FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

An Ontology for Case-based Project Management

Master Thesis

for obtaining the academic degree

Diplom-Ingenieur/in

Software Engineering & Internet Computing

submitted by

Diana Toskova
Matriculation number 1428256

Department
Informatics Faculty, Vienna University of Technology

Advisor
Ao. Univ. Prof. Dipl. -Inf Dr. Ing Jurgen Dorn

Wien, 25.09.2019

(Submitter Signature) (Advisor Signature)

Erklärung zur Verfassung der Arbeit

Diana Toskova, Tautenhayngasee 28/5/7, 1150 Wien

„Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe."

Wien, 25.09.2019

Abstract

In today's highly competitive business world, it is not enough for organizations to only be good at producing some product or service. An effective Project Management can bring benefits to the organization, that go beyond just creating a product or service, but can also help achieving their long-term objectives. Such long-term goals are for example achieved while doing Business Process Management (BPM) projects, where important business processes are optimized, in order to improve the overall way the work is done within the organization. The Project Management makes use of different techniques, tools, methods and best practices, in order to guarantee that projects have fulfilled their objectives and are done in the predefined time and budget.

Since organizations are doing many projects, some of which are similar (or at least some parts of them are similar), it is of huge importance for them to learn from the experiences from the past projects, and use this knowledge while doing new projects. This will ensure that the organization will constantly improve its work and stay competitive on the market. Case-based Reasoning is a method that uses old experiences in order to find better solutions when similar (or the same) problems occur. It can be used to support Project Management, in order to make better decisions when planning and executing projects. Having additional knowledge within the domain can further optimize this process and an ontology on Project Management can be a great means in that direction. Ontologies contain knowledge (usually about a specific domain), including different concepts, their hierarchy and interrelations and a lot of additional metadata. It can be further expanded, in order to cover specific types of projects, such as software projects or BPM projects. Creating such an ontology can support the Case-based Reasoning process in all of its phases, helping the Project Manager making more informed and more optimal decisions.

Kurzfassung

In der heutigen Geschäftswelt reicht es nicht aus, dass Unternehmen ihre Produkte und Dienstleistungen gut produzieren, um wettbewerbsfähig auf dem Markt zu bleiben. Ein effektives Projektmanagement kann der Organisation Vorteile bringen, die über die reine Erstellung eines Produkts oder einer Dienstleistung hinausgehen, aber auch zur Erfüllung ihrer langfristigen Ziele beitragen. Solche langfristigen Ziele werden zum Beispiel bei Business Process Management-Projekten erreicht, bei denen wichtige Geschäftsprozesse optimiert werden, um die Arbeitsweise innerhalb der Organisation insgesamt zu verbessern. Das Projektmanagement verwendet verschiedene Techniken, Werkzeuge, Methoden und Best Practices, um sicherzustellen, dass die Projekte ihre Ziele erreicht haben und in der vorgegebenen Zeit und im vorgegebenen Budget durchgeführt werden.

Da Organisationen viele Projekte durchführen, von denen einige ähnlich sind (oder zumindest manche Teile davon ähnlich sind), ist es für sie von großer Bedeutung, aus den Erfahrungen aus den vergangenen Projekten zu lernen und das Wissen bei der Durchführung neuer Projekte zu nutzen. Dies stellt sicher, dass die Organisation ihre Arbeit ständig verbessert und auf dem Markt wettbewerbsfähig bleibt. Das Case-based Reasoning ist eine Methode, die alte Erfahrungen nutzt, um bessere Lösungen für ähnliche Probleme zu finden. Es kann zur Unterstützung des Projektmanagements verwendet werden, um bessere Entscheidungen bei der Planung und Ausführung von Projekten zu treffen. Durch zusätzliches Fachwissen kann dieser Prozess weiter optimiert werden, und eine Ontologie zum Projektmanagement kann ein hervorragendes Mittel in diese Richtung sein. Ontologien enthalten Wissen (normalerweise über eine bestimmte Domäne), einschließlich verschiedener Konzepte, deren Hierarchie und Zusammenhänge sowie viele zusätzliche Metadaten. Es kann weiter ausgebaut werden, um bestimmte Arten von Projekten abzudecken, beispielsweise Softwareprojekte oder BPM-Projekte. Das Erstellen einer solchen Ontologie kann den Case-based Reasoning-Prozess in allen Phasen unterstützen und den Projektmanager dabei unterstützen, bessere Entscheidungen zu treffen.

Outline

1. Introduction

In order to better structure their work, organizations nowadays almost always organize it in the form of projects. Almost every organisation, regardless of its type and domain (technical, educational, social, governmental, etc.) has to deal with numerous projects. Depending on their final goal, they can be classified in different categories, but the common thing is that projects are timed (have a specific starting point and have to, at some point, end) and at the end they reach some kind of goal (deliver some form of result). Although some projects may often resemble others, the projects are usually unique [13]. Generally people from different areas of the organization take part in a project, which also have different knowledge and competences. In project-oriented organizations, it's everyone's job to help the management and contribute to bettering the projects. They take different measures, in order to learn from the projects in the past and use that knowledge for future projects [12].

Depending on their goal, there exist different types of projects. A software project, for example, has the goal of creating some software as a final product. Business Process Management project (BPM-project) on the other hand deals with picking some important processes within the organization (or processes which include collaborations with other organizations) and applying BPM techniques, in order to optimize those processes in terms of cost, time, people involved, etc. Processes surround us everywhere - from simple activities, performed in our everyday life, to complex business processes in organisations, involving many actors and elaborate actions. Everyone within an organization is part of various processes (sometimes without even realizing it), meaning that the importance of processes is not only for the organization itself, but also affects all the individuals in it. Typical process involves a number of actions (or simple tasks), different actors, which may be human actors or some computer systems/machines, various decision points and at the end some (or many) outcome(s) [13]. Processes are usually repeated many times and some of their steps may even be done in an automated way by some tool/machine, etc., which ideally should contribute to making the process more optimized. This is not so easily done, because of the ever changing environment, surrounding those processes and the organization itself. A well functioning process-oriented organization should ideally be able to always adapt itself and react to those changes, but in the real world this is often not the case [13] [23].

The field of Business Process Management is relatively young discipline, but it is already accepted as a standard manner for improving the way organizations of different industries and sizes perform their everyday work. In the Information and Communication Technologies (ICT) world, it brings together the technical and business point of view, in order to optimize the way companies and organizations do their business. It involves different techniques for managing and monitoring processes, in order to enable process re-execution, automation and optimisation. The BPM projects need to gain comprehensive insight into the way specific processes are performed, in order to later find a suitable way to improve those processes. The execution of business processes is sustained from the so called Workflow Management Systems (WfMS), which support their modeling and execution and the storing and documenting of process execution data. Depending on the prerequisites, different WfMSs are developed in the recent years, which are the result of different techniques and methods, supported by different modeling languages [22].

In the field of Business Process Management exists an extensive amount of standard and trusted practices, which are continuously under development with the goal of improving the existing processes in order to do better than the competitors. In this context, finding new ways for optimizing processes, minimizing repetitive work, dealing with bottlenecks and identifying errors, is essential for optimizing the organization in general. It is essential for every organisation to find a way to constantly improve the way the work is done, in order to increase the quality, time to market and effective delivery of their products. Improvement goals mostly include reduction of costs, production time, errors and personnel involved, but also aim at better consistency and adaptability of processes and making them easily automatable and repeatable. The processes and the actors, involved in them, should be able to adjust themselves easily to new requirements and unexpected events. An important property of a well functioning organization is for it to be able to able to identify problems within processes on time, instead of repeating the same mistakes over and over again. The general goal is to create processes, which are adaptable and durable and to make them and the organisation itself more flexible [13].

Despite all of the work done during BPM-projects, in order to optimize the processes within the organizations, still the majority of projects are doomed to fail due to various reasons. The most common of those reasons include: insufficient planning and risk management, not accurate cost, time and effort estimations, poor project management and leadership, lack of communication. Discenza [10] classifies these reasons in 7 categories and according to him,

failure reasons often include: dealing with unclarities, insufficient customer involvement, lack of motivation and inadequate tools and technology. Even more general, the failure reasons may be categorized into three categories: people, processes and communication. Attkinson [4] also classifies the criteria into three categories - time, cost and quality, to which he referred to as the “Iron triangle”. Defining failure criteria is extremely important, in order to overcome those failures.

As opposed to other projects, Information Technology (IT) projects have to face additional problems concerning the constantly changing requirements and the expansion of the project scope [15]. Furthermore, the ever evolving technology presents uncertainty, which additionally increases the risk levels of a project. According to a study performed by the IT company Geneca in 2011 [18] on 600 executives, approximately 75% of the people believe that the projects they are working on are fated to fail. According to the survey many believe that their projects differ from the requirements, that they spend to much time on reworking things and they even admit that the aims of the project are not fully clear to them [18].

In today’s technologically developed world, computers have the ability to retrieve and store large amounts of data. The data from projects can be stored and this data can be used in the future to learn from the experiences of the past projects and implement the learned lessons into new ones [64]. Case-based Reasoning (CBR) is a technique that uses old experiences for solving new problems. Those problems may vary from very simple everyday actions to very complex ones. The method stores knowledge from problems as cases and then uses those cases to derive some conclusions, that can be used for solving new similar problems. This approach is used in the field of Project Management to support managers (and everybody working on a project) to make better decisions about new projects, based on the experience of previous ones. Having a comprehensive knowledge about the project domain can also be very beneficial for everyone working on a project. In computer science, ontologies are the gathering and representation of knowledge (concepts) from a specific domain (or several domains), which make the understanding of those concepts, their classes and the relations between them a lot more understandable. A domain theory may further help Case-based Reasoning approaches, in order to simplify and optimize the finding of similarities between cases and to derive better conclusions [28][45].

1.1. Case-based Project Management and Domain Knowledge

In order to catch up with always changing and increasing demands for their products and services, organizations nowadays have to be constantly ready for changes. Therefore, disciplines as Project Management have evolved, in order to support them for meeting those requirements. Since most of the work is done in the form of projects from different sizes, having good Project Management methods is crucial for organizations that want to stay competitive in the market. It is not sufficient for them to do a good product, but they also have to be able to do it in time and in the limits of the specified budget. Despite the measures taken in that direction, according to the Project Management Institute (PMI), still an extensive amount of projects are doomed to fail, especially because they fail to meet the requirements regarding costs and time [42][44].

The organizations often make use of processes, in order to achieve better execution of repetitive tasks. Essential characteristic of good processes is for them to be able to adjust themselves, depending on the ever changing environment [3]. Business Process Management is a discipline that involves many other disciplines, concerned with the processes within organisations. It is important for business managers, industrial engineers, people involved in the internet technologies and basically every employee within an organisation. The Business Process Management provides an opportunity to improve quality of products (or services), compliance and the performance of the organisation. It arises in the early 90s with the trend of organising business around processes. The basic idea behind it is that each product or service, delivered to the market, is the result of many actions and decisions taken by different actors. Each process takes some input, performs some tasks on it and some output is expected, which should bring value to the customer. For the business it is extremely important to understand the interactions between those activities, in order to improve the corresponding processes. Basic characteristic of successful company is the ability to adapt existing products to the market demands and to promptly create new functionalities and products, when needed [60].

In almost every organisation there is an enormous gap between the understanding and vision of IT specialists and business management. The Business Process Management is also a bridge between the two and allows IT specialists to create and monitor products in a

way that it corresponds with the vision of the management [13]. The interests of the business administration community from one side include reducing costs of products/services and increasing customer satisfaction. On the other side is the computer science point of view, where scientists try to investigate the structural characteristics of processes, using abstractions of real business processes. Modeling tools are a great way to represent the processes, but often the understanding of the surrounding domain is lacking for many people involved in the project. Software engineers are also interested in bettering the existing and creating improved software systems, that can support the management and ease the understanding of the domain for everyone involved. Especially in the current century of extreme development of computers in general, Internet of Things, Machine Learning and Artificial Intelligence, information systems play even more important role in the area, since they support more and more projects within companies. The Business Process Management Notation (BPMN) and the Business Process Execution Language (BPEL) were both created to make the communication between the business and technology fields easier. Those languages follow the basic programming principles, which makes them easy for programmers to follow and reproduce into code. They are also simple to understand, which ensures that they will be easy for business people to learn. BPMN is widely used for specifying business processes as models. The so called Business Process Management Systems (BPMS) are used to coordinate everything that is involved in a process [26][38].

Business Process (BP) (or Workflow) Modeling is part of the Business Process Management, which is used for creating models and visual representations of processes. Stakeholders and actors that should participate in a process, often come from different domains, meaning that they have different competences and knowledge. For someone with a business background for example, will be a lot more difficult to understand some technical sequence of steps from the process. Therefore, models make it a lot easier for everyone to get the same understanding of the process. During the execution of a process, the modeling makes it easier to specify what changes can be done, in order to improve the current process. In order to outperform competitors, each organisation should aim at polishing and orienting its processes towards its business strategies. The Business Process Modeling ensures the better use of resources and shortening process execution time by giving the opportunity to simulate processes (even from different viewpoints), which makes it easier to spot the problems and come up with ideas for solutions.

As the majority of projects are doomed to fail, it is extremely important for organizations to learn from the past experiences. Constantly creating new knowledge and using that knowledge for future projects is crucial for organizations that want to be competitive and successful [66]. But still many organizations cannot effectively combine the disciplines of Knowledge Management (KM) and Project Management. When organizing the work in projects, each project deals with huge amounts of knowledge (depending on the size and type of the project). In today's world of “knowledge creating” organizations, the way they deal with that knowledge, how effectively they capture it from a project and how they use it for future projects, plays a big role for the success of an organization.

In order to stay competitive on the market and grow, organizations have to constantly strive to improve the way they perform their work. One of the most important things that need to be done for improving the way projects are done within organizations are to analyse the situation before starting the project and not to start in a way that is doomed to fail from the beginning. For that reason, learning from past projects and past experiences is crucial for improvement in the future. But in order to be able to make this comparison at all, the important information from old projects must be stored somewhere. Case-based Reasoning is a method that is used exactly for that purpose. It is a process of finding solutions for problems and approaching new projects, by considering old experiences, that are stored as cases. A case usually contains the appropriate action that is taken as a response of a given problem or situation. It contains a single project instance, the experiences from which can be used when approaching a new project or can even be used by other teams within the organization (or even beyond the organization) [11][16].

Case-based Reasoning is applied when working on a project in a way that similar old projects (or problems) have to be found, in order to learn from the experiences of those projects. For finding similarities between cases, a similarity function has to be chosen. Domain ontologies can be used for defining such functions for similarities by using the relationships between concepts of the ontology. This type of similarity function is based on ontological similarity [33]. Ontologies are an appropriate method to capture the most important domain terms of a project, which can further contribute to improving the phases of Case-based Reasoning. In the field of Computer Science (CS), ontologies are defined as representation of concepts within a specific domain (or combination of domains), which should define the properties of each concept and the relationships between different concepts. Gruber [21] simply defines an ontology as “an explicit specification of

conceptualization". Ontologies combine different methods from software engineering, the World Wide Web (WWW), Artificial Intelligence (AI), Formal Logic (FL) and Automated Reasoning (AR) [20]. They make domain knowledge understandable for machines by representing it as an explicit conceptual knowledge model. In the field of Semantic Web, they serve as a means for making websites understandable from information systems. The goal of this thesis is to create an ontology on Project Management, which should improve the whole CBR life-cycle [11].

