FP7 ICT STREP Project

Deliverable D5.3

Experience Knowledge Mechanisms and Representation

http://www.learnpad.eu

Project Number	:	FP7-619583
Project Title	:	Learn PAd Model Based Social Learning for Public Administrations

Deliverable Number	:	D5.3
Title of Deliverable	:	Experience Knowledge Mechanisms and Representation
Nature of Deliverable	:	Report
Dissemination level	:	Public
License	:	Creative Commons Attribution 3.0 License
Version	:	4.0
Contractual Delivery Date	:	31 January 2016
Actual Delivery Date	:	29 January 2016
Contributing WP	:	WP5
Editor(s)	:	Barbara Thönssen (FHNW), Frieder Witschel (FHNW), Knut Hinkelmann (FHNW), Andreas Martin (FHNW)
Author(s)	:	Barbara Thönssen (FHNW), Sandro Emmenegger (FHNW), Frieder Witschel (FHNW), Emanuele Laurenzi (FHNW), Andreas Martin (FHNW), Congyu Zhang (FHNW), Jean Simard (XWIKI), Nesat Efendioglu (BOC)
Reviewer(s)	:	Sarah Zribi (Linagora), Antonia Bertolino (CNR)

Abstract

Deliverable D5.3 focuses on creating a knowledge base repository consisting of cases and a similarity based semantic case retrieval mechanism. We studied thoroughly the state-of-the-art of experience knowledge mechanism and representation. We did in depth domain analysis with Marche Region to identify requirements for experience management in Public Administrations, particularly in the Titolo Unico process. This reconfirmed that Case Based Reasoning (CBR) is an appropriate approach for experience management as it reflects extremely well the way individuals use former, i.e. existing experience knowledge to solve problems at hand.

Together with Marche Region we determined case descriptions and case content. In order to generalize the approach we enhanced case metadata by a standard of the National Insititute for Statistics (ISTAT).

For implementation we built upon research on ontology based CBR performed in a Swiss national project. For the Learn PAd project the core CBR component of the FHNW ICEBERG Toolkit was reused and adapted to meet the specific requirements and to fit into the Learn PAd platform, i.e. into the ontology and recommender component. The Learn PAd ontology was enhanced by CBR concepts and similarity functions.

For evaluation we inserted 12 former cases in the case base, formalized as instances in the Learn PAd ontology. Representatives of Marche Region created a new (fictitious) case and determined manually the three most similar cases from the case base. The similar cases suggested by the CBR were then compared to them. Based on the result we improved the similarity measures (e.g. weights of attributes) and did a second run. With the newly derived weight vector the ranking that the expert expected / recommended was achieved. This confirms the suitability of our similarity model and similarity functions but to avoid the risk of overturning it needs subsequent plausibility check with the expert. After this early evaluation, focussing on the achieved quality of recommended cases further evaluations will be done through simulation and the comprehensive demonstrator assessment.

Keyword list

Case Based Reasoning, CBR cycle, Similarity-based Semantic Case Retrieval Mechanism, Ontology, Ontology-based CBR, Similarity Models, Similarity Functions, Similar Cases, Adaptation Models, Characterisation Space, Characterisation Elements, CBR Configuration and data import.

Document History

Version	Changes	Author(s)
0.1	First Draft	Barbara Thönssen (FHNW), Sandro Emmenegger (FHNW), Frieder Witschel (FHNW), Emanuele Laurenzi (FHNW), Andreas Martin (FHNW), Knut Hinkelmann (FHNW), Congyu Zhang (FHNW)
0.2	Layout	Barbara Thönssen (FHNW), Emanuele Laurenzi (FHNW)
0.3	Improved styles	Emanuele Laurenzi (FHNW)
0.4	Internal review updates	Emanuele Laurenzi (FHNW), Frieder Witschel (FHNW), Barbara Thönssen (FHNW), Andreas Martin (FHNW), Sandro Emmenegger (FHNW)
1.0	Finalization	Barbara Thönssen (FHNW), (FHNW), Frieder Witschel (FHNW), Emanuele Laurenzi (FHNW), Jean Simard (XWIKI), Nesat Efendioglu (BOC)
2.0 - 3.0	Improvements	Barbara Thönssen (FHNW), (FHNW), Frieder Witschel (FHNW), Emanuele Laurenzi (FHNW)
4.0	Submission	Barbara Thönssen (FHNW)

Document Review

Release	Date	Ver.	Reviewers	Comments
ToC	2 December 2015	Ver 0.1		
Draft	27 December 2015	Ver 1.1		
Internal	10 January 2015	Ver 2.1	Sarah Zribi, Antonia Bertolino	
Candidate Final	27 January 2015	Ver 3.1	Antonia Bertolino	

Glossary, acronyms & abbreviations

Item	Description
ArchiMEO	Enterprise Ontology developed by FHNW based on ArchiMate Standard and enhanced for Learn PAd
BP	Business Process
BPMN	Business Process Model and Notation
Browsing	Learning mode that allows navigating through models
CBR	Case Based Reasoning
FHNW	University of Applied Sciences and Arts Northwestern Switzerland
Learning Material	All entities not representing models but relevant for learning as books, tutorials, learning audio and video file but also browsing and simulation
Learning Objects	All models represented in the wiki are considered learning objects
MOP	Memory Organization Pocket
MR	Marche Region
OWL	Ontology Web Language
PA	Public Administration
SPARQL	SPARQL is a W3C recommendation defines the syntax and semantics of the SPARQL query language for RDF.
SPIN	SPIN is a W3C Member Submission that has become the de-facto industry standard to represent SPARQL rules and constraints
SUAP	Sportello Unico Attivitá Produttive
Titolo Unico	'Titolo unico' means in English standard request to start business activity; here used to name the business process considered
UC	Use Case
UML	Unified Modeling Language
W3C	World Wide Web Consortium

