POLITECNICO DI MILANO

School of Industrial and Information Engineering

Master of Science in Management Engineering

Systematic Literature Review: Ontologies in maintenance and reliability fields

Author: Luca Bertolotti

Student ID: 942099

Advisor: Prof. Marco Macchi

Co-advisor: Ing. Adalberto Polenghi

Academic Year 2020 - 2021

Abstract

The correct development of maintenance policy and reliability analysis is a fundamental activity in order to support the industrial production systems, reducing the risks and the costs. Both maintenance policy and reliability analysis could be improved through different tools, and one of them, recently discussed by the literature, is the ontology. The term ontology refers to an explicit specification of a conceptualization and its importance is due to the fact that it could help the system actors to have a better knowledge of the domain. The main purpose of this thesis is to investigate through the scientific literature the state of the art of ontologies inside maintenance and reliability topics. To this end, a precise methodology, namely the systematic literature review, will be implemented in order to render an objective analysis. Firstly, a number of constraints will be used in order to select some papers, which will be then taken into consideration to proceed with the research. Secondly, a statistical analysis will be conducted, considering features such as the date of release of the papers and the definition of ontology used. Thirdly, a double categorization of ontology is proposed through the statistical analysis: this categorization is based on the application field and the role of the ontology in the selected papers. In order to complete the statistical part, a taxonomy analysis will be developed so that it is possible to compare the terms used to define concepts and the meaning that these terms aim to render. Then, a content analysis of each paper will be performed in order to shed light on the contribution that each one brings. In conclusion, the analysis of the results will provide several suggestions, as hereafter summarized: 1) Ontologies are growing in maintenance and reliability fields; 2) A maintenance advisor and FMEA are the topics which are dealt with the most; 3) The ontologies play many roles in the management of knowledge inside the scientific research; 4) OWL is the standard for the ontology development and several languages are used to support and improve its functionalities; 5) The average number of ontology layer for the domain/subdomain level is between 1 and 4; 6) There is not a common and shared terms used to explicit a concept; 7) A lot of different contents are provided into the analysed papers; the most interesting ones regard the improvement of existing system to reduce the heterogeneity and to create a common and shared knowledge of the system.

Key words: Ontology, Maintenance, Reliability.

Abstract Italiano

Il corretto sviluppo delle politiche di manutenzione e delle analisi d'affidabilità è un'attività fondamentale per supportare i sistemi di produzione industriale, riducendo i rischi ed i costi. Sia le politiche di manutenzione che l'analisi dell'affidabilità potrebbero essere migliorate attraverso diversi strumenti recentemente citati tra le ontologie. Il termine ontologia si riferisce a una specificazione esplicita di una concettualizzazione e la sua importanza è dovuta al fatto che potrebbe aiutare gli attori del sistema ad avere una migliore conoscenza del dominio. Lo scopo principale di questa tesi è quello di indagare attraverso la letteratura scientifica lo stato dell'arte delle ontologie all'interno dei temi della manutenzione e dell'affidabilità. A tal fine, una metodologia precisa definita come revisione sistematica della letteratura verrà implementata, cercando di percorrere un percorso il più oggettivo possibile. In primo luogo, verranno fissati una serie di vincoli per selezionare alcuni articoli, che verranno poi presi in considerazione per procedere con la ricerca. In secondo luogo, verrà condotta un'analisi statistica, considerando caratteristiche quali per esempio, la data di rilascio degli articoli e la definizione dell'ontologia utilizzata. In terzo luogo, verrà proposta una doppia categorizzazione dell'ontologia attraverso l'analisi statistica: questa categorizzazione si basa sul campo di applicazione e sul ruolo dell'ontologia negli articoli selezionati. Per completare la parte statistica, verrà sviluppata un'analisi tassonomica in modo da poter confrontare i termini utilizzati per la definizione dei concetti ed il significato che questi termini vogliono trasmettere. Quindi, verrà eseguita un'analisi del contenuto degli articoli per far luce sul contributo che ciascuno apporta. In conclusione, l'analisi dei risultati fornirà diversi suggerimenti tra cui: 1) L'uso delle ontologie sta decisamente crescendo nei campi della manutenzione e dell'affidabilità; 2) Lo sviluppo di sistemi di gestione e controllo della manutenzione ed l'implementazione di procedure come FMEA sono gli argomenti maggiormente affrontati; 3) Le ontologie hanno molti ruoli nella gestione della conoscenza all'interno della ricerca scientifica; 4) OWL è lo standard per lo sviluppo delle ontologie e diversi linguaggi sono usati per supportare e migliorare le sue funzionalità; 5) Il numero medio di livelli ontologici per il livello dominio/sottodominio è compreso tra 1 e 4; 6) Non esiste un uso comune dei termini per esplicitare un concetto; 7) Nei documenti analizzati vengono forniti molti contenuti diversi, i più interessanti riguardano il miglioramento del sistema esistente riducendo l'eterogeneità e creando una conoscenza comune e condivisa sul sistema.

Parole chiave: Ontologie, Manutenzione, Affidabilità.

Contents

Introduction

Nowadays, a large and fast transformation is occurring. The industry systems are completely evolving to a new stage, in which the operations are intelligent and connected. This transformation is called Industry 4.0. The industry 4.0 is the information-intensive transformation of manufacturing (and related industries) into a connected environment of big data, people, processes, services, systems, and industrial resources enabled for the IoT with generation and use of data that can be used as a way to create smart industries and ecosystems of industrial innovation and collaboration. Industry 4.0 is therefore a broad vision with clear frameworks and reference architectures, mainly characterized by the connection of physical industrial assets and digital technologies in the so-called cyber-physical systems.

The opportunities of Industry 4.0 are well known by the industrial players and some of them are: 1) Optimization and automation of the activities; 2) Real-time data to support real time economy; 3) Advanced and integrated maintenance; 4) Enabling technologies to share the resources between human and robot; 5) Sustainability of the operations; 6) Better understanding of customer's needs. Each one of these points will bring a competitive advance to the players that will be able to develop a competitive strategy. It is also widely known that industries are dealing with a lot of issues. For instance, they need to face with the product life cycle reduction; the high demand variability; the high customization of products; the heterogeneity between systems; and lacks of information.

In order to take advantage of opportunities and to address the challenges, the actors have to approach new technologies that support the industry 4.0. One of the most famous technologies are the micro sensors, that are small devices that could be positioned in any place, and they transmit simple signal to give to the listener an information. Another famous technology that enables the implementation of Industry 4.0 is the cloud. The cloud provides a lot of services for the industries like the computational power and a data storage system. The wide world of the technology that supports industry 4.0 also include the ontologies.

The ontologies were born in the 90s in order to investigate and understand the complexity of a domain. Coming to the present-day, ontologies are the tools that allow domain experts to describe a domain with an explicit conceptualization permitting the system to maintain data integrity and homogeneity. Indeed, an interesting point of view is certainly the increase of the reliability of a system, optimizing the productive time through the studying of the domain and the reduction of the heterogeneity between systems. Therefore, the aim of this thesis is to investigate the ontologies in the maintenance and reliability field to understand which are the most important covered topics and develop a series of statistics to help future works.

The work will be developed following a systematic literature review (SLR) methodology, starting from the selection of the databases to query and arriving to the eligibility criteria and the statistics. The structure of this thesis is divided into four main blocks. The first block is an introduction of the ontology and the SLR methodology; the second block contains the statistics on the final papers; the third block regards the content analysis of the papers; the last block is the discussion of the results and the conclusion.

1. Ontologies description

The purpose of this chapter is to describe ontology, from its definition to its development criteria. This first introduction will be very useful in order to understand the object of the thesis and the next chapters.

1.1. What is an ontology?

Although the term ontology may be defined in many ways, the most acknowledged definition is the one provided by (Gruber 1995):

"An ontology is an explicit specification of a conceptualization. The term is borrowed from philosophy, where an ontology is a systematic account of Existence. For AI systems, what exists is that which can be represented. When the knowledge of a domain is represented in a declarative formalism, the set of objects that can be represented is called universe of discourse. This set of objects, and the describable relationship among them, are reflected in the representational vocabulary with which a knowledge-based program represents knowledge."

This definition certainly includes significant aspects which must be better explained. First of all, what is the meaning of specification of a conceptualization? A conceptualization is a simplified view of the world we want to represent. According to (Al-Baltah and Ghani 2014) , conceptualization is the process of structuring the domain knowledge. The term specification, on the other hand, refers to the conceptualization of a single and well-defined domain with a precise purpose. The second part of the definition refers to the representation of the domain through a declarative formalism, which is a set of objects consisting of:

- Classes, that represent concepts in the real world;
- Relations between the concepts of the domain;
- Axioms, that are rules normally followed in the real world;
- Instances, that are the elements in an ontology.

A brief example might clarify this concept. It is possible to take into consideration the domain of a primary school, in which the classes are represented by the students, the classrooms and the professors. The relations are the one existing between the professors and the students: the professors teach, and the students have a classroom. The axiom is given by the fact that each student has one classroom and as far as the instances are concerned, we can hypothesise that one student is named Simone Rossi and he is ten years old. There are other definitions of ontology, but we will see that this one is the most accepted in the ontology environment. To resume, an ontology is a tool to represent, through the elements listed before, the real world in a unique way. The use of this representation will be explained in the next paragraphs and chapters.

1.2.Design criteria for ontologies

A preliminary set of design criteria is proposed in (Gruber 1995), whose purpose is knowledge sharing and interoperation:

- Clarity means that an ontology should communicate the intended meaning of the defined terms. Definitions should be objective;
- Coherence means that it should sanction inferences that are consistent with the definitions. If a sentence, that can be inferred from the axioms, contradicts a definition, then the ontology is incoherent;
- Extendibility means that an ontology should be designed to anticipate the uses of the shared vocabulary. In other words, one should be able to define new terms for special uses based on an existing vocabulary;
- Minimal encoding bias, which means that knowledge level must be independent from a particular symbol-level encoding;
- Minimal ontological commitment, which refers to the fact that the ontology must support the intended knowledge sharing activities.

These criteria help the authors of ontology models to describe the reality in a unique and complete way, avoiding misunderstandings and errors.

1.3.Standard development methodologies

The main task for the ontology development is usually the ontology management, including scheduling, controlling, quality assurance and in which there is also the feasibility study. All these tasks are common in most part of ontology programs. As far as the development methodology is concerned, we can distinguish two classifications. The first classification is based on the level of the analysed details, which can be divided into micro-level ontology or macro-level ontology.

The micro-level method focuses on the formalization aspect. whereas the macro-level deals with the development process of the ontology. The main used macro-level development methods are imported from the typical development methods implemented in the programming sector, that is the Waterfalls and the Agile. In the document ('Agile Project Management', n.d.) created by the Fondazione Politecnico di Milano it is possible to find the different types of macro-level development process.

- Waterfall was founded in 1970 by W. Royce and was initially applied to the manufacturing sector. It consists in a series of well-defined steps: requirements, design, implementation, verification, and maintenance. Through this approach, each phase must be completed before starting the next phase. It is suitable for a stable development context where it is possible to predict and define the outputs before starting;
- Agile development divides the project in sections and for each section the sequence define, build, test and release process is followed. Agile development means collaboration and fast changes in order to reply to/face with the needs of the market. It is based on the lean approach.
- There is a third methodology called the Lifecycle method. It divides the development into different sections: the requirement development phase, in which the requirements are well defined; the ontological analysis phase, in which the perimeters of the ontology are defined; the design phase, in which all the design actions are decided. In Figure 1 is shown the lifecycle process defined in the ontology summit of 2013.

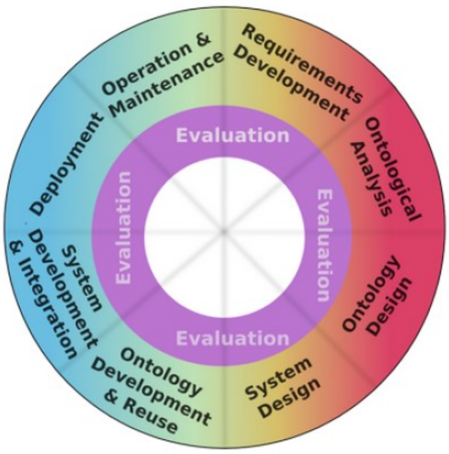

Figure 1: Life cycle process

The waterfall and the agile methodologies are well known methodologies, because they are used in the digital project development. No approach is better than another, the development method must be selected considering the characteristic of the project.

1.4.Specific ontology methodologies

Since the beginning of the ontology science (the 1990 more or less), scientists started to create some model to facilitate and standardize the creation of new ontologies. The article (Al-Baltah and Ghani 2014) tries to classify and evaluate some models proposed over the last 30 years. The study suggests a method based on eight criteria applied to five most famous models. The criteria used to compare the development methods are:

- Specification intended as the process of identifying the purpose of the ontology for which it will be used and scope of ontology by knowing uses and users of the ontology;
- Conceptualization intended as the process of structuring the domain knowledge that has been acquired in a conceptual model and that provides necessary description about the problem and its solution;
- Formalization, that is the activity in which the conceptual model is transformed into a semi-formal model;
- Implementation, that is the ontology codified using a representation language;
- Maintenance intended as the process of updating or correcting the implemented ontology;
- Knowledge acquisition, that is the process of acquiring the ontology knowledge also from different sources;
- Evaluation, that means the evaluation to judge the quality of the ontology;
- Documentation, that is all the necessary manuals to facilitate the use.

The interesting result is that the model which supports the greatest number of criteria is the Methontology like shown in Figure 2.

Activities	Uschold and King	TOVE	Methontology	Brusa *et al.* (2008)	UPON
Specification	√	√	√	√	√
Conceptualization	p	p	√	√	√
Formalization	p	√	√	p	√
Implementation	√	√	√	√	√
Maintenance	x	x	p	p	x
Knowledge acquisition	√	p	√	√	√
Evaluation	x	x	√	√	√
Documentation	√	x	√	√	√

√: Supported; p: Partially supported; x: Unsupported

Figure 2: Specific methodologies comparison

The result of this investigation is that Methontology seems to be the best approach. It is widely investigated in the paper (Fernandez, Gomez-Pearez, and Juristo 1997) that explains all the steps of the method.

1.5.Ontology in manufacturing domain and their requirements

There are many reasons why the ontology is a useful tool in the manufacturing domain. First of all, we have to remember that ontology is a semantic tool used in the data integration. It means that the first scope will be the correct communication and exchange of data between the different technological instruments inside a certain manufacturing environment. There is a second very interesting aspect, that is based on the knowledge. Due to the increase of the pervasive system into the manufacturing industry, we need a method to extract knowledge in a unique and shared way, in order to correctly classify the data arrived from the field. To resume, the two main reasons are the data integration and the knowledge extraction.

In the article (Negri et al. 2016) the authors deeply analyse the available literature and they develop a four-points model in which they explain the main requirements needed to an ontology in a manufacturing domain. The requirements emerged are very interesting:

- The conceptual and data storage model, that means the ability to properly represent the environment through a conceptual model and sharing knowledge through the different actors of the system. The storage model is fundamental to carry out analysis on past phenomena;
- The easy use and maintain, that can be translated into properties like intuitiveness, object-orientation, scalability;
- The support reasoning, that is basically a way to create knowledge;
- The interoperability, that is translated into machine readability.

In the paper they analyse the different type of languages in order to meet the above requirements. However, I believe that, before the selection of the languages, we have to think about the possibilities that an instrument like the ontologies give us in order to understand a domain. The applications in the manufacturing domain are enormous and varied and through the systematic literature review we will better understand the current situation.

1.6.Ontology languages

Ontologies are part of the semantic web, and they could be built with different methodologies, different purposes, and different languages. This paragraph examines the most used and interested languages supporting the implementation process of ontologies. Starting from the paper (Negri et al. 2016) that proposed a systematic literature review on the ontology languages used into the ontology development, a list of the most used languages is proposed In the Table 1to clarify the role of a language and its characteristics.

Language	Description	Reasoning support
KIF	It means Knowledge interchange format, and it is a formal language developed for the interchange of knowledge among disparate computer programs. It allows a conceptualization of the world in terms of object, functions, and relations. It is very simple, and it has a limit expression power.	No
XML	It is one of the most famous informatic languages of the world. It means extensible markup language and it is used to define the content of structured document and it could be used also into the ontology development. It is not only and ontology language, but it used to ontology development. It has a lot of issues like the lacks in semantics or difficulties when a new term is introduced.	No
RDF	It means resource description schema and it provides a simple data model. It is used to describe information about web resources. It is intended for describing data and facilitating data exchange. It is a data model for relation between things which also has an RDF schema and Vocabulary.	Some inference engines mainly for constraint checking
OWL	It means web ontology language and it is the main standard for the ontology development. It is an extension of the RDF but has more expression power. It includes classes, operations, and action between them. It allows mapping between ontologies, and it has reasoning power.	It allows complex reasoning about documents
OWL Lite	It is an OWL with limitation in the constructs.	Guaranteed termination and efficiency in reasoning
OWL DL	DL means description logic and it has more expression power than OWL Lite. It has balance between expressivity and computational completeness.	Decidable reasoning
OWL Full	With OWL Full the classes can be represented as an individual or a property.	Undecidable.

Table 1: Languages of ontologies

Nowadays the most used languages are all based on the OWL family, due to the expression power, the simple usability, and the reasoning engine. The editors like Protégé help the users throughout the development and composition of the ontology.

1.7.Goodness of ontology

The previous chapter dealt with the specification of a conceptualization in a certain domain, but when does this ontology really define and explain the knowledge of a certain domain? It is possible to have good and bad ontology. For the ontology, the equivalent to 'not compile' of a piece of code is when there is a violation of the syntax (C. Maria Keet 2020). Obviously, the ontology must be as near as possible to the real world, and a margin of errors is intrinsic to the development. Syntax violation can occur, for example, when a class named student must not be a relation between two other classes/must not have a relation with two other classes. Another type of error is defined when there are physical differences in the ontology structure and in the analysed real domain. In this case the errors are not conceptual, but physical. In conclusion, ontology models could be classified in the good ontology when what it is represented in the ontology covers the real domain and describes only the real domain.

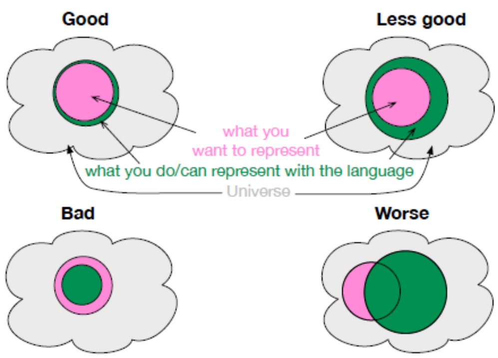

Figure 3 Adapted from C. Maria Keet 2020

Here above the Figure 3 proposed by C. Maria Keet, in which there is the classification of ontology goodness:

- Good, focuses only on the domain with a correct coverage;
- Less good, focuses only on the domain, but it doesn't cover all the domain;
- Bad, focuses also out the domain;
- Worse, focuses out the domain and without a correct coverage.