The goal of the thesis is to answer the following research questions:

- Which are the most important concepts of the management of projects (and in particular the management of Workflow Management System projects) and their interconnections?
- Can an ontology on Project Management help derive better conclusions in Case-based Reasoning (CBR)?

1.2. Expected Results

This thesis should answer the research questions from the previous section by firstly exploring the literature sources regarding the topics. It should investigate the fields of Project Management and Business Process Management and identify common factors for project failures. It should define a set of conclusions, that most often will lead to the failure of the project or its success.

There exist techniques that make use of Case-based Reasoning, which help organizations to learn from the experiences from past projects, in order to achieve more successful projects in the future. The current thesis will propose an ontology, which will include some of the most important concepts of the Project Management domain. The proposed ontology should model the reality of a project in a best possible way and will additionally contribute to understanding the environment surrounding the project and will clear the structural characteristics of a project.

The current thesis will show if such an ontology (and the domain knowledge in general) can help derive better conclusions in existing Case-based Reasoning approaches and if the domain knowledge can contribute for improving some or all of the phases of the Case-based

Reasoning life-cycle. The thesis will define some important conclusions from the Project Management domain and will show that the proposed ontology can help deriving those conclusions and respectively improve the Project Management process, which finally should lead to the improvements within the organization in general.

In the final part of the thesis, it will be evaluated, if the proposed ontology can somehow contribute to improving Case-based Reasoning and Project Management in general.

1.3. Structure of the Thesis

The thesis will begin with exploring the theory of Project Management, Business Process Management, Ontologies, CBR in section 2. Section 3 will describe the practical part of the thesis. It will explore the implementation of the ontology by defining the scope, concepts and properties, defined in the ontology. A set of conclusions will be defined in section 3.1, which will then be derived using the knowledge embedded in the ontology. Section 4 will be an evaluation, which will describe the effects of using the ontology with CBR. Section 5 will be a conclusion.

2. State of the Art

In this section we will do a comprehensive literature review of all of the topics: Knowledge Management and Domain Knowledge, Ontologies, Project Management, Business Process Management and Case-based Reasoning. It will define the most important terms and concepts of each ot those fields. The defining and understanding of the most important terms is crucial for the later practical implementation of the ontology.

2.1. Knowledge Management and Domain Knowledge

In today's world, where every product becomes outdated extremely quickly, the handling of knowledge and the constant creation of knowledge that can be used in the future and that can be learned from, is typical for the organizations that want to remain prosperous. According to Frappaolo [16], Knowledge Management includes:

"practices and technologies that facilitate the efficient creation and exchange of knowledge"

The discipline also deals with the capturing and acquisition of this knowledge, its organization and representation, in order to make it useful. Even in 1991, Nonaka [37] begins his famous article for the Harvard Business Review magazine with the saying: "*In an economy where the only certainty is uncertainty, the one sure source of lasting competitive advantage is knowledge*". In his article he comes up with the differentiation between *explicit* and *tacit* knowledge. He describes *explicit* knowledge as easier to capture, because of its systematic nature. It is easily understandable for machines and includes concrete information, such as, for example, the changes of the costs over time. The *tacit* knowledge is a lot harder to capture and use, because of its subjectivity. Tacit knowledge is the personal knowledge of individuals, which very often is the main impulse for major innovations. According to Nonaka, the constant creation of new knowledge and its use for uppercomming projects and products is the reason why some companies succeed. In order for companies to succeed, they have to find a way for integrating knowledge management into their Project Management practices [37][66].

Since the available knowledge in the world is basically infinite, it is important to classify it and organize it in some way. Domain knowledge is a term, which characterizes knowledge about

a specific field (or domain). Domain may be every aspect of the world. As part of this thesis the Project Management domain will be explored. Different concepts (words) may have different meanings, depending on the domain, from which point of view we are seeing the concept. For example the word “virus” as part of the “computer science” domain has the meaning of type of malicious software, while as part of the “medicine” domain it means a small infectious agent. Often, domain knowledge is very specific, such as mastering some specific technology. People who have extensive understanding of a specific field (domain) are considered as domain experts. Not every person that is part of a project is an expert of every aspect within that project. For example, a frontend engineer may be not familiar at all with the database domain.

In order to make use of the knowledge within projects (and in general), organizations have to find ways for capturing and managing this knowledge. They have to define different methods and make use of appropriate techniques, in order to make the most of their knowledge. Ontologies are a way for representation of knowledge (usually of a specific domain). In the computer science field, ontologies represent “maps” of concepts of a domain. They can be classified in three different types - domain ontologies, upper ontologies and hybrid ontologies. As part of this thesis we will consider the domain ontologies, which represent knowledge, which is part of a specific domain. The upper ontologies represent the relationships between concepts from different domains (a set of random concepts) and the hybrid ontologies are combination between the previous two types. Ontologies can be used in order to better represent the domain knowledge, by defining a variety of domain concepts, which are described by their properties, classified in different categories and classes and are linked together, representing the relationships between those concepts. Ontologies can be used to find the relationships between different concepts and are usually used for reducing the complexity of the understanding of a domain [16][25].

Sometimes while creating ontologies, the amount of concepts, attributes and rules may grow really fast, exposing the possibilities for mistakes and inconsistencies. While developing the ontology, the use of reasoner can make making the ontology error free much easier. Several reasoners come as plugins directly with the ontology creation tools, which makes them very convenient to use without many additional steps to be performed. Famous reasoners include: HermiT, Pellet, FACT++, Racer, etc.. Those have different characteristics in terms of the logic that they support, the algorithm for checking consistency that they use, the language that they are implemented in, etc..

2.2. Project Management

As already mentioned, it is typical for today's organizations, that they organize their work as projects. According to the Project Management Body of Knowledge (PMBOK), created by the Project Management Institute (PMI) [42], a project is defined as:

"a temporary endeavor undertaken to create a unique product, service or result"

No matter if some of the projects within organizations seem to be very similar, but they are still unique, due to some unique characteristics of the current project and its result. Let's take a software development company as an example. They usually make use of the same technologies (that the team knows best), use the same development methodology (ex. Scrum, Waterfall, etc.), and may even repeat the same setup, testing, deployment, etc., but still the product that is produced at the end is different, it is a new product for the company. Projects usually produce some change for the organization as a result, which should bring benefits to it. In order to produce some kind of benefit, the projects are most often concerned with certain levels of risk (risk of losing clients, losing the trust of the clients, losing money, etc.). Important property of a project is that it is timed, meaning that it has some specific beginning point, but also eventually ends (although sometimes can take a lot more than the planned time, or the planned time can be really long from the beginning). The end of a project usually comes when the goals, defined at the beginning, are achieved, but sometimes it can happen that projects are discontinued (for example if the customer is not satisfied with the way the work is done, or if the organization cannot carry out the project further, for some reason). Furthermore, a project is restricted by a predefined set of resources, meaning that the amount of materials, tools, devices, people and their time is specified at the beginning and should ideally be kept in the pre-defined amounts during the execution of the project [35][42][61].

The discipline that deals with such projects is called Project Management. The Project Management Institute (PMI) [42] gives the following definition for Project Management:

"Project management is the application of knowledge, skills, tools and techniques to project activities to meet project requirements"

According to Westland [61], it incorporates different tools, competences and processes, which a manager can use (or possess), in order to achieve project success (Fig. 1). Needed competencies include a good domain knowledge of the manager, as well as leadership, communication, organization, risk management (and many others) capabilities. In today's technologically developed world, the possibilities for software tools that managers can make use of, are endless. Their purpose is to ease the work of the managers and to contribute for better results. Even machine-learning technologies (such as the Case-based Reasoning) can be incorporated, in order to learn from past projects [35][61].

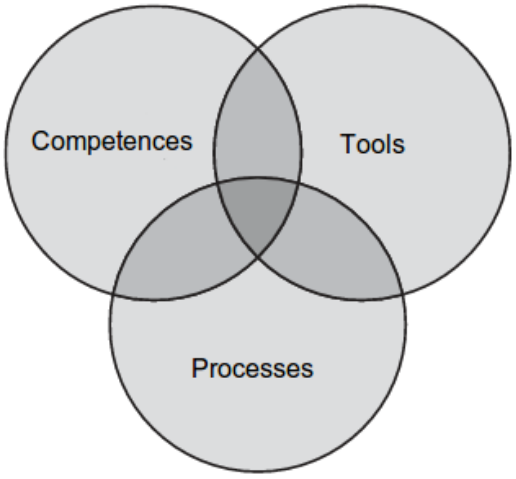

Fig. 1 Project Management Components (adapted from [61])

The most important role of a project is the role of the Project Manager. He/she is the person who is responsible for accomplishing the project goals. The Project Manager is chosen by the organization that will execute the project and should apply different well established techniques and approaches and should possess a variety of competences and knowledge specific to the project domain. Additionally, important characteristics of a good Project Manager include the performance, which he/she can achieve for accomplishing the project goals. Communication and leadership competences are also crucial for performing good Project Management. Other people or organizations that have a connection to the project are called stakeholders. Stakeholders have some effect on the project and they are also affected by its results. Determining all the stakeholders is crucial for good project planning [35][42][61].

2.2.1. History of Project Management

The first evidence of Project Management activities ordinates from ancient Egypt, where the construction of the Great Pyramid at Giza may be seen as one of the first large-scale projects. The modern methods on Project Management, however originate from the times of the Second World War, around the 1950s, when the U.S. navy used some tools and techniques from Project Management for their project, called Polaris, concerned with the launching of ballistic missiles [3][31].

The development of the Project Management field can be divided into four periods. Kwak [31] makes the following division of those periods, shown in Table 1:

Period	Theme
Prior to 1958	Craft system to human relations Administration
1958 - 1979	Application of Management Science
1980 - 1994	Production center: Human resources
1995 to Present	Creating a new environment

Table 1. Four periods of project management [31]

During the first period, the period of Craft system and human relations administration, the hand-labor, which was very time-consuming, was substituted by technological inventions. Especially during (and after) the Industrial Revolution, the invention of powered tools and machines, the creation of production factories and the boom of mass production has tremendously fasten the course of the projects. The invention of the Work Breakdown Structure and the Gantt Diagram further contributed to the development of the modern Project Management methods [31].

During the Application of Management Science period, major technological inventions arose in different industries such as: military and government organizations, telecommunications, airlines or manufacturing. The invention of the foundations of the internet, the Apollo missions of NASA to the moon, the creation of the UNIX programming language, the

invention of the first automatic copier, the creation of the Microsoft company and many others were significant contributions to the development of modern Project Management. Furthermore, many tools for PM were created, such as the Critical Path Method (CPM), which is a statistical technique for managing activities of a project that are well defined, or PERT (Program (Project) Evaluation and Review Technique) - a statistical technique for managing uncertain activities within a project [31].

The third period of the history of Project Management was mostly influenced by the massive developments in the Information Technology area. The computers for personal use became more and more popular in the 80s and the creation of different Project Management made it easier to apply Project Management standards in a regular and convenient way. This tendency continued also in the fourth period (1995 to Present), when the use and importance of Internet is constantly growing in every Project Management aspect [31].

2.2.2. Project Management Life-Cycle

The phases (or a life-cycle) of a project include the activities, that need to be performed from the beginning (or actually from the idea about the project) to its finale (when either the goals of the project are met or the project is terminated). During those phases, a standardized activities are performed and each one of them can be linked to specific time of the project schedule. This does not mean that one phase terminates and the next one begins, some phases can also continue during the following phases (or during the whole life-cycle of a project). According to the PMBOK [42], a project life-cycle can be divided into five phases: Project Initiation, Planning, Execution, Closing and Monitoring & Controlling. While the first four follow a specific schedule (the Closing phase, for example, cannot come before the Initiation phase, or before the project has started), the Monitoring & Controlling phase spans throughout the whole project development [29][42][61]. The whole life-cycle can be seen in Fig. 2:

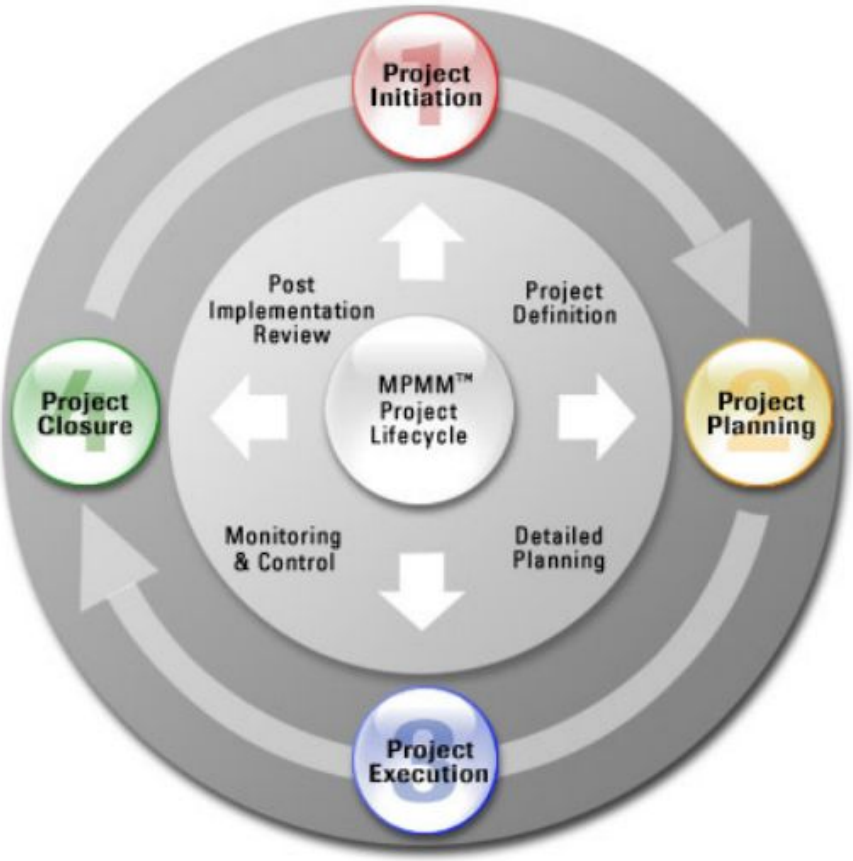

Fig 2. The four phases of the project life-cycle [35]

During the **Initiation Phase**, the idea about the project is considered as a solution for existing problems or as a chance for the organization. The phase represents starting a completely new project, but also starting a phase in an existing project. An important part of the Initiation phase is defining the project scope and objectives (exactly what the project should achieve) and to check if those objectives correspond to the objectives of the organization in general. The project deliverables are also defined, which include the products or services that the project has to produce at the end. Another output of the Initiation phase is the project mission, or the reason why the project will be done. The Project Manager is usually selected during this phase. Although sometimes neglected, the exact definition of the project stakeholders at this point is crucial for the further project success. Important activities of the phase are the creation of documents, such as the Project Initiation Request (from some client or the organization itself), of the Business Case (includes the context of the problem and possible solutions) , or the Project Charter (which specifies requirements about time, money, objectives, scope, etc.) [29][35][40][42][61].

After the objectives and the mission of the project are defined in the Initiation Phase, an exact plan and schedule of actions has to be created during the **Planning Phase**. An output of this phase is the Project plan, which includes planning the costs and the budget, the resources, the time and schedule and all of the possible risks, quality requirements, etc.. The exact Work Breakdown Structure (WBS) has to be defined, which consists of dividing the work that needs to be done, in smaller chunks. Other important things that need to be defined during the Planning process, are the communications (within the team and with all of the stakeholders) and the procurement (all of the products or services from external providers). Because of the sometimes changing environment, surrounding the projects and some unexpected things that may happen during the execution, the Project plan can be adjusted during the next phases of the project [29][35][40][42][61].