Table of Contents

1. Introduction **9**

1.1. Experience Management Mechanism and Representation – Motivation *9*

1.2. Methodology *10*

1.3. Structure of the Deliverable *11*

2. Related Work **12**

2.1. Case-based Reasoning *12*

2.1.1. General Case Structure *13*

2.1.2. CBR Knowledge Containers *14*

2.2. CBR Approaches and Research Directions *15*

2.3. Ontology-based CBR *17*

2.4. Case-based Reasoning and Processes *18*

3. Domain Analysis **20**

4. CBR for Learn PAd **25**

4.1. Case Characterisation and Case Content *25*

4.2. Similarity Model *30*

4.3. Adaptation Model *36*

4.4. User Interface *39*

5. Technical Architecture and Implementation **41**

5.1. Implementation of CBR *41*

5.1.1. Formalization of Case Characterisation and Content *42*

5.1.2. CBR Lifecycle Management *43*

5.2. CBR Integration into Learn PAd System *45*

6. Evaluation **48**

6.1. Early Evaluation *48*

6.2. Simulation and Collaborative Workspace *53*

7. Conclusions & Future Work **54**

List of Figures

Figure 1: CBR reuse principle (adapted from Richter & Weber (2013)) 13
Figure 2: The CBR cycle (adapted from Aamodt & Plaza (1994)) 13
Figure 3: CBR Knowledge Containers (Bergmann & Schaaf 2003) 14
Figure 4: CBR Knowledge Containers (Bergmann & Schaaf 2003) 15
Figure 5: Titolo Unico process 20
Figure 6: Outcome 1st Workshop: Case Template 22
Figure 7: Comparison of a query case with historical cases 30
Figure 8: Taxonomic structure of the value range of attribute "application type" 32
Figure 9: Taxonomy with similarity values assigned to inner nodes 33
Figure 10: The adaptation process in CBR (Leake 1996) 37
Figure 11: User dialogue for testing adaptation hypotheses 39
Figure 12: Characterisation elements user interface 40
Figure 13: ICEBERG implementation (adapted from Martin et al. (2013)) 41
Figure 14: ICEBERG ontology structure (adapted from Martin et al. (2013)) 42
Figure 15: Application Case Characterisation 43
Figure 16: Case lifecycle management frontend 44
Figure 17: Case adaption 45
Figure 18: CBR component integration 45
Figure 19: Learn PAd simulation component integration 46
Figure 20: Event paramters of the simulation component 47

List of Tables

Table 1: Case characterisation 25
Table 2: Super-sub-concept relations of CBR-specific concepts 27
Table 3: Object properties of concepts 28
Table 4: Data type properties of concepts 29
Table 5: Case content 30
Table 6: Some historical and new (query) cases with their values of the “Zone” attribute 34
Table 7: Weights and functions to determine similarity of concepts 35
Table 8: Example of characteristics of a new case 38
Table 9: Characteristics of a retrieved (similar) case 38
Table 10: Characterisation elements of the fictitious case 49
Table 11: Results of two retrieval runs with the Learn PAd CBR system 52

1. Introduction

As defined in the Description of Work, Deliverable D5.3 will describe the experience knowledge base repository consisting of cases and a similarity-based semantic case retrieval mechanism. The deliverable describes how cases need to be characterised in order to capture the relevant information of an example business process (SUAP Titolo Unico) and which underlying ontological concepts and relationships are needed – as derived from an in-depth study of that application scenario. From the case study, we derive special characteristics of the application of experience management in business processes of public administrations and identify corresponding research questions. The document finally discusses how case similarity should be assessed and how users will interact with the Learn PAd system to retrieve, adapt and re-use historical cases.

1.1. Experience Management Mechanism and Representation – Motivation

Learn PAd aims at supporting holistic learning solutions for the Public Administrations, providing collaborative workplace learning centered on Business Processes (BPs) and their context.

In Deliverable D5.2[1], we have described how the context of a learner – comprising both the business process and the organizational context – and his/her level of competence can be used to provide context-sensitive recommendations of suitable learning objects. Such recommendations are based on context-sensitive rules which explicitly codify knowledge: “under which circumstances can a new regulation be applied?”, “which competence level is needed to understand a tutorial?” Recommended learning materials can be studied by the learner and will help to solve problems and make decisions.

However, sometimes it is not easy to construct rules, i.e. to make the knowledge explicit. This might be due e.g. to a high number of potential exceptional situations which cannot be captured by a reasonably small set of rules. In such situations, human judgment is needed and must be based on experience and the ability to transfer old solutions to new situations.

Many civil servants possess a lot of experience – but there are at least two reasons why further support is needed: firstly, civil servants might need support in recalling the experiences that they made – such support can make e.g. the retrieval of related information more *efficient*; secondly, less experienced civil servants should be able to profit from the experience of “veterans” such that they can work more efficiently and effectively – where effectiveness comprises also the ability to make decisions that are *consistent* with previous decisions made within the same public administration.

Case-based reasoning (CBR) is a proven way of documenting, retrieving and re-using experience. It supports knowledge workers in capturing the essential characteristics of a new situation (a case), in retrieving past cases where the situation was similar and in deriving and storing a solution in the current situation by adapting the solution(s) of the similar past case(s).

[1] Delivarable D5.2: Semantics for the Wiki.

In Learn PAd, cases correspond to instances of (parts of) business processes. In order to support civil servants in the re-use of experience from past process instances, we describe in this deliverable

- Which are the essential characteristics of a process instance that are relevant for finding past cases from which to adapt solutions. We develop such ***case characterisations*** for the SUAP Titolo Unico process that was described already in Deliverable D5.2[2]. Characterisations are based on the meta models and ontologies developed in WP3 and D5.1[3]. We also describe a model for the content that is used and created when a process is executed (***case content***)
- How to assess the ***similarity*** between a new case characterisation and historical ones. This allows to retrieve the cases that are most closely related to the current one and from which civil servants can most likely re-use experience.
- How users can be supported in ***adapting*** old solutions to a new situation.

We describe how requirements regarding these steps have been gathered with representatives of Public Administrations, i.e. Marche Region, and which special research questions have emerged due to using CBR in the specific context of business processes in Public Administrations.

1.2. Methodology

As for D5.2, for developing this deliverable we adopt the design science research methodology for information systems research (Hevner et al. 2004). Hence, the research design will follow the stages in the design science research methodology: problem awareness, suggestion, development and evaluation. In the following paragraphs each stage is described.

1. Awareness of Problem – in this phase, we performed a detailed domain analysis to understand which aspects are relevant for characterising cases, what impact these aspects should have on the assessment of similarity between cases and which mechanisms were required for adapting case solutions to new situations. Our analysis was based on a thorough study of relevant literature in the area of experience management and case-based reasoning, as well as on a series of interviews and workshops with SUAP officers from the Marche Region.
2. Suggestion – based on the findings in the problem awareness phase, we derive and describe the conceptual models that facilitate the implementation of a CBR system, namely
    a. The model for case characterisation and case content,
    b. The similarity model that defines how cases will be compared,
    c. The adaptation model that captures the interaction possibilities those civil servants can use to adapt old solutions to new situations.

