A Viewpoint-Based Case-Based Reasoning Approach Utilising an Enterprise Architecture Ontology for Experience Management

Andreas Martin[ab*], Sandro Emmenegger[a], Knut Hinkelmann[ac], Barbara Thönssen[a]

[a] University of Applied Sciences and Arts Northwestern Switzerland FHNW, School of Business, Olten, Switzerland; [b] University of South Africa, School of Computing, Florida Park, Roodepoort, Johannesburg, South Africa; [c] University of Pretoria, Department of Informatics, Pretoria, South Africa

The accessibility of project knowledge obtained from experiences is an important and crucial issue in enterprises. This information need about project knowledge can be different from one person to another depending on the different roles he or she has. Therefore, a new ontology-based case-based reasoning approach that utilises an enterprise ontology is introduced in this paper to improve the accessibility of this project knowledge. Utilising an enterprise ontology improves the case-based reasoning system through the systematic inclusion of enterprise-specific knowledge. This enterprise-specific knowledge is captured using the overall structure given by the enterprise ontology named ArchiMEO, which is a partial ontological realisation of the enterprise architecture framework ArchiMate. This ontological representation, containing historical cases and specific enterprise domain knowledge, is applied in a new ontology-based case-based reasoning approach. To support the different information needs of different stakeholders, this ontology-based case-based reasoning approach has been built in such a way that different views, viewpoints, concerns and stakeholders can be considered. This is realised using a case viewpoint model derived from the ISO/IEC/IEEE 42010 standard. The introduced approach was implemented as a demonstrator and evaluated using an application case that has been elicited from a business partner in the Swiss research project.

Keywords: case-based reasoning, enterprise ontology, enterprise architecture, ontology-based case-based reasoning, experience management, viewpoint

1. Introduction

Retrieving and maintaining existing knowledge and past experience is an important aspect for enterprises, especially for knowledge-intensive and agile activities that typically occur in project management. Powerful experience management systems enable enterprises to reuse experiential project knowledge, which can be crucial for project success. Case-based reasoning (CBR) can be an adequate method to retrieve experiential knowledge in an experience management system (Bergmann 2002). CBR can be adopted for experiential project knowledge, which can be captured in structural cases.

An initial analysis showed that the accessibility of experiential knowledge is absolutely crucial for a company. Furthermore, it was possible to observe that different people performing diverse roles in a company have different information needs. This requires

---

*Corresponding author. Email: andreas.martin@fhnw.ch

an approach that is able to present relevant information according to the varying concerns of different people. A CBR approach should therefore be able to offer a way to retrieve existing knowledge in consideration of the people's specific roles and concerns. Aha, Breslow, and Muñoz Avila (2001) and DAquin, Lieber, and Napoli (2005) mention that CBR systems should consider the different viewpoints of users. A viewpoint[1] "[...] provides a means to focus on particular aspects [...]" of a thing of interest or an information need, "[...] determined by the concerns of the stakeholders [...]" (Lankhorst 2009, p 149). DAquin, Lieber, and Napoli (2005) introduced a CBR approach to support oncologists, for example, providing different viewpoints on oncology. The motivation was the "[...] multi-disciplinary domain of oncology" (DAquin, Lieber, and Napoli 2005, p. 154). Project management is typically multi-disciplinary since several members with different expertise and roles are involved. Unfortunately, no significant attention has been paid to CBR concerning peoples' different viewpoints. *Therefore, a new CBR approach is needed that presents relevant information in the context of the specific concerns and roles of the employees working on projects.*

In CBR, existing knowledge (sometimes called background, contextual or domain knowledge) is used as vocabulary to describe cases (Richter 1998). "Vocabularies define the concepts and relationships (also referred to as 'terms') used to describe and represent an area of concern" (W3C 2012b). This vocabulary needs to be acquired in advance and often leads to a so-called knowledge acquisition bottleneck. To overcome this issue, several approaches suggest the utilisation of ontologies (Díaz-Agudo and González-Calero 2000; Recio-Garía and Díaz-Agudo 2007; Gao and Deng 2010). To provide a CBR system with *enterprise*-specific knowledge, so-called *enterprise ontologies* can be used (Martin, Emmenegger, and Wilke 2013). But enterprise ontologies need to be created first, resulting in a knowledge bottleneck as well. As a solution, Kang et al. (2010); Hinkelmann, Merelli, and Thönssen (2010); Thönssen (2010) and Martin, Emmenegger, and Wilke (2013) suggest reusing enterprise architecture descriptions when formalising an enterprise ontology. Because an enterprise architecture is a description of an "enterprise's organisational structure, business processes, information systems, and infrastructure" (Lankhorst 2009, p. 3), it is an excellent source of enterprise-specific knowledge to describe cases. If an enterprise omits creating and maintaining an enterprise architecture description, selected enterprise models (business process models, organisational charts, etc.) are a potential source of enterprise vocabulary and knowledge as well. This allows reusing of existing vocabulary without storing it redundantly, thus avoiding possible inconsistencies and additional maintenance effort. *This paper introduces a CBR approach for experiential project knowledge, taking into account the specific viewpoints of the users and their concerns.*

The paper is structured as follows: It starts with the literature review in section 2 about CBR, enterprise architecture, enterprise ontology and ontology-based CBR. This is followed by section 3, which describes the method and the research process. Section 4 introduces the integration of an enterprise ontology into a CBR approach, and section 5 shows our research contribution concerning view and viewpoint in CBR. Section 6 depicts an procedure model for this approach, and section 7 provides an evaluation of the implemented artefact based on a specific application case. The paper closes with a discussion in section 8 and a conclusion, including future work, in section 9.

---

[1]Originally, Lankhorst (2009, p. 149) used the term 'viewpoint' in the context of enterprise architectures. Here it is extended in a way that it can be used for CBR to satisfy an information need in accordance with the specifications of ISO/IEC/IEEE (2011); see chapter 5.

2.   Literature Review

In this section we introduce the main concepts starting with a brief overview on CBR, followed by related work concerning enterprise ontologies and enterprise architecture frameworks. Finally, the section ends with related work concerning ontology-based CBR.

2.1.   *Case-Based Reasoning*

According to Leake (1996, p. 2), CBR can be seen as "reasoning by remembering". It is a technology-independent methodology (Watson 1999) for humans and information systems. Kolodner (1993, p. 27) describes CBR in two ways: "Case-based reasoning is both [...] the ways people use cases to solve problems and the ways we can make machines use them". CBR is able to utilize the specific experience of previously solved, concrete problem situations (cases). A new problem is solved by finding a similar past case and reusing the solution in the new problem situation (Aamodt and Plaza 1994). *Problem* and *solution* are traditional CBR terminology.

Bergmann (2002) extends this terminology and view on cases in a way that a potential solution can be seen as a new lesson learned. This is called *lesson space* and contains additional knowledge objects such as links to documents that are used in the described situation. The lesson spaces "[...] contain information that is not the solution itself but useful to find a solution" (Bergmann 2002, p. 50). To describe the lesson, Bergmann (2002, p. 50) uses the term *characterisation space*, which extends the classical problem space with additional information, e.g., "derived descriptions or properties that were not present in the problem solving situations from which the experience emerges". The characterisation space "describes all facts about the experience that are relevant for deciding whether the experience can be reused in a certain situation"(Bergmann 2002, p. 50) and contains certain metadata to characterise the lesson. Apart from that, it "[...] works as a kind of index for the lessons part" (Bergmann 2002, p. 48).
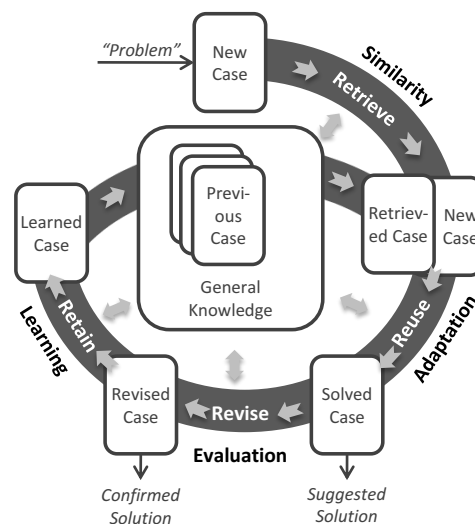

Figure 1. Case-based reasoning cycle (adapted from Aamodt and Plaza (1994)).

This work uses the CBR methodology as the basis for a CBR information system design. The general methodology can be explained using the CBR cycle of Aamodt and Plaza (1994). It shows elementary features of a CBR system and consists of the following steps (see figure 1) using the four Rs (retrieve, reuse, revise and retain):

(1) *Retrieve* the most similar case(s) from the case base, which contains historical cases, based on the characterisation of the current situation used as query, using a *similarity* mechanism.
(2) *Reuse* the lesson from the retrieved case(s) as the suggested solution for the new situation; adapt the retrieved lesson to the new situation , which becomes part of a new case.
(3) *Revise* the new case after evaluating it in the new situation.
(4) *Retain* useful experience for future reuse; store the new case in the knowledge base.

According to Bergmann et al. (2003), it is possible to distinguish three approaches:

(1) *Textual* CBR: Cases are recorded as or derived from free text. The work of Weber, Ashley, and Bruninghaus (2006) provide a good introduction into this.
(2) *Conversational* CBR: The case acquisition takes place in a conversational (dialogue) manner. The work of Aha, Breslow, and Muñoz Avila (2001) can be recommended as an introduction.
(3) *Structural* CBR: Cases are described using a certain vocabulary or domain model (see Bergmann 2002, p. 54). Some examples, among several others, are: Yokoyama (1990), who introduces an object-oriented approach, and Plaza (1995), who uses feature terms.

This work relies on structural CBR, which uses existing knowledge (sometimes called background, contextual or domain knowledge) as vocabulary to describe a case (Richter 1998). In the next section we introduce a possibility that can be used as a source for background knowledge.

2.2. *Enterprise Architecture*

Enterprise architecture descriptions can act as a source of background knowledge. Enterprise architectures contain relevant aspects of an enterpris–business structures, IT structures and their relationships (Ross, Weill, and Robertson 2006; Lankhorst 2009). Lankhorst (2009, p. 3) defines an enterprise architecture as "[...] a coherent whole of principles, methods and models that are used in the design and realisation of an enterprises organisational structure, business processes, information systems, and infrastructure". ISO/IEC/IEEE (2011) distinguishes between architecture descriptios, frameworks and description languages:

- Architecture description (AD): "An architecture description (AD) is an artifact that expresses an Architecture. Architects and other system stakeholders use Architecture Descriptions to understand, analyze and compare Architectures"(ISO/IEC/IEEE 2011). An AD is what is written down as a concrete work product. It could be a document, a model repository or a collection of artifacts.
- Architecture framework (AF): "An architecture framework establishes a common practice for creating, interpreting, analysing and using architecture descriptions within a particular domain of application or stakeholder community"(ISO/IEC/IEEE 2011). Enterprise architecture frameworks (EAF) guide and support the creation and interpretation of a concrete enterprise architecture description. Matthes (2011) describes more than fifty enterprise architecture frameworks. One prominent example is the Zachman Framework (Zachman 1987, 2008), which can be seen as a schema and a classification containing a "[...] total set of descriptive representations relevant for describing an enterprise" (Zachman 2008, p. 1). Another example of such an EAF is The Open Group Architecture Framework (TOGAF) (The Open Group 2009b).

- Architecture description language (ADL): "An ADL is any form of expression for use in architecture descriptions"(ISO/IEC/IEEE 2011). An example of an enterprise architecture description language is ArchiMate (Lankhorst, Proper, and Jonkers 2009; The Open Group 2009a, 2012). There are also description languages for specific parts of an enterprise architecture like BPMN (OMG 2011) for business process modelling or BMM (Object Management Group 2008) for business motivation modelling.