In the next phase - the **Execution Phase**, the actions from the Project Management plan (from the previous phase) are performed, in order to ensure the project objectives. Additionally, the phase includes managing activities on the team, which have to ensure that the work goes smoothly and if some issues arise, they will be resolved. The communication and procurement activities are carefully managed. Furthermore, the Project Manager has to ensure that all of the affected stakeholders take part in the decision making and are aware of the problems that may occur. Different processes for risk and quality management are also performed during the Execution phase [29][35][40][42][61].

After the goals and all of the objectives of the project are met and the customer is satisfied with the result, comes the **Closing Phase** of the project. In this phase, all of the project execution actions are completed and the resources are released, all of the documentation is handed out to the customer and all of the external contracts are terminated. Ideally the project flow is analyzed and some lessons learned are documented, that can be used for future projects [29][35][40][42][61].

The **Monitoring and Controlling Phase** continues through the whole project (and spans over all of the other phases), although is critically important during the Execution phase, when the state of the project has to be monitored constantly. The processes that are performed during the project, are monitored and reviewed and, if needed, optimized. Furthermore, the performance of the project and the quality of the results are constantly checked and if needed, the Project plan is adjusted. This phase is crucial for organizations that strive to always optimizing itself, their processes and projects [29][35][40][42][61].

2.3. Business Process Management

The main goal of each organisation is to constantly improve itself - to adapt to new circumstances, to better deal with errors, to win more venue, to improve customer satisfaction, etc. A basic step that direction can be done by improving the processes of the organisation. Business Process Management (BPM) deals with monitoring the work of everyone within the organisation with the aim of delivering stable results. It is a management discipline from the area of Operations Management, which deals with managing series of interconnected events with the goal of bringing benefits to the customer and the organisation itself, by optimizing its processes and increasing the efficiency of the workflow. In his book, Jeston [24] defines the BPM as:

"A management discipline focused on using business processes as a significant contributor to achieving an organization's objectives through the improvement, ongoing performance management and governance of essential business processes".

The processes coordinate systems, people, information and all of the organisation's assets, in order to establish business results, which promote the business strategy of the organisation. From the BPM viewpoint, the processes within an organisation are seen as essential assets. Therefore, it deals with all of the steps of the process execution. It starts with identification of all processes within the organisation, then deals with modelling them and implementing them. Afterwards processes are analysed and constantly monitored in order to ensure process automation and optimization, which on the other hand should deliver significant benefits to the overall well-being of the organisation. Such benefits may differ, depending on the type of the organisation. In most cases it is about optimizing business results leading to monetary gains, reducing production times, minimising errors, increasing customer satisfaction and the ability of the organisation to react to the constantly changing market [6][13][24][52].

2.3.1. History of Business Process Management

The concept of "process thinking" has its roots long ago in history (around 500 B.C.), when the Chinese military strategist Sun Tzu mentions processes as a way to accomplish some objective in his book "The Art of War" [56]. However, the "father of modern economics" (how he is often referred to) is believed to be Adam Smith, who wrote his book "The Inquiry into

the Nature and Causes of the Wealth Nations" (short: "Wealth of Nations") [54], in which he investigates the reasons why some nations become really wealthy. In his book, he starts the idea of "division of labor", which then evolves in what we today call Business Process Management [46][54]. Another important point in the history of BPM is during the 19th century, when one of the most influential and controversial people in the field, Fredrick Taylor, suggested a set of principles, famous as "Theory of Scientific Management" [57]. The main idea behind those principles was the idea that each worker should only perform one specific task. This idea of labor division placed the roots of the formation of the so called managers, people who should supervise the work of others [13][57].

Another concepts, such as some japanese philosophies, state that the processes are not uniform and the inputs of those processes always change over time, depending on what is happening in the environment at that time. Contrary to the Taylor`s point of view, the japanese organisations believed that their goal should be to keep their processes at their best for as long as possible. Such principles include Lean, main idea behind which is to eliminate the waste and always aim at the highest quality of products and services. Six Sigma is also another point of view, which concentrates on reducing the errors during process execution by constantly checking the quality of outputs. After the emergence of technology, another set of practices arose, called Kaizen, where the main idea was to prompt the people within the company to use their knowledge and competences, in order to propose changes, which will benefit the organisation [8][13].

In today's world, which is dominated by technology, we cannot imagine BPM existing without significant technological support. With the development of Information Technology, business processes within and across organisations have changed. During the 60s each workflow management system was designed to perform a specific workflow within one organisation, because of the restricted functionality of the operating systems. Some decades later, with the improvement of data retrieval and storage, the data-driven approaches started to be more and more incorporated within the workflows of organisations. Information technology eases the BPM by supporting information management, and communication for the different types of information involved in a business. ERP software such as Oracle, SAP or Baan, which also include workflow management elements, also have seen enormous development in that direction. Nowadays some modern BPM solutions even make use of cloud computing in a way that they use tools, which are accessed as services over the network. In that case the BPM logic uses an application server, where the data is stored in the cloud [46].

2.3.2. Business Process Overview

In order to produce some product or deliver a service, organisations (no matter of what type) perform a series of activities, which may include single tasks or complex actions, decision points and should at the end produce an outcome, which is of some benefit to a customer or to the organisation itself. The processes are in most cases performed frequently and therefore can be also automated. The first modern description of a Business Process dates from 1776 and Adam Smith [54] and later leads to the creation of areas such as BPM and the development of different Business Process Management Systems (BPMS) [46]. In his book “Fundamentals of Business Process Management”, Dumas [13] defines a business process as:

“a collection of interrelated events, activities and decision points that involve a number of actors and objects, and that collectively lead to an outcome that is of value to at least one customer”

A process has some underlying goal and ends with some output, which either corresponds with the initial goal or in the other hand gives some indication, that the process may need to be optimized. Processes have a starting point, but then they do not just end at some point, but are rater repeated over and over again. Each single step within a process is dependant on the other ones (previous and following), therefore we can say that the steps happen in a specific order. If the process is more complex, it can be also divided into some subprocesses, each of which has its own flow but is again dependent on the others and should contribute to the overall aim of the process. Those subprocesses can also be categorized and grouped, depending on their function. A process owner is a person, who is responsible for ensuring that this goal is met. Not every process has such process owner, but in general it is beneficial to have someone that oversees the process and makes sure that everything goes smoothly [13][52].

In order to be more understandable, processes may be modeled using some modelling technique, which represents all of the activities, decision points, actors and the data flow of the process. Simply explained, the term Business Process (Workflow) Modeling indicates a (visual or textual) representation of the actors, involved in a process, the steps that they perform and the decision logic and the communication and interaction between them. The Business Process Modeling is part of the Business Process Analysis and there exist many

different modelling techniques. The Business Process Modelling Notation (BPMN) is the most commonly used one of those techniques, where the result is a draft of the business processes flow [30][52][62].

2.3.3. Elements of a Business Process

In Fig. 3 is displayed Dumas's representation of a business process, including its elements and the correlations between them [13]. This subsection describes all of the ingredients involved in a business process:

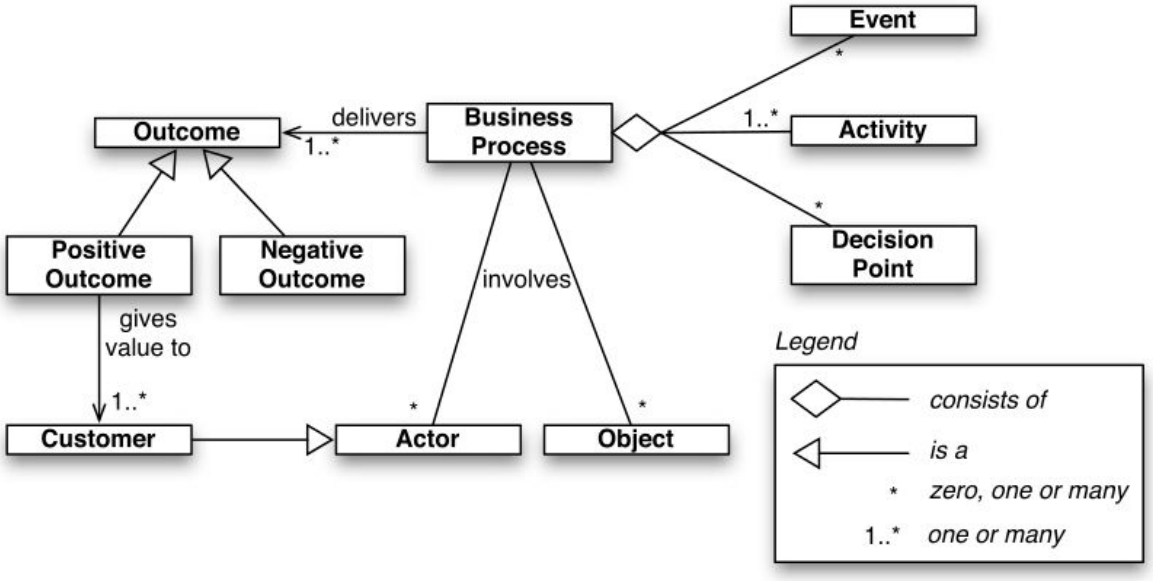

Fig. 3 Ingredients of a Business Process [13]

Actors are the ones who execute the tasks in a process. Actors in a business process may be of different types - human actors, organisations, physical and not physical objects, software systems, etc.. Whole organisations may also play the role of an actor in a process, for example some company that deals with deliveries of goods, or company that submits some type of products. Physical objects (such as computers) and immaterial objects (for example some documents in electronic form) may also be actors in a process [13]. An important actor of each process is the process owner. He/she is in charge of the process and has the mission of inspiring process thinking within the organization and improving its processes.

Events come up during the run of a process and influence its course and outcome. Usually something has to trigger the event, the so called cause. It is something that happens, which

sets off the event. The trigger may also be another event. Each event has also some consequence - something that comes up as a result from the event. Events have no duration, they happen atomically at some time point. Depending on when the events happen, they may be divided in three types - start, intermediate and end events. Start events initiate the process. They may be something that triggers the start of a process (like another event, for example making a purchase for some product or service) or may happen from itself at specific time point (for example at the first date of each month). The intermediate events happen in the course of the process and affect its run, but do not lead to the end of a process. They again may be triggered by something else or always happen at a specific time point. The last type of events (based on when they happen) are the end events, which terminate the process and cannot lead to another event [39]. As already mentioned, events have no duration. If the thing that happens takes time, it is called activity. Activities can be very complex and take extensive amount of time or really simple. In that case they are called tasks. Tasks usually require just one step of work, while activities contain a number of such steps [13].

Furthermore, a process may contain decision points. Those are points during the execution of a process, when some (or more) decision(s) is made. Decision points determine how the process flow goes and the differences between different process runs. The decision points may differ in a way that a different number of alternatives can be picked. In some cases exactly one of the alternatives can be chosen, while in other cases multiple options can be picked [13].

After executing all of the actions and going through all of the decision points, the process finishes with one or more outcomes. Among all the actors involved in the process there is one that plays a special role - the customer. He is the one that uses up the outcome of the process and will be unhappy if the outcome is negative. The customer may be part of the organisation (for example human actors, such as managers or team leads, or organisational actors - lets say a whole division of the company) or be external to it (like the person/organisation who purchases the product/service). There may also be not only one customer, but several, who may differ by type. If the process ends in the best possible way, the outcome would be beneficial for the customer. On the other hand, the outcome of the process may be negative and not bring any benefits to the customer or only partially bring some value to him.

2.3.4. The BPM Project Life-cycle

In the previous sections we have defined what a processes are and how BPM can be applied within the organization, in order to optimize those processes. The projects, which deal with the application of BPM practices, in order to optimize specific processes within the organization and respectively to bring general improvements within the organization, are classified as Business Process Management projects. Generally, we can divide the life-cycle of a Business Process Management project into six phases. Of course, different people from the field could come up with different names for the phases, split them in even more counterparts or even remove some of the phases. As part of this thesis we define the phases as follows: process identification & discovery, process analysis, process modeling and (re)design, process implementation, process monitoring and controlling and finally - process optimisation.

The first phase - **Identification & Discovery**, consists of firstly identifying the most important processes of the organization, which are usually not so hard to identify, because of their importance to the customers. Furthermore, the most problematic processes need to be identified, because not fixing their problems may have crucial consequences on the organization.

After identifying and prioritizing the processes and determining the interconnections between them, comes the second phase of the BPM Lifecycle - the **Process Analysis**. Processes are analysed, in order to determine if the actual business processes are oriented towards achieving the aims of the organisation, if they produce some errors, wastages or redundancies.

Depending on the results of this phase, the third phase includes **(Re-)design** of the selected processes, in order to improve deficiencies or Business Process Reengineering, which seeks at completely redoing the process structure.

After the processes are (re)designed, the next phase of the BPM project is the (systematic or non-systematic) **Process Implementation**. With the developments in the IT sector and the emergence of Workflow Management Systems (WfMSs), such systems are used for managing the processes within the organization. They provided the infrastructure to monitor

a set of related tasks. Those WfMSs made it possible to oversee the business processes and evaluate their outcomes. After the birth of Web Services (WS) WfMSs more easily were linked to (for example) Enterprise Resource Planning (ERP) systems, which were used to keep all of the organisational data at one place, making it easier for everyone (who needs it and has the right) to access it. Such conjuncted systems are called Business Process Management Systems (BPMSs), which make it possible to more easily improve processes [13].

After the implementation of the processes, they need to be monitored throughout their whole life cycle, which brings us to the next BPM phase - the **Monitoring and Controlling** phase, which is not a single activity, but a continuous one and has the purpose of gathering information about if the process is optimal enough or some changes should be made to the process design or the implementation tools. This enables the organisation to always keep track of the state of their processes and eventually leads to the next phase of BPM - the **Process Optimisation** phase. By reengineering and redesigning processes the organisation can achieve process management optimisation and overall optimisation of the organisation [13].

2.4. Case-based Reasoning

Case-based Reasoning is problem-solving and learning discipline, ordinating from the Machine Learning field, which makes use of the knowledge of old situations by finding similarities to the current situation. It is an incremental learning approach, where either the new problem is solved each time or it is registered as a new, so called *case*, which can be again used for the problems to follow. The technique is adopted by the typical way how people solve problems in real life, by repeating the same solutions for the same problems, which makes the CBR related to the field of cognitive psychology [1][17].

2.4.1. History of Case-based Reasoning

The field of Case-based Reasoning is part of the machine learning discipline, but its roots lie in psychology, or more specifically the cognitive psychology, which is, simply put, the study of the way people think (for example: how they use their memory, how they study languages or how they solve problems). In 1953 Wittgenstein [65] in his work “Philosophical Investigations” (german: “Philosophische Untersuchungen”), which is even today believed to be extremely influential book in the field of philosophy, analyses the way people learn

languages. He states that the meaning of each word (or concept) is defined by an instance of its characterizing features, which can be connected to the current meaning of “case” within CBR. Other records of the roots of current Case-based Reasoning can be found in the works of Tulving [59], who first makes the distinction between semantic and episodic memory. He proposes a method, which is related to the current CBR method, where information can be stored and later recalled. Also in the 70s, Rumelhart [47] proposes his *schema theory*, where pieces of existing information are used for making decisions in the present.