[2] Deliveralbe D5.2: Semantics for the Wiki. http://wiki.learnpad.eu/LearnPAdWiki/bin/download/Deliverables/D5_2/LearnPAd_D5.2_V4.0.pdf

[3] Delivarable D5.1: Models for setting the Wiki. http://wiki.learnpad.eu/LearnPAdWiki/bin/download/Deliverables/D5_1/LearnPAd_D5.1_V4.0.pdf

3. Development – at this stage, we define the technical architecture of the CBR system, comprising its integration into the Learn PAd platform. A prototypical implementation has been done.
4. Evaluation – the solution will be fully evaluated in D8.4. We will cooperate with WP8 for a joint evaluation of the functionalities of the Learn PAd platform. In this deliverable, we report results from an early evaluation which was done by comparing results of case retrieval with the expectations of experts on a small case base.

1.3. Structure of the Deliverable

The deliverable is structured as follows: in Chapter 2, we provide a thorough overview of literature related to experience management and case-based reasoning; we focus particularly on approaches that use ontologies for characterisation of cases. Chapter 3 presents the findings of the domain analysis (problem awareness), with a focus on elicited requirements. We describe our conceptual solution – comprising case characterisation model, similarity model and adaptation model and user interface – in Chapter 4 and summarize the corresponding technical architecture in Chapter 5. Chapter 6 discusses some findings of a preliminary evaluation before Chapter 7 concludes and indicates directions for future research.

2. Related Work

In the following sections we provide findings from literature regarding case-based reasoning, tools and frameworks. Additionally we will describe how ontology can be useful for CBR.

2.1. Case-based Reasoning

Case-based reasoning (CBR) can be seen as Leake (Leake 1996) defines “reasoning by remembering” (Leake 1996) or “reasoning from reminding” (Madhusudan et al. 2004) and as a technical independent methodology (Watson 1999) to humans and information systems. *“Case-based reasoning is both […] the ways people use cases to solve problems and the ways we can make machines use them”* (Kolodner 1993). CBR can be seen according to Aamodt and Plaza (Aamodt 1994) as “[…]a recent approach to problem solving and learning”.

Case-based reasoning is strongly influenced by cognitive science – the original idea was derived from the results of several studies concerning the human brain (Lopez de Mantaras et al. 2006). Schank and Abelson (Schank & Abelson 1977) laid a foundation for further case-based reasoning studies by their studies of how humans understands stories and how the memory affects the understanding of certain stories. Schank (Schank 1982) introduced the concept of memory organization packets (MOP’s). These MOP’s tries to explain how humans organize individual scenes in live that are linked to other MOP’s and can be linked to specific context or major goal. Schank (Schank 1999) came up with a revised definition of MOP’s as follows: *“A MOP consists of a set of scenes directed toward the achievement of a goal. A MOP always has one major scene whose goal is the essence or purpose of the events organized by the MOP”.* In order to use these MOP’s as a source for reminding and adapting the memories to a new situation, Schank (Schank 1999) argued that *“[…] there must be structures that capture similarities between situations that occur in different domains”*. These structures are introduced by Schank (Schank 1999) as thematic organization packets (TOP’s) and contain abstract and domain-independent information. MOP’s are a basis for creating cases in a case-based reasoning approach (Lopez de Mantaras et al. 2006) – a person who creates cases in a case-based reasoning system tries to describe individual (set) scenes (MOP’s). A person is often able to use personal reminding’s (MOP’s) in a different situation by abstracting existing information using TOP’s – this as a relationship to case-based reasoning and a basic requirement to the CBR users and the approach as well.

Figure 1 depicts the basic reuse principle, where case-based reasoning (CBR) uses the specific knowledge of previously experienced situations (cases containing old problems and old solutions) in order to propose a solution (reuse) to a new situation (new problem) by comparing the new problem with old problems based on the similarity (Richter & Weber 2013; Aamodt 1994). This refers to the basic assumption of CBR that, "[…] *similar problems have similar solutions*" (Watson 2003). But this also refers to one of the main challenges in CBR: "*If similar problems have very different solutions, a case-based reasoner may give inaccurate advice*" (Watson 2003). Using the traditional CBR terminology a case consists of a *problem space* (problem items / descriptions) that is used for describing a certain *solution space* (solution items) (Bergmann 2002). “*A case-based reasoner solves new problems by adapting solutions that were used to solve old problems*” (Riesbeck & Schank 1989). Apart from the reuse of these historical cases, case-based reasoning also provides the “learning of new cases”.

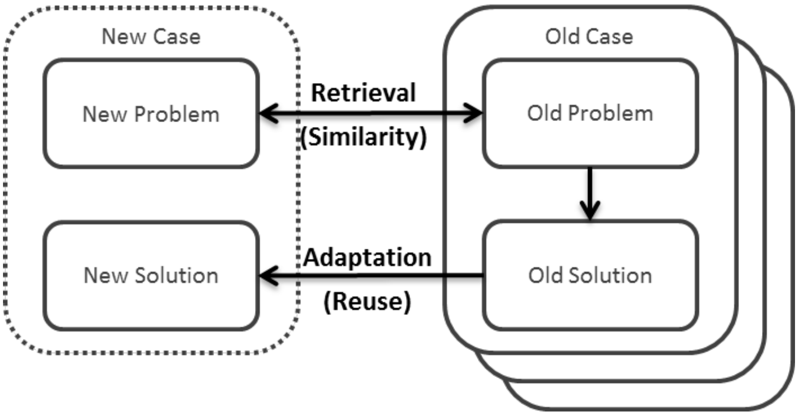

Figure 1: CBR reuse principle (adapted from Richter & Weber (2013))

Based on Aamodt and Plaza (1994) and Madhusudan, Zhao and Marshall (2004) the generic CBR cycle (see Figure 2) consists of the following steps: *1. **Retrieve** the most similar cases from the knowledge base (case-base containing previous cases) based on the problem description of the new case (problem case) using a similarity mechanism. 2. **Reuse** the knowledge in the retrieved case(s) in order to solve the current problem – adapt the historical knowledge to the new problem (adaptation). 3. **Revise** and test the suggested solution e.g. by evaluating it under the real world problem (evaluation). 4. **Retain** useful experience (past solutions and failures) for future reuse and store a new case in the knowledge base (case learning).*

