl:Class ;
	rdfs:subClassOf	bs:Activity , [a owl:Restriction ; owl:onProperty nta:transforms; owl:allValuesFrom maint:Deficiency] ;
	skos:definition	"a task that is carried out in order to keep or get the building in its original condition"@en ;
	skos:prefLabel	"Maintenance task"@en ;
	skos:altLabel	"Onderhoudstaak"@nl .

Subject	*Predicate*	*Object*
Maint:Desired-ConditionValue	rdf:type	owl:Class ;
	rdfs:subClassOf	bs:State ;
	skos:definition	"a condition value that an organization desires that its building element have"@en ;
	skos:prefLabel	"Desired condition value"@en ;
	skos:altLabel	"gewenste conditie waarde"@nl .

Subject	*Predicate*	*Object*
maint:Warranty	rdf:type	owl:Class ;
	rdfs:subClassOf	bs:InformationObject ;
	skos:definition	“a written guarantee, issued to the purchaser of an product or service by its manufacturer or service provider, promising to repair or replace it if necessary within a specified period of time.”@en ;
	skos:prefLabel	“Warranty”@en ;
	skos:altLabel	“Garantie”@nl .

Subject	*Predicate*	*Object*
maint:Building-Project	rdf:type	owl:Class ;
	rdfs:subClassOf	bs:Activity ;
	skos:definition	“A plan for creating, changing or demolishing a (part of a) building”@en ;
	skos:prefLabel	“Building project”@en ;
	skos:altLabel	“Bouwproject”@nl .

Properties

Subject	*Predicate*	*Object*
maint:notifiedAt-Date	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:DeficiencyNotificationReport ;
	rdfs:range	xsd:date ;
	skos:definition	“The date on which the deficiency that the deficiency notification report describes was notified”@en ;
	skos:prefLabel	“Notification date”@en ;
	skos:altLabel	“Notificatie datum”@nl .

Subject	*Predicate*	*Object*
maint:notifiedBy	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:DeficiencyNotificationReport ;
	rdfs:range	xsd:string ;
	skos:definition	“The name of the person that notifies the deficiency that the deficiency notification report describes”@en ;
	skos:prefLabel	“Notifier”@en ;
	skos:altLabel	“Kennisgever”@nl .

Subject	*Predicate*	*Object*
maint:notifiedBy-Organization	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:DeficiencyNotificationReport ;
	rdfs:range	xsd:string ;
	skos:definition	"The name of the organization that the person who notifies the deficiency, that the deficiency notification report describes, belongs to"@en ;
	skos:prefLabel	"Organization of notifier"@en ;
	skos:altLabel	"Organisatie van Kennisgever"@nl .

Subject	*Predicate*	*Object*
maint:inspectedAt-Date	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:ConditionMeasurementReport ;
	rdfs:range	xsd:string ;
	skos:definition	"The date on which the condition measurement report was conducted"@en ;
	skos:prefLabel	"Inspection date"@en ;
	skos:altLabel	"Inspectie datum"@nl .

Subject	*Predicate*	*Object*
maint:InspectedBy	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:ConditionMeasurementReport ;
	rdfs:range	xsd:string ;
	skos:definition	"The name of the person that conducted the inspection"@en ;
	skos:prefLabel	"Inspector"@en ;
	skos:altLabel	"Inspecteur"@nl .

Subject	*Predicate*	*Object*
maint:InspectedBy-Organization	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:ConditionMeasurementReport ;
	rdfs:range	xsd:string ;
	skos:definition	"The name of the organization that is responsible for conducting the condition measurement inspection"@en ;
	skos:prefLabel	"Organization, responsible for the condition measurement inspection"@en ;
	skos:altLabel	"Organisatie verantwoordelijk voor de conditie meting"@nl .

Subject	*Predicate*	*Object*
maint:isOf-DeficiencyType	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:Deficiency ;
	rdfs:range	xsd:string ;
	skos:definition	"A standardized description of the deficiency"@en ;
	skos:prefLabel	"Deficiency type"@en ;
	skos:altLabel	"Gebrek soort"@nl .

Subject	*Predicate*	*Object*
maint:hasDeficiency-Description	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:Deficiency ;
	rdfs:range	xsd:string ;
	skos:definition	"A specific description of the identified deficiency"@en ;
	skos:prefLabel	"Deficiency description"@en ;
	skos:altLabel	"Gebrek omschrijving"@nl .

Subject	*Predicate*	*Object*
maint:hasCondition-Score	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:Deficiency ;
	rdfs:range	[a rdfs:Datatype ; owl:onDatatype xsd:integer ; owl:withRestrictions ([xsd:minInclusive 0][xsd:maxInclusive 6])] ;
	skos:definition	"The condition score of a building element derived from the deficiency"@en ;
	skos:prefLabel	"Condition score"@en ;
	skos:altLabel	"Conditie score"@nl .

Subject	*Predicate*	*Object*
maint:hasSeverity	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:Deficiency ;
	rdfs:range	xsd:integer ;
	skos:definition	"A measurement of the severity of a deficiency"@en ;
	skos:prefLabel	"Severity"@en ;
	skos:altLabel	"Ernst"@nl .

Subject	*Predicate*	*Object*
maint:hasStadium	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:Deficiency ;
	rdfs:range	xsd:string ;
	skos:definition	"A measurement of the stadium of a deficiency"@en ;
	skos:prefLabel	"Stadium"@en ;
	skos:altLabel	"Intensiteit"@nl .

Subject	*Predicate*	*Object*
maint:hasScale	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:Deficiency ;
	rdfs:range	xsd:string ;
	skos:definition	"A measurement of the scale of a deficiency expressed "@en ;
	skos:prefLabel	"Scale"@en ;
	skos:altLabel	"Omvang"@nl .

Subject	*Predicate*	*Object*
maint:hasAspect	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:Deficiency ;
	rdfs:range	xsd:string ;
	skos:definition	"Categorization of the deficiency to a theme"@en ;
	skos:prefLabel	"Aspect"@en ;
	skos:altLabel	"Aspect"@nl .

Subject	*Predicate*	*Object*
maint:maintenance-TaskName	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:MaintenanceTask ;
	rdfs:range	xsd:string ;
	skos:definition	"The name of the maintenance task, this is intended to be a standardized task name"@en ;
	skos:prefLabel	"Maintenance task name"@en ;
	skos:altLabel	"Naam onderhoudstaak"@nl .