1.8.Follow up step

At the end of this chapter the concept of ontology must be quite clear, but it is also true that a lot of variables are involved in its development. To understand which are the real implementations and the most interesting variables, it is necessary to analyse the application of the ontologies in the scientific literature and this will be the aim of the next chapters.

2. Literature review process

The analysis of the literature is fundamental for two main reasons: firstly, it helps us to understand the current situation of a certain topic; secondly, it can be useful to predict the future trend of the same topic. The other reasons why an analysis of the literature is extremely important can be found in the paper (Snyder 2019) and these are, for example, the identification of critical points in past researches or the reusing of some scientific results. The strength of this approach is that it organizes a process in defined steps, and it eliminates the subjective point of view in the research protocol.

The systematic literature review carried out in this thesis aims to investigate the current situation of ontology in the reliability and maintenance field. Ontologies are a well-known topic in the digital technology/informatic sector, in which a lot of people give their contribute to its scientific development. The interest of this research is based on the maintenance and reliability field in the industrial sector, so the analysis will be reduced only on that field. A lot of different applications of ontology in the industry sector are expected to be found, from the knowledge extraction to the digital simulation. The following chapters will provide the definition of the research methodology references and the specific research methodology applied.

2.1.General Systematic Literature Review methodology

As methodology reference, will be followed the one presented in the (Brereton et al. 2007), Instead as guide line will be used the (Polenghi, Roda, Macchi, and Pozzetti 2021). It is a very recent work, since it was published in 2021, and it analyses the information as a key dimension to develop industrial asset management in manufacturing. The aim of the paper is to find gaps in asset management literature and provide a summary of challenges and advice for future development. The research methodology used in the reference article is composed by four main steps: the definition of a framework, the definition of the research protocol, the systematic review implementation, and the research content analysis.

2.1.1. Framework

The framework is an overview of the different areas of reference in which the research could be interested. The creation of the framework is an iterative process that involves the analysis of both scientific literature and the ISO. The analysed work proposed a list of papers related to a specific area of interest with a small description. This part will not be very interesting in the thesis SLR because the thesis has a precise topic, so there are only few areas involved.

2.1.2. Research protocol

The first step of the research protocol is to define the eligibility criteria. The authors of the reference papers proposed three main eligibility criteria:

- Only English-written documents with full text available;
- Peer-reviewed journal papers and conference papers;
- Papers dated 2008 forwards.

In this phase it is also useful to define the scientific libraries to query in order to achieve a good result. The scientific repositories selected were Scopus and Web of Science (WoS).

2.1.3. Systematic review implementation

This part is the most interesting one, because it is necessary to implement the research protocol as its first step. The result of the research protocol is a high number of papers. The second part of this phase includes the duplicates removal, eligibility criteria application, title and abstract screening, full text reading analysis and snowball analysis. To understand the correct flow, Figure 4 shows the systematic review implementation steps.

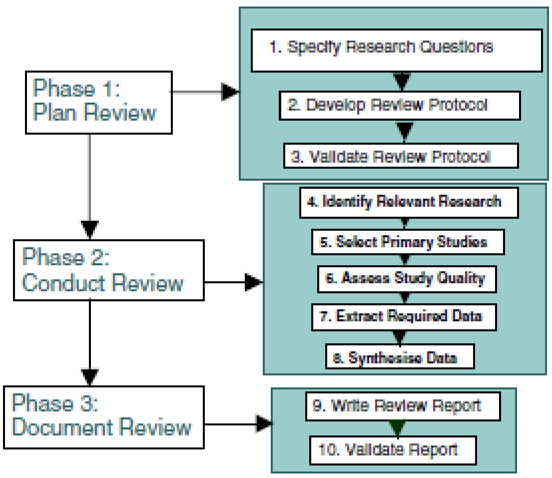

Figure 4: Adapted from Brereton et al. 2007

2.1.4. Research content analysis

The last part of the SLR is the description of the eligible documents. The content analysis is divided into documents statistics, in which all the possible statistics on the document are extracted and the description of the content. The reference paper proposed statistics about the number of documents per period and the type of the document per period, and it proposed statistics about the topic of the papers. The second part of this step is the real content analysis, in which all the paper are deeply analysed to arrive to some final conclusions.

2.2.Specific Systematic Literature Review methodology

The research methodology used in this thesis complies with the steps explained in the research reference paragraph, but with some customization due to the related topics and the structure of the thesis.

2.2.1. Framework

In this case the framework analysis is not relevant, because the addressed topics are the maintenance and the reliability in the ontology field. The perimeter of the analysis will relate only to these two main topics and without other topic constraints.

2.2.2. Research protocol

The first step is the planning of the eligibility criteria. In the current SLR, it was decided to use the following eligibility criteria:

- Only English-written documents with full text available;
- Journal papers and conference papers, book chapters;
- Only engineering, maintenance, industrial documents.

These first defined eligibility criteria will be useful for the research on the online publications archive. The Databases selected for the analysis are Scopus, WoS and IEEE Xplore (IEEEX). The words used in these databases and their combinations are defined in Table 2, where all the combinations are shown.

Ontolog*	AND	Maintenance	AND	Industry
	AND	Maintenance	AND	Manufacturing
	AND	Maintenance	AND	Process
OR				
Ontolog*	AND	Reliability	AND	Industry
	AND	Reliability	AND	Manufacturing
	AND	Reliability	AND	Process

Table 2: Query decomposition

The complete query used in the databases was:

Ontolog* and (Maintenance or Reliability) and (Industry or Manufacturing or Process)

2.2.3. Systematic review implementation

The SLR implementation starts with the application of the research protocol, and, after that, all the activities of selection and cleaning are performed. To simplify, the process is represented in Figure 5.

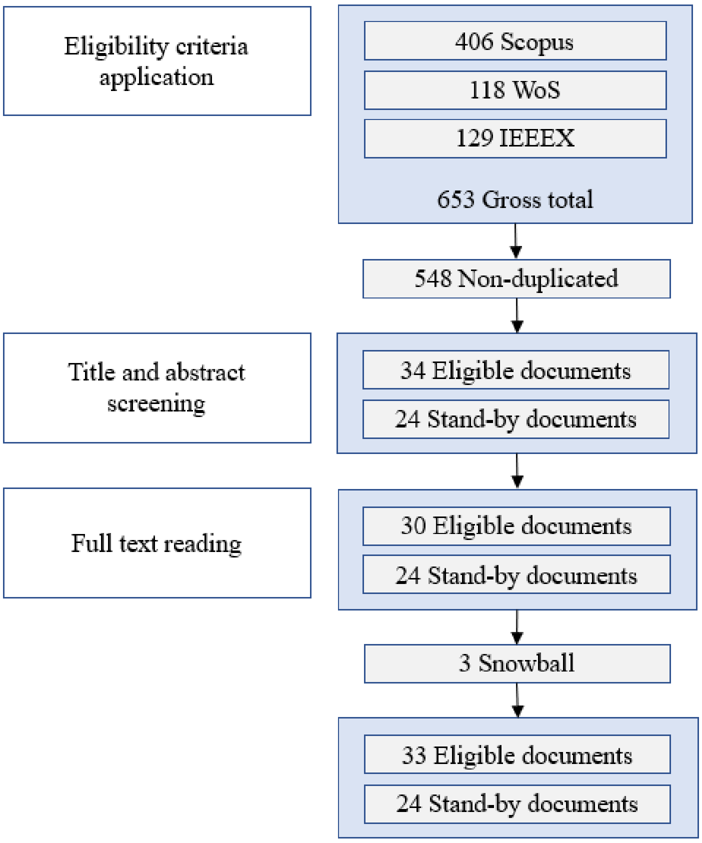

Figure 5: My SLR implementation steps

The procedure followed is the same one applied for the research protocol, except for two main things. The first difference is that the title and abstract screening was not divided into title elimination and abstract elimination, but the process was contiguous. The second main difference is that in this research there is a batch of articles called stand-by documents that is a batch of interesting papers that are not coherent with the research topic but interested in the ontology field. These papers were allocated into a batch called stand-by because they could be useful in the evolution of this work or in another future research.

2.2.4. Research content analysis

The research content analysis will be discussed in a dedicated chapter with a subdivision between the statistical analysis and the content analysis. The Statistical part will show the most interesting trends, instead the content analysis will analyse the content of the papers in a detailed way.

3. Descriptive statistics

Statistical analysis can be defined as an analysis developed on scientific literature with different aims. One of them is emphasizing the most interesting trends and macro variables which occurred, another one could be supporting the understanding of the scientific literature. The articles used in this examination are the results of the methodological research discussed in the previous chapter. Per each article, in the first step of the statistical analysis, a series of data was extracted to have a preliminary and overall point of view. In the second step a deeply analysis was carried out to better understand the ontology structure where possible. Finally, the articles were categorized in two different ways; the first based on the application field and the second on the ontology role.

The descriptive statistics chapter is divided in three main sections. The statistical analytics in which there are all the statistics about the paper characteristics and its content (definition of ontology, period of analysis, languages used, and layers of ontology). The second section is the Taxonomy section, in which an analysis of the terms and composition of the ontologies is provided. The last section is a categorization of the papers with two main methods based on application field and role of ontology.

3.1.Preliminary statistics

This chapter is composed by five main chapters. The first one is the period of analysis that is the chapter in which is explained the period in which the articles were published, and few statistics are carried out to understand the trend of the topics during the years. The second chapter is about the categorization of the paper through two different methods the application field that means the field in which the papers apports its contributes and the role of ontology that it means what is role of the ontology, proposed by the authors inside the paper. The third chapter is called definition of ontology and it is a statistic about the definition used inside the papers to describe the ontology. The fourth statistics is the language used statistics based on the type of languages used inside the papers to develop and to support the ontology. The last but not the least is a statistic about the number of layers that the ontologies have inside the analysed papers.

3.1.1. Period of analysis

The period of analysis is based on the release date of the papers. With this information it is possible to understand in which phase of the topic lifecycle we are today. The years were divided into 4 main period: each period contains 5 years, starting from 2000, and the only period which contains 6 years is the last one, since it is from 2016 to 2021, extremes included. Figure 6 represents the number of articles per period.

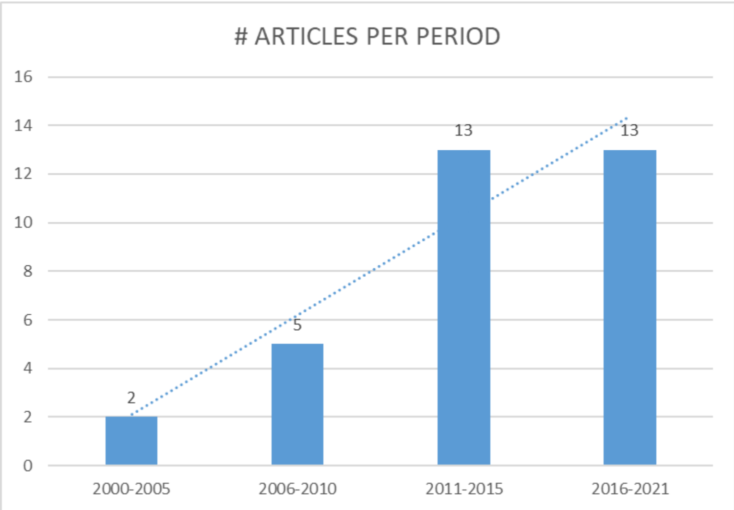

Figure 6: Period of analysis graph

The trend is very clear, there is an expansion of ontologies into the maintenance and reliability field. According to the diagram, the number of cited papers increased over the years, reaching 13 papers after 2010. In addition to this, the diagram does not show a decreasing trend, and it could mean that the topic is not into the maximum expansion yet.

3.1.2. Categorization

The papers were categorized with two different methods. The first method included the application field of the ontology, while the second method included the ontology role in the paper. The application filed categorization is divided into four main sectors, which are maintenance, risk management, data analysis, and process optimization. The ontology role categorization is divided into three main roles, that is sharing knowledge, storage knowledge, and using knowledge.

Application field

Application field refers to the field in which the innovation approach of the papers is applied to bring a new vision. In maintenance field it is possible to find the papers regarding the improvement of a type of maintenance approach, or papers which bring new technology inside the maintenance sectors. The largest part of these articles is regarded to the improvement of prognostic health management. Risk management refers to all those activities that want to mitigate the risk inside the usual operations. The papers covered a lot of sectors, from electrical products to medical sector, but the larger involved field is the FMEA analysis. Data analysis is the smallest batch of papers, and here are categorized the articles that use the existing data to improve the reliability or other properties of the system. An example is the use of the ontology to studying the environment. Last but not least, the process optimization is the biggest section. In process optimization it is possible to find all the articles related to simulation, optimization of existing activities, innovative approaches. All these articles bring to the existing system an innovation related to the ontologies. In Figure 7 the subdivision of articles per role.

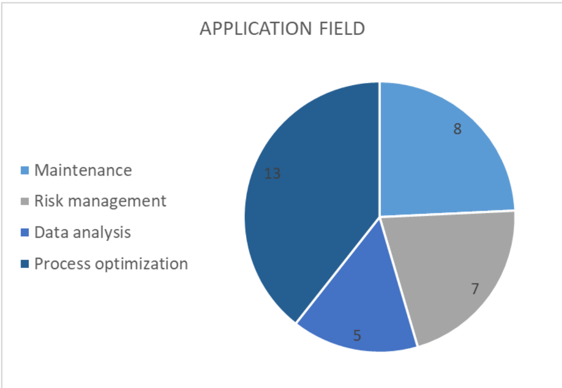

Figure 7: Apllication fiels pie chart

As shown in the graph, most of the articles proposed an innovation of existing process so an optimization.

Role of ontology

In all the articles it is possible to find different roles for the ontology. The main roles were the sharing knowledge, which means to give a common understanding for all the actors on a precise domain; storage knowledge which means to create a repository of knowledge to be used when it is necessary to understand some behaviour of the system; using knowledge, meaning that there is an existing common knowledge, and it is used to improve the system. It is possible that a paper presents an ontology with more than one role, in this case only the principal role is considered for this subdivision. In Figure 8 the ontology role classification of the papers is shown.

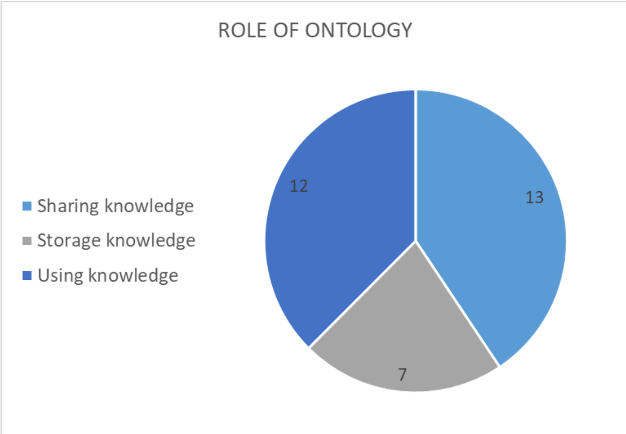

Figure 8: Role of ontology pie chart

The sum of the papers is 32 and it means that one papers is not classified, and it is correct because it has not an ontology model, but it has a petri model. It is included in the analysis because it represents an interesting model, coherent with the thesis topics. The roles are quite balanced, probably the storage is not the first topic because there are other tools used for this function.

3.1.3. Definition of ontology

The expression "definition of ontology" commonly refers to the way in which authors had to explain what an ontology is. There are a lot of definition of ontology, and it is interesting to understand which one is the most used. The definition proposed are listed in the Table 3.

Author	Definition
Araujo	"An ontology formally describes the knowledge through concepts and relationships that exist in a particular area of concentration or specific domain"
Borst	"An ontology is a formal, explicit specification of a shared Conceptualization"

Gruber	"An ontology is an explicit specification of a conceptualization"
Guarino	"Ontology is the specific canonical definition of conceptual model"
Studer	"An ontology is an explicit and formal specification of a shared conceptualization"
Wikipedia	"Ontology is a formal representation of a set of concepts within a domain and the relationships between those concepts. It is used to reason about the properties of that domain and may be used to define the domain"
Zuniga	"Ontology is an axiomatic theory made explicit by means of a specific formal language"

Table 3: Ontology's definitions

Table 4 shows the number of citations per single definition of ontology. The definition is associated to the author of the paper and the one which is most used is the definition of Gruber. The second most used definition is the Studer one, but the gap with the first is very large. Another interesting aspect is that there are 15 papers without a clear reference to an existing definition. It means that inside the field there is not a completely and shared definition.

Araujo	Borst	Gruber	Guarino	Studer	Wikipedia	Zuniga	None
1	1	10	1	1	1	1	16
3%	3%	30%	3%	3%	3%	3%	48%

Table 4: Ontology's definitions statistic

For most authors, ontology is a specification of a conceptualization. But what does this mean? To answer this question, it is important to underline the fact that there is not a unique interpretation of this definition and, as a matter of fact, many authors decided not to use it. Today we can define an ontology as a logical theory and a description logics knowledge base (C. Maria Keet 2020).

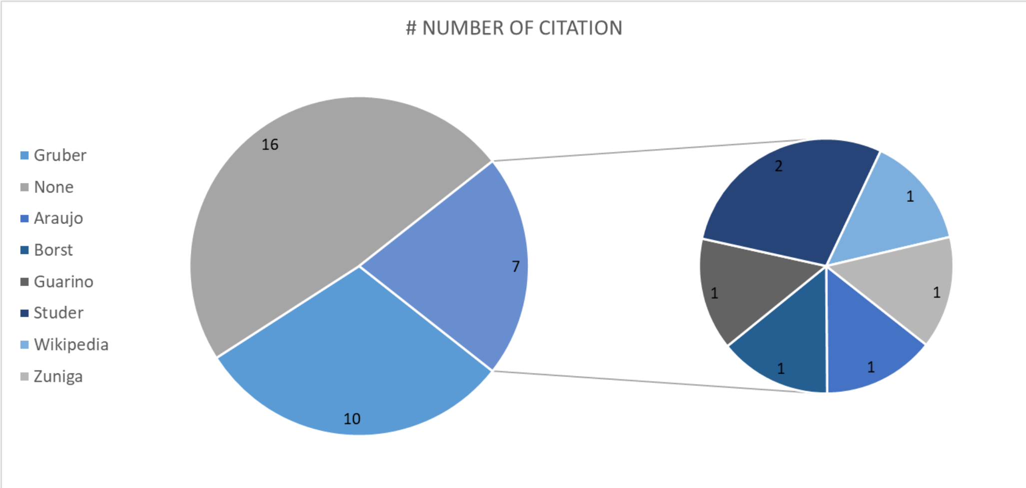

Figure 9: Number of citation pie chart

3.1.4. Languages used

The statistic on the languages is divided into two steps. The first step consists in the analysis of the main language, which is generally the language that creates the ontology. The second one is dedicated to the analysis of the other languages, which are not less relevant, but they are at support of the first main language for the creation of an ontology that meets the requirement of the study. When the first language is not sufficient, it is integrated with the other languages.