Enterprise architecture descriptions are a valuable source of background knowledge for the enterprise. Nevertheless, an enterprise architecture description needs to be created and maintained. If an enterprise decides to omit an enterprise architecture framework, selected enterprise models like business process models or organisation models can be considered alternative sources for background knowledge as well. Enterprise models contain valuable knowledge about the enterprise itself and provide adequate representations for different stakeholders.

2.3. *Enterprise Ontology*

CBR uses existing knowledge (sometimes called background, contextual or domain knowledge) as vocabulary to describe a case (Richter 1998). "Vocabularies define the concepts and relationships (also referred to as 'terms') used to describe and represent an area of concern" (W3C 2012b). Based on the original definition of Gruber (1993, p. 199), Studer, Benjamins, and Fensel (1998, p. 184) define an ontology as "a formal, explicit specification of a shared conceptualisation". "There is no clear division between what is referred to as vocabularies and ontologies" (W3C 2012b). "The trend is to use the word ontology for more complex, and possibly quite formal collection of terms [...]" (W3C 2012b).

In the enterprise domain, several enterprise ontologies have been introduced to describe enterprise models or architectures. "The main purpose of an enterprise ontology is to promote the common understanding between people across enterprises, as well as to serve as a communication medium between people and applications, and between different applications" (Leppänen 2007, p. 273). The Toronto Virtual Enterprise (TOVE) project introduced two foundational and four business ontologies (Fox, Barbuceanu, and Gruninger 1996; Fox et al. 1998). Other prominent ontologies are 'the enterprise ontology' (Uschold et al. 1998), the 'organizational memory' (Abecker et al. 1998) and the 'context-based enterprise ontology' (Leppänen 2007).

2.3.1. *ArchiMEO*

As mentioned, enterprise architecture descriptions are valuable sources of enterprise knowledge that can be used in CBR. To use the knowledge of enterprise ontologies, architectures and models, Kang et al. (2010); Hinkelmann, Merelli, and Thönssen (2010); Thönssen (2010) and Hinkelmann et al. (2015) suggest interlinking or relating enterprise ontologies with enterprise architectures and making these ontologies available in a machine-readable way. The enterprise ontology ArchiMEO[2] has been developed by the University of Applied Sciences and Arts Northwestern Switzerland FHNW. ArchiMEO is an enterprise ontology based on the enterprise architecture framework ArchiMate (Lankhorst, Proper, and Jonkers 2009; The Open Group 2009a, 2012) and Zachman (1987, 2008), and is extended with selected concepts from other enterprise ontologies:

---

[2]ArchiMEO is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License and available for download: http://www.ikm-group.ch/archimeo

TOVE (Fox, Barbuceanu, and Gruninger 1996; Fox et al. 1998), 'the enterprise ontology' (Uschold et al. 1998) and the context-based enterprise ontology (Leppänen 2007).

The foundation of the ArchiMEO ontology was laid during the development of the linked enterprise models and objects (LEMO) approach by Martin (2010) and Brun (2010). ArchiMEO is formalised using RDFS-Plus (Allemang and Hendler 2008), which is expressed in RDF(S) (W3C 2014a,c), with the distinction that certain resources are in the namespace of OWL (W3C 2012a). Apart from that, it is serialised using the RDF 1.1 Turtle (W3C 2014b) format. Finally, ArchiMEO has been successfully applied in risk management (Emmenegger, Laurenzini, and Thönssen 2012), contract management (Thönssen and Lutz 2012) and case-based reasoning (Martin, Emmenegger, and Wilke 2013).

2.4. *Case-based Reasoning Tools and Frameworks*

There exist several frameworks for CBR inter alia jCOLIBRI (Bello-Tomás, González-Calero, and Díaz-Agudo 2004) and jCOLIBRI2 (Recio-García, González-Calero, and Díaz-Agudo 2014a), myCBR (Roth-Berghofer and Bahls 2008; Bach and Althoff 2012) or CAKE (Bergmann, Kolodner, and Plaza 2006; Maximini and Maximini 2007).

2.4.1. *jCOLIBRI*

jCOLIBRI is a framework for creating CBR systems (Bello-Tomás, González-Calero, and Díaz-Agudo 2004). "The main goal of jCOLIBRI2 is to provide a reference platform for developing CBR applications" (Recio-García, González-Calero, and Díaz-Agudo 2014a, p. 3). jCOLIBRI2 is a multi-layer application. There exist two main components the COLIBRI studio and the jCOLIBRI engine itself. The studio component can be used to design, construct and configure a case-based reasoning application. Apart from the CBR core functionalities, the engine and the framework also provide several extensions for textual CBR including information extraction functionalities. In addition, the engine uses a persistence layer that provides connectors to relational databases, plain text and ontologies using an ontology bridge (Recio-García, González-Calero, and Díaz-Agudo 2014a). Recio-García, González-Calero, and Díaz-Agudo (2014b) introduce a template based design approach for the COLIBRI Studio Development Environment.

2.4.2. *myCBR and COBRA*

Roth-Berghofer and Bahls (2008) introduce a CBR plugin to the ontology editor Protégé called *myCBR*. The first version of the myCBR plugin was designed as a rapid prototyping tool and similarity-retrieval engine (Roth-Berghofer and Bahls 2008). myCBR consists of three main features and views: (1) The explanation editor can be used to describe concepts, which are used as case characterization; (2) the retrieval editor provides an interface to the conceptual and backward explanations. The backward explanations are description of the retrieval process, including the local and global similarity features; (3) the similarity measure editor provides a possibility for editing local and global similarity measurements. Stahl and Roth-Berghofer (2008) used the myCBR framework to configure a Web-based CBR application. Similarity and domain model are created with the Protégé-based editor and exported as XML files, which can be used in a Web application containing a CBR engine (Stahl and Roth-Berghofer 2008). The current version of myCBR is version 3. It is a rich client platform (RCP) of the Eclipse development IDE. A software development kit written in Java and allows to build own CBR applications (Roth-Berghofer et al. 2012). This approach has been successfully applied in decision

support, configuration (computer configurations), diagnosis and information composition (Bach and Althoff 2012).

COBRA (Conversational Ontology-based CBR plAtform) is a CBR application platform using an eclipse-based editor (Assali, Lenne, and Debray 2010). It aligns the cases to a domain-specific ontology using inference rules. COBRA uses certain parts of jCOLIBRI and Assali, Lenne, and Debray (2010) add an additional layer to support heterogeneous case bases.

2.4.3. Further Case-based Reasoning Tools and Frameworks

The AIAI *CBR Shell* is a system for "[...] automatic polling of fields for sensitivity to goal finding and the stochastic hill-climbing of ever-fitter combinations of field weights" (Wheeler and Aitken 2000, p. 94). The tool is a closed source available "in demonstration form". It shows the potential of K-nearest neighbour and threshold retrieval, multiple diagnostic algorithms, genetic algorithm weight learning and cross-platform implementation.

Manenti and Sartori (2010) present *CReP* (Case Retrieval Platform), a framework that provides tools to describe cases, similarity functions on case description parts using a hierarchical structure and string matching.

Bogaerts and Leake (2005) introduce an open source CBR framework called *IUCBRF*, which is written in Java for developing modular CBR applications. The system was a basis for several other CBR systems in the research community.

A commercially used CBR system *Information Access Suite* has been developed by Empolis. This system is one of "the most widely used" CBR-based systems in Europe. The CBR component acts as an "underlying methodology" of the Information Access Suite provided by Empolis. This Information Access Suite has been used in a wide range of "commercial application domains" (Bergmann et al. 2009, p. 3).

Bergmann et al. (2006) introduce a generic collaboration support architecture called *CAKE* (Collaborative Agent-based Knowledge Engine). CAKE can be used for the selection of agents and (sub-) workflows with the usage of CBR technology. "The CAKE (Collaborative Agent-based Knowledge Engine) architecture, for instance, combines workflow technology, agent technology, and structural CBR to select appropriate agents and workflows in knowledge-intensive application domains using CBR" (Bergmann et al. 2009, p. 3).

CREEK is a CBR system introduced by Aamodt (1991, 2004). It "[...] is a knowledge intensive approach to problem solving and learning" (Aamodt 1991, p. 137). There exists a Java-based implementation of CREEK called TrollCreek, which has been used in the petroleum industry (Bergmann et al. 2009).

*eXiT*CBR* is a CBR tool for medical prognosis (López et al. 2011; Pla et al. 2013). Pla et al. (2013, p. 1) "[...] propose a user-friendly medically oriented tool for prognosis development systems and experimentation under a case-based reasoning methodology". Besides several generic CBR tools, e.g., jCOLIBRI, eXiT*CBR has been developed under the focus of a particular purpose (medical prognosis) and domain (medicine). eXiT*CBR.v2 extends the isolated CBR system eXiT*CBR.v1 in such a way that collaborative data can be used.

2.5. Ontology-Based Case-Based Reasoning

In order to reduce the knowledge acquisition bottleneck, it is possible to enrich CBR systems with domain knowledge. "The more knowledge is embedded into the system, the more effective [it] is expected to be" (Recio-García, Díaz-Agudo, and González-

Calero 2008, p. 54). In this case, ontologies are helpful and can provide this knowledge. To use the potential of ontologies in a CBR system a combination of ontology-based systems and CBR is essential. Ontology-based systems can benefit from structural CBR and vice versa, as discussed in Bergmann and Schaaf (2003) and Bichindaritz (2004). Ontology-based CBR "[...] can take advantage of this domain knowledge and obtain more accurate results" (Recio-García, Díaz-Agudo, and González-Calero 2008, p. 54). Several approaches, such as the above mentioned jCOLIBRI (see section 2.4.1), myCBR and COBRA (see section 2.4.2) and other approaches by Daz-Agudo and González-Calero (2001); Wang, Hu, and Zhang (2003), combine ontologies and CBR. Unfortunately, only a limited number of approaches go beyond taxonomic CBR (Bergmann 1998), including the properties/relations in ontologies (Hefke 2004; Hefke and Abecker 2006a; Hefke 2008).

Such an approach has been introduced by Hefke (2004), which is part of the Knowledge Management Implementation and Recommendation (KMIR) Framework (see further: Ehrig et al. (2004); Hefke and Abecker (2006a,b); Hefke et al. (2006); Hefke (2008)).

Chen and Wu (2003) introduce a RDF-based markup language for case-based reasoning called CaseML. As Fidjeland (2006) argues, CaseML defines only a small vocabulary with limited expressiveness compared to other approaches. Fidjeland (2006) introduces in his master thesis an OWL vocabulary for Creek (see section 2.4.3). Apart from the vocabulary, Fidjeland (2006) introduces a possibility for sharing the case base and the domain model using an OWL representation. Fidjeland (2006) uses Jena (McBride 2001) and Creek for implementing the proposed OWL vocabulary approach.

Bichindaritz (2004) introduces a case-based reasoning framework called Mmoire for biological and medical cases. That CBR system uses a semantic web standard based interchange language, "[...] bridge the gap between the multiple case based reasoning systems dedicated to a single domain [...]" (Bichindaritz 2004, p.47). Bichindaritz (2004, p.61) points out the benefit of such an ontology-based case-base and inter-change possibility as follows: "The perspective of unlimited cooperation between these systems is extremely promising for the improvement of healthcare and biomedical research [...]".

It is therefore not surprising, that the medical and clinical domain has been prominent in the recent past. In the following we list the four main research paths of recent studies, two domains and two methods, in the field of ontology-based CBR:

- *Medical and clinical domain:* Shen et al. (2015) propose an ontology-based CBR and multi-agent-based clinical decision support system. The used ontology "[...] employs the domain knowledge to ease the extraction of similar clinical cases and provide treatment suggestions to patients and physicians" (Shen et al. 2015, p.307). Sene, Kamsu-Foguem, and Rumeau (2015) propose an ontology-based CBR approach based on taxonomic reasoning for telemedicine in the oncology domain with the inclusion of natural language processing (NLP). Delir Haghighi et al. (2013) introduce a development and evaluation of an ontology-based CBR system in medical emergency management.