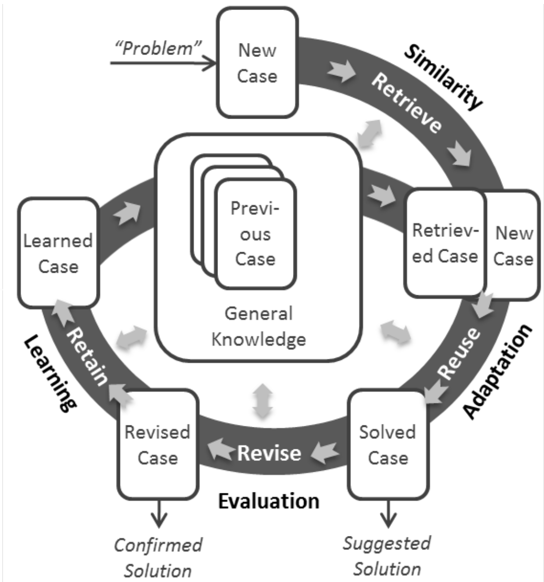

Figure 2: The CBR cycle (adapted from Aamodt & Plaza (1994))

2.1.1.General Case Structure

The basic idea of the 'case' concept is to capture information for problem solving as used in cognitive science (Bergmann, Kolodner, et al. 2006). Traditionally a CBR systems case description consists of a *problem* and a *solution* part. Bergmann (Bergmann 2002) extends the problem and solution view by: the *characterisation part* and the *lesson part*. "*The case characterisation part describes all facts about the experience that are relevant for deciding whether the experience can be reused in a certain situation*" (Bergmann 2002). The characterisation part contains elements that can be seen as index or metadata to the case. In contrast to regular index or metadata, the characterisation part is usually more detailed as it must contain the whole context of a case. In CBR the characterisation part can be seen as problem space. Bergmann (2002) extends this view by the characterisation part that can contain *“[…] derived descriptions or properties that were not present in the problem solving situations […]*”. The lesson part contains elements that are needed to describe the case in order to enable the *user to take actions* based on the suggestion. Based on (Secchi et al. 1999) several space agencies define a lesson or lesson learned as *“[…] knowledge or understanding gained by experience*” (Weber et al. 2000). In traditional CBR terminology the lesson part can be seen as a solution part. Bergmann (2002) extends this view by using the term lesson space that *“[…] can contain information that is not the solution itself but useful to find a solution*”.

2.1.2.CBR Knowledge Containers

The distribution of knowledge in a case-based reasoning system has been introduced by Richter (Richter 1995). In case-based reasoning it is possible to identify four containers of CBR knowledge (Richter 1998). These "*knowledge containers*" are identified as shown in Figure 3: the vocabulary, similarity measure, solution transformation and case base (Richter 1998).

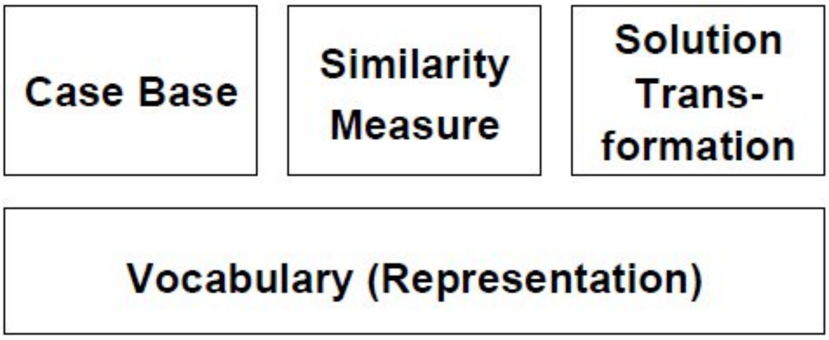

Figure 3: CBR Knowledge Containers (Bergmann & Schaaf 2003)

The *vocabulary* container contains the *background knowledge*, which is "*general and problem independent […]*" and when it *"[…] describes a specific part of the domain it is also called contextual knowledge*" (Richter 1998).

The cases themselves are captured in a *case base*, which is another knowledge container in case-based reasoning (Richter 1998). To retrieve cases in a CBR system, at least one similarity measurement is needed. "*A case base CB is a set of cases which, for retrieval purposes, is usually equipped with additional structures; structured case bases exist also under the name of case memory*" (Richter 1998).

Bergmann and Schaaf (Bergmann & Schaaf 2003) described the knowledge containers of a structural case-based reasoning system. In Figure 4 the elements are presented, which are relevant when describing (characterising) knowledge items (Bergmann & Schaaf 2003).

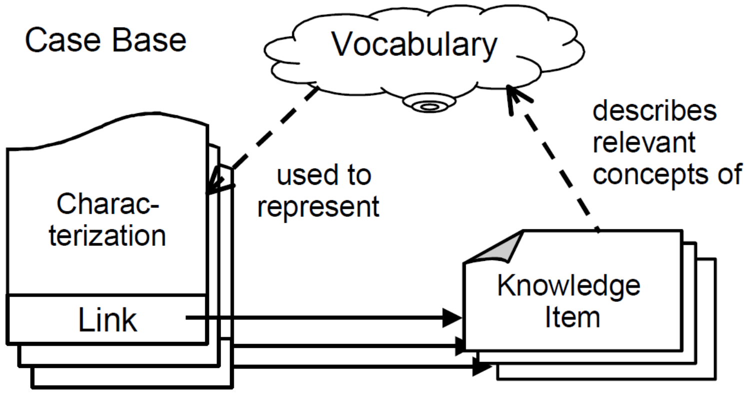

Figure 4: CBR Knowledge Containers (Bergmann & Schaaf 2003)

The *knowledge item* is linked to a case characterisation, which describes to knowledge item using the vocabulary knowledge container. As mentioned before, this vocabulary consists of concepts for describing the knowledge items (Bergmann & Schaaf 2003).

In order to retrieve similar cases at least one *similarity measure* is needed. According to Richter (Richter 1998), this similarity measure is another knowledge container in CBR. "A similarity measure is a container which can store more or less sophisticated knowledge about a problem class" (Richter 1998).

The *solution transformation* knowledge container contains knowledge that is used during adaptation of a retrieved experience item to a new situation. This solution transformation is sometimes called adaptation knowledge (Wilke et al. 1997; Richter 1995; Richter 1998).

2.2. CBR Approaches and Research Directions

It is possible to distinguish three main approaches in case-based reasoning (Bergmann et al. 2003):

1. *Textual CBR* (see e.g. Shimazu (Shimazu 1998) and Weber, Ashley, & Bruninghaus (Weber et al. 2006)) where the cases are recorded as or derived from free text.
2. *Conversational CBR* (see e.g. D. Aha & Breslow (Aha & Breslow 1997), D. Aha, Maney, & Breslow (Aha et al. 1998) and Aha, Breslow, & Muñoz-Avila (Aha et al. 2001)) where the case acquisition takes place in a conversational (dialog) manner.
3. *Structural CBR* (see e.g. Yokoyama (Yokoyama 1990) and Aamodt (Aamodt 1991)) where the cases are described by using a certain vocabulary or domain model (Bergmann 2002).