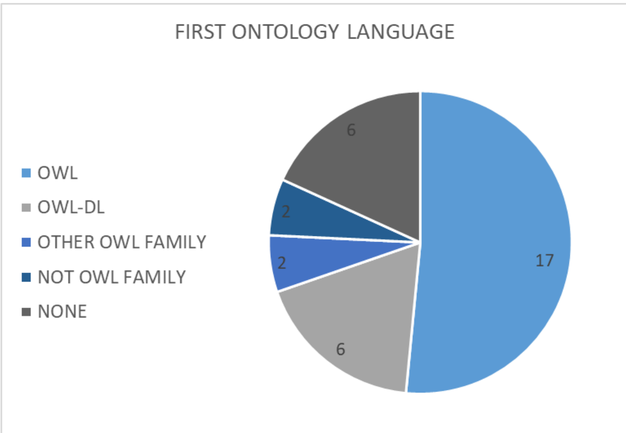

Figure 10: Primary languages pie chart

Figure 10 represents the statistic on the first ontology language. It shows that the most used language is the OWL (Web Ontology Language), which is used in 17 papers out of 33. The second most used language is the OWL-DL: it is part of the OWL family, but in this case there is a description logic improvement that allow the users to have a better expression power. If the analysis is carried out on the family language, OWL counts 25 papers out of 33. The articles without language are those that do not provide an ontology model, or they discuss only a methodological aspect. Now Figure 11 shows the statistic of the second part of the languages analysis.

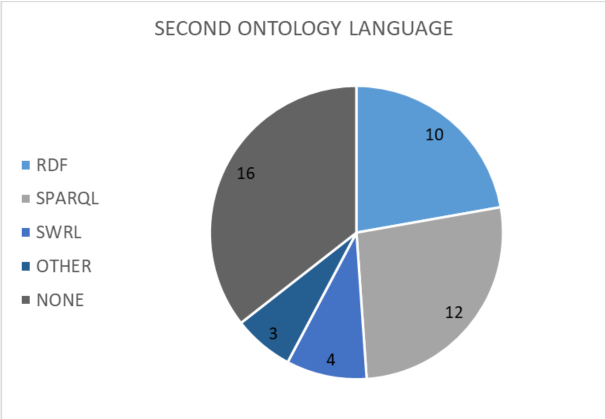

Figure 11: Secondary languages pie chart

The pie chart shows how two of these languages are very important for the support to ontology. The first one is SPARQL. It is a language that allows the user to query a storage repository. It is very powerful to extract information. The second most important support language is the RDF. The RDF is a language which enables the user to better represent the syntax of the model, defining a formal schema. A half of the article doesn't use a secondary language, and it means that there is not a mandatory relationship between a first language and a second language, it depends on the needs and the scope of the authors. It could also mean that the synergies between a first and a second language are not useful into the proposed papers.

3.1.5. Layers of ontology

The layer analysis starts from the analysis of the ontology proposed on the papers. The ontologies could be multi-layers, with a tree structure. All ontologies start with the concept "thing", which is the main concept that will be specified with the other subsequent layer. So, the layers are a sort of clarification of what is the initial thing. Generally, more layers mean more depth analysis, but this is not always true. In this analysis I decided to classify the ontologies by the number of layers that they have. Due to the different fields found in the SLR and to the period of release of the papers, the structure of the layers is quite different per each paper, but this type of problem is not addressed, because the layer analysis will be deeply focused on the following statistical part. In this part of the analysis only the cold number of the ontology layer proposed was investigated.

Another clarification is that in this analysis the upper levels are neglected. The expression "upper level" generally means all the levels from the thing concept to the fist level proposed, excluded. All the ontology levels, such as the Basic Formal Ontology, the Extended Relation Ontology and all the family of Common Core Ontology, are neglected because usually they are not discussed throughout the paper, and they don't produce significant information on the maintenance and reliability domain. To clarify this aspect, it is possible to take as reference the Figure 12 proposed in the paper (Polenghi, Roda, Macchi, Pozzetti, et al. 2021) where all the level are classified.

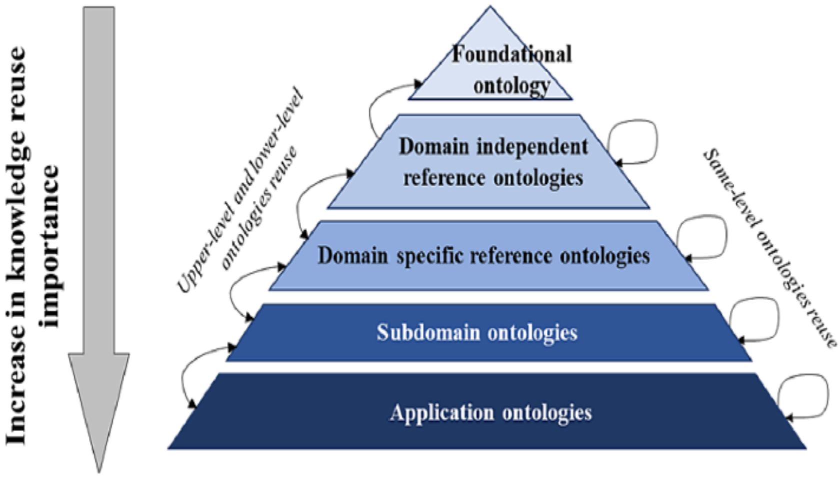

Figure 12: Ontology levels

The first level from the top, included all the foundational ontologies like the BFO, the second level included ontologies like the time or geospatial ontologies. These two first levels are excluded from the analysis. The third level is partially included, some ontologies reuse some domain specific reference ontologies. The most point of interest is about the sub domain ontologies level and part of the application ontologies level. Inside the level of interest, the analysis of layers is carried out. The result of the layer analysis is shown in Figure 13.

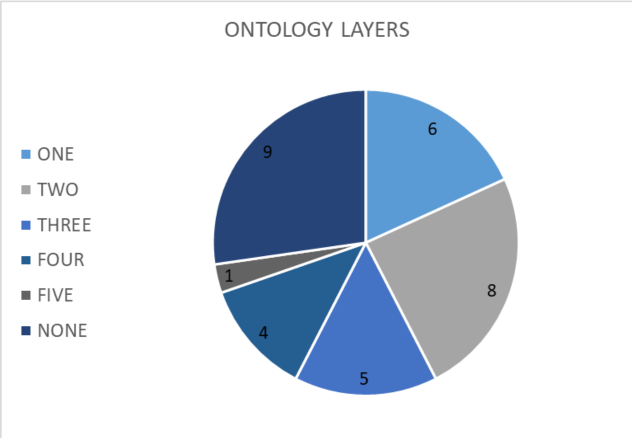

Figure 13: Layers pie chart

According to the figure, the most popular number of layers is two, but there is not a prevalent trend that overwhelm the other number of levels. Inside the *NONE* category there are all the papers that do not have an ontology model or do not specify the number of layers of a hypothetical explained model.

3.2.Taxonomy

The second analysis, as anticipated in the previous chapter, will be focused on the layers of the different ontologies. As explained before, the ontology layers are in an upper part quite standard or at least very similar, and in the bottom part are very customized. For this reason, in the preliminary analysis of this thesis it was decided to study only the two middle layers that have a good grade of similarity between the different models proposed. The idea is to find where the main differences are into the taxonomy of the classes and provide some suggestion for the future work.

The analysed layers are two, in the middle between the upper generic levels and the bottom specific layers, as represented in Figure 14.

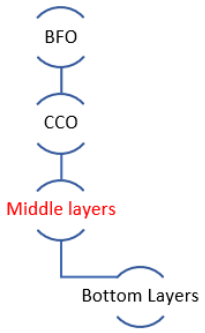

Figure 14: Hierarchy chart

This analysis is carried out only on the most used and general classes found in the analysed papers.

3.2.1. First level analysis

The first level is the first of the two middle level cited before. It is divided into four main parts: in the first part there are the concepts *Index of risk* and *FMEA,* whereas the second part is composed by the four main elements of manufacturing and production system *Product, Process, Person, Asset*. The third section consists of *System, Function, Action, Component* and the last one is composed by *MaintenanceOperation, Failure, StateOfAsset,* and *Solution.* The number of classes for both middle layers is described in Figure 15 On the left there is the first middle layer and, on the right, the second one. The higher number of articles in both the middle layers is restrained in the slot of classes between 4 to 6.

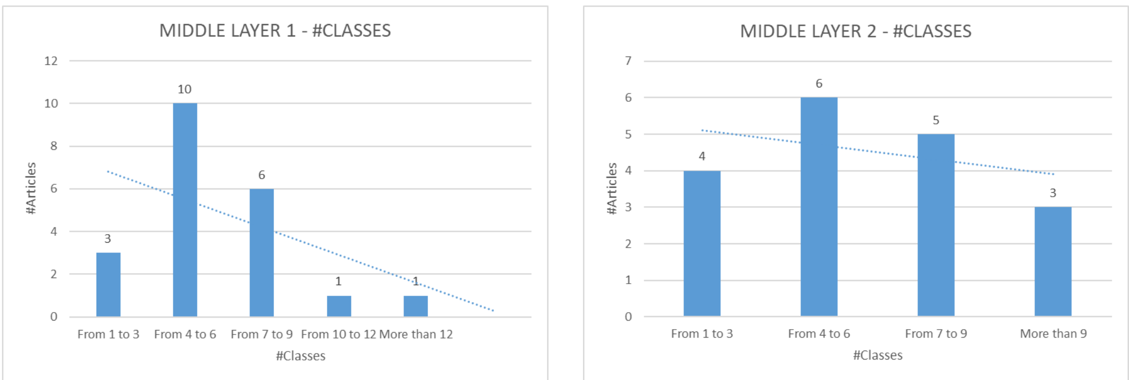

Figure 15: Numbers of classes chart

In the following part of the chapter, the terms used to describe a concept are represented in a table with the related papers. The aim of this procedure is to catalogue the terms in order to achieve a unique way to describe a concept.

Table 5 includes all the terms used to represent the concept of risk index.

Paper	Terms
(Rehman and Kifor 2016)	RPN
(Cho et al. 2020)	RiskPriorityNumber
(Ren, Ding, and Li 2019)	Hazard

Table 5: Taxonomy - Risk index terms

The first two rows represent the risk priority number taxonomy. As shown in the table, there is not a unique way to define it, the first one is the acronym and the second one is the extended word. The last two rows represent the risk index, in one case it is indicated with the word *Hazard* and in the other case with the word *KeyRiskIndicator*.

The concept of FMEA is described by different terms listed in Table 6.

Paper	Terms
(Mikos et al. 2011)	FMEADescription
(Rehman and Kifor 2016)	FMEA
(Rasovska, Chebel-Morello, and Zerhouni 2005)	FMECA

Table 6: Taxonomy - FMEA terms

About FMEA, the descriptions proposed are not very explicative, because they are different and there are numerous types of FMEA. It is not possible to represent the concepts only with the terms FMEA.

In the following tables there are the main elements of production systems, starting from the concept of product in Table 7.

Paper	Terms
(Mikos et al. 2011)	Product
(Harcuba and Vrba 2015)	ProductType
(Li et al. 2017)	Product

Table 7: Taxonomy - Product terms

For the product concept the classes used are *Product* and *ProductType*. The class *Product* identify in a unique way the concept, whereas the *ProductType* seems to be referred to the typology of the product and instead in this case it is referred to the product.

The most used concept inside the analysed paper is the process. The terms that represent the process are listed into the Table 8.

Paper	Terms
(Mikos et al. 2011)	Process
(Hodkiewicz et al. 2021)	Process
(Rehman and Kifor 2016)	Process
(Hodkiewicz et al. 2020)	Process
(Harcuba and Vrba 2015)	Process
(Cho et al. 2020)	Process definition

Table 8: Taxonomy - Process terms

In the case of the process the scientific literature is quite homogeneous. In five papers out of six the term used is *Process* and only in one case a different term is used. For this concept there is a good level of uniformity.

The terms used to represent the concept of person that operate inside the environment are shown in Table 9.

Paper	Terms
(Banujan and Vasanthapriyan 2020)	Actors
(Cho et al. 2020)	Individual person
(Arena et al. 2017)	Operator

Table 9: Taxonomy - Person terms

The identification of the people concept is not uniform. This case is the opposite of the previous one with the process concept. For the identification of the people concept, the words *Actors, Individual person, Operator* are used. Each of these words has a precise and different meaning. In order to have a unique concept, it is necessary to find a more general and transversal term.

The Asset concept is one of the most interesting, because there is not a unique definition of asset, and the reason is that there is not a uniformity in the level of the classes. As shown in the Table 10, one term which is used is the *ManufacturingItems,* that it could be an asset and in fact in the description of the concept there is also a reference to the asset, but it could also represent other manufacturing items. The same reasoning could be done for the *MechanicalSystem*, which it may represent not only the asset. Does this mean that the terms are wrong? No, they are only too general in this case. Regarding *Asset* and *Machine*, there is not a single term proposed.

Paper	Terms
(Hodkiewicz et al. 2021)	MechanicalSystem
(Hodkiewicz et al. 2020)	MechanicalSystem
(Yew, Foong, and Sivarajan 2019)	Machine
(Cho et al. 2020)	Material entity
(Nuñez and Borsato 2018)	ManufaturingItems
(Jin, Xiang, and Lv 2009)	Asset

Table 10: Taxonomy - Asset terms

The concept of system, intended as the physical space that contains the asset and the stakeholder of the process, is represented by the terms in Table 11.

Paper	Terms
(Aarnio, Vyatkin, and Hastbacka 2016)	Context
(Cho et al. 2020)	System
(Li et al. 2017)	Environment
(Aarnio, Seilonen, and Friman 2014)	System

Table 11: Taxonomy - System terms

The terms proposed don't have uniformity and the same precise meaning. It is important to define the higher level in order to have a good uniformity of meaning.

Function concepts is the ability to carry out an activity to achieve a goal. It could be developed by different actors/objects inside the system. In this case there is a good uniformity between terms and meaning as explained in Table 12.

Paper	Terms
(Mikos et al. 2011)	Function
(Hodkiewicz et al. 2020)	PrimaryFunction
(Jin, Xiang, and Lv 2009)	Function

Table 12: Taxonomy - Function terms

Component concept has a very wide range of meanings and taxonomies. In some cases, the component is the part of an asset, whereas in other cases it is the part of a system. Component is a concept without a clear meaning, there are a lot of interpretation.

Paper	Terms
(Hodkiewicz et al. 2021)	Component
(Aarnio, Vyatkin, and Hastbacka 2016)	CElement
(Hodkiewicz et al. 2020)	Component
(Cho et al. 2020)	SystemComponent
(Medina-Oliva et al. 2012)	Component

Table 13: Taxonomy - Component terms

In addition to this, in the taxonomy the concept of component is not so clear as described in Table 13, it is possible to find the same concept of component described by *Component* and *CElement.*

As far as the concept of maintenance activities is concerned, there are two main terms listed in Table 14.

Paper	Terms
(Jin, Xiang, and Lv 2009)	MaintenanceActivities
(Hossayni et al. 2020)	MaintenaceOperation

Table 14:Taxonomy - Maintenance activity terms

Similarly to the Asset concept, the failure concept is one of the most interesting. The terms used are quite different from each other. The difference is due to the level of description of the ontology. In Table 15 there are the terms used in the first middle level, but it is possible to find some failure classes also in the second middle layer.

Paper	Terms
(Mikos et al. 2011)	Failure
(Rehman and Kifor 2016)	FailureMode
(Rasovska, Chebel-Morello, and Zerhouni 2005)	Failure
(Hossayni et al. 2020)	FailureModel

Table 15: Taxonomy - Failure mode terms

The state of an object represents the ability of that object to perform an activity in that precise moment. If the object is not able to perform, the activity is in a bad state, sometimes called fault. The state of an object could be represented by different terms like in Table 16.

Paper	Terms
(Terkaj, Tolio, and Urgo 2015)	ObjectState
(Hodkiewicz et al. 2020)	ObjectInFaultState
(Yew, Foong, and Sivarajan 2019)	Down
(Jin, Xiang, and Lv 2009)	Fault

Table 16: Taxonomy - State terms

The last concept of the first middle layer is the adjustment after a failure. Table 17 shows the main terms used to describe the repair activities.

Paper	Terms
(Rehman and Kifor 2016)	MitigationAction
(Aarnio, Vyatkin, and Hastbacka 2016)	CSolutionPackage
(Zhou and Ren 2011)	Repair

Table 17: Taxonomy - Repair terms

The meanings of the terms used to describe the repair are the same, but the taxonomy is very different.

3.2.2. Second level analysis

The second level of the two middle layers is more specific than the previous one. As written before, the more the layer is near the bottom, the more the layer is specific to that domain. For this reason, there are less similarities in the concepts. It is possible to identify two macro areas in which we can research the similarities, the *Maintenance,* and the *Failure* one.

Starting with the maintenance part, the first concept to define contains the operation carried out to perform the maintenance (lines one and three). The second concept is the task of which an operation is composed (lines two and four).

Paper	Terms
(Viinikkala, Syrjala, and Kuikka 2006)	MaintenanceOperation
	MaintenanceTask
(Aarnio, Vyatkin, and Hastbacka 2016)	MaintWork
	MaintAcitivity

Table 18: Taxonomy - Maintenance operation terms

Table 18 represents on the right the terms used to represent the concept of maintenance operation and of maintenance task. The taxonomy is completely different between the two papers, but the meaning is the same. Clearly, *MaintenanceOperation* is associated to *MaintWork* and *MaintenanceTask* with the *MaintActivity*.

The failure part is more interesting because there are more similarities in this field.

Paper	Terms
(Mikos et al. 2011)	PotentialEffectOfFailure
(Hodkiewicz et al. 2021)	FailureEffect
(Rehman and Kifor 2016)	Effect
(Hodkiewicz et al. 2020)	FailureEffect
(Zhou and Ren 2011)	Failure effect

Table 19: Taxonomy - Failure effect terms

Table 19 represents the terms used to describe the failure effect, and Table 20 represents the terms used to describe the failure mode.

Paper	Terms
(Mikos et al. 2011)	PotentialFailureMode
(Hodkiewicz et al. 2021)	FailureMode
(Hodkiewicz et al. 2020)	FailureModeObservation
(Zhou and Ren 2011)	Failure mode

Table 20: Taxonomy - Failure mode terms

In both cases the used terms are very different from each other without a unique standard.

3.2.3. The relations

At this point, the last part to analyse concerns the relations between the concepts. The average number of relations inside each ontology model proposed is between 7 and 9 per model. In Figure 16 it is possible to see how many articles are into each slot.

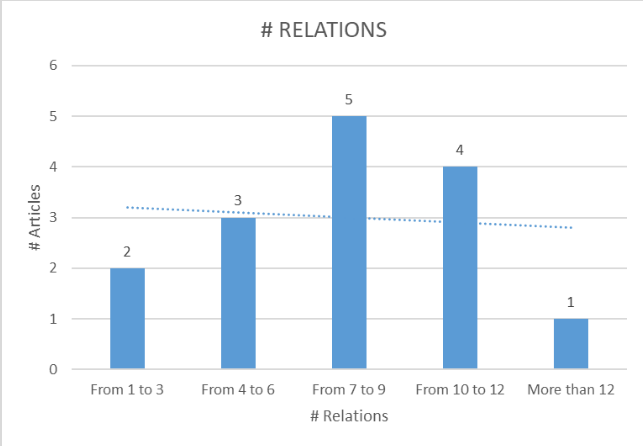

Figure 16: Number of relation chart

Analysing the taxonomy of the relations, it is decided to exclude the relations with less than 3 citations and to include all the relation with more than 2 citations. The most used citations are *hasFunction, hasEffect, hasCause, hasPart.* In this analysis the active and passive form of the verbs are considered the same. If there are two relations that are the same relation, but one in an active form and the other in the passive form, only the active form is considered, and it is considered as one using. In case there is only the passive relation, it is considered as one using at the same level of active relation. Here below the tables with the reference of the relations.