The modern CBR originates from the works of Schank [50], who proposes a model on dynamic memory at the Yale University [1]. His research continued Silvan Tomkins [58] work on his *script theory*. In this theory human behaviour can be summarized in analogically functioning patterns, called “scripts”. Schank’s Memory Organization Packets (MOPs) [49] and Thematic Organization Packets (TOPs) [51] were also important for the creation of current CBR methodologies. MOPs represent a combination of general and specific (based on experience) knowledge structures [49] and TOPs characterize situations by the intentions of the actor [51].

The research on Case-based Reasoning methods started in the United States, but later the researches on memory-based reasoning, analogical reasoning (conducted by Genter [19] in 1983), and other related fields (and their combination) led to the increased fascination by the CBR topic in Europe and Asia. As a result, the International Conference on Case-based Reasoning was created and it was firstly conducted in Portugal in 1995 [1].

The creator of the first CBR system is considered to be Janet Kolodner, who conducted research in the topics of problem solving and learning. She came up with the CBR method and its use for everyday situations. She was the creator of the first CBR system, the CYRUS system [27], which is based on the Schlank’s method [50]. Other CBR systems with her involvement are MEDIATOR, PERSUADER, JULIA, EXPEDITOR, CELIA, MEDIC, etc. Another famous system is Michael Lebowitz’s [32] Integrated Partial Parser (IPP), which is a generalization model, also based on the Schlank’s model on dynamic memory.

2.4.2. Case-based Reasoning Overview

Simple definition for Case-based Reasoning is given by Richter [45], who describes it as:

"Case-based reasoning (CBR) is a methodology for solving problems"

Even more particular, it makes use of old experiences, in order to solve new problems. Aamod [1] also defines Case-based Reasoning as the following:

"To solve a new problem by remembering a previous similar situation and by reusing information and knowledge of that situation"

In order to solve new problems (to make a decision on how to proceed in a given situation), Case-based reasoners use solutions from the past for the given problem. The solutions from the past are either directly used (if the problem is very similar or exactly the same) or are adapted, in order to fit into the new circumstances. This is the same technique that people (or all living creatures) use for solving problems and proceeding in different situations. For example, when a person is making himself/herself a coffee in the morning, he/she will not try to invent a new way for making coffee, but will rather use the coffee machine in the same way as yesterday. If the person buys new coffee machine, he/she will again proceed in a similar way, but will change a few things in the process, depending on the functions of the new machine (adapting old situations to fit new ones) [17].

There is not one specific CBR methodology that exists. Different methods differ, for example, in the way cases are stored (as single ones or classified and combined, based on similar problem). The cases may be simply stored or may be organized in some type of structure. Depending on the system, cases may be retrieved in different ways, some of which may be way more performant than others. There are also different methods for similarity identification. This thesis will deal with one type of CBR, which makes use of domain ontologies. Other differences exist in the way the CBR system works - it may be fully automatic, but can also need help from people when making a decision about choosing the most similar case [1].

In CBR all of the decision-making is made based on the previous cases. Aamod [1] defines a case as "a problem situation". A case is a representation of some problem that has already happened. It contains a set of attributes, or characteristics of the current situation and the way the problem is resolved. Therefore, cases always represent something that has actually happened and not some ideas about solutions (that may work or not). The attributes are most often represented by feature-value pairs. Within a case there are two types of

information, information about the situation that happened and information about the solution. The decision taken in the previous situation may have led to positive outcomes, but can also lead to problems. Depending on the concrete CBR methodology, cases may also contain negative outcomes and not only positive ones [1][28][45].

The attributes that characterize a problem may be further divided into two types: indexed and non indexed. Indexed ones represent information that can be used for solving the new problem and non indexed ones contain additional information, which could help the person (or system) to make a decision. When a problem occurs, it is compared with previous cases. If no similar case is identified, then the new problem is also stored as a case. In order to really be able to make use of Case-based Reasoning, organizations have to make sure that they are constantly storing each new case in a case-base. The method is different from other reasoning methods by the fact that, because the situations (problems) differ and the surrounding circumstances may differ, therefore one solution that has worked perfectly for a problem, may not work again for the current problem [1][28][53][45].

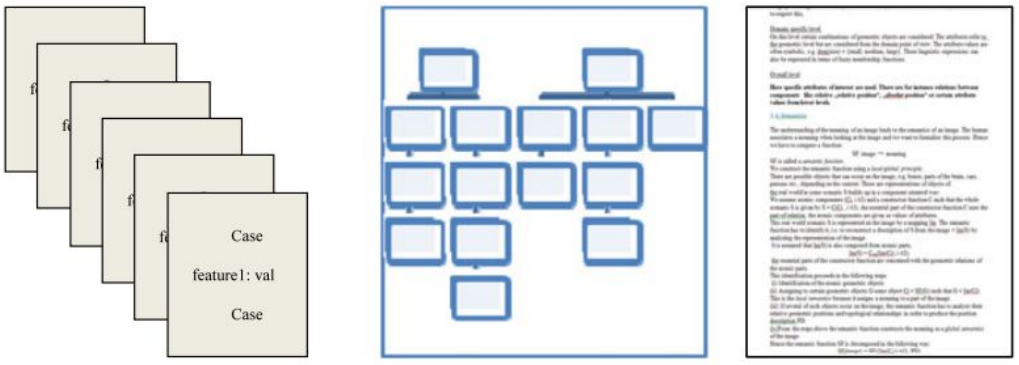

Fig. 4 Types of case organization: flat, structured, unstructured text [45]

The collection of all the cases, that an organization (or a CBR system) stores is called a case-base. A case-base is some form of storage, usually some database. According to Richter [45], there exist three main types of case-bases (which can be seen in Fig. 4): flat, structured and unstructured. Ideally, the case-base should be effective and performant enough, so that the retrieving of cases and searching for similarities functions in the most optimal way. A flat case-base contains individual cases, which have no interconnections between them and is usually used when the number of cases is not too large. Structured case-bases organize the cases in some form of relationship - hierarchical, graph, etc.

Unstructured case-base organization includes cases, represented by some form of unstructured information, such as text or images [1][28][45][53].

2.4.3. Case-based Reasoning Life-cycle

The Case-based Reasoning process can be divided into four major steps, which are repeated every time we want to solve new problems. This can be seen in Fig. 5. The process starts by *retrieving* the most similar case from a case-base. It continues by *reusing* the retrieved case for solving the new problem. In the third step the solution from is *revised*, in order to determine its success and if it is still valuable. Finally, the experiences from the current problem are *retained*, in order to be available for the problems to come.

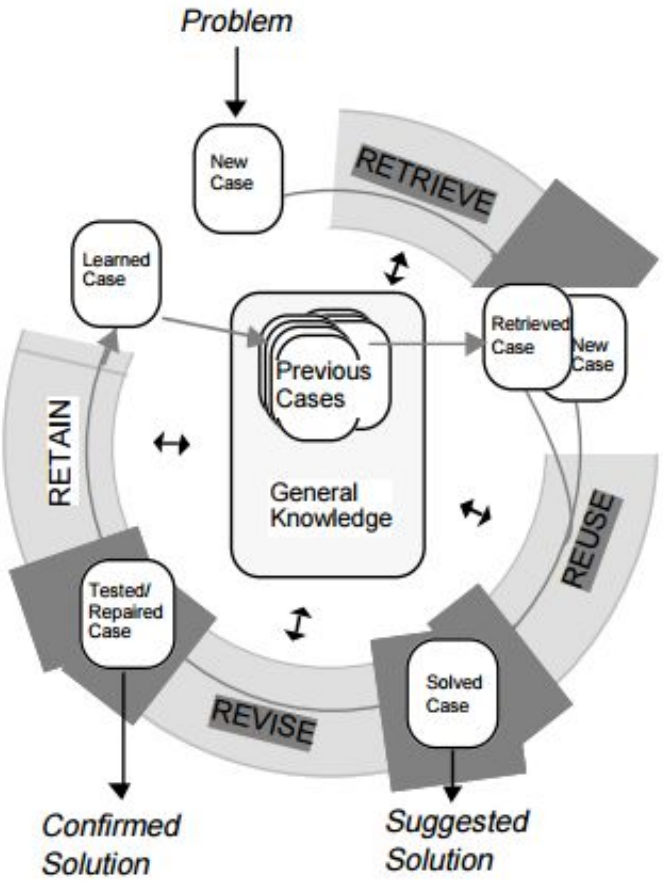

Fig. 5 The CBR cycle [1]

The CBR usually starts when a new problem occurs. The new problem is the input of the first phase: the **Retrieve Phase**. The new problem is described by different attributes, in order for it to be compared to the old ones and for similarities to be calculated. This attributes may be very comprehensive, but can also be limited, making the finding of similarities more difficult. At this point it is important not only to include attributes about the problem itself, but also

about its surrounding circumstances. Then comes the actual matching, which makes use of the attributes of each case (and the current problem). The comparison of the attributes includes not only measuring the similarity between them, but also deciding which attributes are more important for the current problem. Depending on the concrete CBR method, different similarity matching functions may be used. The case-base is then searched for the possible matches of the current problem and the most similar cases are identified. Finally the case, which is most similar to the current one, has to be chosen [1][28][45].

The second phase of the CBR cycle is the **Reuse Phase** (Kolodner [28] refers to it as the "proposing a ballpark solution" phase). In this phase a solution of the selected in previous (retrieve) phase case is used for the new problem. For this reason, attributes of the old case need to be compared to the attributes of the new case. Not all attributes will be relevant for the current situation, therefore only the relevant should be compared. The remaining ones may be used as contextual information. Most often problems do not match exactly, therefore their solutions also are not exactly the same and need to be adapted. Aamodt [1] differentiates between two different types of solution *reuse* - *transformational* and *derivational reuse*. In *transformational reuse* the solution from the old case is directly used (maybe after some adaptations). In *derivational reuse* on the other hand, the reasoner examines the process, the exact methodology of coming to the solution in the past and reuses the method to come up with brand new solutions. The phase ends, when a specific solution is proposed [1][28][45].

The goal of the **Revise Phase** is to analyze the proposed solution and at the end to confirm it. It should be checked, if the old solution can be applied to the new situation. Possible ways to determine that include opposing the chosen in the last step solution to others, that are additionally retrieved from the case-base (maybe the same set of similar solutions that were chosen in the previous step) [28]. Other options include doing a real-life test, where the solution from the past is applied in the current environment [1][45] or some kind of simulation, which is very often the preferred option (because of the financial aspects), but it does not give a real idea about the surrounding circumstances [1][28][45]. Additionally, possible environmental factors may be imagined, in order to check if the solution will work in different situations [28]. Afterwards the problems of the old solution and their causes have to be determined. After the errors are identified, they should be repaired and the solution needs to be altered, in order to ensure that the same errors will not happen again [1]. Furthermore,

possible chances, that have been missed in the past situation may be identified and applied [28].

The final phase of the CBR cycle is the **Retain Phase**, or the learning phase, where the new case, produced from the previous phase is added to the case-base. In this phase the reasoner should choose the attributes that will be saved, how to save them and how the case should be indexed. Depending on the actual CBR method, only a positive solutions are approved and added to the case-base, or faulty ones also. A new case is not necessarily created, but the old case may be updated, in order to incorporate the new circumstances and outcomes. Additional information that may be included in a case are information about the environment, the circumstances, explanation and justification of the solution, description of the methodology, which is used to come up with the new solution, etc.. All of this will be represented by concepts and attributes in our ontology. After deciding what information exactly and how it will be stored, the new case (or the old one, if it will be modified) must be indexed in some way, in order to simplify the retrieval in following CBR cycles. The goal of this phase is the continuous improvement of the CBR system [1][28][45].

2.4.4. Domain Knowledge

In order to constantly improve their Project Management, organizations can make use of Case-based Reasoning, which contributes to the retention of important information from different projects, which can then be used for making decisions during future projects. As already mentioned, CBR supports Project Management by retrieving similar to the current project cases from a case-base, which have to support the decision making. In the retrieving phase of the Case-based Reasoning, similar cases are retrieved based on some kind of similarity function. Those similarity functions may be very simple, but comprehensive domain knowledge can enrich them and make matching, based on semantic similarities between concepts, possible [9][12].

Generally the matching is done either syntactically or semantically. Syntactical matching usually makes sense in domain, where the domain knowledge is very limited. In the Project Management domain ontologies can be used for making the understanding of the matched cases and why are they similar, way more simple. The domain knowledge can also contribute while deciding on the importance of the single matched attributes [1].

Defining various attributes for each project, which will be then used for finding similar cases, can make the similarity identification very difficult, because of the huge amount of different possible attributes. An ontology on Project Management could limit the concepts of the Project Management and therefore the possible attributes, that are used to identify projects (and cases). This could lead to a way more simplified case matching in the future [12].

Furthermore, domain knowledge may be used in the retaining phase of the CBR process, where the new project has to be added as a new case to the case-base. Concepts from the domain knowledge may be chosen as attributes for describing the problem and the solution, which should again make the later retrieval of similar cases, based on those attributes, simpler and faster.

The goal of this thesis is to create an ontology on Project Management, which should include all of the important concepts of a project. This should contribute to bettering several phases of the CBR-cycle. Furthermore, the ontology should be easily extendable, in order to make it useful to other domains or in order to the current Project Management domain knowledge to be extended, if needed.

2.5. Optimization of Processes

In order for organizations to remain successful and improve customer satisfaction, they should constantly strive to improve the way they are doing their work, and therefore they need to constantly try to improve their processes. This improvement is usually done as part of a Business Process Management project. Optimization goals often include reducing the costs of making a product or service by automating some of the processes, reducing the workforce needed, reducing the amount of external resources, minimizing the time to market of products, minimizing the risks, and many others. For that reason, organizations often incorporate famous methodologies like Lean or Six Sigma. Lean, for example, strives at improving the processes within organizations and increasing the production by minimizing resource usage and waste. Six Sigma on the other hand tries to reduce defects and get constant, optimized results. In order to improve their processes, organizations must first identify what the problems are. Therefore, they need to thoroughly analyze their existing processes and learn from them, in order to be able to propose optimization actions. While analyzing the processes, depending on the concrete optimization goals, organizations need to determine some specific performance measures. This important measures are often referred to as Key Performance Indicators (KPIs) [36][48][63].

There exist various possibilities for optimizations within organizations. Process automation is an option for optimisation, in which some steps of the process (or the whole process) are automated. Removing bottlenecks and redundant work is critical for optimizing the processes within the organization. Those improvement options are covered within the BPM-projects. Nowadays the majority of organizations also make use of the technologies, in order to support their projects. IT technology can significantly simplify the management of projects, can contribute for standardizing the processes and allow the storage of huge amounts of data, which can be used in the future. The measurement and analysis of performance is also significantly simplified by some IT systems. This also gives organizations the opportunity to minimize the repeating of mistakes, that have happened before by learning from past experiences. This thesis will examine exactly one such option for optimization of projects - the Case-based Project Management. Furthermore, it will investigate how an ontology on Project Management can improve the CBR method.

Our ontology should improve the performance of the CBR methodology by presenting additional optimisations in each of the CBR life-cycle phases. During the Retrieve and Reuse Phases, ontologies can offer additional support by providing a set of predefined concepts and properties (attributes), which can be used, in order to limit and define the exact attributes, which will be always used for describing (old and new) cases. During the Revise Phase, our ontology may help by providing a set of predefined rules, which will prove a set of conclusions, related to the domain and therefore may simplify the identification of errors. Since the attributes and the rules that concern all the cases are present in our ontology, it will simplify the creation and storing new cases in the Retain Phase, or the learning phase.