- *Early warning and emergency domain:* Ju et al. (2016) propose an incentive-oriented early warning system (EWS) using ontology-based CBR for predicting co-movements of oil price shocks and the macro-economy. They underpinned the importance of having a clear procedure model with an end-user and expert involvement when implementing an ontology-based CBR system. Amailef and Lu (2013) introduce an ontology-based CBR implementation for intelligent m-Government emergency response services. It is notable that this implementation gives end users the possibility to adjust extempore certain similarity weights during retrieval phase and allows them to evaluate the proposed solution (outcome) during retaining phase.

- *Semantic rule based CBR:* Bouhana et al. (2015) introduce an ontology-based CBR approach for personalized itinerary search systems for sustainable urban freight transport with the inclusion of personalized rules. Bulu, Alpkocak, and Balci (2012) describe a CBR approach for ontology-annotated mammographies. They use a "Semantic Query-enhanced Web Rule Language (SQWRL) to process retrieval of similar masses from annotated mammography collection in OWL" (Bulu, Alpkocak, and Balci 2012, p.11194). SQWRL enhances semantic web rule languages with 'SQL-like operations' for the retrieval in an OWL ontology.
- *Fuzzy-ontology oriented CBR:* El-Sappagh, Elmogy, and Riad (2015) introduce a CBR approach using an OWL2 fuzzy ontology that is used in a diabetes diagnosis application scenario. The results are promising: "The resulting system can answer complex medical queries related to semantic understanding of medical concepts and handling of vague terms" (El-Sappagh, Elmogy, and Riad 2015, p.179). They successfully compete against traditional CBR approaches. El-Sappagh, Elmogy, and Riad (2015, p.206) conclude that their "[...] fuzzy-semantic retrieval algorithm outweighs all of the JCOLIBRI algorithms, and it covers their limitations".

2.6. Conclusion

This literature review shows that CBR itself can benefit from ontology-based vocabularies. We mentioned numerous CBR tools and frameworks implementing ontology-based CBR approaches (see section 2.5), which are underlining the potential of accessing and utilizing an ontology-based repository. Among others, Potes Ruiz, Kamsu-Foguem, and Noyes (2013) and Kamsu-Foguem and Noyes (2013) show that an integrated CBR approach can support the sharing of lessons-learned by collaborative experts.

Additionally, recent work shows how enterprise ontologies can benefit from enterprise architecture and models (see sections 2.2 and 2.3). Utilizing an enterprise ontology in an ontology-based CBR approach can be regarded as a next logical step as suggested by Martin, Emmenegger, and Wilke (2013). Unfortunately, no significant attention has been paid in recent work in ontology-based CBR concerning the different viewpoints and concerns of the users. Therefore, we introduce in the following sections a CBR approach taking into account the specific viewpoints and concerns of the users.

3. Method

This work is an outcome of the research project called [sic!], which stands for **s**oftware **i**ntegration using ontology-based **c**ase-based reasoning. The applied research work was conducted together with the ELO Digital Office CH AG, a software and consulting company for enterprise content management.

This work followed the methodology of design science research (DSR) as introduced by Vaishnavi and Kuechler (2004) but was enhanced with elements from Peffers et al. (2008). As shown in figure 2, the process started with (1) an *awareness* phase in which the research problem was addressed and motivated. This was done in close collaboration with the application partner. The outcome of the (2) *suggestion* phase was a tentative design of the approach and the system. Based on iterative and agile development, we decided to execute the (3) *development* in incremental cycles. Iterative and agile approaches split up the requirements into smaller pieces and demonstrated the solution (also called increments) using small use cases (Métrailler 2011). After performing the incremental development phase, the resulting artefacts were evaluated in the (4) *evaluation* phase. Finally, the results were documented in the (5) *conclusion* phase.
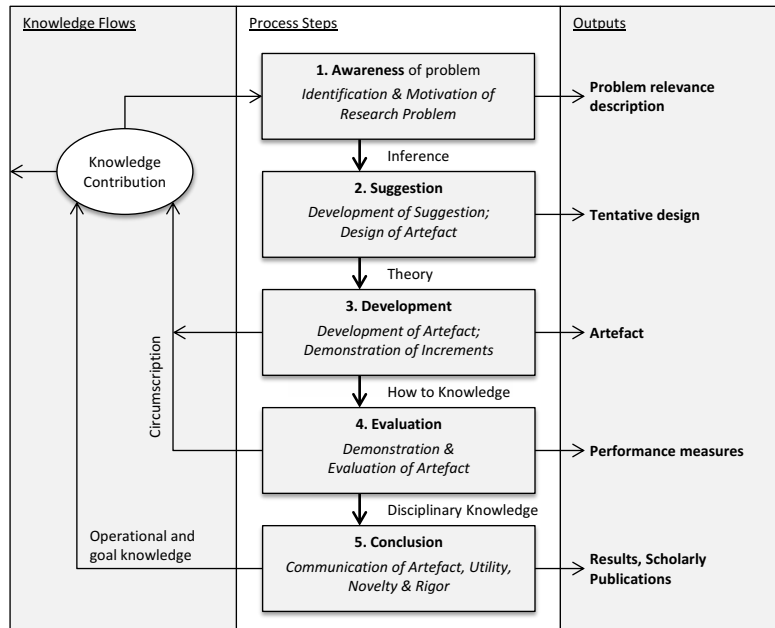
Figure 2. Design science research process adapted from Vaishnavi and Kuechler (2004) and enhanced with elements from Peffers et al. (2008).

Following the *guidelines* of DSR proposed by Hevner et al. (2004), we produced several artefacts. We developed an implementation (artefact A: instantiation) of the ontology-based CBR approach (artefact B: model), including the ontologies and retrieval mechanisms (artefact C: method) that was applied to a business problem given by our application partner. Additionally, we followed the *principle* of action design research in such a way that we introduced an approach that is practice inspired and "[...] can be applied to the class of problems that the specific problem exemplifies" (Sein et al. 2011, p. 40). Our approach was the outcome of 'design as search process'.

After developing the artefact, the approach was evaluated in close collaboration with our application partner using *scenarios* as the design evaluation method. Using these detailed scenarios we could demonstrate the utility of the artefact. In addition, we executed *simulations* using artificial data derived from real-life data.

4. Integrating an Enterprise Ontology into Case-Based Reasoning

This section provides an overview of the approach in sub-section 4.1, including a brief description of the ontologies and the implementation. Sub-section 4.2 presents how the case structure was realised, including a brief description of the case characterisation, similarity mechanism and case lesson.

4.1. *Overall Approach*

An initial and first version of the 'integrated enterprise ontology and CBR approach', presented in this section was previously published in an early and first version by Martin, Emmenegger, and Wilke (2013).

As mentioned in the introduction, CBR is a common methodology for humans and systems to solve problems. Recent work (Bergmann et al. 2003) has shown that the

CBR methodology implemented in information systems can be successfully applied in business and is an adequate methodology for experience management (Bergmann 2002). During the elaboration of the application scenario and the following awareness phase, it was possible to observe that people implicitly applied the CBR methodology. It is therefore possible to conclude that people, at least in our application scenario, were already familiar with this kind of work. CBR relies on knowledge gathered in advance or in former iterations. The knowledge is stored using a certain structure in a case base. Bergmann (2002) argues that structural CBR is best suited for experience management and knowledge-intensive tasks. As mentioned in the literature review in section 2, structural CBR can benefit from ontology-based knowledge representation. In order to exploit the full potential, a structural ontology-based CBR approach was introduced by Martin, Emmenegger, and Wilke (2013). This enabled the reuse of existing domain knowledge during the execution of the CBR cycle.
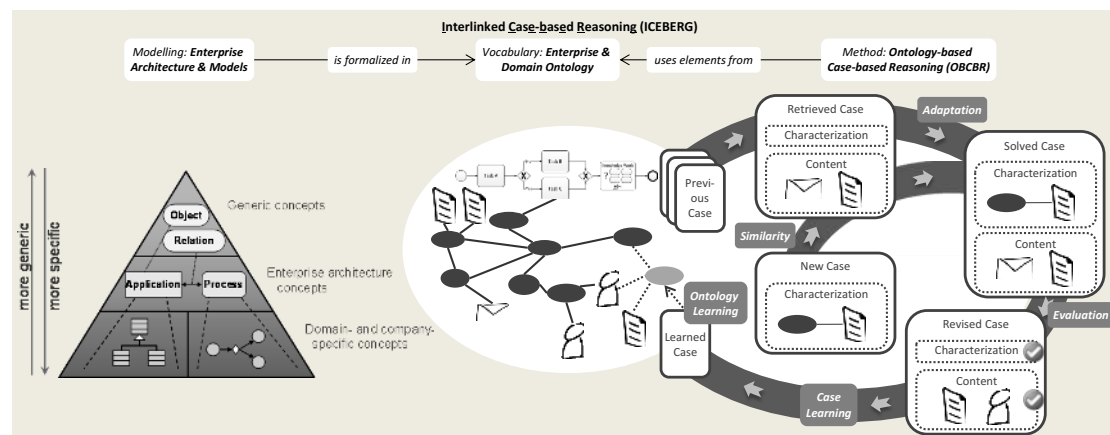

Figure 3. Interlinked CBR approach (adapted from Martin, Emmenegger, and Wilke (2013)).

Bergmann and Schaaf (2003) suggest to use a 'knowledge container' to represent background and domain knowledge. Approaches that do not consider background knowledge "[...] are often isolated and closed in the sense that they are not developed with respect to cooperation with other systems" (Bergmann and Schaaf 2003, p. 622). Ontology-based CBR uses the same technology for the CBR case base and the knowledge container. Our approach uses the enterprise ontology ArchiMEO for this purpose. "Most [ontology-based CBR] systems rely on proprietary, sometimes even XML compliant, languages for the vocabulary and the cases but do not facilitate the exchange of knowledge" (Bergmann and Schaaf 2003, p. 622). The technology used in this approach is based on W3C standards and recommendations.

As mentioned in section 2.2, enterprise architecture frameworks are commonly used in business to model enterprise-specific elements and integrate existing models into a coherent architectural description. These architectural descriptions can be formalised using enterprise ontologies, as mentioned in section 2.3. ArchiMEO interlinks enterprise ontologies with enterprise architectures and provides the possibility to formalise descriptions of enterprise architectures or models. As a consequence, formalised enterprise architectures can deliver existing domain knowledge that improve CBR.

Ontology-based CBR (OBCBR) systems can make use of these formalised descriptions of enterprise architectures or models. As mentioned in section 2.5, there exist some frameworks that implement OBCBR. This approach was extended to allow enterprise ontologies to be applied in a CBR system and tackles the research problem mentioned in section 1.

11

Figure 3 depicts an overview of the elaborated approach, called the **i**nterlinked **c**as**e**-**b**as**e**d **r**easonin**g** approach or, in short, the ICEBERG approach. The iceberg metaphor is commonly used in applied psychology, pedagogic and interpersonal communication. It also visualizes the Pareto principle (80/20). The metaphor was first described by Ernest Hemingway. Later, it was used by Sigmund Freud to explain the differences between consciousness and un-consciousness in human action. In this work, the metaphor has been used to describe the notion and goal of the case-based reasoning approach: *Using interlinked (ontology-based) case-based reasoning to bring* **hidden knowledge to the surface**.

A description of an enterprise architecture is formalized in the enterprise ontology ArchiMEO. The ICEBERG approach enables reusing elements from the enterprise ontology. In the section 4.1.1, we describe how the ontology is structured and in section 4.1.2, we describe how the overall approach is implemented.