Paper	Terms
(Hodkiewicz et al. 2021)	hasFunction
(Hodkiewicz et al. 2020)	hasFunction
(Jin, Xiang, and Lv 2009)	hasFunction

Table 21:Taxonomy - hasFunction terms

Paper	Terms
(Mikos et al. 2011)	hasEndEffect
(Nuñez and Borsato 2018)	hasEffect
(Jin, Xiang, and Lv 2009)	hasEffect

Table 22: Taxonomy - hasEffect terms

Paper	Terms
(Mikos et al. 2011)	hasFailureCause
(Rehman and Kifor 2016)	isCausedBy

(Yew, Foong, and Sivarajan 2019)	Caused_By
(Nuñez and Borsato 2018)	hasCause
(Jin, Xiang, and Lv 2009)	hasCause

Table 23: Taxonomy - hasCause terms

Paper	Terms
(Hodkiewicz et al. 2021)	functionalPartOf
(Nuñez and Borsato 2018)	isPartOf
(Jin, Xiang, and Lv 2009)	hasPart

Table 24: Taxonomy - hasPart terms

Other interesting relations that have less than 3 citations are *hasOperation, hasFeature, hasState, hasTask, hasComponent, hasDuration, relizedIn.* They were analysed but not considered relevant enough to this work due to the small using.

3.3.Mapping tables

This chapter has the only scope to highlight the work done to arrive at the above cited statistics. The content of this chapter will be a series of tables in which are hold all the statistics. The Table 25 represents the mapping process of the preliminary statistic that is the period of analysis, the definition of ontology, the language used and the layers of ontology. About the categorization the Table 26 shown the first five rows of the table used to categorize the articles. At the end the Tables 27, 28, 29 shown the first five rows of the taxonomy tables.

It is mandatory to state that some tables are been reduced or considered only with the most important columns due to space reasons.

TITLE	DB	ONTOLOGY LANGUAGE	OTHER LANGUAGES	ONTOLOGY LEVELS	DEFINITION ONTOLOGY	YEAR
A Maintenance Demand Analyzer - a Web Service based on a Semantic Plant Model	IEEEX	OWL-DT	SPARQL	2	Gruber 1993	2006
A novel way of integrating rule-based knowledge into a Web ontology language framework	SCOPUS	OWL CUSTOMIZED	SPARQL, SPIN, RFD	1	---	2013
A petri-net-based approach to reliability determination	SCOPUS	OWL-S	---	2	---	2013

of ontology-based service compositions						
A system for distributed sharing and reuse of design and manufacturing knowledge in the PFMEA domain using a description logics-based ontology	WOS	OWL-DL	---	1	Zuniga	2011
A virtual factory approach for in situ simulation to support production and	SCOPUS	---	---	---	---	2015

maintenance planning						
An ontology for reasoning over engineering textual data stored in FMEA spreadsheet tables	SCOPUS	OWL-DL	---	4	Studer 1998	2021
An ontology of manufacturing knowledge for design decision support	SCOPUS	---	---	1	Gruber 1993	2008

An Ontology to Support Semantic Management of FMEA Knowledge	WOS	OWL	RDF/XML - SPARQL	1	Gruber 1993	2016
An Ontology-based Semantic Foundation for Flexible Manufacturing Systems	Snowball	OWL	SWRL-SPARQL	1	---	2011
Application of ontology guided search for improved equipment diagnosis in a vehicle assembly plant	IEEEX	OWL	---	---	---	2009

Bridge Ontology Architecture for Knowledge Management in Bridge Maintenance	SCOPUS	OWL	---	1	---	2020
Building an ontological knowledgebase for bridge maintenance	SCOPUS	OWL	SPARQL, SWRL	---	Gruber 1993	2019
Combining and adapting process patterns for flexible workflow	SCOPUS	KIF	---	2	Gruninger 1996	2000
Context modeling with situation rules	IEEEX	OWL	SPARQL	2	---	2016

for industrial maintenance						
Digitalization and reasoning over engineering textual data stored in spreadsheet tables	SCOPUS	OWL-DL	RDF	4	Studer 1998	2020
Failure mode databases and their knowledge-based management	SCOPUS	XML	---	---	---	2014
Failure ontology of board-level electronic	SCOPUS	---	---	3	Gruber 1993	2011

product for reliability design						
Knowledge-Based Improvement of Machine Downtime Management for IR4.0	IEEEX	OWL	RDFS	---	---	2019
Model based FMEA - An efficient tool for quality management of the free lead soldering	SCOPU S	---	---	---	Wikipedia	2012
Ontologies for flexible production systems	IEEEX	OWL	SPARQL AND RFD	3	---	2015

Ontology Based Collaborative Simulation Framework Using HLA and Web Services	IEEEX	OWL	---	3	Guarino	2009
Ontology for strategies and predictive maintenance models	SCOPUS	---	RDF, SPARQL	2	---	2020
Ontology-Based Environmental Effectiveness Knowledge Application System for Optimal Reliability Design	SCOPUS	OWL	SWRL	4	Gruber 1993, Studer, Nico	2017

Ontology-based industrial plant description supporting simulation model design and maintenance	IEEEX	---	---	3	Gruber 1993	2013
OntoProg: An ontology-based model for implementing Prognostics Health Management in mechanical machines	Snowball	OWL	SWRL, SPARQL	---	Araujo	2018
Process of s-maintenance: decision support system	IEEEX	OWL	PYTHON, JAVA	3	---	2005

for maintenance intervention						
Prognostics assessment using fleet-wide ontology	Snowball	OWL	---	5	Gruber 2009	2012
Semantic integrated condition monitoring and maintenance of complex system	IEEEX	OWL DL	RDF, SPARQL	---	Gruber 1993	2009
Semantic repository for case-based reasoning in CBM services	SCOPUS	OWL DL	RDF GRAPHS - SPARQL	4	---	2014

Semantics-driven knowledge representation for decision support and status awareness at process plant floors	IEEEX	OWL	RDF - SPARQL	2	BorstW	2017
SemKoRe: Improving machine maintenance in industrial iot with semantic knowledge graphs	SCOPUS	OWL	RDF	2	---	2020
Towards ontology-based modeling of technical	SCOPUS	---	---	2	Gruber 1993	2015

documentation and operation data of the engineering asset						
Towards Ontology-Based Software Architecture Representations	IEEEX	OWL	---	---	---	2017

Table 25: Mapping table 1

	APPLICATION FIELD				ONTOLOGY ROLE		
TITLE	**MAINTENANCE**	**RISK MANAGEMENT**	**DATA ANALYSIS**	**PROCESS OPTIMIZATION**	**SHARING KNOWLEDGE**	**STORAGE KNOWLEDGE**	**USING KNOWLEDGE**
A Maintenance Demand Analyzer - a Web Service based on a Semantic Plant Model	X				X		

A novel way of integrating rule-based knowledge into a Web ontology language framework		X				X	
A petri-net-based approach to reliability determination of ontology-based service compositions		X					
A system for distributed sharing and reuse of design and manufacturing knowledge in the PFMEA domain using a description logics-based ontology		X			X		
A virtual factory approach for in situ simulation to support				X	X		

production and maintenance planning							

Table 26: Mapping table 2

PAPER	# CLASSES	CLASSES	FMEA	PRODUCT	PROCESS	PERSON	ASSET
A system for distributed sharing and reuse of design and manufacturing knowledge in the PFMEA domain using a description logics-based ontology	7	Action Failure FMEADescription Function Images Process Product	FMEADescription	Product	Process		
A virtual factory approach for in situ simulation to support production and maintenance planning	8	FactoryObject Representation PropertyState Placement ObjectState StateFrequency					

		RatioMeasure FactoryObjectH istory					
An ontology for reasoning over engineering textual data stored in FMEA spreadsheet tables	4	Process Component MechanicalSyst em AnchillarySyste m			Proc ess		MechanicalS ystem
An Ontology to Support Semantic Management of FMEA Knowledge	6	FMEA RPN Process FailureMode MitigationActio n ControlMethod	FMEA		Proc ess		

Table 27: Mapping table 3

PAPER	# CLASSES	CLASSES	Maintenance Event	Maintenance Operation	Failure Effect	Failure Mode
A Maintenance Demand Analyzer - a Web Service based on a Semantic Plant Model.	6	MaintenanceEvent MaintenanceHistory DeviceIndividual MaintenanceConfiguration MaintenanceOperation MaintenanceTask	MaintenanceEvent	MaintenanceOperation		
A system for distributed sharing and reuse of design and manufact	7	ClassificationOptions LocationOfFailure PotentialCausesOfFailure PotentialEffectsOfFailure			PotentialEffecOfFailure	PotentialFailureMode

uring knowledge in the PFMEA domain using a description logics-based ontology		PotentialFailureMode PrimaryIdentifier SecundaryIdentifier				
An ontology for reasoning over engineering textual data stored in FMEA spreadsheet tables	9	FailureEffect FinalFailureEffect FailureMode ComponentFunction PrimaryFunction Malfunction ObjectInFaultState FailureEvent FunctionalFailure			FailureEffect	FailureMode

An Ontology to Support Semantic Management of FMEA Knowledge	2	Failure Effect			Effect	

Table 28: Mapping table 4

PAPER	# RELATIONS	RELATIONS	hasOperation	hasFunction	hasEffect	hasFeature	hasState	hasTask	hasCause
A Maintenance Demand Analyzer - a Web Service based on a Semantic	6	hasPerformedOperation hasPerformedEvent hasMaintenanceHistory hasOperation	hasOperation					hasTask	

Plant Model		hasCriticality hasTask							
A system for distributed sharing and reuse of design and manufacturing knowledge in the PFMEA domain using a description logics-based ontology	7	hasEndEffect hasFailureCause hasFeature hasLocalEffect hasPrimaryID hasSecundaryID isRelatedToFunction			hasEndEffect	hasFeature			hasFailureCause

A virtual factory approach for in situ simulation to support production and maintenance planning	10	hasStateMachine hasHistory hasState hasStateFrequency hasStateRepresentation hasRepresentations hasStayRatio hasObjectPlacement hasPropertySet hasPlacement					hasState		
An ontology for reasoning over engineering textual data stored in FMEA spreadsheet tables	9	Imports x3 hasParticipant hasFunctionalPart realizedIn hasFunction x2 functionalPartOf		hasFunction					

Table 29: Mapping table 5

4. Content analysis

The content analysis is based on the analysis of the content of the papers. This analysis will provide an interesting point of view on the scientific literature in order to discover trends, information and suggestions that will be useful for the definition of the final result of the thesis. The content analysis is divided in four main paragraphs in conformity with the ontology role categorization proposed in the previous chapter.

4.1.Maintenance

Over the last twenty years, the role of maintenance in enterprises has become more and more important from a technological and economical point of view. Technical and management contributes improves the maintenance strategy and the maintenance performance, they increase the reliability, reducing maintenance costs. In this chapter, all the articles related to the maintenance and ontologies will be analysed. The chapter is divided in two main sub-chapters: the first one is related to the support of maintenance decision making, while the second one is related to the predictive and prognostic health management.

4.1.1. Support maintenance decision making

Operations and maintenance activities are very important in a cost saving and optimization perspective, enhancing the productivity. Coordination lacks between operation and maintenance often causes unnecessary costs. Today web services are the tools used to link different and heterogenous systems. The aim of (Viinikkala, Syrjala, and Kuikka 2006) is to provide semantic interoperability in a web compatible form. The idea is to use these technologies to improve maintenance process combining various maintenance and condition monitoring. The authors proposed a plant model ontology divided the maintenance ontology and the process ontology into two sub-ontologies. A semantic plan model is a semantic representation of the process, defining independently the concepts. The lower ontologies are used to represent basic data and to explain the process activities during the execution. With this structure, the ontology will be able to combine the different maintenance policy and analyse the field data. This ontology is built in OWL-DL language using Protégé and it is accessed by services. Focusing on the lower-level ontology, the maintenance ontology contains maintenance history and predictive maintenance schedules; in fact, a maintenance record is created inside the historical repository after a maintenance operation. Instead/On the contrary, the process ontology is a representation of the process structure and device positions. The process plant service implements a specialized interface to query the devices, the maintenance to query the historical maintenance records. In addition to these two specialized interfaces, the author proposed a generic interface that allows more flexibility in query and process analysis. The queries are implemented in SPARQL and are used to access the semantic plant model. On the base of the queries' results, a preventive maintenance schedule is suggested from the system to the operators. The result is that the semantic web ontologies could be used to reduce the heterogeneity of the systems and to explicit implicit knowledge. The same aim of this previous paper can be found with a different proposed solution in (Aarnio, Vyatkin, and Hastbacka 2016). The paper wants to describe how the knowledge modelling approach architecture can be implemented using ontologies. The main contribution of the paper is a context modelling approach that supports information integration from different data sources and easy extension of the model with new annotation concerning any elements of the contextual environment of maintenance worker. After that, the context knowledge represented in its elements is organised in a hierarchy of context classes and predefined queries, situation rules and validation constraints can be attached to related context classes. To create a contextual knowledge is necessary to access at different information inside a system, in fact maintenance operations require information from different and heterogenous sources like plant model information systems, condition monitoring system, document catalogues. The idea is to automate this information retrieval and automatically provide relevant information supporting the task at hand. The solution proposed was the semantic gateway shown in Figure 17.

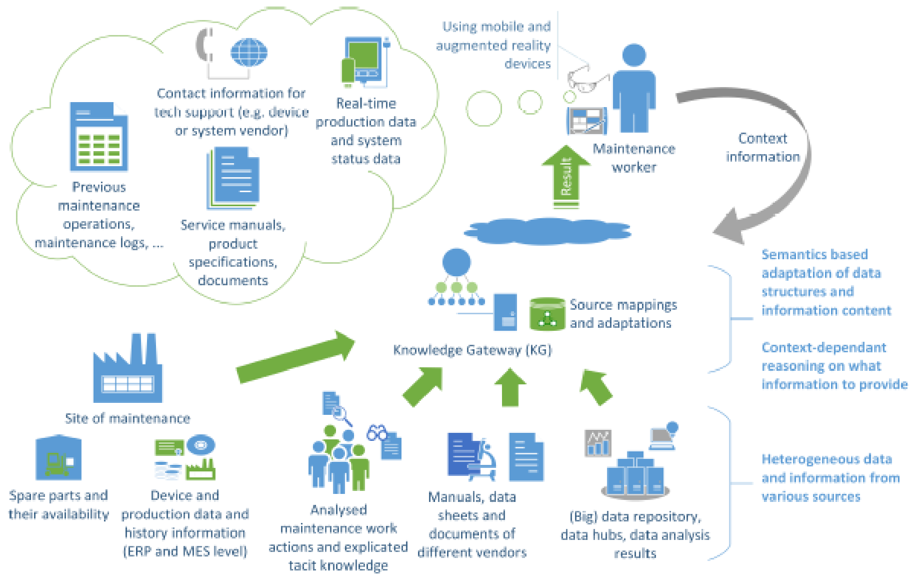

Figure 17: Knowledge gateway functioning

The knowledge gateway acts like a mediator between the parties, on one side there is the field with the field devices or the technical documentation and on the other side there are the maintenance workers. In the middle part of the article the maintenance technician's context was described, highlighting the importance of correctly defining the context and apply this definition to the maintenance field having a maintenance situation context modelling whose aim is to provide relevant situation dependent support information for field service personnel during maintenance work. The authors proposed a situation assessment context (SAC) that categorizes the context element into five main groups, extended to ten new categories for the maintenance context. At this point the main idea of the paper is clear, but it is necessary define some requirement that the knowledge gateway (KG) has to comply with. First of all, the access to KG must be unique via SPARQL with the possibility to filter the information. Other important requirements are that the context model must be update and it could be used to create reports and that the KG have to provide flexibility in the means of maintenance operation. The ontology to support this context model architecture is divided into two parts, that is the upper level (most generic level) and the lower level (the most specific one). A contextual reasoning procedure is carried out into the KG thanks to some rules. To better classify, all the information is in RDF format, and it is accessed in a unique way using SPARQL queries, instead the reasoning provided by KG is carried out for the ontology models. The data are later stored in an RDF repository. In its final part, the paper presents a use-case to demonstrate the technical feasibility of the model.

(Hossayni et al. 2020) presents a knowledge graph developer to improve machine maintenance in the industrial domain. It helps Original Equipment Manufacturers (OEM) to capture, share, and exploit the failure knowledge generated by customers machines located around the world. The authors found interesting motivations for the development of SemKoRe which are, first of all, the already mentioned heterogeneity between different system inside the companies and, secondly, the fact that maybe a single company cannot share the existing knowledge with other companies because they internally stored the knowledge. For these reasons, they adopted a distributed architecture in which knowledge collection is performed on the edge layer and the collected knowledge is shared with other actors through a cloud-base instance. They provided three different configurations for the high-level architecture: the local architecture, in which all machines are connected with a sort of supervisor that stores information; the cloud-based architecture, in which all the machines are connected to the cloud and the hybrid architecture with a first local storage system; and the connection between the local storage and the cloud as shown in Figure 18.

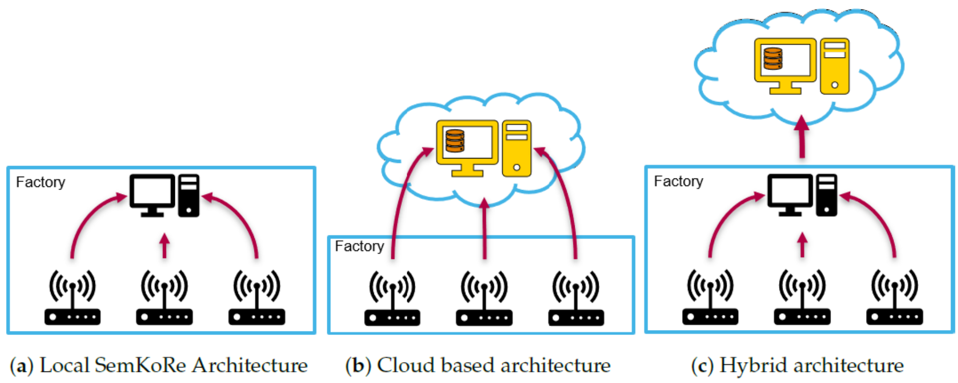

Figure 18: SemKore configurations

The most interesting architecture is the second one, the cloud based. There are three sub-layers that composed SemKoRe: the agent that collects data when failures occur; the server that runs on the cloud and manages failure data producers and it collects data; and a possible local level in which a first SemKoRe server is located. As far as the ontology model is concerned, they defined a data model interacting with the machine builders. At first, the ontology acts as a common data model starting from a flat ontology model with the most common concepts and then acquiring information the ontology grow up thanks to the domain specialists and to the knowledge collection process. A seven-step method followed to construct a first machine failure ontology; furthermore, a machine domain ontology was created to describe the machine components. The SemKoRe process is divided into two main area: the edge area, in which there are the agents, and the cloud where there are the servers. The failure collection starts when a machine failure occurs. After the occurrence of a failure, a survey is carried out to validate the real occurrence of a failure. After the validation, firstly, the data are shared with the server in cloud; secondly, the data are validated in the cloud before the integration with the knowledge graph; finally, data are anonymised to protect the privacy and aggregated to share with the agents the information about similar failures. At the end of the paper an implementation example is provided to describe the go-on steps.