Subject	*Predicate*	*Object*
maint:maintenance-TaskDescription	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:MaintenanceTask ;
	rdfs:range	xsd:string ;
	skos:definition	"A specific description of a maintenance task instance"@en ;
	skos:prefLabel	"Maintenance task description"@en ;
	skos:altLabel	"Onderhoudstaak omschrijving"@nl .

Subject	*Predicate*	*Object*
maint:projectName	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:MaintenanceTask ;
	rdfs:range	xsd:string ;
	skos:definition	"The name of the project"@en ;
	skos:prefLabel	"Project name"@en ;
	skos:altLabel	"Projectnaam"@nl .

Subject	*Predicate*	*Object*
maint:project-Description	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:MaintenanceTask ;
	rdfs:range	xsd:string ;
	skos:definition	"An informative description of the project"@en ;
	skos:prefLabel	"Project description"@en ;
	skos:altLabel	"Project omschrijving"@nl .

Subject	*Predicate*	*Object*
maint:project-Status	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:MaintenanceTask ;
	rdfs:range	xsd:string ;
	skos:definition	"Describes the status of the project"@en ;
	skos:prefLabel	"Project status"@en ;
	skos:altLabel	"Projectstatus"@nl .

Subject	*Predicate*	*Object*
maint:contact-Person	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:MaintenanceTask ;
	rdfs:range	xsd:string ;
	skos:definition	"The name of the person to contact with respect to project specific issues"@en ;
	skos:prefLabel	"Contact person"@en ;
	skos:altLabel	"Contactpersoon"@nl .

Subject	*Predicate*	*Object*
maint:required-Value	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:MaintenanceTask ;
	rdfs:range	[a rdfs:Datatype ; owl:onDatatype xsd:integer ; owl:withRestrictions ([xsd:minInclusive 0][xsd:maxInclusive 6])] ;
	skos:definition	"the condition value that an organization desires that its building element have"@en ;
	skos:prefLabel	"Desired condition value"@en ;
	skos:altLabel	"Gewenste conditie niveau"@nl .

Subject	*Predicate*	*Object*
maint:warranty-Status	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:Warranty ;
	rdfs:range	xsd:string ;
	skos:definition	"Indicates if warranty is expired or not"@en ;
	skos:prefLabel	"Warranty status"@en ;
	skos:altLabel	"Garantie status"@nl .

Subject	*Predicate*	*Object*
maint:warranty-Provider	rdf:type	owl:DatatypeProperty ;
	rdfs:domain	maint:Warranty ;
	rdfs:range	xsd:string ;
	skos:definition	"Name of the organization that provides the warranty"@en ;
	skos:prefLabel	"Warranty provider"@en ;
	skos:altLabel	"Garantiegever"@nl .

Links set

Subject	*Predicate*	*Object*
maint:Deficiency	rdfs:subClassOf	[a owl:Restriction ; owl:onProperty nta:contains; owl:allValuesFrom BuildingElement] .
maint:Maintenance-Task	rdfs:subClassOf	[a owl:Restriction ; owl:onProperty maint:isConditionOf ; owl:allValuesFrom BuildingElement] .
maint:Desired-ConditionValue	rdfs:subClassOf	[a owl:Restriction ; owl:onProperty bs:describes ; owl:allValuesFrom BuildingElement] .
maint:Warranty	rdfs:subClassOf	[a owl:Restriction ; owl:onProperty bs:describes ; owl:allValuesFrom BuildingElement] .
maint:Building-Project	rdfs:subClassOf	[a owl:Restriction ; owl:onProperty bs:transforms ; owl:allValuesFrom Building] .