4.1.1. Overview of the ontology structure

This approach relies on the following ontology structure. Figure 4 (ontology structure overview) shows the dependencies (imports) of the ontologies. These ontologies are formalised using RDFS-Plus (Allemang and Hendler 2008) and extended with certain resources in the namespace of OWL (W3C 2012a).
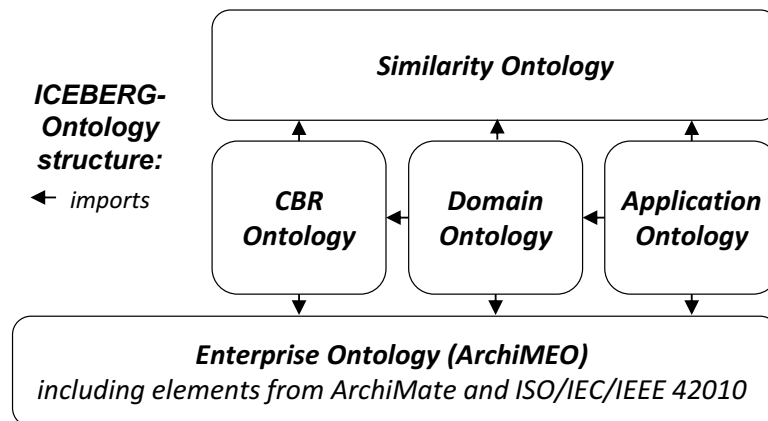

Figure 4. Ontology structure.

As mentioned in section 2, there is no clear differentiation between the term vocabulary and ontology (W3C 2012b). And even a "[...] coincidence of [a CBR] vocabulary and an ontology [...]" can be observed (Bergmann and Schaaf 2003, p. 622). This approach is oriented towards the following statement of Bergmann and Schaaf (2003, p. 622): "Neglecting the fact that an ontology typically serves many purposes one can say that a [CBR] vocabulary is an ontology of the domain of discourse underlying the [CBR] application".

The *enterprise ontology (ArchiMEO)* is the foundation of the ICEBERG approach. ArchiMEO consists of concepts defined in ArchiMate and ISO/IEC/IEEE 42010 (see section 5) and concepts from other enterprise ontologies. This provides the basic ontology to describe an enterprise. Based on ArchiMEO, the following five ontologies were built in coherence to ArchiMEO; no ontology merging was applied. The retrieval and configuration of the CBR system is done using the *similarity ontology* in combination with the *CBR ontology* (see sections 4.2.1 and 4.2.2). A *domain ontology* contains enterprise-specific domain knowledge of ELO Digital Office AG, our application partner. A further

domain ontology, the *project ontology* (not shown in figure 4), contains specific concepts and relations for the project management use case as presented in this work. An *application ontology* specializes the domain ontology with respect to an enterprise idiosyncrasy (van Heijst, Schreiber, and Wielinga 1997).

The ontologies contain a large number of concepts. In this paper we limit ourselves to present only the concepts necessary to explain the CBR approach . We refer to Emmenegger, Laurenzini, and Thönssen (2012); Thönssen and Lutz (2012) and Martin, Emmenegger, and Wilke (2013), in which ArchiMEO is described and used in different contexts; the foundation of the ontology has been presented in Martin (2010) and Brun (2010). This foundation has been enhanced by elements of ArchiMate as described by 'The Open Group (2012)'.

4.1.2. Implementation

Figure 5 shows how the approach previously presented in sections 4.1 and 4.1.1 was implemented. We initially formalised elements from the enterprise models and architecture of our application partner in a domain ontology using ArchiMEO concepts. Additionally, we developed a project management ontology with the help of the application partner's use case. The implementation, which is called ICEBERG-Toolkit, is divided into two main components: a server and a client component. The ontology itself is stored in a triple store: an Apache Jena TDB[3]. The persistence and business logic component uses the ontology and similarity functions to execute the reasoning. This is developed in Java and uses the Apache Jena API[4] for building the semantic Web application. The presentation and business logic components contain views of the CBR system. This is developed in Java using the JavaFX[5] GUI framework and the OpenDolphin[6] open-source library for a lightweight remote model-view-controller separation.
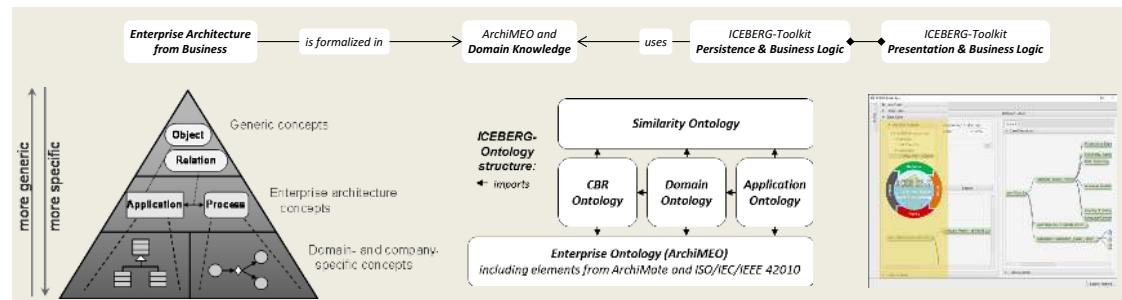

Figure 5. Implementation of the ICEBERG approach.

4.2. Modelling Characterisation and Lesson

For our basic case structure we follow Bergmann (2002), who distinguishes between the case characterisation space and the lesson space. The case characterisation space describes relevant facts of experiences. The lesson part of cases covers the solution to a problem and the decisions made or the steps involved to solve the problem. Both case characterisation and lesson elements are modelled as formal case structures in the

---

[3]Apache Jena TDB: http://jena.apache.org/documentation/tdb
[4]Apache Jena API: http://jena.apache.org
[5]JavaFX: http://www.oracle.com/technetwork/java/javase/overview/javafx-overview-2158620.html
[6]OpenDolphin: http://open-dolphin.org

enterprise and domain ontologies. A triple store serves as the database and knowledge base for the cases and their instances.

4.2.1. Case characterisation and similarity model

In our ICEBERG approach, the characterisations of cases are expressed by a defined part of the enterprise and domain ontology, as shown in figure 6. This explicit knowledge is used for the comparison of a query case with historical cases in the CBR retrieval phase. The applied similarity measures use this knowledge. To define which concepts and relations belong to the case characterisation and which similarity functions and weights shall be applied, a similarity ontology based on OWL annotation elements was created. The similarity ontology can be seen as the similarity model known from structural CBR systems. The similarity approach has the advantage that explicit enterprise and domain knowledge can be modelled together with similarity measures and that a standard ontology development environment can be used to model the case characterisations.
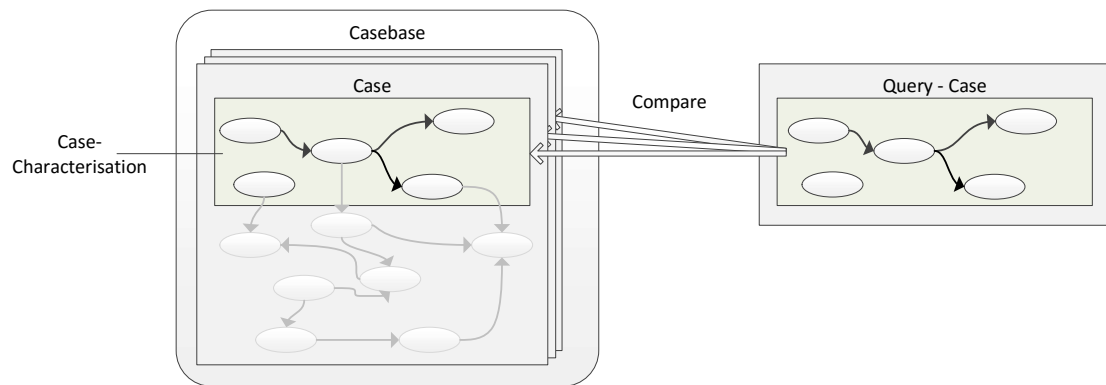

Figure 6. Case characterisation.

The OWL ontology language defines different kinds of properties. With the similarity model we can determine the definitions of similarity measures, functions and weights for annotation, datatype and object properties, as well as for classes of the characterisation space. For the retrieval and ranking of historical cases, the user defines a query based on the case characterisation structure and elements. The query is compared with all characterisations in the case base and the system returns a ranked list of historical cases with a similarity value between 0 and 1.

4.2.2. Local, global and taxonomic similarity

The similarity computation can be reduced to the global-local principle, according to Bergmann (2002). Simple attributes of class instances in the same hierarchy are compared by individually defined similarity functions and are aggregated according to a global similarity. All attributes are individually weighted, and simple attributes are represented by annotation and datatype properties with simple value types such as *Integer* or *String*. Relational attributes are represented by object properties. Since relational attributes might lead to a comparison of multiple referenced instances in the query as well as in the historical case, the global similarity function must be a set function, as proposed in Hefke and Abecker (2006a). For the evaluation case we have implemented a cosine set function, as found in several frameworks (Recio-García, González-Calero, and Díaz-Agudo 2014a; Hefke and Abecker 2006a), and a probabilistic similarity function commonly used in information retrieval.

The framework we provide is flexible enough to be enhanced with new global and local similarity functions. Besides the global-local similarity knowledge, the object-oriented case characterisation might contain taxonomic knowledge, which can be considered in a local similarity measure as well. A first implementation of a similarity function in the framework considers the object positions of query case instances and instances in the same taxonomic tree of the compared historical cases. The measure applied uses the instance distance to express the similarity; i.e., the number of nodes between the instances in the taxonomy.

4.2.3. Case lesson or content

The lesson part is comparable to the solution space or case content in traditional CBR terminology, but it differs, according to Bergmann (2002, p. 50), in the way "[...]the lesson space can contain information that is not the solution itself but useful to find a solution".
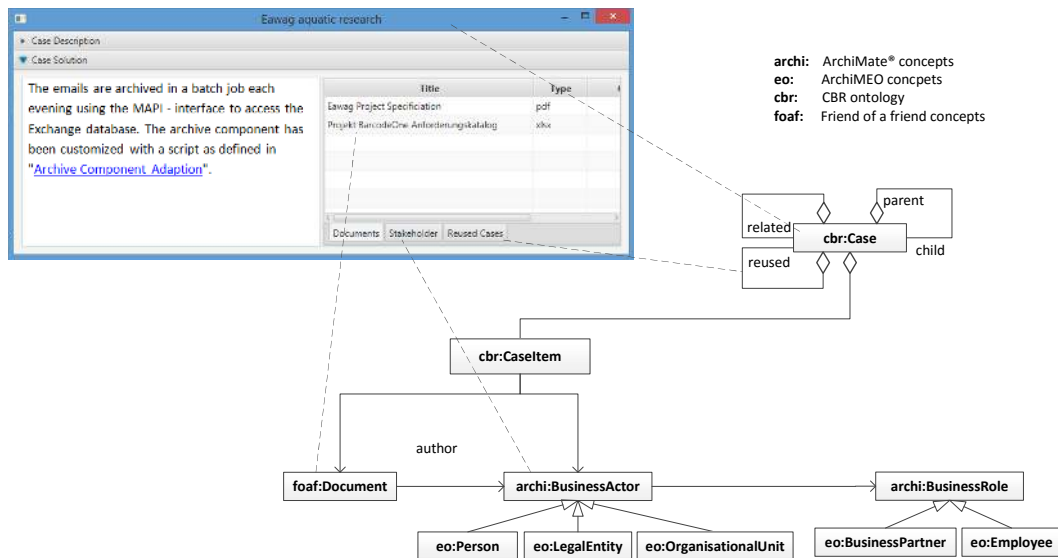

Figure 7. Case lesson model reflected in GUI.