The field analysed by this thesis is not only industrial maintenance, but also the maintenance in general. As a matter of fact, in the SLR it is found (Banujan and Vasanthapriyan 2020) which proposes an ontology driven approach to share the bridge maintenance knowledge through bridge maintenance ontology. To develop the maintenance ontology, the authors started from the contextual information, that is all the data that gives information context to a person, entity, or event. The data collection was done using the grounded theory, which is a technique that involves building hypotheses through methodical data collection and analysis. The relevant data were obtained through formal and informal expert collaboration. A series of competency questions were asked to define and to limit the scope of knowledge that is represented in the ontology. At the end of this definition process, fourteen high level classes were identified with the related relationship between classes and the axioms. The quality of the ontology was verified by an online ontology evaluator, a reasoner and ontology experts. The online ontology evaluator is useful to identify warning when there is a problem into the ontological development, for example when two classes describe the same things. On the other hand, the reasoner is useful to infer logical consequences in the modelled ontology from a set of asserted facts or axioms. The experts were two external people and examined the deficiencies into the ontology model. The conclusion was that knowledge management is the key to share knowledge in a determined field, in this case the bridge maintenance. Another article with the aim to improve the decision making in maintenance field is the (Ren, Ding, and Li 2019). This article is focused on creating a novel technology based on a comprehensive decision-making approach that bridge engineers can employ to obtain knowledge from various fields to make more effective decisions. The authors proposed a unique OWL ontology that represents knowledge in bridge maintenance by managing the interconnected relationships that exist between multiple domains. They decided to use ontologies into the bridge maintenance field, because, as it is commonly known, there are a lot of advantages using ontologies: firstly, the ontologies provide a vocabulary useful to model knowledge; secondly, they give the description of relationships; thirdly, they provide a hierarchy of concepts in a particular domain. After a literature review, the authors defined the resources for the ontology development, that are, for example, the different level of healthy evaluation of a bridge or the computation of technical conditions. The ontology contains four components: 1) the knowledge base stores ontology models and SWRL rules; 2) the ontology management system; 3) the rule engine; 4) the query interface. The development of ontology started with the competency questions to define the domain and the scope and after that, the authors considered reusing the existing ontologies. The third step consisted in defining a list of crucial terms also defining the classes and the hierarchy. The last steps involved the definition of the classes' properties and the facets, and the creation of instances and of SWRL rules. The application of this ontology called BrMontology consists in the automation of bridge evaluation, giving information related to the bridge maintenance, that could be used to help decision makers to check information about big events. The validation of the model was subdivided into three validations, which are the semantic validation, the syntactical validation, and the case study validation in which the authors used an existing bridge to validate the model. Some limitations exist into the proposed model, like the high number of possible damage types that could happen, so the variability of the damages; another limitation is the manual input of technical data. There are a lot of possible improvement to optimize this model and achieve a better automation and reliability of the model.

4.1.2. Predictive and Prognostic maintenance

With the advent of Industry 4.0, the recent trend of advanced maintenance policies is becoming the predictive maintenance (PM) and the prognostic health management (PHM). These two policies are accepted because they could decrease the risk, increase the reliability, and get near to a zero-failure environment. (Cho et al. 2020) describes with the European Union project Z-BRE4K a novel approach, aiming to eliminate unexpected breakdowns. More precisely, Z-BREA4K ontology deals with the design and implementation of the semantic model and the method for intelligent filtering. The first thing that the papers highlights is that in a system it is possible to find different types of storage with different types of data, or maybe the same type of data but with different configurations. In this case the IT department cannot cope with the reconstruction of all the database with a new standard because it needs time, and it is expensive. Data are very useful in order to define a future strategy, so the idea is to use the ontologies to federate the data and make them available for new knowledge development without the knowledge of all different databases. After this step, a series of questions are proposed in order to design the taxonomies to describe the maintenance engineering. With the competency questions, the authors defined the perimeter of the project and the different relationship between the entities. The answers to the competency questions are basically the requirement of the end-users. The final objective of Z-BRE4K is to identify the domain through a semantic model providing a data integration framework between various sources. Z-BRE4K receives as input the data arriving from different sources, the FMECA analysis, the condition monitoring sensors, the machine simulators and so on. The data are stored into an RDF repository, that is a component of a knowledge base system and subsequently these data are manipulated/used into a web service that provide to the final user a predictive maintenance program.

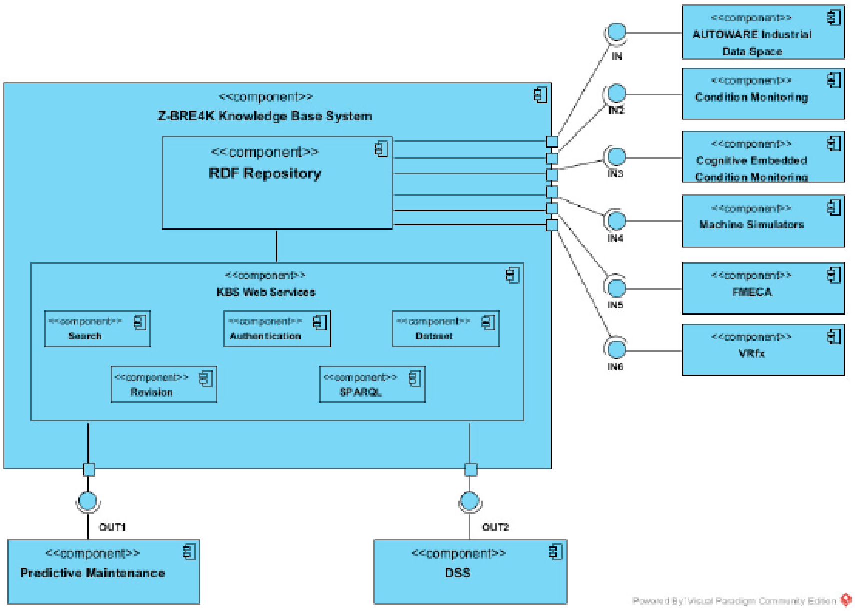

Figure 19: Z-BREAK structure

As shown in Figure 19 a SPARQL query interface is already provided to query the knowledge base system. Z-BRE4K receives in input a lot of information and it gives actions to implement a predictive maintenance strategy as main output.

This system is integrated in a bigger modern environment that is the base of the industry 4.0. In a 4.0 environment is common to find a cyber-physical system (CPS) composed by the interactions between the field devices and the algorithms. In this context, one of the most important needs is to have a standardization into the data management, from the collection to the knowledge creation. A tool used in the CPS environment is the PHM, that is a useful technique for predicting faults and problems inside a manufacturing industry. For the implementation of PHM in the CPS it is necessary to create a formal representation that integrates quantitative and qualitative data and provides a standard terminology and allows a more precise decision in real-time. (Nuñez and Borsato 2018) proposes a study for modelling the PHM using an ontology, able to formally represent its terms, concepts, and hierarchies to create relationship that allows people to have information to take decisions. In order to apply PHM, the authors decided to use a FMECA technique (which is considered the best technique for fault analysis) in order to analyse the mechanical systems. FMECA is useful in this case to examine the context and provide information on the system. The next step is to use the information provided by FMECA and carry out a prognostic management. The PHM means to determine the progression of the degradation of the component, giving indication on the remaining useful life (RUL). Using an ontology called OntoProg with the RUL analysis, the authors estimate the time to failure. The development of OntoProg starts with the determination of the scope, that is defining the remaining useful life. After that, the authors consider to reuse existing research and they re-used some existing concepts and taxonomies. The second part was dedicated to the ontology definition, from the number of classes to the relationships (all was defined by SWRL language). The queries in SPARQL are used to evaluate the feedback information contained in the OntoProg model. The queries are also implemented to evaluate the robustness of the model, comparing the reality with the model results. An interesting and similar approach of PHM is shown in the article (Medina-Oliva et al. 2012). The article proposes the use of a fleet vision on the system because a focused view on a single part of the system could not be enough to reduce risk and failure. A fleet should be viewed as a set of systems, sub-systems and equipment, the idea is to use the individual knowledge to improve PHM at a fleet level. The paper focused on the naval domain, and this is why it is used the term fleet, that can be extended to a production line, vehicles, airplanes. The implementation of fleet is very useful because it is possible to use characteristics or some information arriving from a single unit (sub-system, equipment) and extend this information to all elements inside the fleet. Obviously, the fleet could be composed by identical, similar, or heterogeneous units (the most interesting one). In naval field we are in the third fleet, the heterogeneous one. The fleet dimension can help the heterogeneous environment to develop enough information to create a diagnostic and prognostic model. The idea of the paper is to create sub-fleet grouping a set of units with similar technical characteristics. At a fleet level, engineers must treat different databases with heterogenous structure, so it is necessary the introduction of a semantic model that allows the definition of similarities among units and contexts. The ontology is divided into different contexts: the technical context, in which they explain the technical characteristics of the equipments; the dysfunctional context, in which they create a degradation model; the operational context, in which the mission and the environment of a unit are described; the service context, composed by the operation modes; the application context, formed by the needs of PHM optimization. For this model, a technical case study on different types of engines in naval sector was provided. The result is that contextual information can be useful in a specific context these contexts allow to consider fleet unit similarities and heterogeneities.

4.2. Risk management

The concept of risk inside an industrial environment refers to the likelihood of a specific negative effect within a specified period. It is commonly described by a complex function probability, and it is mitigated by numerous actions of prevention and risk assessment. The analysis of risk is teamwork procedure that involves all the actors inside the company. In the case of this written work, the risk taken into consideration is the risk of failure of an object inside a process. This chapter is divided into two main paragraphs: the first one analyses the FMEA methodology, while the second considers a particular article that uses ontologies in a medical field to prevent the heart failure.

4.2.1. FMEA analysis

Failure mode and effects analysis (FMEA) was formally introduced in the late 1940's in the military environment. The first push for failure prevention came in the 1960's in the aerospace sector and later, in 1970's, Ford Motor Company introduced FMEA in its production strategy improvement. FMEA is a procedure in operation management for analysis of potential failure modes within a system.

Today, FMEA is developed through three steps:

- Step 1: determine all failure modes based on the functional requirement and their effects;
- Step 2: highlight causes and occurrences to each failure mode;
- Step 3: testing and design verification.

At the end it is possible to develop a number called risk priority number (RPN), obtained by multiplying the rates of severity, occurrency and detectability (Molhanec and Povolotskaya 2012). FMEA analysis is divided into four main types of FMEA:

- System FMEA used to analyse a system;
- Design FMEA used to analyse the product preproduction phase;
- Process FMEA used to analyse the manufacturing processes;
- Service FMEA used to analyse services.

This chapter will explain the content of the articles that combine the FMEA tool and the ontologies. The first article (already cited above) focuses on the step 2 of FMEA, which is the cause and occurrences analysis. The authors try to eliminate the dependency of people to the definition of potential causes because non-discovered failure is more hazardous than other possible mistakes. In this section of FMEA they proposed an ontology-based approach creating a domain ontology. Ontological approach helps FMEA in two main ways: firstly, it creates a common understanding inside the domain without mistakes; secondly, the knowledge stored into the ontology can be processed and reused. This thesis will show how the ontology can effectively help the FMEA.

One example is using ontology to reduce the manual effort of normalization of different spreadsheets that have different structures, and they arrive from different sources. The engineering text data captured in FMEA is usually semi-structured or unstructured. The use of ontologies could assist the extraction and alignment of data. Tables are efficient for editing and storing data, but when they are used in schema-less formats cannot be interpret as-is using reasoner. The solution for (Hodkiewicz et al. 2020) is to propose an ontology as a mean of representing the data currently stored in many spreadsheet tables. The disadvantages of data stored in spreadsheet are that it is difficult to create a propagation flow from a bottom failure to a high level, and it is also not easy to create relationship between rows, and typically a spreadsheet has no explicit schema, so it is difficult to reuse and share knowledge. The principles that the authors adhere to are the application of a pattern that allows integration with top-level ontologies, using asset hierarchy from engineering standards, supporting modular ontology development principles keeping ontology separate from asset hierarchy. The development structure is proposed in Figure 20.

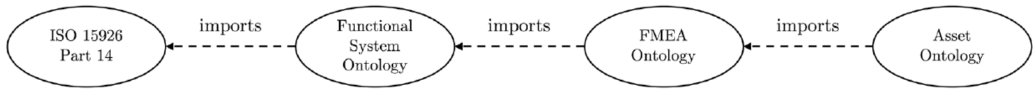

Figure 20: Ontology imports

The functional system ontology imports from the ISO 15926 the four standard classes, which are Process, Component, MechanicalSystem and AnchillarySystem. These allow to represent the functional structure and relationships of an asset. The FMEA ontology is composed of nine classes and the aim is to provide definitions of the vocabulary used in the FMEA process. The asset ontology extends the FMEA ontology, and it introduces classes necessary to represent the hierarchical structure of the assets. Obviously, the asset ontology depends on the asset that we want to describe, in this case heater and ventilation systems. The most interesting part of the model is that a failure in the bottom level creates a propagation effect on the other parts and with this ontology it is possible to create this failure propagation through an OWL-DL reasoner. With the simple excel table it is difficult to recreate the propagation mechanism.

The same idea is developed in the paper (Hodkiewicz et al. 2021), in which, from the unstructured FMEA spreadsheets, an alignment between data and an automated reasoning is proposed. In brief, the authors propose the use of ontologies as a way to represent the data stored in FMEA tables and perform quality checks. When a tool like FMEA is applied, it necessarily starts with an assets decomposition in functional sub-system. When the decomposition level of detail is satisfied, it is usually followed by the failure mode analysis from the lower level to the higher. The impact of the lower levels on the higher level is very important. The propagation of a failure must be considered in an overall systemic analysis like FMEA. The most interesting part is the central one of the paper, in which the description of the ontology design can be found. The ontology has three layers: the first one is called functional system ontology (FSO); the second one is known as the FMEA ontology (FMO); the third one is called asset system ontology (ASO). The first layer represents the functional structure and relationship of an asset, in fact the relationships between the four classes is only an import. The second layer is an ontology that has the challenge to provide definitions for vocabulary used in FMEA. The third layer is used to extend the second layer in a particular domain, in the case of the paper in the heater systems. A use case can be found in the paper to test the ontology and achieve some results. The results after the use case are that the failure effects in the models are correctly propagated, and the FMEA spreadsheet tables become explicit. It is understandable that the knowledge produced during FMEA is really valuable, because it can reduce cost and effort if it is adequately managed and reutilized. An ontology to model the FMEA process can bring to reuse the knowledge.

Regarding the reuse of knowledge, (Rehman and Kifor 2016) in the analysis of literature discover that no specific ontology exists to address the FMEA knowledge sharing and reuse, but there is already research on the storage system of FMEA result. For this reason, the authors proposed an ontology to reuse and share knowledge. To reuse knowledge, they created PFMEA ontology, which means process failure mode and effect analysis that allow them to reach the information stored in a certain repository. PFMEA is composed of six main classes (Mitigation_action, Failure_mode, RPN, Control_method, Process, FMEA) and two sub-classes of Failure_mode (Effect, Cause). There are a lot of relations between the classes that will be not listed here. The authors used SPARQL with a server in order to access information from the ontology. The server provided a graphical interface, and the result of the queries was exported in CVS format. The queries proposed are:

- A query to display FMEA worksheet header information;
- A query to display complete details of FMEA process;
- A query to display causes and recommendations for each failure mode;

- A query to display causes, effects, and recommendation for a specific failure mode;

To recap, this ontology uses an already existing result that is a series of data inside a certain repository, and with SPARQL queries ask to the ontology some details on existing data. With this work we can reuse existing knowledge. An example of reuse of knowledge is also provided by (Zhou and Ren 2011) with the sharing and reusing of reliability design knowledge of board-level electronic products (BLEP). BLEP is an electronic product formed by printed circuit board, connector, and solder joint. It has different level of encapsulation where different components are installed. BLEP failure includes internal causes and external causes: internal means that a part inside the BLEP doesn't work correctly, while external means that BLEP doesn't work correctly due to external stresses like temperature. The target of the reliability design is to identify all the possible local faults and eliminate or prevent them. The most interesting part analysed in the article is the failure ontology, that is an ontology divided into four subsets, BLEP design ontology, failure features ontology, failure propagation ontologies, and stress ontologies.

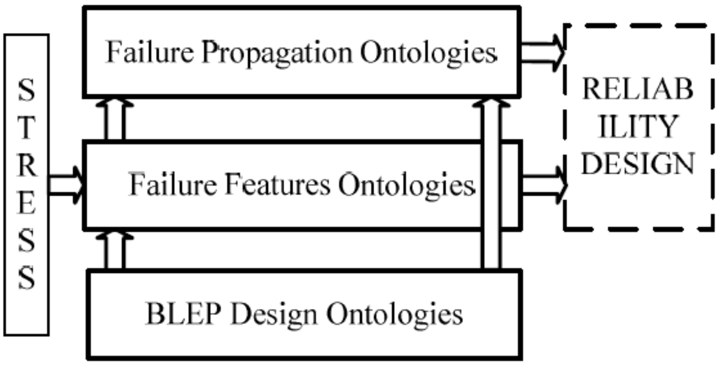

Figure 21: BLEEP ontology structure

BLEP ontology defines the composition of BLEP and its operating characteristics; stress ontology is rooted in failure component units; the failure ontology is based on the previous two ontologies, and it is composed by six main characteristics, which are failure countermeasure, failure mode, failure condition, failure site, failure cause and failure effect. For system-level products they introduce the failure propagation ontology that includes two types of propagation, logical and physical. Physical propagation means that failure spreads from one part to another, whereas logical means that the functional of a circuit board cannot be achieved. The three sub ontologies are integrated in a single entity. A case study is proposed to verify the feasibility of the reliability knowledge representation, comparing the methodology propped with failure modes mechanisms and effect analysis. Basically, the authors perform the FMEA analysis on BLEP and they apply to it the ontology proposed.