Appendix-C Python script

This Appendix shows one of the Python scripts that was used in order to convert the Excel source documents in RDF-data (Turtle syntax) according to the designed Maintenance Ontology.

```
import xlrd
import os
from rdflib import Graph , Literal , Namespace , RDF , RDFS , OWL, URIRef, XSD

#directs to the excel file
currentPath = os.getcwd()
pathFolderData = os.path.abspath( os.path.join( currentPath , os.pardir))
pathToFile = os.path.join( pathFolderData , "Desktop" , "ConditieMeting_11.xlsx" )
wb = xlrd.open_workbook( pathToFile )

#initiates a graph
g = Graph()

#defines namespaces
g.bind( "owl" , OWL )
NTA8035 = Namespace( "https://NTA8035.org/nta#" )
g.bind( "nta" , NTA8035 )
INSTIFC = Namespace( "http://linkedbuildingdata.net/ifc/resources/20191018_114250/" )
g.bind( "inst" , INSTIFC )
MAINT = Namespace( "https://maintenanceontology.org/maint#" )
g.bind( "maint" , MAINT )
INSTMAINT = Namespace( "http://EWI/ConditieMeting.tudelft/" )
g.bind( "instmaint" , INSTMAINT )

#directs to the sheets in the excel file
sh_MetaData = wb.sheet_by_name( "MetaData")
sh_Deficiencies = wb.sheet_by_name( "DeficienciesGemapd" )

#ASSIGNING TYPES
#assigning maint:Deficiency type
for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = RDF.type
    o = MAINT.Deficiency
    g.add ((s, p, o))

#assigning maint:ConditionMeasurementReport type
    ConditionMeasurementReport = str (sh_MetaData.cell_value (0,1))
    s = INSTMAINT[ConditionMeasurementReport]
    p = RDF.type
    o = MAINT.ConditionMeasurementReport
```

```
    g.add ((s,p,o))

#METADATA RELATED TRIPPLES
#connects the ConditionMeasurementReport to the Building instance from the ifc file
ConditionMeasurementReport = str (sh_MetaData.cell_value (0,1))
Building = str( sh_MetaData.cell_value( 5 , 1 ))
s = INSTMAINT[ConditionMeasurementReport]
p = MAINT.inspectsBuilding
o = INSTIFC[Building]
g.add ((s,p,o))

#connects the ConditionMeasurementReport to all the deficiencies it describes
for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    ConditionMeasurementReport = str (sh_MetaData.cell_value (0,1))
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    s = INSTMAINT[ConditionMeasurementReport]
    p = NTA8035.describes
    o = INSTMAINT[Gebrekinstantie]
    g.add ((s,p,o))

#links the code to the ConditionMeasurementReport instance
    ConditionMeasurementReport = str (sh_MetaData.cell_value (0,1))
    InspectionCode = str( sh_MetaData.cell_value( 1 , 1 ))
    s = INSTMAINT[ConditionMeasurementReport]
    p = MAINT.inspectionCode
    o = Literal(InspectionCode)
    g.add ((s,p,o))

#links the inspector to the ConditionMeasurementReport instance
    ConditionMeasurementReport = str (sh_MetaData.cell_value (0,1))
    InspectorName = str( sh_MetaData.cell_value( 2 , 1 ))
    s = INSTMAINT[ConditionMeasurementReport]
    p = MAINT.inspectedBy
    o = Literal(InspectorName)
    g.add ((s,p,o))

#links the responsible organization to the ConditionMeasurementReport instance
    ConditionMeasurementReport = str (sh_MetaData.cell_value (0,1))
    ResponsibleOrganisation = str( sh_MetaData.cell_value( 3 , 1 ))
    s = INSTMAINT[ConditionMeasurementReport]
    p = MAINT.responsibleOrganisation
    o = Literal(ResponsibleOrganisation)
    g.add ((s,p,o))

#links the date to the ConditionMeasurementReport instance
    ConditionMeasurementReport = str (sh_MetaData.cell_value (0,1))
    InspectionDate = str( sh_MetaData.cell_value( 4 , 1 ))
    s = INSTMAINT[ConditionMeasurementReport]
```

```
    p = MAINT.inspectedAtDate
    o = Literal(InspectionDate, datatype=XSD.date)
    g.add ((s,p,o))

#CREERD INSTANTIE DATA GERELATEERDE TRIPLES
#links the deficiency to the building object
for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    Dakinstantie = str( sh_Deficiencies.cell_value( i , 3 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = NTA8035.isConditionOf
    o = INSTIFC[ Dakinstantie ]
    g.add ((s, p, o))

#links the deficiency to the deficiency type/sort
for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    Gebrektype = str( sh_Deficiencies.cell_value( i , 7 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = MAINT.deficiencyType
    o = Literal(Gebrektype)
    g.add ((s, p, o))

for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    Gebrektype = str( sh_Deficiencies.cell_value( i , 8 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = MAINT.deficiencyType
    o = Literal(Gebrektype)
    g.add ((s, p, o))

for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    Gebrektype = str( sh_Deficiencies.cell_value( i , 9 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = MAINT.deficiencyType
    o = Literal(Gebrektype)
    g.add ((s, p, o))

for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    Gebrektype = str( sh_Deficiencies.cell_value( i , 10 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = MAINT.deficiencyType
    o = Literal(Gebrektype)
    g.add ((s, p, o))

#links the deficiency to the deficiency description
```

```
for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    Gebrektoelichting = str( sh_Deficiencies.cell_value( i , 11 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = MAINT.hasDescription
    o = Literal(Gebrektoelichting)
    g.add ((s, p, o))

#links the deficiency to the material that was assigned during the inspection (so not the material according the ifc model...)
for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    Gebrekmateriaal = str( sh_Deficiencies.cell_value( i , 6 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = MAINT.material
    o = Literal(Gebrekmateriaal)
    g.add ((s, p, o))

#links the deficiency to the assigned intensity
for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    Intensiteit = str( sh_Deficiencies.cell_value( i , 13 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = MAINT.hasIntensity
    o = Literal(Intensiteit)
    g.add ((s, p, o))

#links the deficiency to the assigned scale
for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    Omvang = str( sh_Deficiencies.cell_value( i , 14 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = MAINT.hasScale
    o = Literal(Omvang)
    g.add ((s, p, o))

#links the deficiency to the assigned ConditionLevel (CN)
for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    ConditieNiveau = str( sh_Deficiencies.cell_value( i , 15 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = MAINT.hasConditionScore
    o = Literal(ConditieNiveau)
    g.add ((s, p, o))

#links the deficiency to the assigned Aspect
for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
```

```
    Aspect = str( sh_Deficiencies.cell_value( i , 23 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = MAINT.hasAspect
    o = Literal(Aspect)
    g.add ((s, p, o))

#links the deficiency to the assigned Priority
for i in range ( 2 , sh_Deficiencies.nrows , 1 ):
    Gebrekinstantie = str( sh_Deficiencies.cell_value( i , 0 ))
    Aspect = str( sh_Deficiencies.cell_value( i , 24 ))
    s = INSTMAINT[ Gebrekinstantie ]
    p = MAINT.hasPriority
    o = Literal(Aspect)
    g.add ((s, p, o))

#Serializes and prints the created tripples in a file to an assigned destination
g.serialize( destination = "C:\\Users\\User\\Desktop\\testConditieMeting_11.ttl" , format = "turtle" )
print( str( os.getcwd() ) )
```

Appendix-D SPARQL-Queries

D.1 Query 1

Query

```
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX express: <https://w3id.org/express#>
PREFIX maint: <https://maintenanceontology.org/maint#>
PREFIX nta: <https://NTA8035.org/nta#>
PREFIX instifc: <http://linkedbuildingdata.net/ifc/resources/20191018_114250/>
PREFIX instmaint: <http://EWI/ConditieMeting.tudelft/>

SELECT  ?BuildingObjectInstance ?RoofSurfaceArea
WHERE {
 #selecting specific elements
 ?BuildingObjectInstance ?b ifcowl:IfcRoof .
 ?BuildingObjectInstance ifcowl:globalId_IfcRoot ?c .
 ?c express:hasString ?d .

 #selecting specific property-sets that describe the selected elements
 ?IfcRelDefinesByProperties1 ifcowl:relatedObjects_IfcRelDefines ?BuildingObjectInstance .
 ?IfcRelDefinesByProperties1 ifcowl:relatingPropertyDefinition_IfcRelDefinesByProperties
 ?IfcPropertySet1 .
 ?IfcPropertySet1 ifcowl:name_IfcRoot ?1Label .
 ?1Label express:hasString ?PropertySet1 .
 FILTER(?PropertySet1 = "Pset_RoofCommon") .

 #selecting specific properties from the selected property-sets
 ?IfcPropertySet1 ifcowl:hasProperties_IfcPropertySet ?SingleProperty1 .
 ?SingleProperty1 ifcowl:name_IfcProperty ?PropertyNameInstance1 .
 ?PropertyNameInstance1 express:hasString "TotalArea" .
 ?SingleProperty1 ifcowl:nominalValue_IfcPropertySingleValue ?PropertyValueInstance1 .
 ?PropertyValueInstance1 express:hasDouble ?RoofSurfaceArea .
 }
```

Query results

BuildingObjectInstance	RoofSurfaceArea
instifc:IfcRoof_340	"1609.83049059465e0"^^xsd:double
instifc:IfcRoof_635	"1209.44151388422e0"^^xsd:double
instifc:IfcRoof_13946	"4580.65537459811e0"^^xsd:double
instifc:IfcRoof_22636	"168.764250001216e0"^^xsd:double
instifc:IfcRoof_22803	"108.61102173021e0"^^xsd:double