Bergmann (Bergmann 2002) made an analysis in order to compare the different approaches deeply. He also shares the experience of creating a case base in a company including the difficulties they might occur. Further he points out three main requirements and efforts that should be taken into account when implementing a case-based reasoning system:

1. Frist the *initial material* that is required to set up the case base.
2. Second the *effort to maintain* the case base.
3. And third the *effort to control the accuracy* of the case based reasoning system.

When implementing a case-based reasoning system, there exists usually no ideal situation in a company where the needed material is available, and is in that structure that is needed.

Most of the time a pre- processing step is needed – here the effort differs from one approach to another.

Based on the assessment of Bergmann (Bergmann 2002) it is possible to conclude that the structural approach is appropriate for *complex problem solving* in comparison to the textual and conversational approaches. The structural approach enables a more accurate retrieval as the textual approach and the maintenance effort is lower as it would needed by the conversational approach when scaling the case base up (Bergmann 2002).

As introduced, case-based reasoning can be basically seen as independent and generic method to humans and information systems. This generic method is the basis for several case-based reasoning applications and research themes. To get a general view on the extensive field of case-based reasoning themes, Greene et al. (Greene et al. 2008; Greene et al. 2010) did a research directions and trends analysis using a case-based reasoning conference literature corpus.

Major CBR Research Topics by co-citation: Greene et al. (Greene et al. 2008) identified the following then major research topics by co-citation view (year(s) of major publications):

1. Recommender systems and diversity (2000 - 2004).
2. Case-base maintenance (1995 - 2000).
3. Case retrieval (1991 - 2004).
4. Learning similarity measures (2000 - 2005).
5. Adaptation (1998 - 2005).
6. Image analysis (1995 - 2004).
7. Textual CBR (1997 - 2007).
8. Conversational CBR (1997 - 2005).
9. Feature weighting & similarity (1994 - 2004).
10. Creativity & knowledge-intensive CBR (1991 - 1997).

Promising CBR Research Trends by co-citation and discriminating terms: a more recent analysis of Greene et al. (Greene et al. 2010) based on co-citation and text-based clustering uncovered a number of potential CBR research themes, which might become more prominent in future (beside classical CBR themes: Recommender systems & diversity, textual CBR, case-base maintenance and conversational CBR).

- *Knowledge-intensive CBR* is a rather recent research theme in CBR and it is characterised as a CBR approach where domain knowledge is available and provided in advance (see e.g. Bergmann & Mougouie and Recio-García (Bergmann & Mougouie 2006) and González-Calero (Recio-García et al. 2012)).
- *Planning* in CBR refers to diverse planning tasks as e.g. route planning or project planning (see Mukkamalla & Muñoz-Avila (Mukkamalla & Muñoz-Avila 2002)).
- *Confidence* in CBR is a calculation that measures the accuracy of the solution (see Cheetham & Price (Cheetham & Price 2004)).
- *Tutoring systems* is an education-oriented research field that is more published outside of the CBR conferences (see Gómez-Martín et al. (Gómez-Martín et al. 2005)).

- *Explanation* in CBR is a post-retrieval step that elucidates the results of CBR, which can themselves be realized as CBR (see (Rissland 2006)).
- *CBR and music* deals with the scheduling of music e.g. in radio or performance (see e.g. (Baccigalupo & Plaza 2007)).
- *CBR in medicine* is not widely recognized as a research theme itself, but it is a significant application area (see e.g. (Marling & Whitehouse 2001) and Sene, Kamsu-Foguem, & Rumeau (2015)).

2.3. Ontology-based CBR

In order to reduce the knowledge acquisition bottleneck, it is advisable to provide the case-based reasoning system with domain knowledge beforehand. *"The more knowledge is embedded into the system, the more effective [it] is expected to be"* (Recio-García et al. 2008). This is exactly were ontologies can come into place. Ontologies can provide this knowledge. To use the power of ontologies in a Case-based reasoning system a combined ontology-based and case-based reasoning approach is needed. Ontology-based systems can benefit from structural CBR and vice versa as discussed in Bergmann and Schaaf (Bergmann & Schaaf 2003) and Bichindaritz (Bichindaritz 2004). Ontology-based CBR *"[…] can take advantage of this domain knowledge and obtain more accurate results"* (Recio-García et al. 2008).

There exist approaches ((Bello-Tomás et al. 2004), (Recio-García et al. 2012), (Roth-Berghofer & Bahls 2008), (Bach & Althoff 2012), (Dıaz-Agudo & González-Calero 2001), (Wang et al. 2003), (Dıaz-Agudo et al. 2005)) that implement a combined Ontology-based CBR and only a limited number of approaches that go beyond taxonomic CBR (Bergmann 1998) including properties / relations in ontologies. Such an approach has been introduced by Hefke (2004), which is part of the Knowledge Management Implementation and Recommendation (KMIR) Framework (Ehrig et al. 2004; Hefke & Abecker 2006a; Hefke & Abecker 2006b; Hefke et al. 2006; Hefke 2008). Chen & Wu (2003) introduced an rdf-based markup language for case-based reasoning called CaseML. As Fidjeland (2006) argued, CaseML defines only a small vocabulary with limited expressiveness compared to other approaches. Fidjeland (2006) introduced in his master thesis an OWL vocabulary for Creek. Creek is a case-based reasoning system introduced by Aamodt (2004;1991). Apart from the vocabulary, Fidjeland (2006) introduced a possibility for sharing the case base and the domain model using an OWL representation. Fidjeland (2006) used Jena (McBride 2001) and jCreek for implementing the proposed OWL vocabulary approach.

Bichindaritz (Bichindaritz 2004) introduced an case-based reasoning framework called *Mémoire* 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). Bichindaritz (Bichindaritz 2004) pointed out the benefit of such an ontology-based case-base and interchange possibility as follows: "*the perspective of unlimited cooperation between these systems is extremely promising for the improvement of healthcare and biomedical research […]*". Dıaz-Agudo and González-Calero (2001) proposed an ontology for describing case-based reasoning processes in a domain in depended way called CBROnto.

Martin, Emmenegger, & Wilke (Martin et al. 2013) introduced an ontology-based CBR approach for maintaining project knowledge, that utilizes an enterprise ontology. They introduce a novel case retrieval mechanism that emphasizes enterprise specific domain knowledge by reusing the enterprise ontology called ArchiMEO.