The paper (Mikos et al. 2011) analyses the existing context and it proposes an architecture in which agents are used to share and retrieve knowledge resulting from the solution of previous non-conformance problems, together with the Potential Failure Mode Effects Analysis (PFMEA). The developed distributed system uses an ontology based on description logics for the knowledge representation in the PFMEA domain. This system seeks to provide means to share and reuse current knowledge in PFMEA tables in support to the management of the organizational knowledge. The authors followed the methontology structure, deeply described in (Fernandez, Gomez-Pearez, and Juristo 1997), to create a model for the PFMEA ontology. They started from the conceptualization and formalization of the PFMEA-DL ontology (DL means description logics) through seven concepts: product, process, function, failure, action, PFMEA images, PFMEA description, as shown in Figure 22.

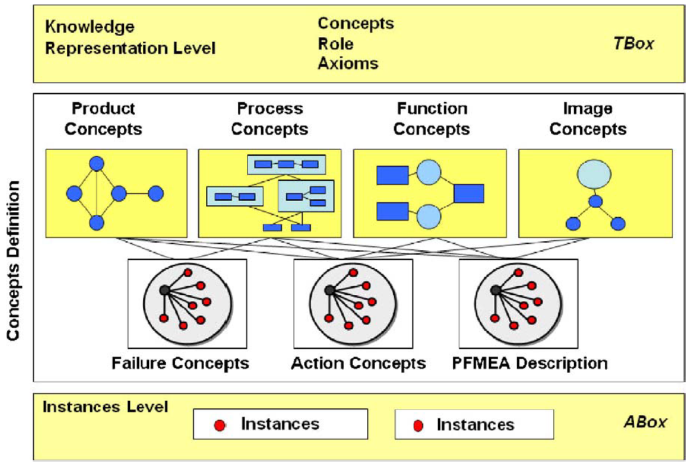

Figure 22: Architecture proposed by Mikos et al. 2011

The second part related to the implementation was done through the OWL-DL language, which combine expressiveness and possibilities to inference. As shown in Figure 22 there are two boxes: the TBox and the ABox. The TBox is a box in which there is the taxonomy and the definition of concepts roles and axioms, while the ABox is a series of combo box in which it is possible to insert queries. An inference service was provided by the RacerPro Server System, it provides reasoning engine services, and it is responsible for the inference service and knowledge retrieval. It is also implemented as a graphical interface, which allows the visualization of the TBox and ABox, the manipulation of knowledge bases and the submission of queries. The innovative approach is not only the model, but the distributed approach proposed. Indeed, the inference engine with the two boxes communicates with an agent that is connected in an internet/intranet network. The agents are the elements of the architectures that allow the sharing and reuse of knowledge. At the end a prototype was developed with Java Agent Development Framework.

4.2.2. Other points of view

FMEA is not the only way to prevent failure, evaluate the risk and improve the reliability, therefore it is very useful to consider the other tool and methodologies that exist. It is also important to take into consideration domains which are different from the mechanical, industrial, and productive one. An interesting type of application of risk management is explained in (Gamberger 2013), where an ontology was proposed to regulate rules of failure into a medical domain, more precisely into a heart failure domain. The article presents a series of rules that defines the guidelines for heart failure management. In a following part there is the definition of actionable classes that correspond to a rule defined before. The elements of the conditions for actionable classes include concepts defined by other actionable classes and properties from the heart failure ontology that may be attributed to patients. In this way the descriptive knowledge available in heart failure ontology may be used directly in the decision-making process. The last part is composed by a reasoner that helps the decision makers to take decisions. The extracted data are stored into OWL formalism, and next the reasoning process places the instance of the patient into appropriate actionable classes. The last step is the interpretation of the results so that each actionable class is tested for including the target patient instance. If true, a message is sent by a user interface.

The last paper of this section changes a little bit the final goal, going from the analysis of failures and reusing the knowledge to the reliability of a system. This goal is pursued by (Yunni Xia et al. 2013). The authors create an ontology for services divided into three perspectives: service profile the high-level description of the service, the service grounding that maps the abstract representation and the service model that serves as the control of interaction flows. The field of application of this paper is the reliability of OWL-S systems with a non-Markovian stochastic petri net (NMSPN). NMSPN is a graphical and mathematical tool used for modelling and analysing the form of a system (parallel, synchronous, etc..). The most interesting part is not the translation from OWL-S model to NMSPN, but the reliability evaluation. In order to study the reliability of ontology-based service compositions, the authors used the concept of reliability block diagram (RBD), in which all configurations of a system can be represented, and the reliability of a single system can be defined. The reliability was defined with the standard formulas of RBD based on the composition of the single tasks. Through the article the analysis is validated by a tool that executes the OWL-S and provides logs of execution time. To identify the bottleneck in the system, a sensitivity analysis was applied varying the parameters of +30% and -30%. This last paper is very interesting, because it applies ontology to the petri network. It is considered to bring a different view on the ontology application for reliability and maintenance.

4.3.Data analysis

Data analysis is a process of inspecting, cleaning, transforming, and modelling data with the goal of discovering useful information. This section contains all the papers that are part of the data analysis process improving some aspects through the ontologies.

4.3.1. Using data

The data analysis can be interfaced with all the steps, from the data requirement to the communication and it could be applied to different sources of data. The databases on which this method can be applied may be different because there are public databases, like the American reliability analysis center (RAC), and private databases that each company/institute can create internally. The public databases have the aim to create a standard set of distributions to be used in the reliability engineering industry. (Chen, Ye, and Li 2014) proposed the use of three different sources of public databases to analyse the data: the failure mode distribution of U.S. Reliability analysis center; the Failure mode and classification of automobile of China automobile industry corporation; and the Reliability prediction handbook for electronic equipment of General armaments department of the people's liberation army. As far as private databases are concerned, each company can create failure report analysis and corrective action system to increase the reliability and maintenance prediction. To reuse failure mode effectively, the paper proposed two key points: the first is an ontology-based annotation which is used to add ontological knowledge to existing databases; the second is to search and retrieve data based on semantic similarity. The objective of this work is to propose an ontology to facilitate the sharing of information; support integration of tools; provide same perspectives with collaborating teams and tools; create a common vocabulary; describe unambiguous definitions. Regarding the similarity, the strategy proposed is to use a semantic similarity index between zero and one. Using an XML editor, it is possible to understand and compute the level of similarity between different sources and the output is the possibility to reuse the failure mode in existing database.

Ontologies can cope also with the problems depicted in (Li et al. 2017): first of all, the application of environmental data lack in systematic process and normalization due to partially used data; secondly, the environmental effectiveness not sufficiently considered in the process of product reliability design and analysis. The author proposed a knowledge application system architecture as shown in Figure 23.

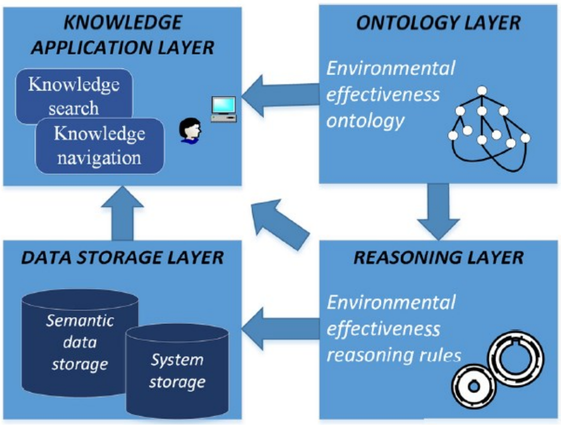

Figure 23: Definition of layer interactions

The most interesting part for this thesis is the ontology layer. Ontology layer acquires and classifies concepts and, together with the reasoning layer, it can extract highly valuable knowledge from a large amount of available information. With ontology interpretation and reasoning, it is possible to discover hidden knowledge. The environmental effectiveness ontology has two main concepts, that is the product and the environment. These two concepts have different sub-concepts like electrical product, mechanical product, soft product, environmental mode, environmental factor. The reasoning layer was implemented with a series of rules written in SWRL. These rules serve as the conjunction between the concept and the attribute with variables. The data storage layer involves semantic storage and system storage. The system storage is the basis of operations and the semantic one is the basis to environmental effectiveness, product, and failure knowledge. All the storage systems are implemented with MySQL, and they are queried with SPARQL to retrieve information. Obviously, the application layer can call the data stored into the storage system and combine the other three layer into a unique integrated system. In order to have a better usability, the total system is composed by a graphical interface that allows the user to query the databases without problem. The author proposed an application of the methodology on a rubber seal to study the interaction between product and environment. The final result is that it is possible to select the better condition in which the rubber seal can operate and as a consequence it is possible to understand when the stress can create a failure. This ontology puts together the product knowledge and the environment knowledge, increasing the accuracy of reliability analysis through the analysis of the data.

4.3.2. Tools in data analysis

Computerized maintenance management system (CMMS) software is used to help managers to make better decisions about incidents of downtime. However, CMMS data are often underutilized for downtime analysis and the quality of output depends on the quality of input. The paper (Yew, Foong, and Sivarajan 2019) proposed an improvement of CMMS brought by knowledge-based component. The component improves utilization of data to generate requirement downtime parameters with better timeliness. The first step is to capture the right knowledge and as common in knowledge field, the authors used competency questions, distributed through a form to employees of a semiconductor manufacturing company. After the competency questions, the information received were used to build an entity structure and relationship diagram. Part of this hierarchical structure was demanded to the ontology. The papers didn't describe the ontology, but it is interesting the application in the structural representation of knowledge in order to optimize the data. The hierarchical structure defined through Jena was imported in Java where the developed structure is applied to a series of data arriving from a SQL server.

Some machine learning indexes are applied to the methodology to validate the results like the precision index, the recall index and the F1 score index. The advantages of this approach of knowledge base in downtime management system is the acquisition and the recommendation of important information for effective decision-making.

A different tool is proposed in (Chougule and Chakrabarty 2009) that is a Variation Reduction Advisor (VRA). The VRA is a system that contains information related to the problems encountered during the process, their root causes, and possible solutions. It is proposed because a knowledge management system can support maintenance in diagnosis in preventing quality issues. VRA was used in General Motors vehicle assembly plant with the aim to improve the activities. In General Motors the functioning was to communicate between teams about the problem of the assembly lines. The ontologies into VRA have the role to describe the existing problems and their solutions through a thesaurus. Thesaurus is a set containing sets of phrases, where phrases contain one or more words. All the phrases belonging to a set are considered synonyms. The most interesting part is the diagnostic process. The diagnostic process is very simple: a user inserts a search phrase, and the algorithm takes as input the typed phrase, the thesaurus and gives the diagnosis as answer. In the first phase, the algorithm searches the synonyms for each phrase, and in the second one all synonyms are retrieved into the phrases and at the end all record retrieved are shown. The ontology is very useful for the search in order to guide the algorithm to the correct synonyms. At the end, a series of indexes are provided to control the precision of the thesaurus, in fact with the computation of these indexes the authors come to the conclusion that the use of ontology ensure that relevant records are not missed.

4.4.Process Optimization

This chapter is very heterogeneous because there are a lot of different solutions proposed to improve industrial process. The first chapter will discuss about maintenance optimization, the second one about the representation of ontologies and simulation and the last one is a content for different types of process optimization.

4.4.1. Maintenance improvement

The first chapter of Process Optimization regards the maintenance optimization. The papers discussed in this chapter are not into the maintenance part, because in the maintenance part there are only the papers that deal with the maintenance policy and the maintenance decision making. The aim of this chapter is improving the process of maintenance.

The first article that can be taken into consideration is (Rasovska, Chebel-Morello, and Zerhouni 2005), which proposes a similar model to KG cited in maintenance chapter. The model is called Proteus and it is the architecture structure provided by the authors, at the center of an environment constituted by field data, and strategy level. Proteus platform like KG is based on and ontology to eliminate heterogeneity between different data sources. The aim of the article was to develop an open system enabling connection and collaboration of different maintenance systems and application through a software architecture for web-based e-maintenance. The difference between KG and Proteus is that in Proteus there is the use of the network instead of an internal application, but the aim is similar, that is provide knowledge on maintenance. Proteus is based on the concept of s-maintenance that is e-maintenance (remote maintenance systems connected to web and using the network for data transfer) based on the notion of semantics. Using s-maintenance is possible to create a corporate memory, and this memory supports the knowledge management. Proteus wants to create a corporate memory of enterprise that store expert knowledge. But what does corporate memory mean? With corporate memory, the authors identify the structured set of knowledge related to the firm experience in a given domain. They followed a cycle model to capitalize knowledge that was based on four main parts, the detection, the actualization, the capitalization, and the preservation. To represent and manage the knowledge, an ontology was proposed. The ontological model is divided into three parts, which are the equipment decomposition, the equipment analysis carried out with FMECA and the description of repair and help system. An application of the decision support system was done on a pallet transfer system that represents a flexible production system. Proteus as a decision support system provides information about cases already stored. This information is the description of the failure (symptoms, localization, state) and of the solution (equipment, actors, tools). In this case, ontology is a facilitator tool to knowledge sharing for various actor of the e-maintenance platform.

In the case of (Jin, Xiang, and Lv 2009), ontologies are used to improve interoperability between heterogeneous enterprise applications and to describe the semantics of condition monitoring and maintenance domain. Condition monitoring is the main element in the condition base maintenance (CBM) policy. A CBM approach uses tools like vibration monitoring, thermograph etcetera to obtain the actual operating condition of critical plant systems. In the discussed paper, there are two graphical explanations of CBM and of ontologies that are interesting for this thesis and showed in Figure 24.

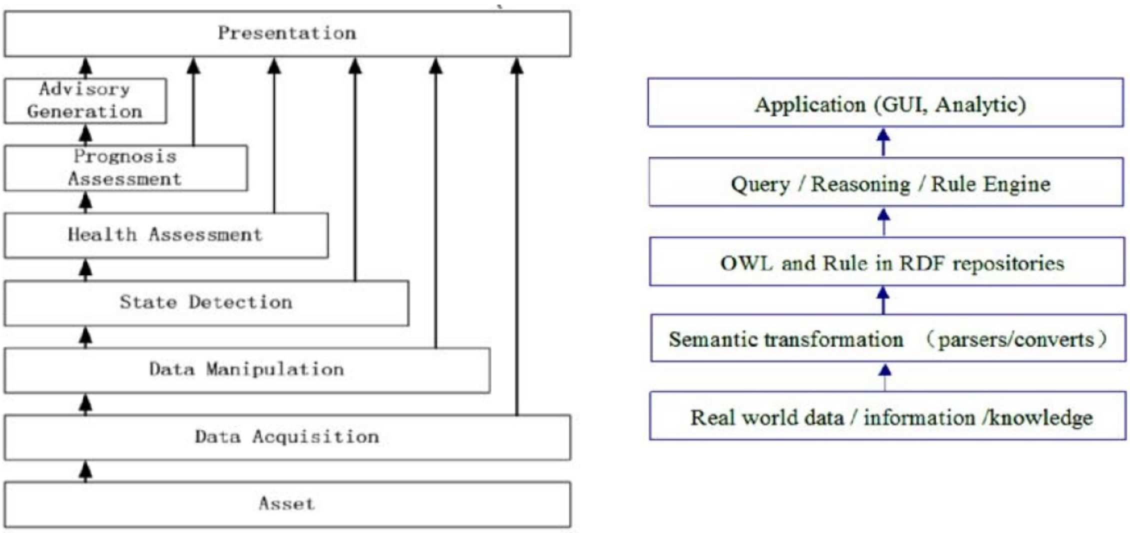

Figure 24: CBM explanation

On the left, there is the representation of the CBM structure proposed by the ISO-13374, whereas on the right there is the representation of the standard structure of an ontology proposed by the current paper. The authors used the semantic transformation layer to solve the heterogeneity problem. Regarding the ontology model, they started from the competency questions to acquire knowledge and then they acquired knowledge from the different assets. When knowledge had been acquired, they created a model to monitor the condition of a system. The CBM can be supported by a case-based reasoning (CBR) that is used to automate part of the CBM decision process. A case knowledge base (CKB) is built over a semantic repository with an interference engine supporting ontology-based information integration and data access using SPARQL queries. This is the main concept of (Aarnio, Seilonen, and Friman 2014) that provide a solution with CKB at the heterogeneity storage problem of classic databases. The architecture of the system is based on two main layers: the HMI level, that is the connection point between the people and the CBM system; and the second layer, that is composed by the reasoning environment with data fusion, knowledge mining, and CBR analysis. The reasoning level acquires from the process and maintenance some data and it acquires data also from field devices (in this case valves are used). The CKB is built over a semantic repository storing case knowledge named RDF graphs, this type of storage system has some functionalities including providing SPARQL query access, enabling reasoning, mapping external data in RDF format.

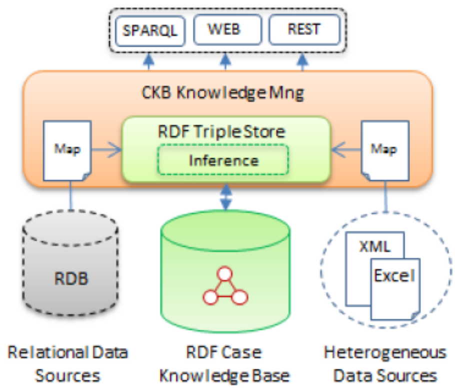

Figure 25: RDF storage system

The Figure 25 shows the architecture of the RDF storage system. Following a methodology proposed by Guarino that divides ontologies into four main types: the top-level ontologies, the domain ontologies, the task ontologies, and the application ontologies. The authors decided to create an ontology model that contains all these four levels. An ontology called OntoCape represents the top-level ontology in which the system was described, then a CBRCase task ontology is provided to represent the case-based part of the ontology and, finally, an OntoCape Plant ontology and a ControlValve ontology are built to represent the domain ontology and the application field. The main reasoning tasks are executed into two different levels, the low graph storage level when a query was asked, and on high CBRApp level to support taxonomy-based similarity calculations with in-memory ontology models. An evaluation phase was carried out to evaluate the goodness of the project. The result is that the performances of the system are considered adequate.

In the (Arena et al. 2017) is proposed an approach called knowledge-enabled supervisory monitoring, which is useful for the analysis of maintenance and nominal operations that are performed in a process plant. The objective is to share process status with information analysis and knowledge-based decision support systems. The existing infrastructure consists in HMIs, that are the input/output instrument for the operators, the SCADA, and the computerized maintenance management system (CMMS), that are the coordinators of the system, and they contain both static and dynamic data of the plant. The article proposes an extension of this structure with a semantic framework composed by the extraction transformation and loading (ETL) engine, a triple store, and an ontology. The two worlds are connected with a middleware that exchange data between the parties. The functions of the semantic framework are to provide a platform to store, maintain and manage the information arriving from the existing structure. The triplets-based information flow provides new knowledge, carrying out semantic analysis aimed to aware the workers on the process status and to estimate the down time. The estimation is done through some indexes that esteem the plant condition level basing the analysis on the P&ID of the plant and on the worker's knowledge. Few examples of indexes are the matchable topology condition (MTC) and the suitable readiness state (SRS). These indexes are used both to support the decision procedure through a proper notification and to determine when a maintenance operation occurs and as a consequence to estimate the remaining time till the process reaches an optimal state. To validate the model, the authors applied the solution proposed to a synthesis gas pilot plant. The results shown give an indication on the indicative maintenance operation time and on the stand-by time subdivided by topology. The aim is not the substitution of the existing IT technologies, but the enrichment of the existing environment.