D.2 Query 2

Query

```
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX maint: <https://maintenanceontology.org/maint#>
PREFIX nta: <https://NTA8035.org/nta#>
PREFIX instifc: <http://linkedbuildingdata.net/ifc/resources/20191018_114250/>
PREFIX instmaint: <http://EWI/ConditieMeting.tudelft/>

SELECT  ?BuildingObjectInstance ?RoofFinishingType
WHERE {
 ?BuildingObjectInstance  ?b ?Type .
 ?BuildingObjectInstance ?b ifcowl:IfcRoof .

 ?IfcRelDefinesByProperties2 ifcowl:relatedObjects_IfcRelDefines ?BuildingObjectInstance .
 ?IfcRelDefinesByProperties2 ifcowl:relatingPropertyDefinition_IfcRelDefinesByProperties
?IfcPropertySet2 .
 ?IfcPropertySet2 ifcowl:name_IfcRoot ?Label2 .
 ?Label2 express:hasString ?PropertySet2 .
 FILTER(?PropertySet2 = "Other") .

 ?IfcPropertySet2 ifcowl:hasProperties_IfcPropertySet ?SingleProperty2 .
 ?SingleProperty2 ifcowl:name_IfcProperty ?PropertyNameInstance2 .
 ?PropertyNameInstance2 express:hasString "Type" .
 ?SingleProperty2 ifcowl:nominalValue_IfcPropertySingleValue ?PropertyValueInstance2 .
 ?PropertyValueInstance2 express:hasString ?RoofFinishingType .
 }
```

Query results

BuildingObjectInstance	RoofFinishingType
instifc:IfcRoof_340	"Basic Roof: 47_EHA_EPS-isolatie_100mm/var_dakbed"
instifc:IfcRoof_635	"Basic Roof: 47_EHA_EPS-isolatie_100mm/var_dakbed"
instifc:IfcRoof_13946	"Basic Roof: 47_EHA_minerale wolplaat + dakdekking"
instifc:IfcRoof_22636	"Basic Roof: 47_EHA_minerale wolplaat + dakdekking(WKO afschot ntb)"
instifc:IfcRoof_22803	"Basic Roof: 47_EHA_minerale wolplaat + dakdekking(WKO afschot ntb)"

D.3 Query 3

Query

```
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX maint: <https://maintenanceontology.org/maint#>
PREFIX nta: <https://NTA8035.org/nta#>
PREFIX instifc: <http://linkedbuildingdata.net/ifc/resources/20191018_114250/>
PREFIX instmaint: <http://EWI/ProjectenLijst.tudelft/>

SELECT  ?BuildingObjectInstance ?WarrantyStatus ?WarrantyEndDate ?WarrantyProvider
WHERE {
 ?BuildingObjectInstance ?b ?Type .
 ?BuildingObjectInstance ?b ifcowl:IfcRoof .

 GRAPH <http://localhost:3030/IFC-TUdelft/data/GarantieInformatie_1>{
 ?Warranty nta:describes ?BuildingObjectInstance ;
      maint:warrantyStatus ?WarrantyStatus ;
      maint:warrantyEndDate ?WarrantyEndDate ;
      maint:warrantyProvider ?WarrantyProvider .}
 }
```

Query results

BuildingObjectInstance	WarrantyStatus	WarrantyEndDate	WarrantyProvider
instifc:IfcRoof_340	"Actief"	"2025-01-01 12:00:00.0"^^xsd:datetime	"Geconsolideerd Lichtervoorden"
instifc:IfcRoof_635	"Actief"	"2020-01-01 12:00:00.0"^^xsd:datetime	"Geconsolideerd Lichtervoorden"
instifc:IfcRoof_13946	"Actief"	"2025-01-01 12:00:00.0"^^xsd:datetime	"Geconsolideerd Lichtervoorden"
instifc:IfcRoof_22636	"Actief"	"2025-01-01 12:00:00.0"^^xsd:datetime	"Geconsolideerd Lichtervoorden"
instifc:IfcRoof_22803	"Verlopen"	"2015-01-01 12:00:00.0"^^xsd:datetime	"Ultradak Gorrinkem"

D.4 Query 4

Query

```
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX maint: <https://maintenanceontology.org/maint#>
PREFIX nta: <https://NTA8035.org/nta#>
PREFIX instifc: <http://linkedbuildingdata.net/ifc/resources/20191018_114250/>
PREFIX instmaint: <http://EWI/ConditieMeting.tudelft/>

SELECT  ?BuildingObjectInstance ?Deficiency ?DeficiencyType ?Description
WHERE {
 ?BuildingObjectInstance ?b ?Type .
 ?BuildingObjectInstance ?b ifcowl:IfcRoof .

 GRAPH <http://localhost:3030/IFC-TUdelft/data/ConditieMeting_11>{
 ?Deficiency nta:isConditionOf ?BuildingObjectInstance ;
          maint:deficiencyType ?DeficiencyType ;
          maint:hasDescription ?Description .}
 }
```

Query results

BuildingObjectInstance	Deficiency	DeficiencyType	Description
instifc:IfcRoof_635	instmaint:Gebrek_12	"Aangroei, mos, algen (G)"	"De dakbedekking is plaatselijk vervuild en er komt plantengroei voor"
instifc:IfcRoof_635	instmaint:Gebrek_12	"Erosie, verweering, verzanding (S)"	"De dakbedekking is plaatselijk vervuild en er komt plantengroei voor"
instifc:IfcRoof_635	instmaint:Gebrek_12	"Ontbreken schutlaag (S)"	"De dakbedekking is plaatselijk vervuild en er komt plantengroei voor"
instifc:IfcRoof_22636	instmaint:Gebrek_19	"Vocht, doorslaand (E) / Blazen, Plooien"	"De dakbedekking is plaatselijk vervuild en er komt plantengroei voor"
instifc:IfcRoof_22636	instmaint:Gebrek_19	"Vuil, aanslag, verkleuring (G)"	"De dakbedekking is plaatselijk vervuild en er komt plantengroei voor"
instifc:IfcRoof_22636	instmaint:Gebrek_20	""	"Op veel plaatsen zijn waterblazen en zonken in de ondergrond onder de dakbedekking