The knowledge of the lesson space is expressed in the enterprise and domain ontology. Elements of the case characterisation are not excluded from the lesson. Knowledge items like documents are represented as links in the ontology knowledge base. Figure 7 depicts this extended model and shows how the lesson elements are reflected in the graphical user interface. *Case* is the main class, and with the relational attributes (parent-child, related, reused), a hierarchical case structure can be built and reused cases can be kept as a reference. Each *Case* consists of *CaseItems*. Up to now we linked the ArchiMEO concept *BusinessActor* with *BusinessRole* as well as the *Document* concept from the FOAF vocabulary to the *CaseItem*. Besides an HTML free text description, these concepts represent the lesson items of our cases.

5. Case View and Viewpoints

During our analysis we recognised that different stakeholders have different concerns and therefore have diverse interests regarding experienced case knowledge. We recognised that a single case characterisation with similarity measures and weights cannot support all

these different interests. Therefore, we introduced the concept of views on a case-by-case characterisation.
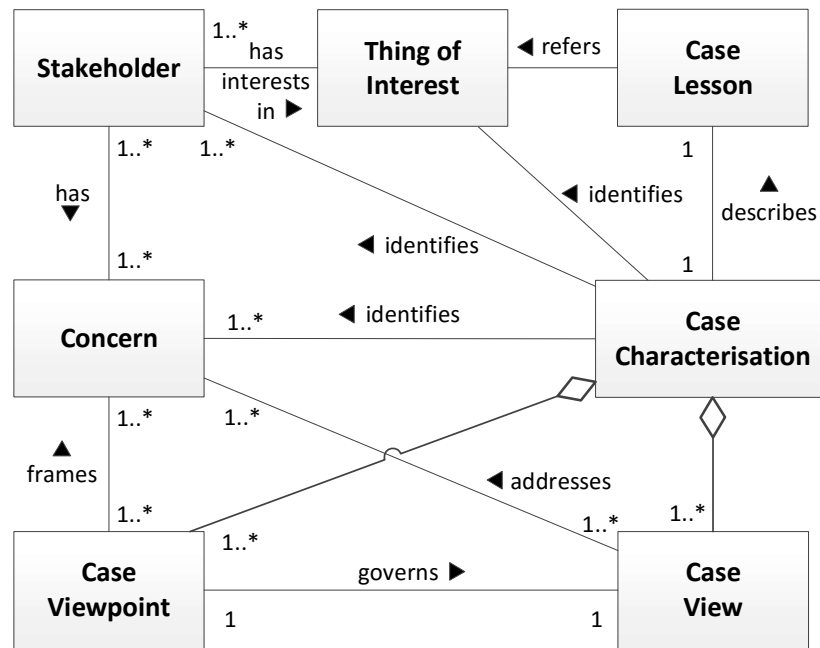

Figure 8. Case viewpoint model.

In line with the view in enterprise architectures, we consider viewpoints to allow stakeholders to define their specific views on cases and case characterisations. As emphasised by The Open Group (2012), the problem of how to satisfy the needs of many actors has been researched since the 1990s. According to ISO/IEC/IEEE 42010 an architecture viewpoint represents the interest of one or more stakeholders, whereas a view expresses the "[...] Architecture of the System of Interest from the perspective of one or more Stakeholders to address specific Concerns [...]" (ISO/IEC/IEEE 2011). The notion of views, introduced by the CIMOSA enterprise architecture framework (ESPRIT Consortium AMICE 1989), and viewpoints can be found in several enterprise architecture frameworks, e.g., TOGAF (The Open Group 2009b) and ArchiMate (The Open Group 2009a, 2012).

ArchiMate (The Open Group 2012, p. 74) defines a view "as a part of an architecture description that addresses a set of related concerns and is addressed to a set of stakeholders. A view is specified by means of viewpoint [...] Simply put, a view is what you see and a viewpoint is where you are looking from". Analogous to ArchiMate (The Open Group 2012, p. 75), we define a viewpoint as a means to focus on particular aspects of case characterisations, respectively case descriptions. With this approach we are able to significantly improve case retrieval since a stakeholder's concern is taken as a selection criterion. In case two or more stakeholders share the same concern, most likely they will weigh them differently. Figure 8 shows an adaptation of the conceptual model of architectural description. We adapted the conceptual model from ISO/IEC/IEEE (2011) as follows (see figure 8):

- *Thing of interest*: The *thing of interest* is something in which a *stakeholder* has interest, which might be described as part of a *case lesson* or *case characterisation*. It can be a system, experiential knowledge, information need, etc.; the term is used as a place holder.

- *Case lesson*: The *case lesson* "[...] can contain information that is not the solution itself but useful to find a solution [...]" (Bergmann 2002, p. 50), including links to information sources containing a potential solution.
- *Case characterisation*: The *case characterisation* "describes all facts about the experience that are relevant for deciding whether the experience can be reused in a certain situation" (Bergmann 2002, p. 50).
- *Stakeholder*: "*Stakeholders* are individuals, groups or organisations holding *concerns* for the [*thing of interest*]".
- *Concern*: A *concern* is any interest in the experiential knowledge system or *thing of interest*.
- *Case viewpoint*: A *case viewpoint* is "[...] a set of conventions for interpreting, using and analysing one type of [...]" *case view* and is derived from an architectural viewpoint. A *case viewpoint* frames a specific set of *concerns*. Examples of case viewpoints include: technical, project management or business management.
- *Case view*: A *case view* expresses the *case* (including the lesson) of the *thing of interest* "[...] from the perspective of one or more *stakeholders* to address specific *concerns*, using the conventions established by its *viewpoint*".

Furthermore, ArchiMate (The Open Group 2012, p. 76) provides a framework to classify viewpoints based on two dimensions: purpose and content, "i.e., the purpose that view must serve and the content it should display".
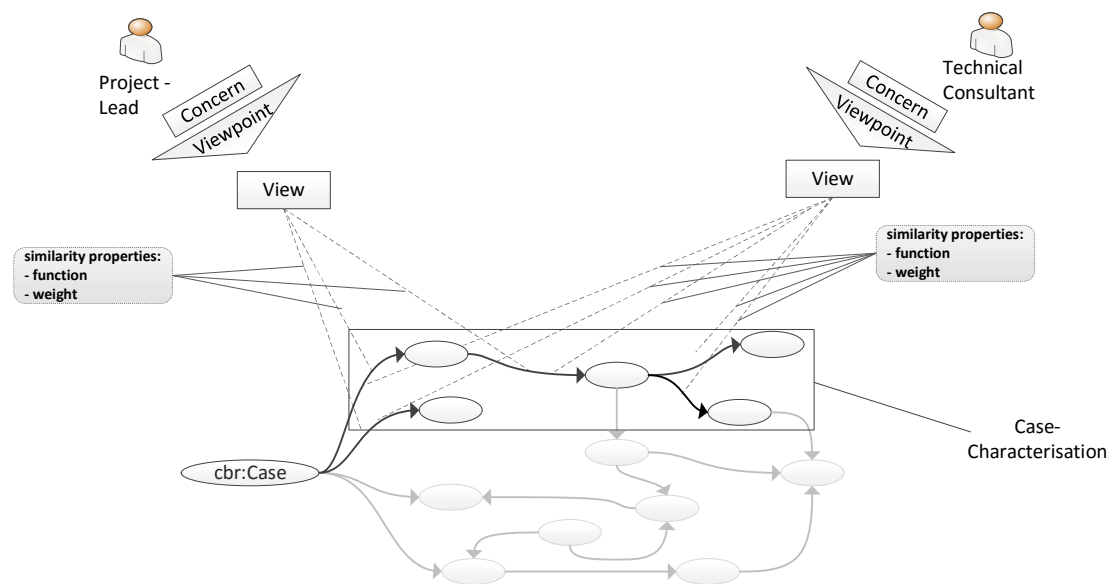

Figure 9. Example views on case characterisation.

Figure 9 shows an example of different stakeholders and concerns leading to different viewpoints and views on the case characterisations. A different view means that part of the case characterisations (classes and relations), the similarity measures (functions) and the weights of the properties (simple and relational ones) differ, depending on the stakeholder's viewpoint and concerns. For the retrieval, a specific characterisation viewpoint is chosen and only those characterisation elements assigned to the corresponding view are used to formulate a query statement. To characterise learned cases released to the knowledge base, the full characterisation defined for the case would be available again.

6. Procedure Model

To apply the described approach in practice, a procedure model has been elaborated called the ICEBERG procedure model. As mentioned in section 2.5, Ju et al. (2016) underpin the importance of having a clear procedure model with an end-user and expert involvement. To apply the ICEBERG approach in practice, an eight-step ICEBERG procedure model is suggested as follows (see figure 10):
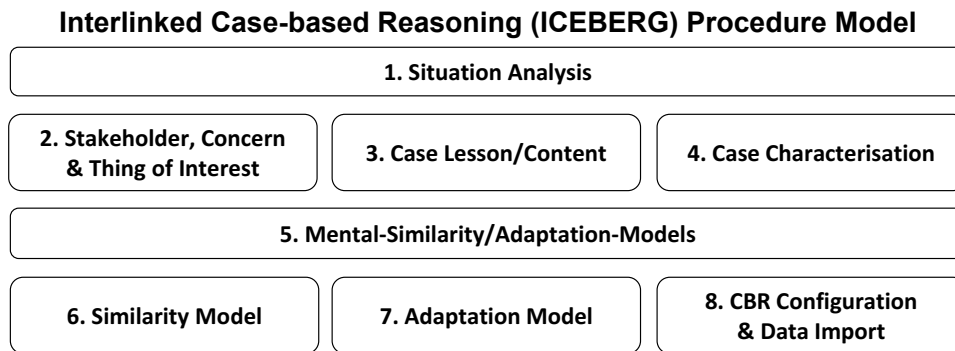
Interlinked Case-based Reasoning (ICEBERG) Procedure Model

1. Situation Analysis

2. Stakeholder, Concern & Thing of Interest	3. Case Lesson/Content	4. Case Characterisation

5. Mental-Similarity/Adaptation-Models

6. Similarity Model	7. Adaptation Model	8. CBR Configuration & Data Import

Figure 10. Interlinked Case-based Reasoning Procedure Model.

(1) *Situation Analysis:* In order to fully understand the current use case, where the CBR approach is to be implemented, a first situation analysis is performed. Requirement elicitation and creativity techniques are applied. It is advisable that this situation analysis is done in a focus group and workshop-like setting. We observed that with story-telling impressive results can be achieved. However, it is additionally advisable that a process landscape, in relation to the existing enterprise architecture description, is elaborated, from which further use cases can be derived.

(2) *Stakeholder, Concern and Thing of Interest:* Based on the situation analysis, stakeholders, their concerns and finally the thing(s) of interest are derived. This identification will then be used to elaborate the various mental-models of the different stakeholders to finally derive the view and viewpoint-based similarity model as introduced in section 5.

(3) *Case Lesson/Content:* Based on the situation analysis, the case lesson/content is elaborated together with the focus group member. It is advisable that exemplary cases with initial data are created and discussed. Additionally, it is advisable to define how this initial data is acquired or imported.

(4) *Case Characterisation:* In conjunction with the case content, a generalized case characterization is created. At this stage, it is essential that an existing enterprise architecture is used to build an initial case characterisation vocabulary as described in section 4.2.1.

(5) *Mental-Similarity/Adaptation-Models:* In essence, CBR is about representing mental similarity and adaptation models in a consolidated form as a configurable model. Before such a configurable model can be implemented, the different mental models are elicited and consolidated from the focus group members. As an example, the following question might be asked: which characterisation element is more important than the others to find or differentiate a certain case? These mental models are consolidated but still relative to the stakeholders and their concerns as depicted in Figure 9.