2.5.1. Critical Success Factors in Project Management

In order to ensure constant project success, it is important to first define what the success factors of a project are. There exist many definitions for successful projects, but they mostly are concerned with achieving the project goals. Lim and Zain [34] define two categories of success factors: micro and macro. Micro viewpoint specifies project as successful, depending on the completion of the project by the achievement of the goals. Macro viewpoint also considers the completion, but also includes the customer satisfaction as an important factor when deciding if a project is successful. Pinto and Slevin [41] also state that the success factors also depend on the current phase of the project, where customer satisfaction is crucial at the end of the project, but at the beginning of the project, the factors

are more connected to the performance of the team, the budget and the spent time. Furthermore, Radujkovic and Sjekavica [43] divide the success factors of projects in three categories: competences of the people, characteristics of the organization and the following of Project Management principles and standards. We can also make a distinction between the success of a project and the success of the Project Management [2]. The success of the project itself is concerned with the accomplishment of the project goals and the Project Management seeks at ensuring the cost, time and quality effectiveness [5].

In order for organizations to achieve constant improvement of their projects, they first need to consider, which are the most common factors that lead to project success and also which are the most common reasons that projects fail. No matter of the categorization of the success/failure factors, the organization has to decide which are its own criteria for success and what is important for it to be achieved. For that purpose, when the organization starts a new project, it can make use of Case-based Reasoning, in order to compare the characteristics of the project with similar projects from the past and to identify possible factors that can lead to the success/failure of the new project. The planning of a project is undoubtedly a phase of the project life-cycle, which is extremely important to ensure continuous project improvement, but the closing phase of a project is also of great importance in that direction. In order to be able to learn from old projects during the planning phase of a project, the lessons learned from many old projects need to be documented (in the case of Case-based Reasoning, to be stored as cases). Therefore the closing phase is extremely important, because managers can evaluate their project and learn valuable lessons for what to do or not do the next time they face a similar situation [5][7][12].

The Project Manager has one of the most crucial roles within projects. He/she has to have strong leadership and communication competences and should be familiar and be following the Project Management principles. Furthermore, the knowledge and competences of the team, that will be working on the product/service is of extreme importance for the project success. Having clearly separated roles within the team is also extremely important. But even if the most competent and motivated team is present, the communication between the team members and also their constant communication with stakeholders is of crucial importance [5][7].

Realistic planning of the cost and time needed are also crucial for successfully carrying out a project. Very often team members are very confident in their competences, that they make

too optimistic estimations or even promise to deliver something that is not doable, or at least not in the predefined time/budget. Other frequently appearing reason for project failure can be the insufficient risk planning, not preparing for critical circumstances or experimenting with new technologies, in which the team is not experienced. Yet technology can be a strong ally when implementing Project Management standards and techniques, which can support the Project Manager in the organization and the team in the communication and carrying out the project [5][7].

The structures, defined within the organization can also have positive/negative impact on the outcomes of the project. The culture that the organization follows is also crucial for achieving project success. Furthermore, the goals of the project have to be explicitly determined and have to be aligned with the goals of the organization in general. The specific requirements have to be defined at the beginning and changes to those requirements must be as minimal as possible during the project execution, in order to avoid the so called “requirements creep” (or “change creep”). This is also one of the typical problems that may occur during the execution of BPM-projects, alongside with insufficient employee motivation for change and the not clearly defined project goal.

3. Ontology

This section will represent the practical part of the thesis. We will make a comprehensive description of the creation of the ontology and will describe the concepts that will be part of the ontology. We will also go into the reasons of choosing the exact languages and technologies that will be used. In the final part of the section, we will also go into detail about what the possible uses of the ontology for each phase of the CBR life-cycle.

3.1. Scenario

In order to create valuable information, that can be used in Case-based reasoning for improving the Project Management within organizations, we will define a set of conclusions that are derived from the domain theory of Project Management. Those conclusions are based on best practices, that have been established in the domain and have been practiced by many successful organizations. Following those best practises does not ensure the success of the project, but contributes to the higher possibility of success and reduces the percentage of failures. Our conclusions will also be based on typical mistakes, which can often contribute to project failure. Our conclusions, of course, do not cover all of the aspects of Project Management, but a limited number of them will be defined as part of this thesis, in order to demonstrate the effects of the ontology.

Conclusion Nr. 1: Projects, where communication is not sufficient, have higher chances to fail.

Good communication is crucial not only in projects, but also in every aspect of our lives. It helps build relationships (personal and professional) and ensures the better working environment and higher motivation of the employees. Furthermore, the increased trust between colleagues leads to improving their performance. Frequent enough and effective communication contributes to every stakeholder within a project to have a clear idea of the goal and the measures that are taken to achieve this goal and to always be on track with the current project situation (or at least with the parts of the project that concern him/her). Good communication between team members is usually easier to achieve, but it is also important that communication happens on all levels. The team should have enough opportunities for communication with the management. The management on the other hand should always

communicate the state of the project and the current problems and drawbacks with other stakeholders and ideally also with the users.

There are different types of communication that exist. Written communication includes communication through emails, chats, sms, etc. Verbal communication may include one to one conversations, telephone conversations, different types of meetings, where different people working on the project may be present. One way of ensuring the communication will happen through the life-cycle of the project is to introduce communication plan at the beginning. When different types of communications are planned early, it is more probable for them to really happen during the course of the project. This plan should include all types of communication, even the simplest meetings between two people.

Conclusion Nr. 2: Projects, where the Project Manager lacks the needed Project Management competences, have higher chances to fail.

The role of the Project Manager is the most crucial role within a project. He/she is the person who is responsible for carrying out the project to success and is also blamed, when the project fails. The Project Manager has to make sure that all of the phases of the project life-cycle go smoothly. In order to achieve a successful project, the Project Manager has to have plenty of competences, to know many tools and techniques. There are plenty of such competences, but some of the important ones include: leadership competences, communication competences, planning competences, risk management competences, negotiation competences, time management competences, etc.

Since the Project Manager is leading the project, he/she of course has to have great leadership competences, because he/she will be leading a (sometimes very big) team. Ideally, the Project Manager has to be able to make everyone in the team feel good with their work, which will further motivate them to give their best. But he/she has to ensure that the work goes as quick and as qualitative as possible. In order to ensure that the Project Manager has to possess great communication competences, which will help him/her to be on the same page with his team, but also the client to be always updated about the current situation and problems. Good planning competences are also especially crucial, because the planning part of the project is usually the most important one. Without a good planning, the projects are most likely doomed to fail. No matter how good the planning was, a good Project Manager always expects that not everything can go according to it. Problems

happen and the manager should be able to deal with them and to plan for their possible occurrence in advance. Since the Project Manager is the person who actively participates in the decision making about time, budget, success criteria, etc., he/she has to have strong negotiation competences, in order to negotiate the best for his/her organization and team.

Conclusion Nr. 3: Projects, where documentation is lacking, have higher chances to fail.

Project documentation is a crucial deliverable of each project and may consist of different documents (such as Project Plan, Requirements Plan, Project Charter, Work Breakdown Structure (WBS), Communication Plan, Lessons Learned, etc.), but also other artefacts such as emails, phone records, fax records, etc.. The importance of each documentation item depends on the organization itself, on the type of project, if it likely that similar projects will be done by the organization, the level of accountability needed, etc. It is typical for people to simply forget things and documentation can help to always have the option to revise things, it is a great tool for keeping track of the done work, of the problems that may arise.

If the project fails, the documentation can be revised and used for figuring out what the reason for the failure was. Furthermore all of the documentation can be used for lessons learned for when doing similar projects in the future. Even the best and most successful projects need their documentation, in order to prove it. No matter that the project was successful and the client is satisfied with the outcome, it can look very bad for the Project Manager and his/her competence, if the project is not well documented. Project Managers often create a broad set of templates, which can be adjusted to fit to new projects, which makes keeping the required documentation quite easier.

Conclusion Nr. 4: Projects, where unrealistic deadlines are set, have higher chances to fail.

While planning a project, it is crucial to clearly specify the scope, cost and time (deadlines) for the project and each phase/aspect of it. If some of those are poorly defined (or not defined at all), it is usually a recipe for disaster. Considering the deadlines, a good project plan should have realistic timeline, where each milestone is clearly defined, and this plan should be followed throughout the whole project. Clearly specifying the final delivery date of the product/service is also extremely important, in order to prevent possible arguments with

the clients in the later project phases. Therefore, realistic deadlines, which are consolidated with the people, who are actually working on a project, are of huge importance. For example, if the project is a project for creating a software, then the Project Manager has to discuss and agree on the deadlines with the programmers team.

While planning the deadlines for a project, the duration of each task must be planned in advance. Depending on the complexity of a task, different durations are specified. Generally speaking, overestimating the duration of a task is always better than underestimating it. In that way, the team will always have some buffer for unexpected situations and risks. Although some managers believe that setting shorter deadlines for tasks can motivate their team to work in a more efficient way, that can often lead to stressful situations and decrease the quality of the products. Therefore the manager have to always strive to find the perfect median of the two. If some time is left, the team can always use it to improve the quality or to invest it in additional testing.

Conclusion Nr. 5: Projects, where the scope is not clearly defined, have higher chances to fail.

One of the most important things (and maybe the first thing) a Project Manager has to do is to define the scope of the project. He/she should set its goals, budget and deadlines right at the beginning and they should be clear for every stakeholder. The problem is that no project is developed in an environment, which is not dependent on outer factors. Therefore the scope of projects often can change during the course of the project, the requirements can grow and therefore the costs and time. For that reason, Project Managers have to do their best, so the scope can be clearly defined from the beginning and agreed with the client. The goals should be discussed in detail and defined in the project plan.

Furthermore, Project Managers have to try to prevent the constant deviations from the original scope as much as possible. While amendments from the original plan are very common in projects (depending on their type), it is important to have every change documented. At the beginning the agreed scope must be documented and signed by the client and also each new change should also be documented somehow and the timeline and budget have to be changed accordingly.

3.2. Ontology Description

This subsection provides a comprehensive description of the ontology with all of the contained classes and their interconnections and properties. Each class (or concept) will be described with a short explanation of its meaning. The ontology will have the primary goal to support the Case-based Reasoning in the Project Management domain.

3.2.1. Domain and Scope

The ontology will represent the domain of Project Management and a variety of concepts around the topic of projects. The concepts will spread around different phases of the Project Management process. It will ease the understanding of different concepts and will represent their correlations via simple hierarchical structure and variety of properties. It will serve for classification purposes for different concepts around the topic of projects. The described ontology includes general concepts of the Project Management domain and therefore may be extended as needed in a variety of subdomains, in order to cover more specific areas of the domain, including some specific types of projects. For example, the ontology is extended to cover BPM-projects by including concepts from the Business Process Management domain.

3.2.2. Ontology Concepts

In order to ensure the correctness of our concepts, all of the included concepts are taken from trusted sources within the domains. Since the PMBOK (Project Management Body of Knowledge) [42] is one of the most reliable sources of information for Project Management and serves as a Project Management Standard, we have used concepts, presented there. We have taken the most important concepts of Project Management, included in Chapter 1. (“Introduction”) of the PMBOK [42], such as 'Project', 'Project Management', 'Project Manager', etc.. Furthermore, we have taken Chapter 2 (“Project life cycle and organization”) into account when creating concepts about the life-cycle of a project. Going into more detail, we have taken some important concepts for each of the phases, inspired by Chapter 3 (“Project Management Process for a Project”), including people involved in a project, important resources and outcomes, etc.. Further concepts are taken from the next chapters, including concepts concerning inputs, outputs, cost, schedules, etc.. The ontology can be further extended, in order to fully cover the standard.

In order to cover more specific types of projects, the BPM-projects, we have extended our basic Project Management concepts into the domain of Business Process Management. Our main reference for those concepts is the book of Dumas - " *Fundamentals of Business Process Management"* [13]. There he covers the most important concepts, which describe what a business process is, what the main ingredients of such a process are and what are the phases of the BPM-projects life-cycle. All of the ontology concepts can be seen in Fig. 6.

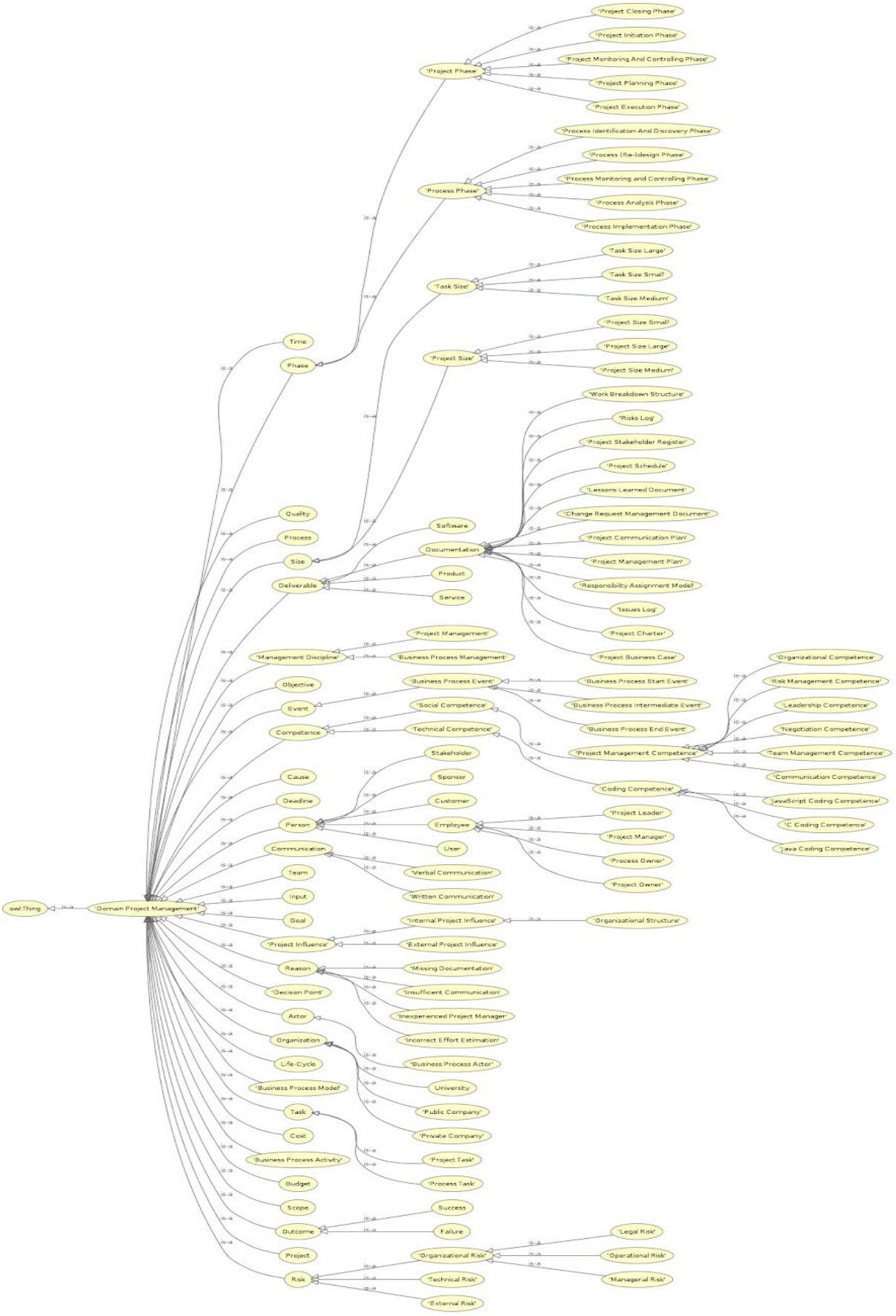

Fig. 6 Ontology

Concept:'Thing'

The 'Thing' concept will be the core concept of the ontology. This will be the parent of all other concepts. It has no direct effect of the current ontology, but is a standard practice while creating ontologies on any domain, in order to ease the eventual integration of the ontology into other ontologies of different domains.