D.5 Query 5

Query

```
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX maint: <https://maintenanceontology.org/maint#>
PREFIX nta: <https://NTA8035.org/nta#>
PREFIX instifc: <http://linkedbuildingdata.net/ifc/resources/20191018_114250/>
PREFIX instmaint: <http://EWI/ConditieMeting.tudelft/>

SELECT  ?BuildingObjectInstance ?Deficiency ?Intensity ?Scale ?ConditionScore
WHERE {
 ?BuildingObjectInstance ?b ?Type .
 ?BuildingObjectInstance ?b ifcowl:IfcRoof .

 GRAPH <http://localhost:3030/IFC-TUdelft/data/ConditieMeting_11>{
 ?Deficiency nta:isConditionOf ?BuildingObjectInstance ;
         maint:hasIntensity ?Intensity ;
         maint:hasScale ?Scale ;
         maint:hasConditionScore ?ConditionScore .}
 }
```

Query results

BuildingObjectInstance	Deficiency	Intensity	Scale	Condi
instifc:IfcRoof_340	instmaint:Gebrek_23	"Gevorderdstadium"	"Plaatselijk"	"4.0"
instifc:IfcRoof_340	instmaint:Gebrek_24	"Eindstadium"	"Aanzienlijk"	"4.0"
instifc:IfcRoof_340	instmaint:Gebrek_25	"Eindstadium"	"Plaatselijk"	"3.0"
instifc:IfcRoof_635	instmaint:Gebrek_12	"Gevorderdstadium"	"Plaatselijk"	"2.0"
instifc:IfcRoof_635	instmaint:Gebrek_13	"No intensity has been specified"	"No size has been specified"	"2.0"
instifc:IfcRoof_13946	instmaint:Gebrek_6	"Gevorderdstadium"	"Regelmatig"	"1.0"
instifc:IfcRoof_22636	instmaint:Gebrek_19	"Gevorderdstadium"	"Plaatselijk"	"4.0"
instifc:IfcRoof_22636	instmaint:Gebrek_20	"Eindstadium"	"Regelmatig"	"4.0"
instifc:IfcRoof_22636	instmaint:Gebrek_21	"Eindstadium"	"Incidenteel"	"1.0"
instifc:IfcRoof_22636	instmaint:Gebrek_22	"No intensity has been specified"	"No size has been specified"	"1.0"
instifc:IfcRoof_22803	instmaint:Gebrek_26	"Gevorderdstadium"	"Plaatselijk"	"1.0"

D.6 Query 6

Query

```
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX maint: <https://maintenanceontology.org/maint#>
PREFIX nta: <https://NTA8035.org/nta#>
PREFIX instifc: <http://linkedbuildingdata.net/ifc/resources/20191018_114250/>
PREFIX instmaint: <http://EWI/ConditieMeting.tudelft/>

SELECT  ?BuildingObjectInstance ?Deficiency ?Aspect ?Priority
WHERE {
 ?BuildingObjectInstance ?b ?Type .
 ?BuildingObjectInstance ?b ifcowl:IfcRoof .

 GRAPH <http://localhost:3030/IFC-TUdelft/data/ConditieMeting_11>{
 ?Deficiency nta:isConditionOf ?BuildingObjectInstance ;
          maint:hasAspect ?Aspect ;
          maint:hasPriority ?Priority .
 ?DeficiencyReport nta:contains ?Deficiency .}
 }
```

Query results

BuildingObjectInstance	Deficiency	Aspect	Priority
instifc:IfcRoof_340	instmaint:Gebrek_23	"Technische motieven"	"7.0"
instifc:IfcRoof_340	instmaint:Gebrek_24	"Technische motieven"	"7.0"
instifc:IfcRoof_340	instmaint:Gebrek_25	"Veiligheid"	"3.0"
instifc:IfcRoof_635	instmaint:Gebrek_12	"Technische motieven"	"7.0"
instifc:IfcRoof_635	instmaint:Gebrek_13	"Technische motieven"	"7.0"
instifc:IfcRoof_13946	instmaint:Gebrek_6	"Bruikbaarheid"	"6.0"
instifc:IfcRoof_22636	instmaint:Gebrek_19	"Technische motieven"	"7.0"
instifc:IfcRoof_22636	instmaint:Gebrek_20	"Bruikbaarheid"	"3.0"
instifc:IfcRoof_22636	instmaint:Gebrek_21	"Technische motieven"	"7.0"
instifc:IfcRoof_22636	instmaint:Gebrek_22	"No aspect has been specified"	"not specified"
instifc:IfcRoof_22803	instmaint:Gebrek_26	"Technische motieven"	"7.0"

D.7 Query 7

Query

```
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX maint: <https://maintenanceontology.org/maint#>
PREFIX nta: <https://NTA8035.org/nta#>
PREFIX instifc: <http://linkedbuildingdata.net/ifc/resources/20191018_114250/>
PREFIX instmaint: <http://EWI/ConditieMeting.tudelft/>

SELECT  ?BuildingObjectInstance ?Deficiency ?Date ?Inspector ?Organisation
WHERE {
 ?BuildingObjectInstance ?b ?Type .
 ?BuildingObjectInstance ?b ifcowl:IfcRoof .

 GRAPH <http://localhost:3030/IFC-TUdelft/data/ConditieMeting_11>{
 ?Deficiency nta:isConditionOf ?BuildingObjectInstance .
 ?DeficiencyReport nta:contains ?Deficiency ;
            maint:inspectedAtDate ?Date ;
            maint:inspectedBy ?Inspector ;
            maint:responsibleOrganisation ?Organisation .}
 }
```

Query results

BuildingObjectInst	Deficiency	Date	Inspector	Organisation
instifc:IfcRoof_340	instmaint:Gebrek_23	"2016-08-01"^^xsd:date	"Person Name"	"Royal Haskoning"
instifc:IfcRoof_340	instmaint:Gebrek_24	"2016-08-01"^^xsd:date	"Person Name"	"Royal Haskoning"
instifc:IfcRoof_340	instmaint:Gebrek_25	"2016-08-01"^^xsd:date	"Person Name"	"Royal Haskoning"
instifc:IfcRoof_635	instmaint:Gebrek_12	"2016-08-01"^^xsd:date	"Person Name"	"Royal Haskoning"
instifc:IfcRoof_635	instmaint:Gebrek_13	"2016-08-01"^^xsd:date	"Person Name"	"Royal Haskoning"
instifc:IfcRoof_13946	instmaint:Gebrek_6	"2016-08-01"^^xsd:date	"Person Name"	"Royal Haskoning"
instifc:IfcRoof_22636	instmaint:Gebrek_19	"2016-08-01"^^xsd:date	"Person Name"	"Royal Haskoning"
instifc:IfcRoof_22636	instmaint:Gebrek_20	"2016-08-01"^^xsd:date	"Person Name"	"Royal Haskoning"
instifc:IfcRoof_22636	instmaint:Gebrek_21	"2016-08-01"^^xsd:date	"Person Name"	"Royal Haskoning"