2.4. Case-based Reasoning and Processes

Case-based reasoning has been applied successfully in workflow and process environments. Madhusudan, Zhao and Marshall (2004), and Kim, Suh and Lee (2002) use CBR to support workflow design. The work of Kaster, Medeiros and Rocha (2005) uses CBR in combination with a decision support system. Weber and Wild (2005) enable ad-hoc modifications of workflows using CBR and Rinderle, Weber, Reichert and Wild (2005) proposed a CBR system enabling ad-hoc modifications of workflows using semantic information. Van der Aalst et al. (2005) proposed a case handling system using explicitly and implicitly structured cases. Wargitsch et al. 1997) proposed a system called WorkBrain. This system's idea is the elaboration of workflows using certain workflow elements or fragments – called workflow building blocks. In this approach CBR is used at the beginning to configure the instance of the workflow – it does not provide the flexibility to determine the process during runtime. Van Elst et al. (2003) describe a system, which has been elaborated during a research project called FRODO. In FRODO they introduce a task concept ontology for weakly-structured workflows. These workflows can be modified during run-time (instance level). Weber et al. (2004) introduced in their work a system called CBRFlow, which combines workflow execution and conversational case-based reasoning (CCBR). CBRFlow uses a case-based reasoning component to handle exceptions to business rules during run-time – it enables *"[…] modifications to a predefined workflow model and to provide incremental learning capabilities*" (Weber et al. 2004). Bergmann et al. (2006) and Maximini and Maximini (2007) described a case-based reasoning system 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.

Further notable CBR research directions and approaches related to processes and workflows are:

- *CBR for Workflow Retrieval*: Bergmann & Gil (2011) introduced an approach for retrieving existing workflows based on a query. In contrast to other approaches, this approach the workflows are graph based and semantically annotated (Bergmann & Gil 2011).
- *CBR for Workflow Adaptation & ad-hoc changes to workflows*: Minor et al. (2010) introduced an process oriented approach using case-based reasoning *"[…] to support the reuse of change experience*". The introduced approach acts as workflow enactment service, which supports the workflow modifications in terms of *"[…] ad-hoc changes in order to fulfill change request and late-modeling*" (Minor et al. 2010).
- *CBR for Workflow Construction*: the work of Leake & Kendall-Morwick (2008) is a way to support scientist that are creating scientific workflows *"[…] by suggesting additions to workflow designs under construction*" (Leake & Kendall-Morwick 2008). The approach of Leake & Kendall-Morwick (2008) is implemented as a plugin to the Xbaya software, which is a graphical modeler for scientific workflows.
- *CBR for Workflow Monitoring*: Kapetanakis et al. (2010) introduced a CBR based monitoring system of workflows. The system informs the process owner or managers about potential issues and gives advice how to deal with the issue.
- *CBR for Process Life Cycle Support*: Weber et al. (2009) developed an approach to support process-aware information systems (PAISs) using case-based reasoning. This ProCycle approach tracks *"[…] changes of individual process instances and the propagation of process type changes […]*" (Weber et al. 2009). Process participants

are able to retrieve and reuse contextual *"[…] knowledge about previously performed changes.*" (Weber et al. 2009).

- *CBR for Process Variant Management*: Cognini, Hinkelmann, & Martin (2015) introduced a case-based reasoning approach that supports the manual generation and refinement of generalized process variants using a process variant modeling language. This work is uses the approach of Martin, Emmenegger, & Wilke (2013) as underlying method and technology. Cognini, Hinkelmann, & Martin (2015) came to the conclusion that CBR is a valid method to support knowledge-intensive processes and workflows.

3. Domain Analysis

To fully understand the case where the CBR approach is to be implemented we did an in-depth domain analysis. Therefore we conducted interviews and workshops with representatives of Marche Region. From it, we derived requirements that are described in the subsequent sections.

To analyse the requirements for experience management we focus again on the Titolo Unico process. As described in detail in D5.2 the business process starts with a request of a citizen for opening a new business and ends with the acceptance or rejection of such. Again from the previous deliverable we know that the most challenging activity of this process is that of identifying all parties that should be involved in the decision (see red circle No 2 in Figure 5). The reason for that is mainly related to aspects like the broad Italian law as well as the high number and different types of potential parties to involve.

However, while conducting the first interview with PA experts, it became clear that for experience management the focus on this activity only would have been too narrow. In fact, there are further aspects that deserve attention when considering experience management. These belong to other activities such as assessing application (see red circle No 1 in Figure 5), which includes the aspect of dealing with mistakes occurring in an instance form (e.g. declarations that are in contrast to each other or missing documents that require further material from the entrepreneur, leading to time delays). A further activity is that of arranging a service conference (see red circle No 3 in Figure 5) that includes the aspect related to the behavior of an involved party (e.g. did it or did not attend the Service Conference). The aspect behavior is also extended to the reply to a request within a given time, which relates to the more challenging activity aforementioned (red circle No 2 of Figure 5). Hence, one aspect can span more than one activity.

Another aspect that takes place in experience management is the fact of accepting or not the activity request and the reason for the rejection. The latter is useful to the PA officer to improve the acceptance rate of next activity requests. This information is reported in the final transcript that is produced either after one (or more) service conference or after the request of opinions, which happen respectively in the activity with the red circle No 3 and the red circle No 2 of Figure 5.