Concept: 'Project'

The concept of 'Project' is a main concept for the current ontology on the domain of Project Management. Since the goal of the ontology will be to support Case-based reasoning on Project Management, this is a core concept, which has to be included in the hierarchy. The 'Project' concept will be crucial while writing the rules of the ontology and deriving the conclusions from the previous subsection.

Concept: 'Management Discipline'

The concept will represent an area of management knowledge and practices. Further subdivision will differentiate between the following disciplines:

'Management Discipline'

- 'Project Management'
- 'Business Process Management'

Concept: 'Size'

We have defined the 'Size' concept which will be used in a more general way, in order to represent the dimension of something. Additionally we have defined the concepts 'Project Size' and 'Task Size' in order to be able to measure the size of a project/task. The size will depend on the duration of the project, but the ontology may also be extended, so that the size of the project is also defined by other things, such as budget, complexity, risk level, etc.. The division that we make is as follows: 'Project Size Small' will represent any project size, where the project has a duration of no more than 12 months (or one year). 'Project Size Medium' will represent all projects with duration between 13 and 36 months (or between 1 and 3 years). 'Project Size Large' will represent all projects with a duration of more than 36

months (or more than three years). Similarly we have defined the size of tasks, where we again use the small, medium and large division. The reason behind the division into 'Project Size' and 'Task Size' is that the size of different things (projects, tasks, processes, products, etc.) may not only be divided in those three categories.

Concept: 'Communication'

The concept will represent all types of information that may occur during the course of a project. The communication includes verbal communication (one-to-one conversations, meetings, events, etc.) or written communication (emails, faxes, sms-s, etc.). In our ontology we will define the number of communication occurrences, which will contribute for deciding if the project lacks the needed communication, which can contribute to the project failure. This will be defined using a data property: 'has communication occurrences'.

Concept: 'Outcome'

The concept represents the two possible outcomes of a project/process/etc.: success and failure.

Concept: 'Person'

The concept represents people in general. The concept is used in different domains (not only the Project Management domain) and usually has the same meaning. In our ontology this concept will be from primary importance, since it will represent all of the individuals, working on a project and having effects on its outcome. The subdivision of the class will be more specific to the Project Management domain.

'Person'
- 'Customer'
- User
- 'Sponsor'
- 'Stakeholder'
- 'Employee'
  - 'Project Manager'
  - 'Project Leader'

'Project Owner'

'Process Owner'

The subclasses of the 'Person' hierarchy are not disjoint, meaning that one individual can belong to more of those classes. 'Employee' will represent people employed by an organization. As subclasses, we will define 'Project Manager', 'Project Owner', 'Project Leader', 'Process Owner', but the organization can extend them according to the different people working in it.

Concept: 'Competence'

The concept of 'Competence' will represent the ability of a person to do something to some extent. His/her expertise in the specific competence will be characterized by some competence level. The competences, abilities and experience of the project team members has huge importance on the run of the project. In some cases can even lead to the failure of the project (for example when the Project Manager is lacking the needed Project Management competences). Possible classification of competences is the following:

'Competence'

'Social Competence'

'Technical Competence'

Of course the types of competences may be extended. We will further define different project management competences for our purposes, including:

'Social Competence'

'Project Management Competence'

'Communication Competence'

'Leadership Competence'

'Negotiation Competence'

'Organizational Competence'

'RiskManagement Competence'

'Team Management Competence'

A good Project Manager has to possess all of the sub competences of the 'Project Management Competence' in sufficient extend. Furthermore example technical competences may include:

'Technical Competence'
    'Coding Competence'
        'Java Coding Competence'
        'C Coding Competence'
        'JavaScript Coding Competence'

Concept: 'Team'

The 'Team' concept represents a group of people, who have collective objective for the organization and have a specific set of common responsibilities. Teams are formed by a group of 'Person' individuals. A person can belong to one or many different teams.

Concept: 'Process'

The concept represents processes of all kinds. The 'Process' concept extents the domain of Project Management into the domain of Business Process Management. In a general sense, processes may be independent, but they can also be part of a project. A process is created from a series of tasks, which come in particular order and depend on each other. Since the work in a project is performed by different teams, a problem with some of those can lead to problems of the project as a whole.

Concept: 'Task'

The concept represents an activity, or a piece of work (a single unit of work) that is done as part of a project or a process. A task has specific duration and its completion has to contribute to the project goal in general. Tasks can be very small pieces of work with little to no effect on the project outcome, but can also be pretty important for the final goal of the project, which makes the concept extremely important. This division is made by defining the 'has task size' property, which has a 'Task Size' object as a domain. Task with small size will be fairly simple units of work, which can be done in less than 8 hours. A task of medium size is a work that can be done in 9 to 40 hours (or 2 to 5 working days). And a complicated task

that will take more than 40 hours (or roughly more than a week) will be defined a task with large size.

Concept: 'Deliverable'

The concept of 'Deliverable' represents everything that can be derived as a result of a project. Deliverables are what the customer expects to get after a project is finished, meaning that the missing of some deliverable will mean that the project is not successful. The following subclasses are defined:

'Deliverable'
- 'Product'
- 'Service'
- 'Documentation'
- 'Software'

'Product' is everything material that comes as a result of a project. 'Services' are immaterial and are performed by some person or machine. 'Document' represents documents in textual, visual, etc. form that are created during the project and can be submitted to the customer or used for internal purposes. 'Software' is a specific type of deliverable, which represents some type of software program, tool, etc.

Concept: 'Organization'

The concept represents a company, institution, association, etc.where people work (or volunteer). An 'Organization' usually has a specific industry (or several), in which it is specified and a clear purpose. Different 'Person' individuals can work for an 'Organization' and therefore also belong to the 'Employee' class. In our ontology we have divided the subclasses of organizations into: 'Private Company', 'Public Company', 'University'.

Concept: 'Budget'

The 'Budget' concept represents the full monetary cost of a project (or part of it), which includes all different types of costs and has a selected time span. The purpose of a budget is not only to decide how much the project will cost, but also for what the money will be used

for. The proper planning of the project budget is of extreme importance for the success of a project. A project, which delivered the product at the end, but needed twice the money of the planned budget, may as well be considered as failed project (especially for companies, who do not have unlimited funds).

Concept: 'Cost'

The 'Cost' concept represents the full amount of money that a project (or some smaller amount of work) will cost for the company. It is important for successful projects that the costs are within the planned budget and that the company can actually afford it.

Concept: 'Deadline'

The concept represents the final point in time to which some task/project/process/phase etc. has to be completed. For some projects (processes, tasks, etc.) is of extreme importance, that it is finished in the predefined deadline.

Concept: 'Goal'

The 'Goal' concept represents the final result that needs to be reached. The goals usually include specific product/service that needs to be produced, at a specific time and within a specific budget. The term is of extreme importance, since achieving the goals with the desired performance plays a huge role while deciding if the project was indeed successful.

Concept: 'Input'

The term represents anything (resource, product, license, service, etc.) that is needed for starting a specific project, process, task, etc.. Clearly specifying the inputs of a project/process, etc. is crucial for the successful completion.

Concept: 'Life-Cycle'

The 'Life-Cycle' concept represents the sequence of different phase of a project or a process. The smooth flow of each individual phase of the life cycle contributes to the success of the project in general.

Concept: 'Phase'

The life cycle of a project/process consists of different phases. The individual phases have different importance for the project success. For example, if during the planning phase everything is not clearly planned, this can lead to problems during the implementation phase. Therefore we have defined 'Project Phase' and 'Process Phase' (and their corresponding phases) as sub concepts, in order to better represent the reality.

Concept: 'Objective'

The 'Objective' class consists of all small products and concrete deliverables of the project/process. If the objectives are actually delivered will be main factor while deciding if the project is successful. If the objectives are missing (or not in the same form, amount, etc. as expected), the project will be considered as failed.

Concept: 'Project Influence'

'Project Influence' concept refers to external or internal factors that have some effect on the outcome of the project (or a part of it, or even a small task that is part of the project). Such factors should not be neglected, since they can sometimes have huge consequences on the project outcome, although they may be not even directly connected to the project.

Concept: 'Quality'

The concept represents the extent to which the product or service, produced by the project is corresponding to its objective. The goal of each company is (or at least should be) to produce products or services of great quality, in order for their customers to be satisfied.

Concept: 'Result'

The Result concept represents all consequences that the product or service (or any deliverable of a project) brings to the customers (or anyone affected by it). The results have to be positive, in order for the project to be considered as successful.

Concept: 'Risk'

The concept will represent everything that may have negative effects or cause problems to the project. Managing risks and being able to react adequately if some problems happen is crucial for successful projects, because usually there aren't projects that go exactly as expected. We will furthermore define the following types of risks (further risks may exist):

'Risk'
- 'External Risk'
- 'Technical Risk'
- 'Organizational Risk'
  - 'Legal Risk'
  - 'Managerial Risk'
  - 'Operational Risk'

Further types of risks may be defined to correspond to the most common risks that can happen depending on the organization itself.

Concept: 'Scope'

The term combines different goals, budget, deliverables, deadlines, etc. which should be achieved. The clear definition of the scope during planning and everyone being clear what this scope is, is crucial for having a successful project.

Concept: 'Time'

The concept of 'Time' represents the elapsed time of completing a task/project/process/phase, etc.

Concept: 'Actor'

The concept will represent someone (or something) that is taking part in some project/process/etc. Since an actor is not necessarily a person, we will define this separate concept. Therefore we have made the 'Actor' and 'Person' concepts not distinct (because some individual can be a person and an actor at the same time).

Concept: 'Business Process Activity'

The term represents a series of steps which are part of a business process. These steps usually have a specific order.

Concept: 'Business Process Model'

The concept represents a (visual or textual) representation of the actors, involved in a process, the steps that they perform and the decision logic and the communication and interaction between them.

Concept: 'Cause'

The term 'Cause' represents a reason for something to happen. In our case it often will be used for cause of specific Business Process action to begin.

Concept: 'Decision Point'

The concept will represent a point during the execution of a process/project/etc. when some (or more) decision(s) is(are) made. Decision points determine how the flow goes and the differences between different run instances. The decision points may differ in a way that a different number of alternatives can be picked.

Concept: 'Event'

The concept represents things that happen. Events happen simultaneously and have no duration. We will often use this concept in the Business Process domain, where events are part of processes.

3.3. Implementation

After determining the domain and scope of the ontology and defining some concepts, which will be contained in the ontology, the next step would be the selection of the ontology language, which will be used. Since the Web Ontology Language (OWL) is the World Wide Web Consortium's (W3C) recommended language for building ontologies (since 2009) this

will be our language of choice. The newest version is the OWL 2 Web Ontology Language. OWL is a markup language. OWL is a XML-based language, which is strongly influenced by the DARPA Agent Markup Language (DAML) + OIL language, which introduces additional formal semantics to RDFS, by making use of description logic [55].

The syntax that will be used will be the OWL/XML syntax, since it can conform to XML schema and can be easily parsed. Furthermore, the syntax is easy to read and understand. Possible downside of the chosen syntax can be the size of the created owl files, which can grow as the ontology grows.

Our ontology editor of choice will be the Protégé tool. The advantages of the selected tool include the user friendly interface. The tool is developed in Java and is open-source tool. It supports the newest standards and provides a variety of plugins, many of which come directly with the tool. Since it is widely used, support and information is available online.

In order to check the validity and consistency of the ontology and to derive logical conclusions from the explicit rules that will be defined using the rule language, we will make use of a reasoner. During the creation of our ontology our reasoner of choice will be the Pellet reasoner. Pellet is an open-source reasoner, which was written in Java and was the primary reasoner which supports OWL description logic and the newest version of OWL - the OWL 2 language. Another reason for choosing this particular reasoner was that it supports rules (SWRL), which will be used for extending the domain theory. Furthermore, Pellet provides reasons for inconsistencies, which make it easier to fix those inconsistencies. Pellet comes as a Protege plugin, which can be easily downloaded and installed.

Our rule language of choice will be the Semantic Web Rule Language (SWRL), which will be used for including additional rules, which will contribute to further enrichment of the ontology hierarchy and will provide additional opportunity for deriving conclusions and making recommendations based on the attributes of individuals. SWRL was created as a combination of OWL Lite (subset language of OWL) and OWL DL, providing additional axioms. The SWRL rules are easy to write and understand. They are formed by a body, called antecedent and a head, a consequent. Basically, if the body part is true, then the head part is derived and also true.

body (antecedent) -> head (consequent)

Protege has additional SWRL tab, which provides a convenient user interface for writing SWRL rules.

Since we are using SWRL as our rule language, we will make use of the Semantic Query-Enhanced Web Rule Language (SQWRL) as our query language. SQWRL is constructed as an addition to SWRL and has a very SQL similar syntax, which makes obtaining additional data from ontologies very convenient. Furthermore, the language contains two types of operators: based on SWRL rules (core operators) and advanced operators (collection operators), including additional and more advanced functionalities, which SWRL lacks, such as aggregation, negation, etc.. The SWRL conclusions can be enhansed by SQWRL queries, which can be used for deriving additional information.

In order to simplify the distinction between the concepts of our Project Management ontology and other concepts with the same identifier from other ontologies, we have created a prefix, which will be used in every mention, rule, etc. where the corresponding concept/property/etc. is used. Our prefix will be ‘pm’ (short for Project Management) and will be used in the following way: pm:<concept identifier>, pm:<property identifier>/etc..

3.4. The Use of the Ontology for CBR

An ontology can be useful in every phase of the Case-base reasoning life-cycle. Our ontology will present additional opportunities while building a case-base, storing and representing cases, while comparing cases and searching for similar cases within the case-base. It will contribute to adapting existing cases to fit the problem more adequately. The following subsections will describe the role of the ontology in the different CBR phases.

3.4.1. Use of the Ontology for Case Representation

Case-representation is not really a commonly mentioned phase in the CBR cycle, but in order for CBR to be effective and achieve the desired results, a solid case-base must be present. After applying the approach for longer, that becomes more and more easier, because with (usually) each new problem, a new case is added to the case-base and therefore the case-base grows with each new case. But when an organization starts out with Case-based reasoning, it will be harder for it to get most of it without having a good case-base to begin with, containing well structured cases with the most important attributes

stored. For that reason our ontology can be extremely useful for storing the cases as a case-base. Our ontology contains important domain knowledge, which will serve as a guideline for the information that will be stored for each case. Since a lot of information will be already in the ontology (structured in the desired way), that will ease the organization by minimizing the possibilities for information acquisition bottleneck, additionally making the CBR method Knowledge Intensive.