(6) *Similarity Model:* Based on the mental models and generalized case characteri-sation, a CBR expert configures the similarity model relative to the views and viewpoints. This configuration is done by determining global and local similarity functions and assigning weights as described in section 4.2.2.

(7) *Adaptation Model:* Apart from the similarity model, the adaptation behaviour is configured by a CBR expert as well. Currently, manual or semi-manual adaptation using rules are supported.

(8) *CBR Configuration and Data Import:* Finally, the entire system is configured by a CBR expert and initial data is imported.

7. Evaluation and Application Case: Project Knowledge

In this section, we describe two application scenarios derived from the analysis of the business of our application and business partner, ELO Digital Office CH AG. We applied our approach to these scenarios and verified the utility with a prototype (instantiation).
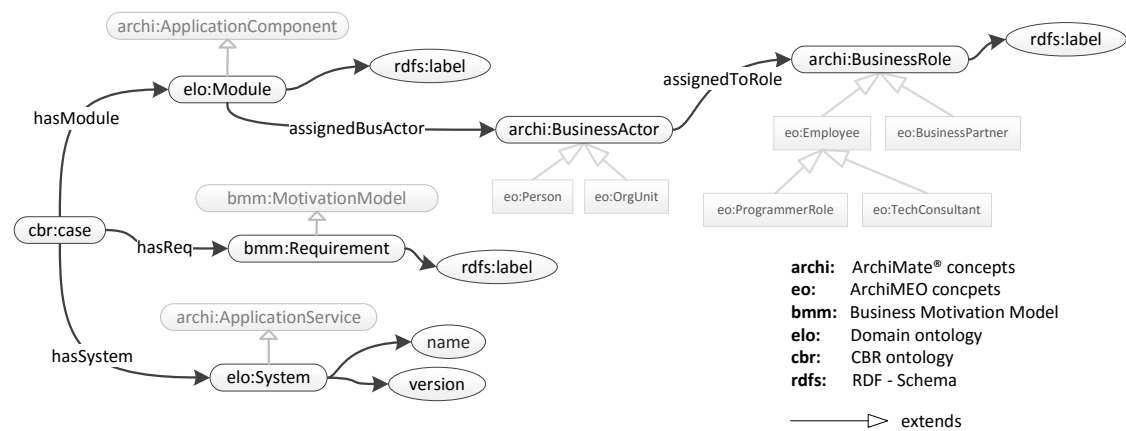

Figure 11. Application case characterisation.

During the sales and offer phases, ELO experts have to analyse a large set of require-ments and answer questionnaires provided by their customers. The offer is made based on this analysis. Another service provided by ELO is the management of projects with the target of integrating its standard software in the IT environment of its customers. For both cases, we have defined application scenarios where a CBR system can provide knowledge for the given cases and tasks to be performed.

The main knowledge items recognised to characterise such cases are depicted in figure 11. In every project some modules of the ELO software have to be integrated with legacy systems in the customer's IT environment. The characterisation shown in figure 11 has been defined with annotations of the similarity ontology applied to the enterprise and domain ontology. We can see that concepts of the business partner domain ontology extend concepts of ArchiMEO and the business motivation model.

7.1. *Offering Viewpoint*

During the offer phase, a technical consultant answers questions about requirements specified by potential customers. While the ELO software modules are well known by the consultant, the questions about the integration of legacy systems are more critical and the technical consultant is looking for experiences made with integrated systems in previous

19

projects. These concerns lead to a specific viewpoint on previous cases and is reflected in a view on the case characterisation, as shown in figure 12. Weights and similarity functions defined in this view are applied when a query based on this view is compared with the previous cases, with respect to their characterisations. The background knowledge of the defined scenarios and the stakeholder's concerns regarding the system are made explicit with the higher weight defined for systems in this view.
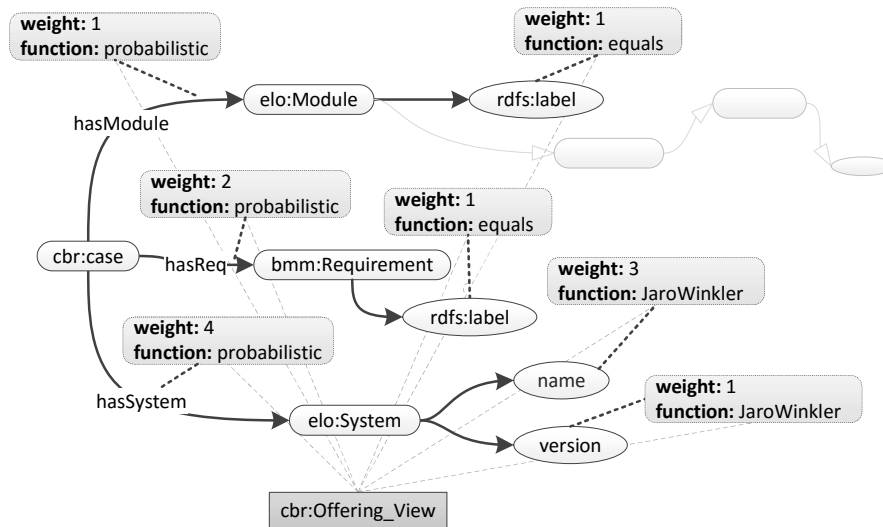
Figure 12. Viewpoint offering.

7.2. *Project Management Viewpoint*

In another business scenario a project manager is staffing his project team. The manager knows that for specific modules some adaptations have to be programmed to fit the target environment. Therefore, he is looking for previous cases where equal modules have been integrated and programmers have been assigned. This specific view is reflected in figure 13, which shows the same case characterisation but with different weights (the similarity of modules is now of higher interest) than in the offering viewpoint.
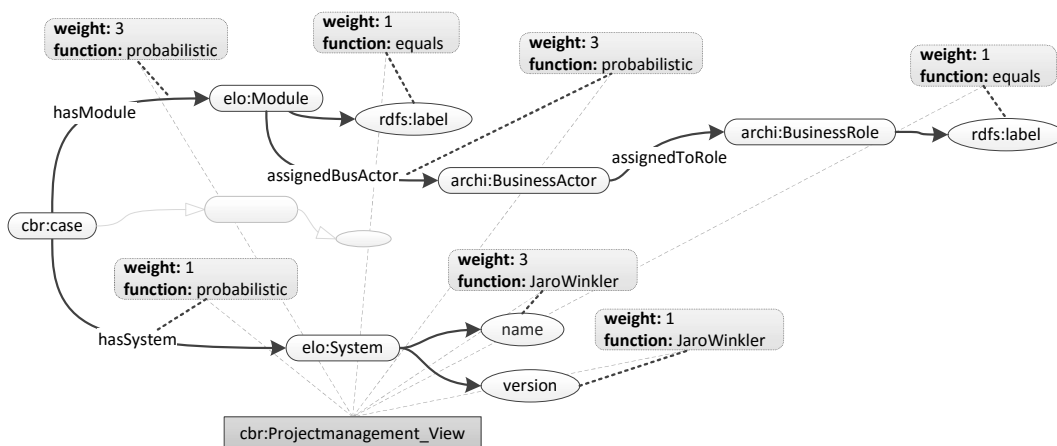
Figure 13. Viewpoint project management.

7.3. *Retrieval Example*

To show the differences between the previously described viewpoints, we apply the following simple query to sample cases with different views on the case characterisations:

- System: name="Oracle", version="12g"
- Module: label="Archive"

The previous cases stored in our case base are:

Case 1

- System: name="SAP", version="ECC 6.0"
- Requirement: label="Archiving"
- Module: label="Archive"
- Business Actor Role: label="Business Analyst"

Case 2

- System: name="Oracle", version="10g"
- Module: label="Barcode"
- Business Actor Role: label="Business Analyst"

As previously explained, global similarity functions can be defined on concepts to aggregate local similarity values from the attribute comparisons or on relational attributes where multiple instances are compared. In our examples, the global similarity measure applied to all relational attributes is the probabilistic function. Since we did not define dedicated functions on concepts, implicitly the defined functions on relational attributes are also applied to aggregate the similarity values of the attributes encapsulated by the referenced class. The probabilistic function, as defined in formula 1, aggregates local similarities in a more conjunctive way than other functions.

$$P(i_1|i_2) = \prod_{k=1}^{n} P(J_{k1}|J_{k2})^{\alpha_k} \qquad (1)$$

where J_{kl} are the set of instances that are linked to instance i_l and the weights α_k reflect the relative influence of relationship (or attribute) r_k for determining the overall similarity.

Local similarity functions applied in the example are the *Equals* and *JaroWinkler* functions. The equality is only used in cases where we expect a closed list like the ELO modules or the set of possible requirements defined in the ELO terminology. For the other text attributes, we applied the JaroWinkler function described in Cohen, Ravikumar, and Fienberg (2003). Many other functions have been implemented and could also be used, but for the example, we kept it simple.

The given query applied with different viewpoints and views would lead to the following similarity results:

Case:	Offer view:	Proj.-Mgmt. view:
Case 1	0.20	0.75
Case 2	0.67	0.21

Table 1. Computed case similarities for the different viewpoints and corresponding views.

As expected, case 2 has a higher similarity and rank with the offering viewpoint due to the matching system and a lower similarity with the project management viewpoint, where the module with possible programmers assigned is of higher interest. To explicitly search for cases with programmers for specific modules, the query must be extended. But this would only be possible from the project management viewpoint with the corresponding view on the *BusinessActor* characterisation concepts.

8.   Discussion

The approach presented in this paper addresses the research problem that *there is a new approach needed which presents relevant information in the context of the specific concerns and roles of the employees working on projects.* Section 5 addressed this research problem by introducing the case view and viewpoint method that has been transferred to the ontology-based CBR domain. That was possible using the ICEBERG approach described in section 4, which laid the foundation for solving the research problem.

Limitations: The proposed approach provides automatic retrieval (similarity), semi-automatic reuse (rule-based adaptation), manual revision (evaluation) and automatic retention (case learning, adding the case to the case base).

Contribution to teh Body of Knowledge: This paper makes a contribution to knowledge on CBR concerning case views and viewpoints, which allows for the considering of information needs in relation to the specific concerns and roles of employees working on projects. In addition, this paper makes a further contribution to knowledge concerning the realisation of the ISO/IEC/IEEE 42010 standard in an enterprise ontology and the subsequent application to a CBR context. This research also makes a contribution to experience management by enabling the retrieval of project knowledge from an ontology-based CBR case base. This ontology-based CBR approach uses the knowledge of an enterprise architecture description in a CBR system using the enterprise ontology ArchiMEO.

Contribution to Practice: During this research work it became clear that in addition to the research problem, the focus on standardised methods was important for business users. Therefore, the research work contributes to practice (business) by proposing standardised methods and technologies. Since the enterprise concepts and relations of ArchiMEO are based on ArchiMate, which is a technical standard from The Open Group (2012) based on the IEEE 1471 standard, we could fulfil this business requirement. Additionally, ArchiMEO is an RDF(S)/OWL ontology, which is a W3C semantic Web standard. Thus, it is possible to provide standardised representation and content as well. The most important contribution is the possibility to address different concerns of business users having different organisational roles in a CBR system.

9.   Conclusion and Future Work

The paper presenta a new approach to retrieve historic project knowledge considering the different viewpoints and concerns of employees working in different roles. The novelty is the ontology-based CBR approach that embeds the knowledge of the enterprise architecture description and enterprise models to present to the users, based on their concerns, tailored results.

Future work: As future work we plan to implement semi-automatic reuse (Adaptation OWL/Rule reasoning/inferencing and machine learning) and enhanced automatic retention (case learning and ontology learning, adding to elements to domain ontology, OWL/rule reasoning). Additionally, we plan to study the artefact in depth based on

a case study with the help of our application partner. Apart from that we plan to enhance the ontology-based CBR with natural language-processing technology to overcome incomplete case descriptions.