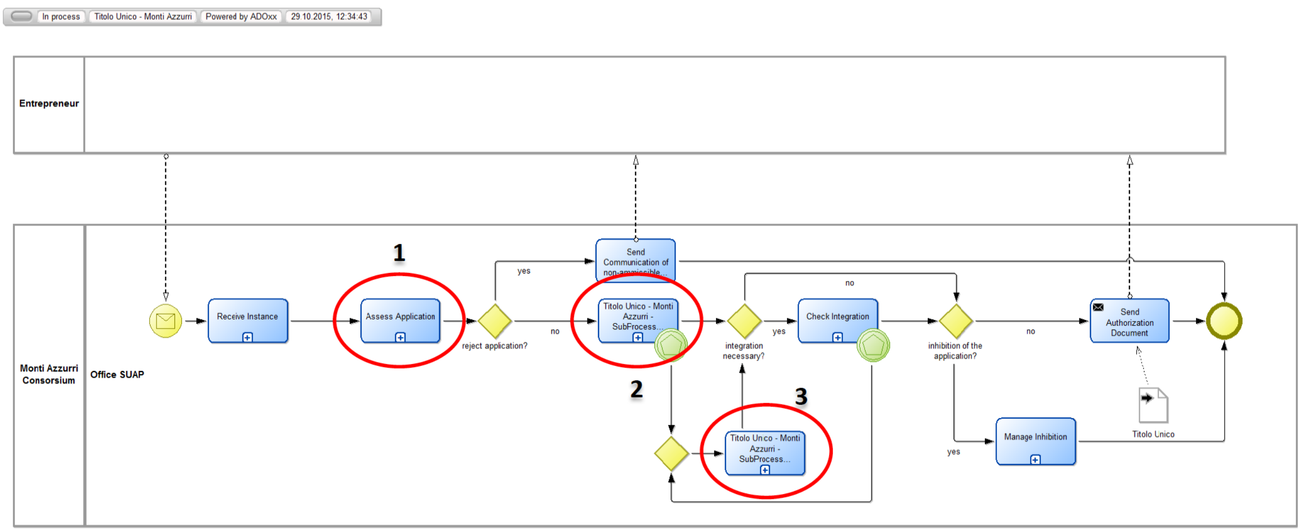

Figure 5: Titolo Unico process

Therefore, interviewed PA experts emphasized the importance of experience management, i.e. of considering former cases to find advice for handling new ones. Up to now, some PA offices are equipped with a web-based application that offers a keywords-based query functionality to retrieve former cases. However, this might be of little help if a PA officer is not experienced as he/she doesn't know what to search for and which of the former cases retrieved are similar to the new case.

We addressed this issue as we created a case template where the main aspects along the Titolo Unico process were taken into account for experience management, i.e. a template containing attributes - or metadata - that describe cases.

The case template was then assessed and refined in a workshop together with PA experts. More in detail, former cases were analysed in order to identify those aspects in the case template that were relevant. In a next step the importance of the attributes (metadata) to determine similarity of cases was defined. Therefore the PA officers were asked to identify those metadata they consider most important for ranking cases. Figure 6 shows the first page of the case template.

Case Number	e.g. 192487
Municipality	e.g. Cingoli, Sarnano, Serrapetrona, Senigallia etc.
Province	e.g. Macerata, Ancona, Pesaro Urbino, Fermo, Ascoli Piceno etc.
Region	e.g. Marche, Umbria, Lazio, Emilia Romagna etc.
Zone	e.g. National Park, Regional Park, Beach area
Application type	**New productive systems** - Localization - realization **Modification of existing productive systems** - restructuring - transformation - reconversion - expansion - transferring - reactivation **Quit**
Sector	**Code and Label** - 1 Trade in Fixed Locations - 2 Newspaper and Magazines - 3 Phone Center -Internet Point - 4 Retail Trade on Public Areas - 5 Administering Food and Beverage - 6 Filling Station - 7 Administration Police - 8 Tourism - 10 Building - 11 Other sectors - 12 Services to people - 13 Sport - 14 Fire Fighters - 15 Funerary - 16 Public Security - 17 Driving Schools - 18 Environment - 19 Agriculture - 20 Health - 21 Public Land - 22 Food
Business Activity	- Industrial - Crafts - Commercial - Farming - Receptive tourism - Telecommunication - Others

Figure 6: Outcome 1st Workshop: Case Template

The metadata of the case template (partially visible in Figure 6) are as follows:

- the case number (see example in Figure 6),
- municipality, province, region (see list in Figure 6),
- zone (see list in Figure 6),
- application type (see list in Figure 6),
- sector (see the most used classification among SUAP officers in Figure 6),
- business activities (these are already embedded in the application form, see Figure 6),
- description of application purpose (e.g. realizing a chalet on the beach area of Senigallia etc.),
- organisational units (parties to involve, e.g. Superintendency for the artistic, architectural, historical and environment goods; ARPAM; ASUR; building office, environment office etc.),
- declarations (e.g. qualifying the intervention; current status of the property; calculation of the construction tax; architectural barriers; security of systems etc.),
- decisions (e.g. activity request accepted or rejected and why),
- mistakes (e.g. no mistakes or yes and a description of them).

In order to have a sound classification of an application case we suggested enriching the case description by a standard. Therefore we agreed on the national standard of economic activities ATECO[4] (see table 1) as a further metadata of an application.

In a second workshop, PA officers were asked to use former cases for advice on how to handle a new case at hand. The exercise was conducted based on an example. The example case was about restructuring a chalet with the purpose of embedding a bar into it. This was located along the coast of Senigallia. In order to create realistic conditions, we invented some errors in the application. For example in the instance form contradicting declarations were made: we ticked that the intervention was not subject to the hygienic opinion, although the realization of a bar always requires it. Additionally, we left unticked the declarations concerning the zone constraints, although the extension of the chalet definitely requires declaring that the intervention takes place within a protected area, i.e. beach area (in Italian "area arenile"). Then, twelve old cases were given to the PA officers and they were asked to identify the ones that could support work on the new (example) case, i.e.:

- recognizing the inconsistency of the declarations and correcting them,
- identifying all the organisational units to involve,
- learning how the organisational units behaved in the past,
- learning from both accepted and rejected cases.

Among the 12 cases, 2 cases were identified as the most similar ones, and thus as the most useful to handle the new case. The PA officers were then asked to determine the aspects relevant for their choice. These aspects were then considered the ones most relevant to determine similarity between cases.

[4] ATECO is a standard provided by The National Institute for Statistics (ISTAT) who is the main supplier of official statistical Information in Italy. Since January 2008 ISTAT has adopted ATECO 2007 as a national standard to classify the economic activities. URL: http://www.istat.it/it/strumenti/definizioni-e-classificazioni/ateco-2007

Within the two workshops we identified the relevant attributes to describe a case and which of these attributes are appropriate for determine similarity between cases. Based on the workshop results case content and case characterisation of cases, describing the Titolo Unico Process, was defined. After this PA officers were asked to attach weights (in the racne 1 - 10, 1 being the less important and 10 the most important attribute) to the attributes in order to determine the importance of characterisation elements.

In the following we list the derived requirements:

Requirement CBR1:

Experience Management should reduce the time for identifying the involved organizational units (current effort).

Requirement CBR2:

Experience Management should reduce the errors in identifying the involved organizational units (current no of errors).

Requirement CBR3:

Experience Management can help identifying personal contacts for organizational units (current effort).

Requirement CBR4:

Experience Management should help improve the acceptance rate of an activity request (current acceptance rate).

Requirement CBR5:

Experience Management should help identifying the incorrect selection of declarations (current effort).