How each case looks (what information is stored in it and how is this information represented) is crucial for the efficiency of the reasoning. Usually the way each case is represented while performing CBR is strongly dependent on the domain and the information itself and the reason for the CBR. Our Project Management domain gives the opportunity for storing each project as a case, extending the case-base with each new project that an organization does. Using the OWL 2 language, each case will be stored as an (project) individual of the ontology. By making use of the data and object properties, specified in the ontology, important features of each case will be described. In that way cases can be extended with additional domain information, which will make the reasoning not only based on syntax, but also will allow the semantic reasoning. This will contribute to the additional Case-base flexibility by allowing reasoning, which is based not exclusively on a specific set of properties [14].

3.4.2. Use of the Ontology for Case Retrieval

In the retrieval part of the Case-based reasoning, new problems occur and must be represented as new cases. It is extremely important to choose the right attributes (or features) of the new case, in order to describe it properly. Our ontology provides the opportunity to describe cases using a variety of attributes, making the later similarity search easier. Each case (project) will contain various simple data properties (with values of simple types as numbers, strings, dates, etc.). This is a minimal condition for thoroughly describing cases. Via the object properties, our ontology will provide additional options for attributes which are not limited to only simple name-value properties, but can also be other types (other individuals). By using classes, that are part of the ontology, this will contribute to making the description of the separate cases similar (containing the same attributes), which will improve the later similarity functions.

While searching for similar cases, not only the attributes of the case are important, but the ontology provides additional information concerning the surrounding environment. Those

additional knowledge will contribute for better similarity functions. Our ontology also provides rules language, which we will use for defining additional rules, which will even more enrich the domain knowledge, and therefore the similarity functions. Furthermore, it provides a query language, which can be used for retrieving cases with specific properties. Making use of such queries can help the retrieval of similar cases. The similarity functions may be based on classes of the ontology and on attributes of those classes.

3.4.3. Use of the Ontology for Case Reuse

After a set of possible solutions (cases) is selected in the retrieval phase, the most similar one of them must be selected and proposed as a possible solution to the new problem. This comparison is made, based on the attributes of each case. Since all cases stored in our ontology will have similar attributes and will follow the same structure of describing the cases, this will ease the reuse part of the CBR method. By making use of our ontology, we can also use rules to define the most important attributes of the selected cases. This is important, since not all attributes have the same weight while comparing cases.

3.4.4. Use of the Ontology for Case Revising

The purpose of the REVISE step is to determine if a solution is suitable for the current problem. In order to do that, our ontology will contribute to identifying possible errors in cases, so that those errors can be avoided in the future. This will contribute to making better decisions next time a similar problem occurs. In order to achieve that, we have identified the conclusions in section 3.1., which represent typical problems while doing projects. In our ontology we've defined some SWRL rules, which will help us identify cases (projects), where the problems, defined in our conclusions, occur. We have taken a few such conclusions as an example, but further ones have to be defined by the organization itself, while implementing the method. In our rules the identifier of the concepts/properties is used, but we have also defined a label, which will allow us to see every concept/property in its natural language variant while using the ontology.

Deriving conclusion Nr. 1: Projects, where communication is not sufficient, have higher chances to fail.

So how much communication is enough for a project? This question is difficult to answer in general. The amount of communication needed depends strongly on the size of the project. Of course that a huge project, which continues for 5 years will need more communication

occurrences than a really small and short-timed project. As part of this thesis, in order to measure the communication that happens during a project in a simple manner, we will simply count the communication occurrences that happen during a project. Projects will be divided in three different sizes: small, medium and large projects. We consider small projects to be projects that have duration less than a year (or less than or equal to 12 months). Medium-sized projects will have duration between 12 and 36 months (between one and three years). Every project that has duration above 36 months will be considered a large project. This sizes will be defined by the organization itself and will depend on its own understanding of the project size, but we will take the current values as an example. Using the SWRL language, we will define the following rules, that will describe the three types of project sizes:

Rule “R01 Small Project”:
pm:Project(?p) ^ pm:hasProjectDuration(?p, ?d) ^ swrlb:lessThanOrEqual(?d, 12) ^ pm:hasProjectSize(?p, ?s) -> pm:ProjectSizeSmall(?s)

Rule “R02 Medium Project”:
pm:Project(?p) ^ pm:hasProjectDuration(?p, ?d) ^ swrlb:greaterThan(?d, 12) ^ swrlb:lessThanOrEqual(?d, 36) ^ pm:hasProjectSize(?p, ?s) -> pm:ProjectSizeMedium(?s)

Rule “R03 Large Project”:
pm:Project(?p) ^ pm:hasProjectDuration(?p, ?d) ^ swrlb:greaterThan(?d, 36) ^ pm:hasProjectSize(?p, ?s) -> pm:ProjectSizeLarge(?s)

The Rule “R01 Small Project” explained: If we have a project p and this project has a duration d and d is less than 12 months (or less than a year), and the project p has size s, then s will be a size of type 'Small'. The explanation of the next two rules follows the same logic.

After we’ve defined the size of the project, we will define the communication occurrences using a data property. Our property will be called 'has communication occurrences' (identifier: pm:hasCommunicationOccurrences) and will have an object of type 'Project' as a Domain and an xsd:int as a Range. The int type is not defined by our ontology and therefore does not have the 'pm' prefix. By making use of annotations, we will define an rdfs:comment annotation, which will serve as a description of the property. This will serve as an

explanation to everyone, who is not fully familiar with the property. Depending on the project size and the communication occurrences, we will define the further rules, which will prove our conclusion and will show the corresponding reason while using the Pellet reasoner:

Rule “R01 Small Project Insufficient Communication”:
pm:Project(?p) ^ pm:hasProjectDuration(?p, ?d) ^ swrlb:lessThanOrEqual(?d, 12) ^ pm:hasCommunicationOccurences(?p, ?c) ^ swrlb:lessThan(?c, 5) ^ pm:hasProjectOutcome(?p, ?o) -> pm:OutcomeFailure(?o)

We have also defined rules in similar ways for our projects of medium and large size. The “R01 Small Project Insufficient Communication” rule states that any project p, which is a small size project (defined by our previous rules) and has communication occurrences count of c and c is less than 5, will have a higher chance to fail.

Deriving conclusion Nr. 2: Projects, where the Project Manager lacks the needed Project Management competences, have higher chances to fail.

In order to define that a person is a Project Manager of a specific project, we will define the 'manages' object property. It will have a 'Project Manager' as a Domain and 'Project' class as Range. Furthermore, we will define the following rule, which will mean that every person p, which manages some project x is to be considered a Project Manager.

Rule “R04 Project Manager”:
pm:Person(?p) ^ pm:manages(?p, ?x) ^ pm:Project(?x) -> pm:ProjectManager(?p)

Following our 'Project Management Competence' sub hierarchy, we will define the following Data Properties:

'has competence level'
    'has project management competence level'
        'has communication competence level'
        'has leadership competence level'

The 'has competence level' (and all of its descendents) will have a 'Person' as Domain and an integer between 1 and 5 as Range (xsd:int[>= "1"^^xsd:int , <= "5"^^xsd:int]). For

simplification, we will assume that there are 5 levels (1-5) of competence, what a person can poses. Furthermore, we will define the following rules:

Rule “R05 Project Manager Communication”:
pm:Person(?p) ^ pm:Project(?x) ^ pm:manages(?p, ?x) ^ pm:hasCommunicationCompetenceLevel(?p, ?m) ^ swrlb:lessThan(?m, 3) ^ pm:hasProjectOutcome(?x, ?o) -> pm:OutcomeFailure(?o)

Rule “R06 Project Management Leadership”:
pm:Person(?p) ^ pm:Project(?x) ^ pm:manages(?p, ?x) ^ pm:hasLeadershipCompetenceLevel(?p, ?m) ^ swrlb:lessThan(?m, 4) ^ pm:hasProjectOutcome(?x, ?o) -> pm:OutcomeFailure(?o)

Rule “R07 Project Management Competence”:
pm:Person(?p) ^ pm:Project(?x) ^ pm:manages(?p, ?x) ^ pm:hasProjectManagementCompetenceLevel(?p, ?m) ^ swrlb:lessThan(?m, 4) ^ pm:hasProjectOutcome(?x, ?o) -> pm:OutcomeFailure(?o)

The “R07 Project Management Competence” rule will mean that every project x, managed by a person p (project manager), where the person p has 'Project Management Competence' level less than 4 will have a higher chance to have a failure outcome. The “R05 Project Manager Communication” and “R06 Project Management Leadership” rules will go even further by stating that in order for a project to not have a high chance of failure, its project manager has to have communication competence level of more that 3 and leadership competence level of more than 4. This will imply that the leadership competences will be very important for our organization. The exact numbers can be specified by the organization itself.

Deriving conclusion Nr. 3: Projects, where documentation is lacking, have higher chances to fail.

Sufficient and high quality documentation plays a big role during each phase of a project. It is especially important for preserving the knowledge from one project and transferring and using that knowledge in following projects. Furthermore, it is important for accountability, for

the management to always know how the project goes, what everyone does, is everything going according to plan.

In order to represent the documentation artefacts with the help of our ontology, we create a class called 'Deliverable'. The documentation will be represented as a subclass (amongst others, such as 'Product', 'Service', 'Software'). Furthermore, we will define different types of important documents as subclasses of the 'Documentation' class:

'Deliverable'
    'Documentation'
        'Project Business Case'
        'Project Charter'
        'Project Communication Plan'
        'Project Management Plan'
        'Project Schedule'
        'Project Stakeholder Register'
        'Change Request Management Document'
        'Issues Log'
        'Lessons Learned Document'
        'Responsibility Assignment Model'
        'Risks Log'
        'Work Breakdown Structure'

Those are a few of the most essential documents, which should be delivered at the end (and during) a good planned and executed project. A 'Project Business Case' document is essential for the beginning of the project and the smooth further going of the project. We will take it as an example and will try to represent that knowledge for a project in our ontology.

In order to describe that a 'Project' is missing a 'Project Business Case', we will define the 'has no project business case' data property. It will take a 'Project' as a domain and will have a boolean value as a range. The person using the ontology will have to explicitly define that the particular 'Project' has no 'Project Business Case'. We will also define the following rule:

Rule “R10 Missing Project Business Case”:

pm:Project(?p) ^ pm:hasNoProjectBusinessCase(?p, true) ^ pm:hasProjectOutcome(?p, ?o) -> pm:OutcomeFailure(?o)

The rule states that each project p which has no business case document, will be more likely to have a failure outcome.

Deriving conclusion Nr. 4: Projects, where unrealistic deadlines are set, have higher chances to fail.

The planning of the deadlines of individual tasks are important for time, cost and effort estimation of the whole project. In order to express the tasks in terms of complexity and duration, we have divided them into three different subclasses: small, medium and large task. This division is done using the 'Size' concept and by defining an object property called 'has task size'. Small tasks are considered to take up to 8 hours or one person day (1 PD), which is often a metric of measuring effort within projects. Medium sized tasks take between 9 and 40 hours and big tasks are considered tasks that will take more than 40 hours (or more than one working person week). We will also define the data property 'has estimated task duration', which will have a 'Task' individual as a domain and an integer as a range. We will make use of this property to state that effort underestimation can lead to project failure:

Rule “R08 Incorrect Effort Estimation Large Task”:
pm:Project(?p) ^ pm:ProjectTask(?t) ^ pm:consistsOf(?p, ?t) ^ pm:hasTaskSize(?t, ?s) ^ pm:TaskSizeLarge(?s) ^ pm:hasEstimatedTaskDuration(?t, ?d) ^ swrlb:lessThanOrEqual(?d, 40) ^ pm:hasProjectOutcome(?p, ?o) -> pm:OutcomeFailure(?o)

Rule “R09 Incorrect Effort Estimation Medium Task”:
pm:Project(?p) ^ pm:ProjectTask(?t) ^ pm:consistsOf(?p, ?t) ^ pm:hasTaskSize(?t, ?s) ^ pm:TaskSizeMedium(?s) ^ pm:hasEstimatedTaskDuration(?t, ?d) ^ swrlb:lessThan(?d, 9) ^ pm:hasProjectOutcome(?p, ?o) -> pm:OutcomeFailure(?o)

Conclusion Nr. 5: Projects, where the scope is not clearly defined, have higher chances to fail.

Defining the project scope right at the beginning of a project is crucial for the further development of the project. It contributes to everyone being on the same page and

understanding the goals and purpose of the current project. In order to prove our statement, we will have to state that for a given project the scope is not defined or is poorly defined. Proving that the project scope definition is missing can be done in a similar way as proving that the project business case is missing, which is still not ideal. As for proving that the scope is clearly defined will be tricky. Estimating the quality and fullness of the project scope definition cannot be done via our ontology, except in some very faint way, depending on the judgement of some person, which cannot express the real life situation in 100% trusty way.

3.4.5. Use of the Ontology for Case Retaining

In the RETAIN step of the Case-based Reasoning, the new project has to be saved as a new case (or an old case must be updated accordingly). This step is of particular importance for constant improvement of the method, because with each new case, new information is saved, which can contribute for future projects. In this step, it is important to decide what information about the current project to keep. Our ontology will help by using its concepts and properties (data and object properties) as a guideline for the information that has to be stored and will be useful for future iterations of the method. If no new case is to be created, then an old case can be enhanced with new properties, which will increase the amount and quality of the stored information.

4. Evaluation

In this section we will evaluate the created ontology and its benefits for the Case-based Reasoning process and the Project Management in general. The ontology contains a variety of the most important concepts of the Project Management domain, including concepts representing different actors, properties and important ingredients of a project, amongst others. Furthermore, the ontology is extended, in order for it to cover more specific types of projects, such as Business Process Management projects. In terms of accuracy, all of the ontology concepts are inspired by definitions by the PMI (Project Management Institute), which guarantees their correctness. All of the concepts included in the ontology are of importance for the domain and for the CBR method, where the ontology will be applied. This means that the ontology is concise enough and does not include unnecessary concepts. The ontology can be freely extended by the organization that will implement Case-based Reasoning with its help, in order to better fit its culture, organizational structure and its way of working. Although its main purpose is to be used for supporting the Case-based Reasoning and improving the Project Management of organizations, it can be freely used for other purposes, which may make use of the Project Management domain knowledge, making the ontology adaptable for other uses. Using the rdfs:label annotation, a simple natural language name is given for every concept, which increases the readability of the ontology. All of the concepts (and properties) defined within the ontology are described using the rdfs:comment annotation, in order to make them understandable and unambiguous. Both the labels and comments of each concept are translated to german language (in addition to englush), in order to make the ontology useful for german speaking users.

In order to evaluate how beneficial the ontology is for the Case-based Reasoning (CBR), it is important to evaluate its impact on the separate phases of the CBR methodology. Although Case Representation is not part of the Case-based Reasoning life-cycle that we use in this thesis, it is important to mention, because the methodology would not be effective, without having a good case-base to work with. Our ontology provides option for storing cases (project experiences) as individuals within the ontology. During our implementation, we have created different individuals of various types: projects, people, organizations, etc. which are all present at one place and can be used for the next iterations of the CBR method. With each iteration even more information is stored, improving the process even more.

Since the ontology comprises the most important concepts of the domain, it improves the Case Retrieval step of the Case-based Reasoning process. While creating new cases, we have made use of those concepts and properties, which guaranties that exactly the same attributes will be present for all cases, which will make the similarity search simpler. In order to retrieve the most similar case during the Case Reuse phase, the similarity function will be based on those same attributes and concepts, which will ensure its improved accuracy. The CBR method will not have to deal with missing or renamed attributes, which would decrease the correctness and effectiveness of the similarity function.