Acknowledgement

Funding

References

Aamodt, Agnar. 1991. "A Knowledge-Intensive, Integrated Approach to Problem Solving and Sustained Learning." Doctoral dissertation. University of Trondheim. Trondheim, Norway, Norway.

Aamodt, Agnar. 2004. "Knowledge-Intensive Case-Based Reasoning in CREEK." In *7th European Conference, ECCBR 2004, Madrid, Spain,* Vol. 3155 of *Lecture Notes in Computer Science* edited by Peter Funk and PedroA. González Calero. 1–15. Springer Berlin Heidelberg.

Aamodt, Agnar, and Enric Plaza. 1994. "Case-based reasoning: Foundational issues, methodological variations, and system approaches." *AI Communications* 7 (1): 39–59.

Abecker, A, A Bernardi, K Hinkelmann, O Kühn, and M Sintek. 1998. "Toward a Well-Founded Technology for Organizational Memories." *IEEE Intelligent Systems and their Applications* Vol.13 (No. 3): 40–48.

Aha, David W., Leonard A. Breslow, and Héctor Muñoz Avila. 2001. "Conversational Case-Based Reasoning." *Applied Intelligence* 14 (1): 9–32.

Allemang, Dean, and James A Hendler. 2008. *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL.* Morgan Kaufmann/Elsevier.

Amailef, Khaled, and Jie Lu. 2013. "Ontology-supported case-based reasoning approach for intelligent m-Government emergency response services." *Decision Support Systems* 55 (1): 79–97.

Assali, Amjad Abou, Dominique Lenne, and Bruno Debray. 2010. "Heterogeneity in Ontological CBR Systems." In *Successful Case-based Reasoning Applications - I SE - 5,* Vol. 305 of *Studies in Computational Intelligence* edited by Stefania Montani and LakhmiC. Jain. 97–116. Springer Berlin Heidelberg.

Bach, Kerstin, and Klaus-Dieter Althoff. 2012. "Developing Case-Based Reasoning Applications Using myCBR 3." In *Case-Based Reasoning Research and Development SE - 4,* Vol. 7466 of *Lecture Notes in Computer Science* edited by BelénDíaz Agudo and Ian Watson. 17–31. Springer Berlin Heidelberg.

Bello-Tomás, Juan José, Pedro A. González-Calero, and Belén Díaz-Agudo. 2004. "JColibri: An Object-Oriented Framework for Building CBR Systems." In *Advances in Case-Based Reasoning SE - 4,* Vol. 3155 of *Lecture Notes in Computer Science* edited by Peter Funk and PedroA. González Calero. 32–46. Springer Berlin Heidelberg.

Bergmann, Ralph. 1998. "On the Use of Taxonomies for Representing Case Features and Local

Similarity Measures." In *Proceedings of the Sixth German Workshop on CBR,* edited by Lothar Gierl and Mario Lenz.

Bergmann, Ralph. 2002. *Experience Management: Foundations, Development Methodology, and Internet-Based Applications.* Vol. 2432 of *Lecture Notes in Computer Science.* Berlin, Heidelberg: Springer Berlin Heidelberg.

Bergmann, Ralph, Klaus-Dieter Althoff, Sean Breen, Mehmet Göker, Michel Manago, Ralph Traphöner, and Stefan Wess. 2003. *Developing Industrial Case-Based Reasoning Applications.* Vol. 1612 of *Lecture Notes in Computer Science.* Berlin, Heidelberg: Springer Berlin Heidelberg.

Bergmann, Ralph, Klaus-Dieter KD Althoff, Mirjam Minor, Meike Reichle, and Kerstin Bach. 2009. "Case-Based Reasoning - Introduction and Recent Developments." *Künstliche Intelligenz: Special Issue on Case-Based Reasoning* 23 (1): 5–11.

Bergmann, Ralph, Andrea Freß mann, Kerstin Maximini, Rainer Maximini, and Thomas Sauer. 2006. "Case-Based Support for Collaborative Business." In *8th European Conference, ECCBR 2006 Fethiye, Turkey, September 4-7, 2006 Proceedings,* Vol. 4106 of *Lecture Notes in Computer Science* edited by ThomasR. Roth-Berghofer, MehmetH. Göker, and H.Altay Güvenir. 519–533. Springer Berlin Heidelberg.

Bergmann, Ralph, Janet L. Kolodner, and Enric Plaza. 2006. "Representation in case-based reasoning." *The Knowledge Engineering Review* 20 (03): 209–213.

Bergmann, Ralph, and Martin Schaaf. 2003. "Structural Case-Based Reasoning and Ontology-Based Knowledge Management: A Perfect Match?." *Journal of Universal Computer Science* 9 (7): 608–626.

Bichindaritz, Isabelle. 2004. "Mémoire: Case Based Reasoning Meets the Semantic Web in Biology and Medicine." In *Advances in Case-Based Reasoning SE - 5,* Vol. 3155 of *Lecture Notes in Computer Science* edited by Peter Funk and PedroA. González Calero. 47–61. Springer Berlin Heidelberg.

Bogaerts, Steven, and David Leake. 2005. *IUCBRF: A framework for rapid and modular CBR system development.* Tech. rep.. Technical Report TR 617, Computer Science Department, Indiana University, Bloomington, IN.

Bouhana, Amna, Amir Zidi, Afef Fekih, Habib Chabchoub, and Mourad Abed. 2015. "An ontology-based CBR approach for personalized itinerary search systems for sustainable urban freight transport." *Expert Systems with Applications* 42 (7): 3724–3741.

Brun, Roman. 2010. "Linked Enterprise Models and Objects representing Content and Context of an Enterprise." Master thesis. University of Applied Sciences Northwestern Switzerland FHNW.

Bulu, Hakan, Adil Alpkocak, and Pinar Balci. 2012. "Ontology-based mammography annotation and Case-based Retrieval of breast masses." *Expert Systems with Applications* 39 (12): 11194–11202.

Chen, Huajun, and Zhaohui Wu. 2003. "On case-based knowledge sharing in semantic Web." In *Proceedings. 15th IEEE International Conference on Tools with Artificial Intelligence,* 200–207. IEEE Comput. Soc.

Cohen, William W., Pradeep D. Ravikumar, and Stephen E. Fienberg. 2003. "A Comparison of String Distance Metrics for Name-Matching Tasks." *Proceedings of IJCAI-03 Workshop on Information Integration on the Web* 73–78. 11.

DAquin, Mathieu, Jean Lieber, and Amedeo Napoli. 2005. "Decentralized Case-Based Reasoning for the Semantic Web." In *The Semantic Web ISWC 2005 SE - 13,* Vol. 3729 of *Lecture Notes in Computer Science* edited by Yolanda Gil, Enrico Motta, V.Richard Benjamins, and MarkA. Musen. 142–155. Springer Berlin Heidelberg.

Daz-Agudo, B, and PA González-Calero. 2001. "Knowledge Intensive CBR Made Affordable." In *Proceedings of the Workshop Program at the Fourth International Conference on Case-Based Reasoning,* .

Delir Haghighi, Pari, Frada Burstein, Arkady Zaslavsky, and Paul Arbon. 2013. "Development and evaluation of ontology for intelligent decision support in medical emergency management for mass gatherings." *Decision Support Systems* 54 (2): 1192–1204.

Díaz-Agudo, Belén, and Pedro A. González-Calero. 2000. "An Architecture for Knowledge Intensive CBR Systems." In *Advances in Case-Based Reasoning,* Vol. 1898 of *Lecture Notes in Computer Science* edited by Enrico Blanzieri and Luigi Portinale. 37–48. Berlin, Heidelberg:

Springer Berlin / Heidelberg. Jan..

Ehrig, Marc, Peter Haase, Mark Hefke, and Nenad Stojanovic. 2004. "Similarity for Ontologies - a Comprehensive Framework." In *In Workshop Enterprise Modelling and Ontology: Ingredients for Interoperability, at PAKM 2004,* .

El-Sappagh, Shaker, Mohammed Elmogy, and A M Riad. 2015. "A fuzzy-ontology oriented case-based reasoning framework for semantic diabetes diagnosis.." *Artificial intelligence in medicine* 65 (3): 179–208.

Emmenegger, Sandro, Emanuele Laurenzini, and Barbara Thönssen. 2012. "Improving supply-chain-management based on semantically enriched risk descriptions." In *Proceedings of 4th Conference on Knowledge Management and Information Sharing (KMIS2012),* .

ESPRIT Consortium AMICE, ed. . 1989. *Open System Architecture for CIM.* Berlin, Heidelberg: Springer Berlin Heidelberg.

Fidjeland, Mikael Kirkeby. 2006. "Distributed Knowledge in Case-Based Reasoning : Knowledge Sharing and Reuse within the Semantic Web." Master thesis. Norwegian University of Science and Technology. http://urn.kb.se/resolve?urn=urn:nbn:no:ntnu:diva-8696.

Fox, Mark S, Mihai Barbuceanu, and Michael Gruninger. 1996. "An organisation ontology for enterprise modeling: Preliminary concepts for linking structure and behaviour." *Computers in Industry* 29 (1-2): 123–134.

Fox, Mark S., Mihai Barbuceanu, Michael Gruninger, and Jinxin Lin. 1998. "An Organisation Ontology for Enterprise Modeling." In *Simulating Organizations: Computational Models of Institutions and Groups,* edited by M. Prietula, K. Carley, and L. Gasser. 131–152. Menlo Park CA: AAAI/MIT Press.

Gao, Junjie, and Guishi Deng. 2010. "Semi-automatic Construction of Ontology-based CBR System for Knowledge Integration." *International Journal of Electrical and Electronics Engineering* 297–303.

Gruber, Thomas R. 1993. "A Translation Approach to Portable Ontology Specifications." *Knowledge Acquisition* 5 (2): 199–221.

Hefke, Mark. 2004. "A framework for the successful introduction of KM using CBR and semantic web technologies." *Journal of Universal Computer Science* 10 (6): 731–739.

Hefke, Mark. 2008. "Ontologiebasierte Werkzeuge zur Unterstützung von Organisationen bei der Einführung und Durchführung von Wissensmanagement." Ph.D. thesis. Universität Fridericiana zu Karlsruhe. http://digbib.ubka.uni-karlsruhe.de/volltexte/1000009186.

Hefke, Mark, and Andreas Abecker. 2006a. "A CBR-Based Approach for Supporting Consulting Agencies in Successfully Accompanying a Customer's Introduction of Knowledge Management." In *8th European Conference, ECCBR 2006 Fethiye, Turkey, September 4-7, 2006 Proceedings,* 534–548.

Hefke, Mark, and Andreas Abecker. 2006b. "KMIR - a Knowledge Management Implementation and Recommendation Framework using CBR and Semantic Web Technologies." In *CBR and Semantic Web Technologies, Proc. MKWI-2006,* .

Hefke, Mark, Valentin Zacharias, Andreas Abecker, and Qingli Wang. 2006. "An extendable java framework for instance similarities in ontologies." In *ICEIS,* Vol. 123–27.

Hevner, A.R., S.T. March, Jinsoo Park, and Sudha Ram. 2004. "Design science in information systems research." *Mis Quarterly* 28 (1): 75–105.

Hinkelmann, Knut, Aurona Gerber, Dimitris Karagiannis, Barbara Thoenssen, Alta van der Merwe, and Robert Woitsch. 2015. "A new paradigm for the continuous alignment of business and IT: Combining enterprise architecture modelling and enterprise ontology." *Computers in Industry* .