D.8 Query 8

Query

```
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX maint: <https://maintenanceontology.org/maint#>
PREFIX nta: <https://NTA8035.org/nta#>
PREFIX instifc: <http://linkedbuildingdata.net/ifc/resources/20191018_114250/>
PREFIX instmaint: <http://EWI/OnderhoudsHistorie.tudelft/>

SELECT  ?BuildingObjectInstance ?Deficiency ?DeficiencyType ?Description ?Date ?Person
?Organisation
WHERE {
 ?BuildingObjectInstance ?b ?Type .
 ?BuildingObjectInstance ?b ifcowl:IfcRoof .

 GRAPH <http://localhost:3030/IFC-TUdelft/data/OnderhoudsHistorie_11>{
  ?NotificationReport nta:describes ?Deficiency ;
         maint:notifiedAtDate ?Date ;
         maint:notifiedBy ?Person ;
         maint:notifiedByOrganisation ?Organisation .

   ?Deficiency nta:isConditionOf ?BuildingObjectInstance ;
         maint:deficiencyType ?DeficiencyType ;
         maint:deficiencyDescription ?Description .}
 }
```

Query results

BuildingObjectInstance	Deficiency	DeficiencyType	Description
instifc:IfcRoof_340	instmaint:Deficiency_79	"Lekkage (dak)"	"Malfunction"
instifc:IfcRoof_635	instmaint:Deficiency_71	"Lekkage (dak)"	"Malfunction"
instifc:IfcRoof_635	instmaint:Deficiency_70	"Lekkage (dak)"	"Malfunction"
instifc:IfcRoof_13946	instmaint:Deficiency_69	"Lekkage (dak)"	"Malfunction"
instifc:IfcRoof_13946	instmaint:Deficiency_68	"Lekkage (dak)"	"Malfunction"
instifc:IfcRoof_22636	instmaint:Deficiency_77	"Lekkage (dak)"	"Malfunction"
instifc:IfcRoof_22636	instmaint:Deficiency_74	"Lekkage (dak)"	"Complaint"
instifc:IfcRoof_22636	instmaint:Deficiency_75	"Lekkage (dak)"	"Malfunction"
instifc:IfcRoof_22636	instmaint:Deficiency_73	"Lekkage (dak)"	"Malfunction"
instifc:IfcRoof_22636	instmaint:Deficiency_76	"Lekkage (dak)"	"Malfunction"
instifc:IfcRoof_22636	instmaint:Deficiency_78	"Lekkage (dak)"	"Malfunction"
instifc:IfcRoof_22803	instmaint:Deficiency_72	"Lekkage (dak)"	"Malfunction"

D.9 Query 9

Query

```
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX maint: <https://maintenanceontology.org/maint#>
PREFIX nta: <https://NTA8035.org/nta#>
PREFIX instifc: <http://linkedbuildingdata.net/ifc/resources/20191018_114250/>
PREFIX instmaint: <http://EWI/ProjectenLijst.tudelft/>

SELECT  ?Building ?Project ?BeginDate ?EndDate ?ProjectDescrition ?ProjectStatus
WHERE {
 ? Building ?b ?Type .
 ? Building ?b ifcowl:IfcBuilding .

 GRAPH <http://localhost:3030/IFC-TUdelft/data/ProjectenLijst_1>{
  ?Project nta:transforms ?Building ;
        nta:begins ?BeginDate ;
        nta:ends ?EndDate ;
        maint:projectDescription ?ProjectDescrition ;
        maint:projectStatus ?ProjectStatus .}
 }
```

Query results

Building	Project	BeginDate	EndDate	ProjectDescritic	ProjectStatu
instifc:IfcBuilding_115	instmaint:Project_01	"2020-02-19"^^xsd:date	"2021-08-19"^^xsd:date	"ESP-Lab"	"Uitvoering"
instifc:IfcBuilding_115	instmaint:Project_18	"2021-08-26"^^xsd:date	"2023-05-30"^^xsd:date	"DEMO"	"ON HOLD"
instifc:IfcBuilding_115	instmaint:Project_19	"2022-12-11"^^xsd:date	"2023-09-24"^^xsd:date	"Uithuizing Hoogbouw Gebouw 36 EWI"	"ON HOLD"
instifc:IfcBuilding_115	instmaint:Project_54	"2025-04-17"^^xsd:date	"2028-08-01"^^xsd:date	"Afstootstrategie gebouw 36"	"ON HOLD"

D.10 Query 10

Query

```
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX maint: <https://maintenanceontology.org/maint#>
PREFIX nta: <https://NTA8035.org/nta#>
PREFIX instifc: <http://linkedbuildingdata.net/ifc/resources/20191018_114250/>
PREFIX instmaint: <http://EWI/ProjectenLijst.tudelft/>

SELECT  ?BuildingObjectInstance ?DesiredConditionLevel
WHERE {
 ?BuildingObjectInstance ?b ?Type .
 ?BuildingObjectInstance ?b ifcowl:IfcRoof .

 GRAPH <http://localhost:3030/IFC-TUdelft/data/AssetManagementStrategy_1>{
        ?BuildingObjectInstance maint:desiredConditionLevel ?DesiredConditionLevel .}
 }
```

Query results

BuildingObjectInstance	DesiredConditionLevel
instifc:IfcRoof_340	"2.0"
instifc:IfcRoof_635	"1.0"
instifc:IfcRoof_13946	"2.0"
instifc:IfcRoof_22636	"1.0"
instifc:IfcRoof_22803	"2.0"

D.11 Query 11

Query

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX instifc:  <http://eigenhaard.net/ifc/20191122/>
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX express: <https://w3id.org/express#>
PREFIX maint: <https://maintenanceontology.org/maint#>
prefix instmaint: <http://EigenHaard/MaintenanceInformation.nl/>
prefix nta: <https://NTA8035.org/nta#>

SELECT ?BuildingObjectInstance ?Material ?SurfaseArea ?NLSfBcode ?WarrantyStatus ?Organisation

WHERE{
GRAPH<http://localhost:3030/IFC_Rivierenhuis/data/IFC_Model_Rivierenhuis>{
?BuildingObjectInstance rdf:type ifcowl:IfcRoof .
?IfcRelDefinesByProperties1 ifcowl:relatedObjects_IfcRelDefines ?BuildingObjectInstance .
?IfcRelDefinesByProperties1 ifcowl:relatingPropertyDefinition_IfcRelDefinesByProperties ?IfcPropertySet1 .
?IfcPropertySet1 ifcowl:name_IfcRoot ?1Label .
?1Label ?Express ?PropertySet1 .
FILTER(?PropertySet1 = "ILS") .