In the Case Revise step of the Case-based Reasoning the solutions have to be evaluated, in order to find possible errors in previous cases. We have defined a set of conclusions, which include typical problems that can cause increased chance of project failure, and have used different SWRL rules, in order to find specific cases, where those problems occur. While this brings benefits for the Case-based Reasoning process, it also comes with some limitations. First of all, some failures can be hard to capture objectively. For example, in order to prove our conclusion Nr. 5 (“Projects, where the scope is not clearly defined, have higher chances to fail.”) the quality of the scope definition must be somehow defined. This is very hard to be done in a fully objective way, but rather depends on the judgement of a person. Further limitation was discovered while proving the conclusion Nr. 3 (“Projects, where documentation is lacking, have higher chances to fail. “). This limitation comes not from the ontology itself, but from the OWL language. In OWL, it is impossible to say that a project, where a document not defined as present, is likely to fail, because that will contradict the Open World Assumption. In the Open World Assumption, it is not enough that a statement is not defined, in order to conclude that the contrary statement is true. The only possible workaround is to explicitly state that the specific document is missing. During the Case Retain step of the Case-based Reasoning, a new case has to be created and added to the case base. Similarly to the Case-Retrieval step, our ontology will help with improving the selection of attributes that will be included in that case.

To sum up, our ontology can have benefits for all of the Case-based Reasoning phases in some way. A downside for the organization will be the additional effort needed for creating the ontology. So every organization has to decide if the benefits prevail over the resources that will be needed. This effort will be mostly done once, while creating an ontology (or adapting existing one, such as the one that we propose) and then can be useful for many projects in the future.

5. Conclusion

Project Management is a discipline that deals with organizing the work in the form of projects and applying different techniques, tools and methods, in order to optimize those projects. In order to improve the effectiveness of the Project Management practices, scientists started to explore different options for taking advantage of other disciplines, such as Knowledge Management or Machine Learning. In the context of this thesis we have explored how an ontology on Project Management can improve the Case-based Reasoning. The Case-based Reasoning is used in the context of Project Management, in order to enable organizations to use the experiences from old projects, which are stored as cases in a case-base, in order to improve their upcoming and ongoing projects. The field of Business Process Management deals with processes, which are repeated over and over again. Our ontology includes the most important concepts of the Project Management domain, which can be used for all types of projects, but it is also extended to the domain of Business Process Management, in order to also be able to cover BPM-projects in more detail.

After defining a series of concepts and the properties that represent the interconnections between those concepts, our ontology is extended with the creation of some rules, making use of the SWRL language. Those rules help us while using the ontology during the whole CBR life-cycle. We have determined that the ontology can be useful for all of those phases. Since it contains specific concepts and properties, it can be useful for the Case Retrieval phase of the CBR, in order to easily decide what attributes to use when searching for similar cases. While comparing cases during the Case Reuse phase, the CBR can also make use of ontologies for choosing the specific attributes to compare. The SWRL rules are useful during the Case Revise phase, where important conclusions can be checked and reasons for errors within projects can be identified. The ontology also simplifies the Case Retain phase, because the predefined concepts and properties can be again used while creating new cases to store in the case-base. In general, the ontology can also serve as case-base itself, because different cases can be stored as individuals within the ontology. Therefore we can conclude that an ontology domain can improve the Case-based Project Management and offers possibilities for simple extension by the corresponding organization which will use it, which will offer even more possibilities aligned with the organization’s structure and goals.

6. Literature

[1] Aamodt, A.,Plaza, E. (1994). *Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches*. AI Communications, Vol 7, Nr. 1.

[2] Almamlook, R., Frefer, A., Mahmoud, M., Haleema, H. (2018). *Overview Success Criteria and Critical Success Factors in Project Management*. Journal of Industrial Engineering and Management, Vol. 7

[3] Antoni, M., Forsberg, T., Kammerlind, P. (2000). *Process management in a project environment - the relation between different roles*. Project Management Institute Annual Seminars & Symposium, Houston, TX, Newtown Square, PA, Project Management Institute

[4] Atkinson, R. (1999). *Project Management: cost, time and quality, two best guesses and a phenomenon, it's time to accept other success criteria*. International journal of project management Vol. 17, No. 6, pp. 337-342

[5] Beleiu, I., Crisan, E., Nistor, R. (2015). *Main factors influencing project success*. Interdisciplinary Management Research (journal), Vol. 11 (2), pp 59-72

[6] Brocke, J., Mathiassen, L., Rosemann, M. (2014). *Business Process Management*. Business & Information Systems Engineering, Vol.6(4), pp. 189-189 [Peer Reviewed Journal]

[7] Cao, T., Swierczek, F. (2010). *Critical success factors in project management: Implication from Vietnam*. Asia Pacific Business Review. 16. pp. 567-589

[8] Chinvigai, Ch., Dafaoui, E.M., Mhamedi, A. (2007). *An approach for enhancing process and process interaction capability*. In Proceedings of the 19th International Conference on Production Research, Valparaiso, Chile, July 29-August 2, pp. 1-6

[9] Delany, S.J, Cunningham, P. (2000). *The Application of Case-Based Reasoning to Early Software Project Cost Estimation and Risk Assessment*. Trinity College Dublin, Department of Computer Science.

[10] Discenza, R. & Forman, J. B. (2007). *Seven causes of project failure: how to recognize them and how to initiate project recovery*. PMI Global Congress 2007—North America, Atlanta, GA., PA: Project Management Institute

[11] Dorn, J. (2019). *Managing Experience in Business Process Management*. Proceedings - 10th Conference Professional Knowledge Management, P Heisig (Hrg.)

[12] Dorn, J. (2016). *Sharing Project Experience through Case-based Reasoning*. Proceedings - 12th International Conference on Knowledge Management, ICKM 2016, Vienna, Austria

[13] Dumas, M.; La Rosa, M.; Mendling, J.; Reijers, H. A. (2013). *Fundamentals of Business Process Management*. 1st ed. Berlin: Springer

[14] El-Sappagh, S., Elmogy, M. (2015). *Case Based Reasoning: Case Representation Methodologies*. International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 6, No. 11

[15] Fenech, K., De Raffaele, C. (2013). *Overcoming ICT project failures - A practical perspective*. 2013 World Congress on Computer and Information Technology, WCCIT 2013

[16] Frappaolo, C. (2006). *Knowledge Management*. Capstone Publishing Ltd, Southern Gate Chichester, England

[17] Friedrich, R., Iglezakis, I., Klein, W., Pregizer, S. (2002). *Experience-based decision support for project management with Case-based Reasoning*. Proceedings of the 1st German Workshop on Experience Management, pp. 139–150

[18] Geneca LLC (2011). *Doomed from the Start? Why a Majority of Business and IT Teams Anticipate Their Software Development Projects Will Fail.* Industry Survey

[19] Gentner, D. (1983). *Structure mapping - a theoretical framework for analogy*. Cognitive Science (journal), Vol.7. pp.155- 170

[20] Grimm S., Abecker A., Völker J., Studer R. (2011). *Ontologies and the Semantic Web*. Appears in: Domingue J., Fensel D., Hendler J.A. (eds) *Handbook of Semantic Web Technologies*. Springer, Berlin, Heidelberg

[21] Gruber, T. R. (1993). *A Translation Approach to Portable Ontology Specifications*. Appears in: Knowledge Acquisition, 5(2):199-220

[22] Gruber, W. (2003). *Modeling and Transformation of Workflows with Temporal Constraints*. Klagenfurt, Univ., Diss., Berlin

[23] Horine, G. (2005). *Absolute Beginner's Guide to Project Management (Absolute Beginner's Guide)*. Que Corp., Indianapolis, IN, USA

[24] Jeston, J., Nelis, J. (2014). *Business Process Management. Practical guidelines to successful implementations*. 3rd ed., Published by: Routledge, New York, NY, USA

[25] Jurisica, I., Mylopoulos, J., Yu, E. (1999). *Using Ontologies for Knowledge Management: An Information Systems Perspective*. Proceedings: Annual Conference of the American Society for Information Science, Washington, D.C., November 1-4

[26] Ko, R.K.L., Lee, S.S.G, Lee, E.W. (2009). *Business process management (BPM) standards: A survey*. Business Process Management Journal, Vol. 15(5), pp. 744-791 [Peer Reviewed Journal]

[27] Kolodner, J. (1983). *Maintaining organization in a dynamic long-term memory*. Cognitive Science, Vol.7, Issue 4, pp.243- 280

[28] Kolodner, J. (1992). *An Introduction to Case-based Reasoning*. Appears in: Artificial Intelligence Review 6 (journal), pp. 3–34

[29] Kourounakis, N., Maraslis, A. (2016). *The PM² Project Management Methodology Guide – Open Edition*; v.0.9, The European Commission, DIGIT Centre of Excellence in Project Management (CoEPM²)

[30] Krogstie J. (2012). *Quality of Business Process Models*. In: Sandkuhl K., Seigerroth U., Stirna J. (eds) (2012). *The Practice of Enterprise Modeling*. PoEM 2012. Lecture Notes in Business Information Processing, vol 134. Springer, Berlin, Heidelberg

[31] Kwak, Y.H. (2005). *A brief History of Project Management*. Appears in: Carayannis, E.G. et al. *The story of managing projects*. (9 eds), Greenwood Publishing Group

[32] Lebowitz, M. (1983). *Generalization From Natural Language Text*. Cognitive Science, Vol.7, Issue 1, pp. 1-40

[33] Lee, W., Shah, N., Sundlas, K., Musen, M. (2008). *Comparison of Ontology-based Semantic-Similarity Measures*. AMIA Annual Symposium Proceedings, pp. 384–388

[34] Lim, C.S, Zain, M.. (1999). *Criteria of project success: An exploratory re-examination*. International Journal of Project Management. Vol. 17, No. 4. pp. 243-248

[35] Method123 (Company) (2009). *Project Management Guidebook*. Method123

[36] Meyer, L. (2006). *Business process optimization: combining project management and six sigma best practices to better understand and optimize critical business processes*. Paper presented at PMI® Global Congress 2006—North America, Seattle, WA. Newtown Square, PA: Project Management Institute

[37] Nonaka, I.. (1991). *The Knowledge-Creating Company*. Harvard Business Review (magazine)

[38] Object Management Group (2013). *Business Process Model and Notation*. BMI Standard Specification 2.0.2

[39] Owen, M., Raj, A. (2004). *BPMN and Business Process Management Introduction to the New Business Process Modeling Standard*. Popkin Software

[40] Petersen, Ch. (2013). *The Practical Guide to Project Management*. 1st ed., Published by BookBoon Publishers

[41] Pinto, J., Slevin, D. (1988). *Project success: definitions and measurement techniques*. Project management journal, Vol. 19, No. 3, pp. 67-73

[42] Project Management Institute (2008). *A guide to the project management body of knowledge: (PMBOK guide); an American National Standard ANSI/PMI 99-001-2008*; Newtown Square, Pa: PMI, Project Management Inst.

[43] Radujkovic, M., Sjekavica, M. (2017). *Project Management Success Factors*. Procedia Engineering. Vol. 196, pp. 607-615, Proceedings: Creative Construction Conference (CCC 2017), Primosten, Croatia

[44] Rosato, M. (2018). *Go Small for Project Success*. Vol. VII, Issue V, St. Joseph's College, Patchogue, NY, USA

[45] Richter M., Weber, R. (2013). *Case-Based Reasoning: A Textbook*. Springer Science & Business Media, Berlin Heidelberg

[46] Rosing, M.; Scheel, H.; Scheer, A.W. (2014). *The Complete Business Process Handbook: Body of Knowledge from Process Modeling to BPM*. 1st ed. London: Morgan Kaufmann

[47] Rumelhart, D. E. (1978). *Schemata: The building blocks of cognition*. In Spiro, R., Bruce, B. & Brewer, W.(Eds.). *Theoretical issues in reading comprehension.* (pp. 33–58). Hillsdale, NJ: Lawrence Erlbaum

[48] Rummler, G; Brache, A. (2012). *Improving Performance: How to Manage the White Space on the Organisation Chart.* 3rd ed. John Wiley & Sons

[49] Schank, R. (1980). *Language and Memory*. Cognitive Science, Vol.4, pp. 243-284

[50] Schank, R. (1983). *Dynamic Memory: A Theory of Reminding and Learning in Computers and People*. Cambridge University Press, New York, NY, USA

[51] Schank, R. (1999). *Thematic Organization Packets*. In: *Dynamic Memory Revisited* (book), pp. 137-154, Cambridge: Cambridge University Press, New York, NY, USA

[52] Sharp, A.; McDermott, P. (2008). *Workflow Modeling. Tools for Process Improvement and Applications Development*. 2nd ed. Norwood, MA

[53] Slade, S. (1991). *Case-based reasoning: A research paradigm*. AI Magazine, pp. 42-55

[54] Smith, A., Cannan, E. (1903). *An inquiry into the nature and causes of the wealth of nations*. edited by: Cannan, E., in 2 vols., originally published: 1776

[55] Smith, M., Horrocks, I., Krötzsch, M., Glimm, B., eds. W3C Recommendation (2012) *OWL 2 Web Ontology Language: Document Overview (Second Edition)*

[56] Sun Tzu, Hart, B. (Foreword), Griffith, S. (Translation) (1971). *The Art of War*. Oxford University Press

[57] Taylor, F. (1911). *The Principles of Scientific Management*. New York, NY, USA and London, UK, Harper & Brothers

[58] Tomkins, S. (1979). *Script Theory: Differential Magnification of Affects*. Nebraska Symposium On Motivation 1978. Ed. Richard A. Deinstbier. Lincoln, NE: University of Nebraska Press, 1979. 201–236

[59] Tulving, E. (1972). *Episodic and semantic memory*. In: Tulving, E., Donaldson, W. (eds.) (1972). *Organization of memory,* New York: Academic Press, pp. 381-403, Department of Psychology. Yale University, New Haven, CT

[60] Weske, M. (2012). *Business Process Management. Concepts. Languages. Architectures.* 2nd ed. Berlin, Heidelberg: Springer

[61] Westland, J. (2006). *The Project Management Life Cycle: A Complete Step-by-step Methodology for Initiation, Planning, Execution & Closing a Project Successfully*. Kogan Page

[62] Williams, S. (1967). *Business Process Modeling Improves Administrative Control. Automation*

[63] Williams, M. (2008). *The Principles of Project Management*, 1st ed., SitePoint Pty. Ltd., Australia

[64] Withoff, M. (2007). *Neue Ansätze und Methoden der Einführung von Lessons Learned Management im Projektmanagement*. Technische Universität Wien

[65] Wittgenstein, L. (1958). *Philosophical investigations*. (book) 2nd ed., first published: 1953, Translated by Anscombe, G.E.M., Blackwell Publishers Ltd., Oxford, UK, pp. 31-34

[66] Yeong, A.; Lim, T.T. (2010). *Integrating knowledge management with project management for project success*. Journal of Project, Program & Portfolio Management 1, pp. 8-19

7. Abbreviations

AI - Artificial Intelligence

AR - Automated Reasoning

BP - Business Process

BPEL - Business Process Execution Language

BPM - Business Process Management

BPMN - Business Process Model and Notation

BPMS - Business Process Management System

CBR - Case-based Reasoning

CS - Computer Science

FL - Formal Logic

ICT - Information and Communication Technologies

IT - Internet Technologies

KM - Knowledge Management

MOPs - Memory Organization Packets

PM - Project Management

PMBOK - Project Management Body of Knowledge

PMI - Project Management Institute

TOPs - Thematic Organization Packets

WfMS - Workflow Management System

WWW - World Wide Web