Another interesting topic of application into the optimization of the maintenance field is the analysis of the technical documentation, that it is often poorly written and constructed. Technical documents are very important to exploit the best potentialities of an assets and for this reason the (Koukias, Nadoveza, and Kiritsis 2015) proposes the use of a generic asset management ontology as a form of documentation that helps companies to reach full potential of an asset. The aim of technical documentation is to describe the technical product and make available the technical know-how and product history for the subsequent users of the information such as the engineer or operators. More precisely, the cited paper suggests the use of an upper ontology model concerning the documentation of operation and maintenance phases of the asset that manufactures can use to extend and create their own asset ontology for their products. Once the technical documentation ontology and the asset ontology are created, the following step is to merge these two ontologies into one single entity. The aim is to make sure that the asset behaves in a correct way in relationship to the technical documentation ontology. The asset management ontology is formed by some generic classes subdivided into static data and dynamic data. An example of static data is the asset, which doesn't change its technical characteristics during the useful life, while an example of dynamic data is the operation data. The authors prefer to present a generic work in order to be adapted to different situations with a lower specific ontology layer. In the next chapter they present the technical documentation asset ontology that is a guide line in the asset management. It helps decision-makers in the asset configuration, operation, and maintenance to guarantee assets' performance and availability. The merging is done taking the union of the terms and axioms creating a completely new ontology. The key to this process is the ontology translation that translate datasets and queries. The advantage of this procedure is to have in a single ontology the guide line (technical references) and an asset management tool.

4.4.2. Ontological representation and simulation

A software architecture plays an important role in the software development lifecycle, especially in the decomposing phase of a system when the big structures are decomposed in a coherent set of interacting components. Today, a lot of standards are proposed, but practitioners trying to describe a software architecture have trade-offs between efforts to put and simplicity. The paper (Yuan 2017) proposed the use of W3C OWL as the basis for architecture representation and interchange. Actually, OWL was not an architecture description language (ADL), but in this case it is adapted to be so. Some benefits deriving from the using of OWL is that it can give to the software architecture community the interoperability without centrality that it means open standard and common notation, formal semantics, decoupled architecture task that means separate knowledge representation to manipulation, rich tooling support, architecture as data. The semantic architecture proposed in the article is composed of sensors and extractors at the bottom level, the triple store in the middle and editors, reasoners, and query engines at the top level. The triple store is used as a database for storing software architecture axioms and facts. The authors proposed four use cases for architectural purpose. The first one is the architecture discovery that involves the discovering of relationships between components; the second use case is about architecture style validation; the third one regards architecture consistency checking, which means that if a configuration is consistent with the system design; and the four one is about architecture visualization. The second example of ontology used to represent a process is (Uddin et al. 2011). The mentioned paper presents an ontology-based knowledge representation for flexible manufacturing systems (FMS), providing a comprehensive semantic foundation of the facility. FMS consists of automated material handling devices connected by rail guided vehicles or industrial robots. Maintaining a high level of automation and facing the changes is a very challenging activity. Generally, a reactive scheduling and simulation methods are utilized to address this problem. Throughout the paper, an FMS use case producing assembly parts for automotive industries is considered. An ontology-based knowledge representation (KR) is addressed to provide a clear semantic of the facility. The architectural point of view of the system is based on an application that dialogues with Microsoft services. The FMS domain ontology is modelled upon the main concept of production order taxonomy associated with the required part manufacturing master data. The ontology model contains all the concepts and attribute of the FMS domain. In the run-time process the web services are invoked and the ontology is populated with relevant run-time instances. The ontology update is addressed by loading the axioms and instances using a source document that contains the web services log of production order. An integration of OWL is done by the SWRL to increase the expressivity. As far as the queries are concerned, SPARQL is used to retrieve information from the modelled FMS domain knowledge. The steps of the process are shown in Figure 26.

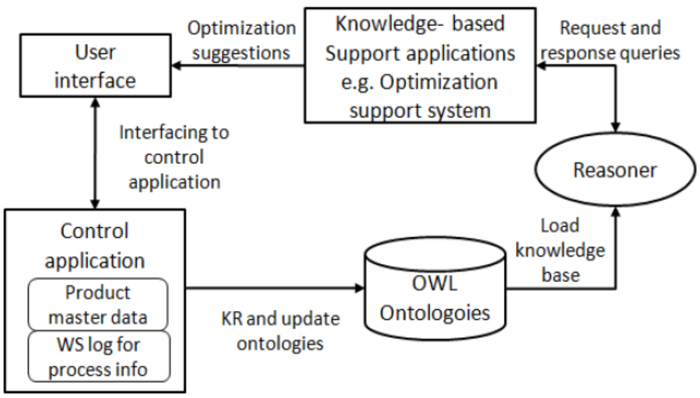

Figure 26: Steps of FMS model

The benefits of using an FMS application with the support of ontology is to avoid overload of centralized software application and provide a common and well-defined knowledge based.

The simulation (Terkaj, Tolio, and Urgo 2015) presented an ontology-based virtual factory approach to evaluate the impact of planning and maintenance decisions during the operation phase of a manufacturing system. The main idea is to create a digital model of the factory continuously synchronized with the real one. Obviously, if the idea is to recreate the factory in a digital model, there is a big problem of system heterogeneity. To avoid this problem, ontologies are perfect. The authors proposed a generic already existing ontology composed of a factory object, the representation, the properties, and the placements. These are the four main classes, but they added other new four classes called object state, state frequency, factory object history and ratio measures. In this case the ontology model is an instrument to support the simulation because it has the aim to group and harmonize the historical distributed information. The implementation of ontology helps the simulation model to have a digital continuity, with all the information available from the past cycles. The goal of in situ simulation is assessing the impact of management decisions on the short-term performance of the production lines. In the final part of the paper a case is proposed to validate the model. A different scope for ontologies in simulation could be used to overcome the semantic barriers of different systems in a simulation and modelling environment. Effective interoperability between distributed and heterogeneous simulation teams and tools is an important issue to develop collaborative simulation in fact (Hu and Zhang 2009) proposed an ontology to solve this problem. The fundamental objective of ontology is to capture the knowledge of certain domain and offer a common understanding of these knowledge, concepts, in order to determine the terminology and give out a formal definition of terms and relations. In collaborative simulation the content of ontology should include model interaction and model mapping. The proposed architecture is developed through two logic layers for collaborative simulation. The first level is composed by a simple ontology called SO, that has all the concepts and interactions inside, it could be considered as a cross-disciplinary ontology. The second layer is composed by a collaborative simulation ontology called DO. About the ontology structure, there are three layers: the basic ontology layer that defines a common semantic model for all ontologies; the domain ontology composed by multiple domain ontologies which are used to express domain detailed knowledge; and the applied ontology that is derived from the domain ontology, and it contains the properties and the data attributes. The three layers structure is shown in Figure 27.

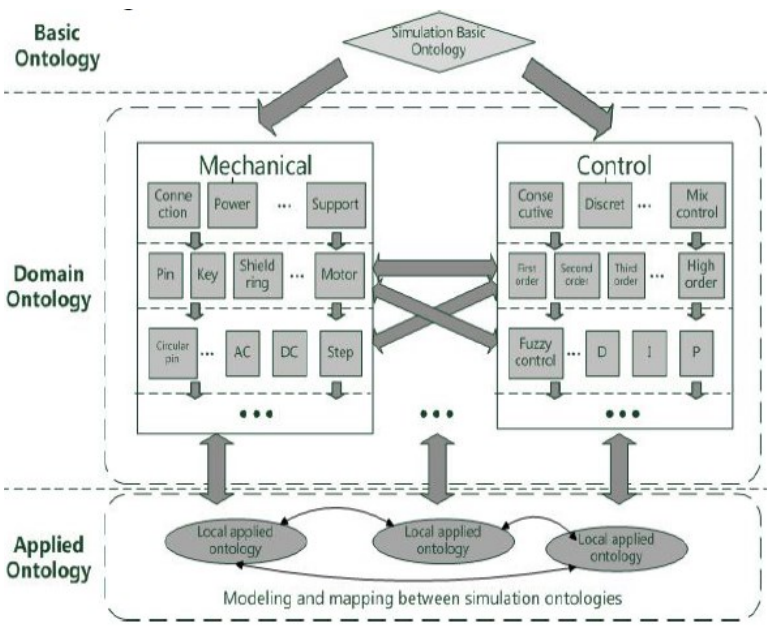

Figure 27: Structure proposed by Hu and Zhang 2009

After the ontology construction, a model transformation, construction, and mapping are needed to change applied ontology into object model. All transformation activities are done through HLA/RTI collaborative framework, and through web services which are open source and accessible from the internet. HLA and web services are powerful tools for modelling and simulation while ontology grants semantic interoperability. The combination of these two models could be interesting for exploiting all the synergies.

The last simulation approach is proposed in (Novak and Sindelar 2013) that is focused on the description of complex and flexible automation systems. The description is needed for optimization of the process structure and parameters, and it is useful to design a simulation model. The simulation model has to dialogue with the real system composed by different instruments like SCADA or ERP, so semantic integration becomes a very interesting solution to create links between instruments. In terms of automation systems, semantic integration is focused on representing meaning of data and roles of each tool to reach efficient interoperability and information transformation. The ontology proposed is an ontology-based industrial plant description, with the goal to store knowledge in a structured way that enables the extraction of information. The first step is to analyse the plant workflow and understand the plant characteristics with domain experts. The next step is the creation of the ontology plant that is useful for simulation specialist in order to implement a simulation library comprising the simulation blocks. The third step is to create a block-based design of simulation. It consists in the creation of simulations block with some characteristics that are extracted from a simulation library (if exists), but the selection of the correct block is not so easy because it needs a lot of attention on properties, characteristics, and I/O of the blocks. A block must approximate the real device and the connection with another device must be correct. The following step is the parameter management, in which the parametrization of the block is fundamental for the correct functioning of the simulation model. To demonstrate the goodness of the model a use case was proposed.

4.4.3. Other optimization

The last paragraph includes all the papers that proposed particular approaches to optimize the process. In this section there are three papers: the first is about the heterogeneity of the projects, the second regards the product design and the last one deals with the optimization of the development process of complex product.

The first, (Moore et al. 2000) proposed an ontology to avoid semantic problems in part of the project named Task-Based Process Management (TBPM). This project aims to create an adapting pattern for flexible workflows. The idea is that there is not a single process model which can be applied to all projects, in fact the dynamic nature of engineering project is the main challenge. It is also true that in some projects there are similarities. The TBPM use a sort of database called plan library in which the structures of the processes are stored, and each plan specifies a series of tasks. Per each given task there are multiple possible plans expressing different ways. One problem introduced by this flexible modelling approach is that there is not consistency between the different plans. An example can be the use of different words in the different plans (artifact, pump, element). The approach taken in the article is to develop an ontology to structure the knowledge about the process context. The development of the ontology was based on two steps, the informal ontology and then in a formal way. They create a hierarchical definition of terms that is able to answer the question: is X an example of Y? recursively testing the base terms they achieve an unambiguous determination. The article was written in the 2000, when the ontologies were only at the initial stage, but the hierarchical decomposition of terms it is a first example of an early ontology.

The second paper examines the product design field. The field and competition intensify the competition in fact manufacturing knowledge is one of the most important sources to support this process. As already stressed, ontologies are increasingly important because they provide the critical semantic foundation for the rapidly expanding field of knowledge. Therefore, the application of the ontological approach in the manufacturing knowledge for support product design can bring some advantages, knowledge sharing, knowledge reuse, and a common understanding of the domain. (Wang and Tong 2008) aims to investigate the manufacturing knowledge needs of different product design decisions, and then to develop an ontology for representation of design decisions and manufacturing knowledge needs. The methodology used to develop the ontology is divided into seven steps: 1) the determination of the domain and the scope, 2) the consideration of the existing ontologies; 3) the enumeration of the important terms; 4) the definition of the classes and the hierarchy; 5) the definition of the properties; 6) the definition of the values of the properties; 7) the creation of the class instances. The paper doesn't propose a new model, but a combination of two existing models and the authors focused only on the definition of the classes and the hierarchy and the definition of the values properties. For the definition of classes and the hierarchy, they started from a requirement list and then they translated the requirement into the development of concepts. The definition of properties had three main parts: the definition of intrinsic properties, extrinsic properties, and the physical decomposition. As far as the definition of values for the properties is concerned, they used a slot editor to edit the value of a properties. This approach proposed an ontological approach which identifies what kind of manufacturing knowledge the different design decision needs.

The last paper discussed ARUM, a project described in (Harcuba and Vrba 2015) its aim was to increase effectiveness of production of complex products. The main goal of the ARUM is to develop a new software solution for organization of work, including tools for long-term planning, operative scheduling, and decision-making support for the production management. All these activities require a high level of internally and externally integration. To manage the integration problem, an ontology solution is proposed in addition to an enterprise service bus. The ARUM ontology was developed through three ontology modules, the ARUM core ontology defines an abstract model for manufacturing processes that introduces generic terms and classes for description of discrete production processes. ARUM scene ontology extends knowledge model of the core ontology with concepts necessary for long to medium term activities, it provides data structures for capturing the real situation in the production. The last one is the ARUM event ontology that is used to model the expected and unexpected events. The combination of these three different ontologies creates a very complex system that is presented in a use-case of rescheduling of running production.

4.5.Summary

During the content analysis and the methodological selection of the papers, I focused my attention on the papers' topic, aim and possible useful usages. Both the considerations I made, and the content analysis led me to the results shown in Table 30. The table is divided into four columns, which are: *Application Field*, *paper*, *Topic, Aim.* A brief explanation of the topic and of the aim are provided in order to better explain the results of the content analysis.

App. field	Paper	Topic	Aim
Maintenance	(Viinikkala, Syrjala, and Kuikka 2006)	Innovative approach to support operations and maintenance decision making.	To improve maintenance and operations suggesting maintenance schedule.
	(Banujan and Vasanthapriyan 2020)	Using the knowledge of bridge to better understand infrastructural problems and take care of them.	To use maintenance ontology model to share maintenance knowledge into bridge industry.
	(Ren, Ding, and Li 2019)	Analysis of reliability and technical maintenance requirements about bridges.	To improve the decision-making process in bridge maintenance.
	(Aarnio, Vyatkin, and Hastbacka 2016)	A context modelling approach that supports the informative system.	To reduce the heterogeneity through the system to share maintenance advice.
	(Cho et al. 2020)	Building an ontology-support system for maintenance prediction.	To create a tool that is able to suggest the correct maintenance policy.
	(Nuñez and Borsato 2018)	Prognostic health management.	To support the decision-making process in intelligent manufacturing systems.
	(Medina-Oliva et al. 2012)	Prognostic health management.	To reduce the risk of failures with a fleet wide ontology.
	(Hossayni et al. 2020)	Development of a vendor agnostic solution to better describe the domain.	To create a shared semantic repository to improve maintenance failure analysis.
Risk Management	(Gamberger 2013)	Heart failure	To improve and prevent the heart failure
	(Yunni Xia et al. 2013)	Development of a quality of service through a petri network.	To improve reliability
	(Mikos et al. 2011)	PFMEA.	To create a distributed model to improve sharing and reuse of knowledge in PFMEA.
	(Hodkiewicz et al. 2021)	FMEA.	To create a homogeneous expression in FMEA results.
	(Rehman and Kifor 2016)	FMEA.	To provide more information to the decisiors.
	(Zhou and Ren 2011)	Reliability of electric products.	To create a failure ontology for electronic products.

	(Molhanec and Povolotskaya 2012)	FMEA.	To improve quality management using ontology in FMEA.
Data analysis	(Chougule and Chakrabarty 2009)	Equipment diagnosis in automotive plant.	To support a searched semantic guide into phrases.
	(Hodkiewicz et al. 2020)	Reasoning over engineering textual data.	To improve the reasoning in spreadsheet tables.
	(Chen, Ye, and Li 2014)	Knowledge stored in databases.	To create a common knowledge in failure mode databases.
	(Yew, Foong, and Sivarajan 2019)	Knowledge based procedure in down time analysis.	To improve CMMS systems with ontology.
	(Li et al. 2017)	Environment data analysis	To create a connection between product and environment to increase the reliability.
Process optimization	(Terkaj, Tolio, and Urgo 2015)	Simulation.	To improve the short-term efficiency of companies.
	(Wang and Tong 2008)	Manufacturing knowledge.	To support the design decision of products.
	(Uddin et al. 2011)	Knowledge management in flexible manufacturing systems.	To support flexible manufacturing systems.
	(Moore et al. 2000)	Activities of a process.	To create unambiguous word classification.
	(Harcuba and Vrba 2015)	Organization of work and production management.	To support flexible production systems.
	(Hu and Zhang 2009)	Simulation.	To create interoperability in a simulation context.
	(Novak and Sindelar 2013)	Simulation.	To introduce an ontology-based approach for plant description.
	(Rasovska, Chebel-Morello, and Zerhouni 2005)	Optimization of maintenance.	To improve maintenance in flexible automated systems.
	(Jin, Xiang, and Lv 2009)	Optimization of maintenance.	To improve the monitoring of an asset.
	(Aarnio, Seilonen, and Friman 2014)	Condition based maintenance.	To create a model to store data in a flexible way.
	(Arena et al. 2017)	Knowledge extraction.	To create a model to give information about downtime and maintenance.

	(Koukias, Nadoveza, and Kiritsis 2015)	Knowledge extraction.	To support asset management with technical documentation.
	(Yuan 2017)	Software architecture.	To improve the understanding of architectural structure of software.

Table 30: Context analysis result table

5. Discussion on Results

After having studied the systematic literature review, it is now possible to consider the relevant points and conclude this research. In particular, the aim of this final chapter is to discuss the results emerged during the analysis, so that it can positively contribute to future investigations. In order to do that, this chapter is divided into three main sections: the first one, named descriptive analysis results, focuses on the macro trends of the ontology field; the second one, called taxonomy analysis, explains the results of the taxonomy and the meaning of the terms used in ontologies to explain the main concepts; the third section, named content analysis result, is a brief summary on the topics emerged in the previous content analysis chapter.

5.1.Descriptive analysis results

This first section aims to illustrate the results of the statistic part, namely preliminary statistics. For each statistic, there will be a brief comment on the results, the future trends, and their implementation.

5.1.1. Period of analysis

The period of analysis shows how ontology is becoming more and more important in the maintenance and reliability field. Ontologies started to be used as an instrument to solve the heterogeneity problem and to share knowledge at the beginning of the 2000s and nowadays they are also widely recognized in the industrial sector. Considering the increasing trend of pervasive systems and industry 4.0, it is possible to foresee an increase of ontologies in the maintenance and reliability sectors.

5.1.2. Categorization – Application field

The first categorization methodology shows how the ontologies are used in order to improve the process in different ways. In particular, it is interesting to notice their implementation in order to a create a) a sort of historical guide line on events and b) a connected and shared system where knowledge is widespread along the process. As far as the maintenance is concerned, their use is based on the improvement of predictive maintenance through the knowledge of the systems or on the prognostic health management of the systems. With regard to risk management, ontologies are relevant for FMEA and risk mitigation. In particular, it will be possible to better investigate risk mitigation through ontologies and the domain knowledge in the next years. Ontologies are also used in data analysis, although in a marginal way. In this case, the main topic is the analysis of spreadsheet and technical documentation.