?IfcPropertySet1 ifcowl:hasProperties_IfcPropertySet ?SingleProperty1 .
?SingleProperty1 ifcowl:name_IfcProperty ?PropertyNameInstance1 .
?PropertyNameInstance1 express:hasString "Oppervlaktebescherming" .
?SingleProperty1 ifcowl:nominalValue_IfcPropertySingleValue ?PropertyValueInstance1 .
?PropertyValueInstance1 express:hasString "Bitumen" .
?PropertyValueInstance1 express:hasString ?Material .
?IfcRelDefinesByProperties2 ifcowl:relatedObjects_IfcRelDefines ?BuildingObjectInstance .
?IfcRelDefinesByProperties2 ifcowl:relatingPropertyDefinition_IfcRelDefinesByProperties ?IfcPropertySet2 .
?IfcPropertySet2 ifcowl:name_IfcRoot ?2Label .
?2Label ?Express ?PropertySet2 .
FILTER(?PropertySet2 = "ILS") .

?IfcPropertySet2 ifcowl:hasProperties_IfcPropertySet ?SingleProperty2 .
?SingleProperty2 ifcowl:name_IfcProperty ?PropertyNameInstance2 .
?PropertyNameInstance2 express:hasString "Oppervlakte" .
?SingleProperty2 ifcowl:nominalValue_IfcPropertySingleValue ?PropertyValueInstance2 .
?PropertyValueInstance2 express:hasDouble ?SurfaseArea .
?IfcRelDefinesByProperties3 ifcowl:relatedObjects_IfcRelDefines ?BuildingObjectInstance .
?IfcRelDefinesByProperties3 ifcowl:relatingPropertyDefinition_IfcRelDefinesByProperties ?IfcPropertySet3 .
?IfcPropertySet3 ifcowl:name_IfcRoot ?3Label .
?3Label ?Express ?PropertySet3 .
FILTER(?PropertySet3 = "ILS") .

?IfcPropertySet3 ifcowl:hasProperties_IfcPropertySet ?SingleProperty3 .
?SingleProperty3 ifcowl:name_IfcProperty ?PropertyNameInstance3 .
?PropertyNameInstance3 express:hasString "Nlsfb4" .
?SingleProperty3 ifcowl:nominalValue_IfcPropertySingleValue ?PropertyValueInstance3 .
?PropertyValueInstance3 express:hasString ?NLSfBcode .}

GRAPH<http://localhost:3030/IFC_Rivierenhuis/data/DakotaGarantie>{
   ?Warranty rdf:type maint:Warranty .
          ?Warranty nta:describes ?BuildingObjectInstance .
          ?Warranty maint:warrantyStatus ?WarrantyStatus .
          ?Warranty maint:warrantyProvider ?Organisation .} }
```

Query results

BuildingObjectInstance	Material	SurfaseArea	NLSfBcode	WarrantyStatus	Organisation
instifc:IfcRoof_2774062	"Bitumen"	"1575.45429999998e0"^^xsd:double	"27.22"	"Geen"	"Consolidated"
instifc:IfcRoof_2777048	"Bitumen"	"19.6536000000024e0"^^xsd:double	"27.22"	"Geen"	"Consolidated"
instifc:IfcRoof_2782731	"Bitumen"	"35.3520000000009e0"^^xsd:double	"27.22"	"Geen"	"Consolidated"
instifc:IfcRoof_2783251	"Bitumen"	"9.89999999999964e0"^^xsd:double	"27.22"	"Geen"	"Consolidated"
instifc:IfcRoof_2788159	"Bitumen"	"777.829099999991e0"^^xsd:double	"27.22"	"Geen"	"Consolidated"
instifc:IfcRoof_2790365	"Bitumen"	"21.8584000000002e0"^^xsd:double	"27.22"	"Geen"	"Consolidated"
instifc:IfcRoof_2799713	"Bitumen"	"108.883447569365e0"^^xsd:double	"27.22"	"Geen"	"Consolidated"
instifc:IfcRoof_2802802	"Bitumen"	"82.6345004048225e0"^^xsd:double	"27.22"	"Geen"	"Consolidated"

D.12 Query 12

Query

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX instifc:  <http://eigenhaard.net/ifc/20191122/>
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX express: <https://w3id.org/express#>
PREFIX maint: <https://maintenanceontology.org/maint#>
prefix instmaint: <http://EigenHaard/MaintenanceInformation.nl/>
prefix nta: <https://NTA8035.org/nta#>

SELECT ?BuildingObjectInstance ?ConditionScore ?Inspector ?Organisation ?DesiredConditionValue

WHERE{
GRAPH<http://localhost:3030/IFC_Rivierenhuis/data/IFC_Model_Rivierenhuis>{
?BuildingObjectInstance rdf:type ifcowl:IfcRoof .}

GRAPH<http://localhost:3030/IFC_Rivierenhuis/data/DakotaConditieScore>{
   ?Deficiency rdf:type maint:Deficiency .
   ?Deficiency nta:isConditionOf ?BuildingObjectInstance .
   ?Deficiency maint:hasConditionScore ?ConditionScore .

   ?ConditionMeasurementReport rdf:type maint:ConditionMeasurementReport .
   ?ConditionMeasurementReport nta:contains ?Deficiency .
   ?ConditionMeasurementReport maint:inspectedBy ?Inspector .
   ?ConditionMeasurementReport maint:responsibleOrganisation ?Organisation .}

GRAPH<http://localhost:3030/IFC_Rivierenhuis/data/DakotaGewensteConditieScore>{
   ?DesiredCondition nta:describes ?BuildingObjectInstance .
   ?DesiredCondition rdf:type maint:DesiredConditionValue .
   ?DesiredCondition maint:hasConditionScore ?DesiredConditionValue .}
}
```

Query results

BuildingObjectInstance	ConditionScore	Inspector	Organisation	DesiredCon
instifc:IfcRoof_2774062	"4.0"	"Sjors SchutWild"	"Consolidated"	"2.0"
instifc:IfcRoof_2777048	"5.0"	"Sjors SchutWild"	"Consolidated"	"2.0"
instifc:IfcRoof_2782731	"2.0"	"Sjors SchutWild"	"Consolidated"	"2.0"
instifc:IfcRoof_2783251	"2.0"	"Sjors SchutWild"	"Consolidated"	"2.0"
instifc:IfcRoof_2788159	"2.0"	"Sjors SchutWild"	"Consolidated"	"2.0"
instifc:IfcRoof_2790365	"3.0"	"Sjors SchutWild"	"Consolidated"	"2.0"
instifc:IfcRoof_2799713	"2.0"	"Sjors SchutWild"	"Consolidated"	"2.0"
instifc:IfcRoof_2802802	"1.0"	"Sjors SchutWild"	"Consolidated"	"2.0"