Hinkelmann, Knut, Emanuela Merelli, and Barbara Thönssen. 2010. "The Role of Content and Context in Enterprise Repositories." In *2nd International Workshop on Advanced Enterprise Architecture and Repositories - AER 2010,* .

ISO/IEC/IEEE. 2011. "ISO/IEC/IEEE 42010: Conceptual Model." http://www.iso-architecture.org/42010/cm/.

Ju, Keyi, Bin Su, Dequn Zhou, and Yuqiang Zhang. 2016. "An incentive-oriented early warning system for predicting the co-movements between oil price shocks and macroeconomy." *Applied Energy* 163: 452–463.

Kamsu-Foguem, Bernard, and Daniel Noyes. 2013. "Graph-based reasoning in collaborative

knowledge management for industrial maintenance." *Computers in Industry* 64 (8): 998–1013.

Kang, Dongwoo, Jeongsoo Lee, Sungchul Choi, and Kwangsoo Kim. 2010. "An ontology-based Enterprise Architecture." *Expert Systems with Applications* 37 (2): 1456–1464.

Kolodner, Janet L. 1993. *Case-based reasoning.* San Mateo, CA: Morgan Kaufmann Publishers.

Lankhorst, Marc. 2009. *Enterprise Architecture at Work.* Berlin, Heidelberg: Springer Berlin Heidelberg.

Lankhorst, M.M., H.A. Proper, and H. Jonkers. 2009. "The Architecture of the ArchiMate Language." In *Enterprise, Business-Process and Information Systems Modeling,* Vol. 29 of *Lecture Notes in Business Information Processing* edited by Terry Halpin, John Krogstie, Selmin Nurcan, Erik Proper, Rainer Schmidt, Pnina Soffer, Roland Ukor, Wil Aalst, John Mylopoulos, Michael Rosemann, Michael J Shaw, and Clemens Szyperski. 367–380. Springer Berlin Heidelberg.

Leake, David B. 1996. "CBR in Context: The Present and Future." In *Case-Based Reasoning: Experiences, Lessons, and Future Directions,* edited by David B. Leake. 1–35. Menlo Park: AAAI Press/MIT Press.

Leppänen, Mauri. 2007. "A Context Based Enterprise Ontology." In *10th International Conference on Business Information Systems (BIS 2007),* edited by Abramowicz W.. 273–286. Poznan, Poland: Springer Berlin / Heidelberg.

López, Beatriz, Carles Pous, Pablo Gay, Albert Pla, Judith Sanz, and Joan Brunet. 2011. "eXiT*CBR: A framework for case-based medical diagnosis development and experimentation." *Artificial Intelligence in Medicine* 51 (2): 81–91.

Manenti, Lorenza, and Fabio Sartori. 2010. "Exploiting CReP for Knowledge Retrieval and Use in Complex Domains." *Metadata and Semantic Research* 160–171.

Martin, Andreas. 2010. "Linked Enterprise Models and Objects providing Context and Content for creating Metadata." Master thesis. University of Applied Sciences Northwestern Switzerland FHNW.

Martin, Andreas, Sandro Emmenegger, and Gwendolin Wilke. 2013. "Integrating an enterprise architecture ontology in a case-based reasoning approach for project knowledge." In *Proceedings of the First International Conference on Enterprise Systems: ES 2013,* 1–12. IEEE. Nov..

Matthes, Dirk. 2011. *Enterprise Architecture Frameworks Kompendium.* Xpert.press. Berlin, Heidelberg: Springer Berlin Heidelberg.

Maximini, Kerstin, and Rainer Maximini. 2007. *Collaborative Agent-Based Knowledge Engine.* Tech. rep.. Universität Trier / Lehstuhl für Wirtschaftsinformatik II.

McBride, Brian. 2001. "Jena: Implementing the RDF model and syntax specification." In *Proceedings of the 2nd International Workshop on the Semantic Web.*, Hongkong.

Métrailler, Alexandre. 2011. "Integration of change and evolution in the lifecycle of SMEs information systems." In *Proceedings of the CAiSE Doctoral Consortium 2011,* edited by Cornelia Boldyreff, Shareeful Islam, Michel Leonard, and Bernhard Thalheim.

Object Management Group. 2008. "Business Motivation Model." http://www.omg.org/spec/BMM/1.0/PDF.

OMG. 2011. "Business Process Model and Notation (BPMN)." http://www.omg.org/spec/BPMN/2.0.

Peffers, Ken, Tuure Tuunanen, Marcus A. Rothenberger, and Samir Chatterjee. 2008. "A Design Science Research Methodology for Information Systems Research." *Journal of Management Information Systems* 24 (3): 45–77.

Pla, Albert, Beatriz López, Pablo Gay, and Carles Pous. 2013. "eXiT*CBR.v2: Distributed case-based reasoning tool for medical prognosis." *Decision Support Systems* 54 (3): 1499–1510.

Plaza, Enric. 1995. "Cases as terms: A feature term approach to the structured representation of cases." In *First International Conference, ICCBR-95 Sesimbra, Portugal, October 2326,* Vol. 1010 of *Lecture Notes in Computer Science* edited by Manuela Veloso and Agnar Aamodt. 265–276. Springer Berlin Heidelberg.

Potes Ruiz, Paula Andrea, Bernard Kamsu-Foguem, and Daniel Noyes. 2013. "Knowledge reuse integrating the collaboration from experts in industrial maintenance management." *Knowledge-Based Systems* 50: 171–186.

Recio-García, Juan A, Belén Díaz-Agudo, and Pedro González-Calero. 2008. "jCOLIBRI2 Tutorial." .

Recio-García, Juan A., Pedro A. González-Calero, and Belén Díaz-Agudo. 2014a. "jcolibri2: A framework for building Case-based reasoning systems." *Science of Computer Programming* 79 (Group 910494): 126–145.

Recio-García, Juan A., Pedro A. González-Calero, and Belén Díaz-Agudo. 2014b. "Template-Based Design in COLIBRI Studio." *Information Systems* 40: 168–178.

Recio-Garía, Juan A., and Belén Díaz-Agudo. 2007. "Ontology based CBR with jCOLIBRI." In *Proceedings of AI-2006, the Twenty-sixth SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence,* 149–162. Springer-Verlag London.

Richter, Michael M. 1998. "Introduction." In *Case-Based Reasoning Technology SE - 1,* Vol. 1400 of *Lecture Notes in Computer Science* edited by Mario Lenz, Hans-Dieter Burkhard, Brigitte Bartsch-Spörl, and Stefan Wess. 1–15. Springer Berlin Heidelberg.

Ross, Jeanne W., Peter Weill, and David Robertson. 2006. *Enterprise architecture as strategy: creating a foundation for business ….* Harvard Business Press.

Roth-Berghofer, Thomas, and Daniel Bahls. 2008. "Explanation Capabilities of the Open Source Case-Based Reasoning Tool myCBR." In *UK Workshop on Case-Based Reasoning UKCBR 2008,* 23–34. http://mycbr-project.net/downloads/ukcbr08.pdf.

Roth-Berghofer, Thomas, Juan Antonio Recio Garcia, Christian Severin Sauer, Kerstin Bach, Klaus-Dieter Althoff, Belen Diaz-Agudo, and Pedro A. González Calero. 2012. "Building Case-based Reasoning Applications with myCBR and COLIBRI Studio." In *Proceedings of the 17th UK Workshop on Case-Based Reasoning,* edited by Miltos Petridis, Thomas Roth-Berghofer, and Nirmalie Wiratunga. 71–82. School of Computing, Engineering and Mathematics, University of Brighton, UK.

Sein, Maung, Ola Henfridsson, Sandeep Purao, Matti Rossi, and Rikard Lindgren. 2011. "Action design research." *MIS Quarterly* 35 (1): 37–56.

Sene, A., B. Kamsu-Foguem, and P. Rumeau. 2015. "Telemedicine framework using case-based reasoning with evidences." *Computer Methods and Programs in Biomedicine* 121 (1): 21–35.

Shen, Ying, Joël Colloc, Armelle Jacquet-Andrieu, and Kai Lei. 2015. "Emerging medical informatics with case-based reasoning for aiding clinical decision in multi-agent system.." *Journal of biomedical informatics* 56: 307–317.

Stahl, Armin, and Thomas R. Roth-Berghofer. 2008. "Rapid Prototyping of CBR Applications with the Open Source Tool myCBR." In *9th European Conference, ECCBR 2008,* Vol. 5239 of *Lecture Notes in Computer Science* edited by Klaus-Dieter Althoff, Ralph Bergmann, Mirjam Minor, and Alexandre Hanft. 615–629. Springer Berlin Heidelberg.

Studer, Rudi, Richard V. Benjamins, and Dieter Fensel. 1998. "Knowledge engineering: Principles and methods." *Data & Knowledge Engineering* 25 (1-2): 161–197.

The Open Group. 2009a. "ArchiMate 1.0 Specification." http://pubs.opengroup.org/architecture/archimate-doc/ts_archimate/.

The Open Group. 2009b. *The Open Group Architecture Framework (TOGAF) Version 9.* Van Haren Publishing.

The Open Group. 2012. "ArchiMate 2.0 Specification." http://pubs.opengroup.org/architecture/archimate2-doc/.

Thönssen, Barbara. 2010. "An Enterprise Ontology Building the Bases for Automatic Metadata Generation." In *Proceedings of the 4th International Conference on Metadata and Semantics, MTSR1200,* 195–210. Madrid.

Thönssen, Barbara, and Jonas Lutz. 2012. "Semantically enriched obligation management." In *Proceedings of 4th Conference on Knowledge Management and Information Sharing (KMIS2012),* .

Uschold, Mike, Martin King, Stuart Moralee, and Yannis Zorgios. 1998. "The Enterprise Ontology." *The Knowledge Engineering Review* 13 (1): 31–89.

Vaishnavi, Vijay, and Bill Kuechler. 2004. "Design Research in Information Systems." http://www.webcitation.org/6BTaPrr1j.

van Heijst, G, A.Th. Schreiber, and BJ Wielinga. 1997. "Using explicit ontologies in KBS development." *International Journal of Human-Computer Studies* 46 (2-3): 183–292.

W3C. 2012a. "OWL 2 Web Ontology Language." http://www.w3.org/TR/owl2-overview/.

W3C. 2012b. "Vocabularies." http://www.w3.org/standards/semanticweb/ontology http://www.webcitation.org/6LubPal3Q.

W3C. 2014a. "RDF 1.1 Concepts and Abstract Syntax." http://www.w3.org/TR/rdf11-concepts/.

W3C. 2014b. "RDF 1.1 Turtle." http://www.w3.org/TR/turtle/.

W3C. 2014c. "RDF Schema 1.1." http://www.w3.org/TR/rdf-schema/.

Wang, Yinglin, Tao Hu, and Shensheng Zhang. 2003. "Ontology-based reconfigurable case-based reasoning system for knowledge integration." In *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme - System Security and Assurance (Cat. No.03CH37483),* Vol. 54878–4883. IEEE.

Watson, I. 1999. "Case-based reasoning is a methodology not a technology." *Knowledge-Based Systems* 12 (5-6): 303–308.

Weber, Rosina O., Kevin D. Ashley, and Stefanie Bruninghaus. 2006. "Textual case-based reasoning." *The Knowledge Engineering Review* 20 (03): 255.

Wheeler, R., and S. Aitken. 2000. "Multiple algorithms for fraud detection." *Knowledge-Based Systems* 13 (2-3): 93–99.

Yokoyama, T. 1990. "An object-oriented and constraint-based knowledge representation system for design object modeling." In *Sixth Conference on Artificial Intelligence for Applications,* 146–152. IEEE Comput. Soc. Press.

Zachman, John A. 1987. "A framework for information systems architecture." *IBM Systems Journal* 26 (3).

Zachman, John A. 2008. "John Zachman's Concise Definition of the Zachman Framework." .