With regard to maintenance, risk management and data analysis, the topic could be deeply investigated. Firstly, in the maintenance the application of domain ontologies to study the correct balance between the predictive and the condition-based maintenance can be investigated. Regarding risk management, the analysis can be focused on the propagation of the risk inside the industry applying techniques oriented to system analysis such as the reliability block diagram. In conclusion, in data analysis a deepening in the categorization of the technical documentation through standard policies and terms is possible.

In the next years, it is possible that ontologies will provide a lot of alternatives, especially in the industrial sector.

5.1.3. Categorization – Role of ontology

In order to make things easier and clearer, this thesis identified only three main roles of ontologies. However, it is important to point out that ontologies have many roles, which are different from each other in many ways.

The three main identified roles were sharing, storage and using knowledge. It is quite common that an ontology has more than one role. Indeed, the trend is to use ontology (supported by other tools) for more activities such as reasoning, sharing knowledge, storing knowledge and the linking of parties.

5.1.4. Definition of ontology

The most acknowledged definition of ontology is the one provided by Gruber. According to the author, an ontology is an explicit specification of a conceptualization and in particular, with these few words, it is possible to link the term ontology with its aim. However, it is necessary to point out that, on the one hand, there are many definitions of ontology and, on the other hand, a lot of researches don’t consider the definition of ontology in the literature review.

5.1.5. Languages used

Nowadays OWL can be identified as the standard and most used language. However, a lot of other languages are used to support the ontology activities. It is very common to find a secondary language to support the ontology activity, but it is not always assured.

5.1.6. Layers of ontology

With the reference to the domain and subdomain levels represented in the Figure 12 the average number of ontology layers is between one and four. Considering the ontologies with one and two layers, it is already possible to discuss half of the ontologies proposed. The trend is to have one or two layers for the domain/subdomain ontologies.

5.2.Taxonomy analysis results

The results of the taxonomy analysis are resumed in Table 31 which is structured in four columns. The column called *concept* describes which is the idea that the author of the paper want to transmit using a certain term, the column *single term* will contain YES if the concept is explained by one single term or NO in the opposite case; the column *single meaning* will contain YES if the terms used has a single meaning (it is possible that more terms are used to explain one single concept) and in the opposite case it will contain NO. The column *suggested term* illustrates a more suitable term for the concept.

Two clarifications must be made. The first one is that the column *single terms* could contain YES also in cases in which a term is the most used with a very popular consensus in comparison with another. The second clarification regards the s*uggested term* column. This will be the most used and suitable in relation to the concept that the authors want to explain in the papers. In the case in which the taxonomy analysis doesn’t give a unique answer to the suggested term or in case there is not sufficient information, the cell will be populated by “---“.

Suggested term	Concept/Relation	Single term	Single meaning
---	An index to define the risk of an event that could occur, creating a negative condition.	NO	NO
FMEA	The description of FMEA.	NO	YES
Product	The concept of an object that is the output of a process or an asset.	NO	YES
Process	A series of actions that are related each other with a single scope.	YES	YES
---	The human entities that interact with the process.	NO	YES
---	The entity that inside a process composed by sub entities that performs material actions to modify the characteristics of the product.	NO	YES
System	The entity that contains all the other entities, forming an environment that interact with other environments.	NO	YES
Function	The active scope that is given by an entity.	YES	YES
Component	A sub part of a bigger entity.	NO	NO
MaintenanceActivities	Activities that allow an entity to perform a correct function in a determinate moment with a determinate goal.	NO	YES

Failure	The state of inability of an entity to perform activities.	NO	YES
---	The condition of an entity.	NO	YES
---	The activities to bring an entity to the health state.	NO	YES
FailureEffect	The events that occur after a failure.	NO	YES
FailureMode	The typology of a failure.	NO	YES
hasFunction	The possibility of an entity to have a certain function.	YES	YES
hasEffect	An entity that creates some effects to another entity.	YES	YES
hasCause	An action that create consequences.	NO	YES
hasPart	The membership to something.	NO	YES

Table 31: Terms / Meaning table

The column *single term* has a higher number of NO than YES, which means that the concept is not expressed by a single term and there is not a completely uniformity in the scientific literature. On the contrary, the column *single meaning* contains more YES than NO. It means that the authors intended more or less the same concept, but they express it in different ways.

5.3.Content analysis results

The Table 32 shows the results of the content analysis. The first column represents the application field; the second column includes the most covered topics; and the third column shows the most popular aim.

Field	Topic	Aim
Maintenance	The most discussed topics are the PHM and the supporting of	There is not a unique and common aim in the examined papers. Some proposed aims

	decision-making process in maintenance.	are: to improve the maintenance decision making; to improve the maintenance operations; to reduce the risk of failure.
Risk management	The most discussed topics are the FMEA and the reliability of the system.	The aim is to improve the reliability of a system.
Data analysis	In this field there is not a prevalent topic. Some proposed topics are: Reasoning over engineering textual data; Equipment diagnosis in automotive plant; Environment data analysis.	The aim of the proposed papers is to use data to improve the system with reasoning procedures or particular analysis.
Process optimization	In this case there are two main discussed topics: the simulation and the improvement of maintenance activities.	The most common aim is to improve the activities of the processes through the creation of dedicated instruments.

Table 32: Content analysis results

5.4.Future developments

There are two possible future developments of this thesis. The first one and the most feasible development could be the comparison between the results of this thesis and the standards proposed by the ISO policies. The second one is based on a deeper SLR on the risk management and data analysis fields, with the aim to discover the existing lacks in the scientific literature. This second possibility is based on the fact that the reliability and data fields are not completely covered by this research and there are a lot of not cited tools like the reliability block diagram that could be improved with the using of ontologies. In case of the risk management the prevalent topic was FMEA without considering other possible tools, and in case of data analysis there are not sufficient papers to develop a complete analysis on the application field.

Bibliography

Aarnio, Pekka, Ilkka Seilonen, and Mats Friman. 2014. 'Semantic Repository for Case-Based Reasoning in CBM Services'. In *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*, 1–8. Barcelona, Spain: IEEE. https://doi.org/10.1109/ETFA.2014.7005195.

Aarnio, Pekka, Valeriy Vyatkin, and David Hastbacka. 2016. 'Context Modeling with Situation Rules for Industrial Maintenance'. In *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*, 1–9. Berlin, Germany: IEEE. https://doi.org/10.1109/ETFA.2016.7733539.

'Agile Project Management'. n.d.

Al-Baltah, Ibrahim Ahmed, and Abdul Azim Abdul Ghani. 2014. 'A Comparative Study on Ontology Development Methodologies towards Building Semantic Conflicts Detection Ontology for Heterogeneous Web Services', 7.

Arena, Damiano, Dimitris Kiritsis, Chrysovalantou Ziogou, and Spyros Voutetakis. 2017. 'Semantics-Driven Knowledge Representation for Decision Support and Status Awareness at Process Plant Floors'. In *2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, 902–8. Funchal: IEEE. https://doi.org/10.1109/ICE.2017.8279979.

Banujan, Kuhaneswaran, and S Vasanthapriyan. 2020. 'Bridge Ontology Architecture for Knowledge Management in Bridge Maintenance'. In *2020 Moratuwa Engineering Research Conference (MERCon)*, 1–6. Moratuwa, Sri Lanka: IEEE. https://doi.org/10.1109/MERCon50084.2020.9185335.

Brereton, Pearl, Barbara A. Kitchenham, David Budgen, Mark Turner, and Mohamed Khalil. 2007. 'Lessons from Applying the Systematic Literature Review Process within the Software Engineering Domain'. *Journal of Systems and Software* 80 (4): 571–83. https://doi.org/10.1016/j.jss.2006.07.009.

C. Maria Keet. 2020. *An Introduction to Ontology Engineering*. V 1.5. 1 vols.

Chen, Ying, Cui Ye, and Guoqi Li. 2014. 'Failure Mode Databases and Their Knowledge-Based Management'. In *2014 10th International Conference on Reliability, Maintainability and Safety (ICRMS)*, 732–36. Guangzhou, China: IEEE. https://doi.org/10.1109/ICRMS.2014.7107294.

Cho, Sangje, Marlène Hildebrand-Ehrhardt, Gokan May, and Dimitris Kiritsis. 2020. 'Ontology for Strategies and Predictive Maintenance Models'. *IFAC-PapersOnLine* 53 (3): 257–64. https://doi.org/10.1016/j.ifacol.2020.11.042.

Chougule, Rahul, and Sugato Chakrabarty. 2009. 'Application of Ontology Guided Search for Improved Equipment Diagnosis in a Vehicle Assembly Plant'. In *2009 IEEE International Conference on Automation Science and Engineering*, 90–95. Bangalore, India: IEEE. https://doi.org/10.1109/COASE.2009.5234132.

Fernandez, Mariano, Asuncion Gomez-Pearez, and Natalia Juristo. 1997. 'Methontology: From Ontological Art Towards Ontological Engineering', 8.

Gamberger, Dragan. 2013. 'A Novel Way of Integrating Rule-Based Knowledge into a Web Ontology Language Framework', 5.

Gruber, Thomas R. 1995. 'Toward Principles for the Design of Ontologies Used for Knowledge Sharing?' *International Journal of Human-Computer Studies* 43 (5–6): 907–28. https://doi.org/10.1006/ijhc.1995.1081.

Harcuba, Ondrej, and Pavel Vrba. 2015. 'Ontologies for Flexible Production Systems'. In *2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA)*, 1–8. Luxembourg, Luxembourg: IEEE. https://doi.org/10.1109/ETFA.2015.7301482.

Hodkiewicz, Melinda, Johan W. Klüwer, Caitlin Woods, Thomas Smoker, and Tim French. 2020. ‘Digitalization and Reasoning over Engineering Textual Data Stored in Spreadsheet Tables’. *IFAC-PapersOnLine* 53 (3): 239–44. https://doi.org/10.1016/j.ifacol.2020.11.039.

Hodkiewicz, Melinda, Johan W. Klüwer, Caitlin Woods, Thomas Smoker, and Emily Low. 2021. ‘An Ontology for Reasoning over Engineering Textual Data Stored in FMEA Spreadsheet Tables’. *Computers in Industry* 131 (October): 103496. https://doi.org/10.1016/j.compind.2021.103496.

Hossayni, Hicham, Imran Khan, Mohammad Aazam, Amin Taleghani-Isfahani, and Noel Crespi. 2020. ‘SemKoRe: Improving Machine Maintenance in Industrial IoT with Semantic Knowledge Graphs’. *Applied Sciences* 10 (18): 6325. https://doi.org/10.3390/app10186325.

Hu, Jialun, and Heming Zhang. 2009. ‘Ontology Based Collaborative Simulation Framework Using HLA and Web Services’. In *2009 WRI World Congress on Computer Science and Information Engineering*, 702–6. Los Angeles, California USA: IEEE. https://doi.org/10.1109/CSIE.2009.547.

Jin, Guiyang, Zhanqin Xiang, and Fuzhai Lv. 2009. ‘Semantic Integrated Condition Monitoring and Maintenance of Complex System’. In *2009 16th International Conference on Industrial Engineering and Engineering Management*, 670–74. Beijing, China: IEEE. https://doi.org/10.1109/ICIEEM.2009.5344503.

Koukias, Andreas, Dražen Nadoveza, and Dimitris Kiritsis. 2015. ‘Towards Ontology-Based Modeling of Technical Documentation and Operation Data of the Engineering Asset’. In *Engineering Asset Management - Systems, Professional Practices and Certification*, edited by Peter W. Tse, Joseph Mathew, King Wong, Rocky Lam, and C.N. Ko, 983–94. Lecture Notes in Mechanical Engineering. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-09507-3_84.

Li, Yu, Bo Sun, Zili Wang, and Yi Ren. 2017. ‘Ontology-Based Environmental Effectiveness Knowledge Application System for Optimal Reliability Design’. *Journal of Computing and Information Science in Engineering* 17 (1): 011005. https://doi.org/10.1115/1.4034129.

Medina-Oliva, Gabriela, Alexandre Voisin, Maxime Monnin, Flavien Peysson, and Jean-Baptiste Leger. 2012. ‘Prognostics Assessment Using Fleet-Wide Ontology’, 11.

Mikos, Walter L., João C.E. Ferreira, Paulo E.A. Botura, and Leandro S. Freitas. 2011. ‘A System for Distributed Sharing and Reuse of Design and Manufacturing Knowledge in the PFMEA Domain Using a Description Logics-Based Ontology’. *Journal of Manufacturing Systems* 30 (3): 133–43. https://doi.org/10.1016/j.jmsy.2011.06.001.

Molhanec, Martin, and Evgeniya Povolotskaya. 2012. ‘Model Based FMEA - An Efficient Tool for Quality Management of the Free Lead Soldering’. In *2012 35th International Spring Seminar on Electronics Technology*, 230–36. Bad Aussee, Austria: IEEE. https://doi.org/10.1109/ISSE.2012.6273078.

Moore, J., R. Inder, P. Chung, A. Macintosh, and J. Stader. 2000. ‘Combining and Adapting Process Patterns for Flexible Workflow’. In *Proceedings 11th International Workshop on Database and Expert Systems Applications*, 797–801. London, UK: IEEE Comput. Soc. https://doi.org/10.1109/DEXA.2000.875116.

Negri, Elisa, Luca Fumagalli, Marco Garetti, and Letizia Tanca. 2016. ‘Requirements and Languages for the Semantic Representation of Manufacturing Systems’. *Computers in Industry* 81 (September): 55–66. https://doi.org/10.1016/j.compind.2015.10.009.

Novak, Petr, and Radek Sindelar. 2013. ‘Ontology-Based Industrial Plant Description Supporting Simulation Model Design and Maintenance’. In *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*, 6866–71. Vienna, Austria: IEEE. https://doi.org/10.1109/IECON.2013.6700269.

Nuñez, David Lira, and Milton Borsato. 2018. 'OntoProg: An Ontology-Based Model for Implementing Prognostics Health Management in Mechanical Machines'. *Advanced Engineering Informatics* 38 (October): 746–59. https://doi.org/10.1016/j.aei.2018.10.006.

Polenghi, Adalberto, Irene Roda, Marco Macchi, and Alessandro Pozzetti. 2021. 'Information as a Key Dimension to Develop Industrial Asset Management in Manufacturing'. *Journal of Quality in Maintenance Engineering* ahead-of-print (ahead-of-print). https://doi.org/10.1108/JQME-09-2020-0095.

Polenghi, Adalberto, Irene Roda, Marco Macchi, Alessandro Pozzetti, and Hervé Panetto. 2021. 'Knowledge Reuse for Ontology Modelling in Maintenance and Industrial Asset Management'. *Journal of Industrial Information Integration*, October, 100298. https://doi.org/10.1016/j.jii.2021.100298.

Rasovska, I., B. Chebel-Morello, and N. Zerhouni. 2005. 'Process of S-Maintenance : Decision Support System for Maintenance Intervention'. In *2005 IEEE Conference on Emerging Technologies and Factory Automation*, 2:679–86. Catania, Italy: IEEE. https://doi.org/10.1109/ETFA.2005.1612741.

Rehman, Zobia, and Claudiu Vasile Kifor. 2016. 'An Ontology to Support Semantic Management of FMEA Knowledge'. *International Journal of Computers Communications & Control* 11 (4): 507. https://doi.org/10.15837/ijccc.2016.4.1674.

Ren, Guoqian, Rui Ding, and Haijiang Li. 2019. 'Building an Ontological Knowledgebase for Bridge Maintenance'. *Advances in Engineering Software* 130 (April): 24–40. https://doi.org/10.1016/j.advengsoft.2019.02.001.

Snyder, Hannah. 2019. 'Literature Review as a Research Methodology: An Overview and Guidelines'. *Journal of Business Research* 104 (November): 333–39. https://doi.org/10.1016/j.jbusres.2019.07.039.

Terkaj, Walter, Tullio Tolio, and Marcello Urgo. 2015. 'A Virtual Factory Approach for in Situ Simulation to Support Production and Maintenance Planning'. *CIRP Annals* 64 (1): 451–54. https://doi.org/10.1016/j.cirp.2015.04.121.

Uddin, M. Kamal, A. Dvoryanchikova, A. Lobov, and J.L. Martinez Lastra. 2011. 'An Ontology-Based Semantic Foundation for Flexible Manufacturing Systems'. In *IECON 2011 - 37th Annual Conference of the IEEE Industrial Electronics Society*, 340–45. Melbourne, Vic, Australia: IEEE. https://doi.org/10.1109/IECON.2011.6119276.

Viinikkala, M., S. Syrjala, and S. Kuikka. 2006. 'A Maintenance Demand Analyzer - a Web Service Based on a Semantic Plant Model'. In *2006 4th IEEE International Conference on Industrial Informatics*, 486–91. Singapore: IEEE. https://doi.org/10.1109/INDIN.2006.275604.

Wang, Keqin, and Shurong Tong. 2008. 'An Ontology of Manufacturing Knowledge for Design Decision Support'. In *2008 4th International Conference on Wireless Communications, Networking and Mobile Computing*, 1–5. Dalian, China: IEEE. https://doi.org/10.1109/WiCom.2008.2510.

Yew, Kwang-Hooi, Oi-Mean Foong, and Tharavink Prasad Sivarajan. 2019. 'Knowledge-Based Improvement of Machine Downtime Management for IR4.0'. In *2019 9th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, 94–98. Penang, Malaysia: IEEE. https://doi.org/10.1109/ICCSCE47578.2019.9068584.

Yuan, Eric. 2017. 'Towards Ontology-Based Software Architecture Representations'. In *2017 IEEE/ACM 1st International Workshop on Establishing the Community-Wide Infrastructure for Architecture-Based Software Engineering (ECASE)*, 21–27. Buenos Aires, Argentina: IEEE. https://doi.org/10.1109/ECASE.2017.5.

Yunni Xia, Xin Luo, Jia Li, and Qingsheng Zhu. 2013. 'A Petri-Net-Based Approach to Reliability Determination of Ontology-Based Service Compositions'. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 43 (5): 1240–47. https://doi.org/10.1109/TSMCA.2012.2227957.

Zhou, Xuan, and Yi Ren. 2011. ‘Failure Ontology of Board-Level Electronic Product for Reliability Design’, 6.

List of figures

List of tables

Acknowledgement

These years at Politecnico di Milano have been the most challenging of my life. There have been many obstacles and problems, but I never gave up. Today, I am very proud of the journey I made and that led me write this final thesis.

I have to say thanks to all my family, my father, my mother, and my brother. They are a constant support in all steps of my life, without them I would never have become the person that I am today. This final result is completely dedicated to them.

I have to say thanks to all the people who gave me moments of joy over these years, especially to my friend Pietro and my partner Morena. Her insecurities and fragilities are my strength.

Last but not least, I want to say thanks to both my supervisor Professor Marco Macchi for the opportunity he gave me to develop this work and to the co-supervisor the Engineer Adalberto Polenghi who supported me during the last steps of my university career.