D.13 Query 13

Query

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX instifc:  <http://eigenhaard.net/ifc/20191122/>
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX express: <https://w3id.org/express#>
PREFIX maint: <https://maintenanceontology.org/maint#>
prefix instmaint: <http://EigenHaard/MaintenanceInformation.nl/>
prefix nta: <https://NTA8035.org/nta#>

SELECT ?BuildingObjectInstance ?DeficiencyType ?Date ?CounterMeasure ?Status

WHERE{
GRAPH<http://localhost:3030/IFC_Rivierenhuis/data/IFC_Model_Rivierenhuis>{
?BuildingObjectInstance rdf:type ifcowl:IfcRoof .}

GRAPH<http://localhost:3030/IFC_Rivierenhuis/data/DakotaGebrekNotificatie>{
   ?Deficiency rdf:type maint:Deficiency .
   ?Deficiency nta:isConditionOf ?BuildingObjectInstance .
   ?Deficiency maint:isOfDeficiencyType ?DeficiencyType .
   ?Deficiency maint:notifiedAtDate ?Date .

   ?CounterMeasure rdf:type maint:MaintenanceTask .
   ?CounterMeasure nta:transforms ?Deficiency .
   ?CounterMeasure nta:transforms ?BuildingObjectInstance .
   ?CounterMeasure maint:taskStatus ?Status .}
}
```

Query results

BuildingObjectInstance	DeficiencyType	Date	CounterMeasure	Status
instifc:IfcRoof_2774062	"Lekkage"	"2019-03-15"^^xsd:date	instmaint:CounterMeasure_1	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2014-03-15"^^xsd:date	instmaint:CounterMeasure_10	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2012-03-15"^^xsd:date	instmaint:CounterMeasure_11	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2009-03-15"^^xsd:date	instmaint:CounterMeasure_12	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2005-03-15"^^xsd:date	instmaint:CounterMeasure_13	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2000-03-15"^^xsd:date	instmaint:CounterMeasure_14	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2019-03-15"^^xsd:date	instmaint:CounterMeasure_2	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2019-03-15"^^xsd:date	instmaint:CounterMeasure_3	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2018-03-15"^^xsd:date	instmaint:CounterMeasure_4	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2018-03-15"^^xsd:date	instmaint:CounterMeasure_5	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2017-03-15"^^xsd:date	instmaint:CounterMeasure_6	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2017-03-15"^^xsd:date	instmaint:CounterMeasure_7	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2016-03-15"^^xsd:date	instmaint:CounterMeasure_8	"Uitgevoerd"
instifc:IfcRoof_2774062	"Lekkage"	"2014-03-15"^^xsd:date	instmaint:CounterMeasure_9	"Uitgevoerd"

D.14 Query 14

Query

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX instifc:  <http://eigenhaard.net/ifc/20191122/>
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX express: <https://w3id.org/express#>
PREFIX maint: <https://maintenanceontology.org/maint#>
prefix instmaint: <http://EigenHaard/MaintenanceInformation.nl/>
prefix nta: <https://NTA8035.org/nta#>

SELECT ?Building ?Project ?Name ?Description ?Status ?PlannedDate

WHERE{
GRAPH<http://localhost:3030/IFC_Rivierenhuis/data/IFC_Model_Rivierenhuis>{
?Building rdf:type ifcowl:IfcBuilding .}

GRAPH<http://localhost:3030/IFC_Rivierenhuis/data/PlannedProjects>{
   ?Project rdf:type maint:RenovationProject .
   ?Project nta:transforms ?Building .
   ?Project nta:startDate ?PlannedDate .
   ?Project maint:projectDescription ?Description .
   ?Project maint:projectStatus ?Status .
   ?Project maint:projectName ?Name .}
}
```

Query results

Building	Project	Name	Description	Status	PlannedDate
instifc:IfcBuilding_115	instmaint:BuildingProject_45	"CE180431"	"Dakrenovatie"	"Planned"	"2021-07-01"^^xsd:date
instifc:IfcBuilding_115	instmaint:BuildingProject_46	"CE180432"	"Schoorstenen"	"Planned"	"2030-02-04"^^xsd:date
instifc:IfcBuilding_115	instmaint:BuildingProject_47	"CE180434"	"Veiligheidsvoorziening"	"Planned"	"2030-09-18"^^xsd:date

D.15 Query 15

Query

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX instifc:  <http://eigenhaard.net/ifc/20191122/>
PREFIX ifcowl: <http://standards.buildingsmart.org/IFC/DEV/IFC2x3/TC1/OWL#>
PREFIX express: <https://w3id.org/express#>
PREFIX maint: <https://maintenanceontology.org/maint#>
prefix instmaint: <http://EigenHaard/MaintenanceInformation.nl/>
prefix nta: <https://NTA8035.org/nta#>

SELECT ?Building ?StrategicInformation ?Topic ?Content

WHERE{
GRAPH<http://localhost:3030/IFC_Rivierenhuis/data/IFC_Model_Rivierenhuis>{
   ?Building rdf:type ifcowl:IfcBuilding .}

GRAPH<http://localhost:3030/IFC_Rivierenhuis/data/VimStrategischeInformate>{
   ?StrategicInformation rdf:type maint:StrategicInformation .
   ?StrategicInformation nta:describes ?Building .
   ?StrategicInformation maint:strategicInformationTopic ?Topic .
   ?StrategicInformation maint:strategicInformationContent ?Content .}
}
```

Query results

Building	StrategicInformation	Topic	Content
instifc:IfcBuilding_115	instmaint:StrategicInformation_1	"SellingPlans"	"Niet meer (2016); Na directieberaad besluit over verkoop september 2016 is complex uit verkoopvijver gehaald; doorverhuren"
instifc:IfcBuilding_115	instmaint:StrategicInformation_2	"SustainabilityPlans"	"Ja; Dit complex staat op de planning voor het opstarten van een SaVE (project 'Samen verduurzamen') aanpak."
instifc:IfcBuilding_115	instmaint:StrategicInformation_3	"AmountOfOwnerOccupiedUnits"	"21.0"