4. CBR for Learn PAd

From the domain analysis we learned that retrieving former cases and comparing them to current cases is an important aspect of learning in public administrations. This is particularly true for knowledge-intensive tasks as we highlighted in Figure 5. Experience management contributes significantly to successfully handling cases and hence, Learn PAd should support PA officers in finding and reusing the best fitting former cases.

As shown in the related work section several approaches for experience management are available, one of them is case-based reasoning (CBR). Drawing upon existing research - for example conducted within the national research project [SIC!][5] - CBR is implemented in Learn PAd for experience management. The CBR approach implemented in Learn PAd utilizes the Learn PAd ontology for case retrieval and similarity determination. For details on the ontology-based CBR see section 2.3.

In the following we describe the representation of the case characterisation content in the Learn PAd ontology. Next, we introduce the similarity model and the adaptation model (see section 2.1 for a brief introduction on similarity and adaptation). We close the chapter providing a mockup of the CBR user interface of Learn PAd.

4.1. Case Characterisation and Case Content

The attributes (metadata) identified within the domain analysis characterise a case (see Figure 6). Based on this outcome a sound and formal structure for the case characterisation and the case content was defined.

Case Characterisation

The case characterisation describes the cases stored in a case base. Learn PAd ontology was then enhanced by concepts representing the attributes (metadata) describing the case as listed in Table 1.

Table 1: Case characterisation

Case Concepts	Description
Applicant	Person that submitted the application.
Application type	Application type can be new productive systems such as realization and (de)localization, or modification of existing ones, such as restructuring, transformation, reconversion, expansion etc., or quitting the activity. These concepts take the form of a tree to assign similarity calues to inner nodes (see Figure 9).
ATECO	ATECO is an italian standard classification of economic activities proposed by the National Insititute for Statistics (ISTAT) (http://www.istat.it/it/archivio/17888 available in Italian only).

[5] **S**oftware **I**ntegration using ontology-based **C**ase-based reasoning [SIC!], project no 14575.1 PFES-ES, is a Swiss national research project supported in part by the Commission for Technology and Innovation (CTI) of the Swiss confederation.

Description of application purpose	E.g. realizing a chalet on the beach area of Senigallia.
Physical Location	The location where an activity should take place. We consider the city, and then its province and region are inferred (like for application type concepts city, province, region are in form of a hierarchy and similarity values are assigned to inner nodes, i.e. 1, 0.6, 0.4 respectively).
Zone	A zone can span one or more cities, provinces and regions. One example is the National Park of Monti Sibillini located across the regions of Marche and Umbria, encompassing the provinces of Macerata, Fermo, Ascoli Piceno and Perugia, and the some cities of these provinces.
Public Administration	A PA can be either a single PA (e.g. Senigallia) or an aggregated PA (e.g. Monti Azzurri).
Organisational Units	Organisational units as for example ARPAM, ASUR, municipal offices etc. These are relevant elements if the organisational unit to involve is known, which generally happens after the red circle n. 2 in Figure 5.
Sectors	One or more business sectors can be affected by an application case. The reference list that is currently adopted by SUAP officers is taken into consideration. That is 21 sectors are provided and each of them are assigned to an identification code. Instances of the concept sector can be tourism, health, building etc.
Business Activity	A business activity is the targeted activity of the application case, e.g. receptive toursim, commercial, farming etc. (see Figure 6).
Decisions	A decision is taken at the end of the process Titolo Unico and it can be either positive or negative referring to the acceptance or rejection of the application respectively.

For Learn PAd the case-specific concepts were related to already existing concepts of the Learn PAd ontology (see section 5 for the technical architecture details). As described in D5.1 the Learn PAd ontology is an extension of the ArchiMEO ontology consisting of several parts, indicated by namespaces:

- archi = concepts and relations representing the Archimate standard (The Open Group 2012).
- eo = concepts and relations representing ArchiMEO specific enhancement.
- top = concepts and relations representing a Top Level Ontology (e.g. for locations).
- cbr = concepts and relations representing CBR-enhancements (made within the aforementioned [SIC!] project).
- ldp = newly introduced name space indicating concepts and relations specific for Learn PAd CBR.

Table 2 shows the 'is-a-hierarchy' for CBR-specific concepts of ArchiMEO.

Table 2: Super-sub-concept relations of CBR-specific concepts

super concept	CBR-specific sub concepts	
archi:BusinessRole	ldp:Applicant	
cbr:Case	lpd:ApplicationCase	
nco:Category	lpd:ApplicationType	lpd:Application_SubType
	lpd:ATECO_Category	lpd:ATECO_SubCategory
eo:OrgUnit	lpd:PublicAdministration	lpd:Single_PA
		lpd:Aggregated_PA
top:PhysicalLocation	lpd:Zone	ldp:Agricultural_Zone
		ldp:Costal_Zone
		ldp:Forrest_Zone
		ldp:Industrial_Zone
		ldp:Living_Zone
		ldp:National_Park_Zone
		ldp:Regional_Nature_Park_Zone
		ldp:Regional_ Protected_Area_Zone
		ldp:Urban_Zone
	top:City	
top:PartOfCountry	top:Province	
	top:Region	
archi:BusinessObject	lpd:BusinessSector	
archi:BusinessBehaviourElement	lpd:BusinessActivity	

Table 3 shows the object properties of concepts, i.e. relations between concepts, used for case characterisation and description.

Table 3: Object properties of concepts

Domain class	Object Property	Range
archi:BusinessActor	assignedToRole	archi:BusinessRole
lpd:ApplicationCase	isSubmittedByApplicant	archi:BusinessActor (subconcepts already existing are eo:Person and eo:OrgUnit)
	isManagedBy	lpd:PublicAdministration
	affectsZone	lpd:Zone
	affectsSector	ldp:BusinessSector
	affectsTargetBusinessActivity	ldp:BusinessActivity
	affectsTargetATECO	ldp:ATECO_Category
	requiresOrganisationalUnits	eo:OrgUnit
	hasType	lpd:ApplicationType
eo:OrgUnit	hasLocation	top:PhysicalLocation
top:PhysicalLocation	isPartOf	top:PhysicalLocation (to represent that City is part of Province, Province is part of Region)
	encompasses	lpd:Zone
lpd:PublicAdministration	administratesLocation	top:PhysicalLocation
	actsOnLaws	nco[6]:LawAndRegulation
lpd:Zone	isRegulatedBy	

Table 4 shows the data type properties of concepts.

[6] noc stands for not categorized objects.

Table 4: Data type properties of concepts

Domain Class	Datatype Property
lpd:ApplicationCase	Description (text)
	Decision (Boolean)
lpd:BusinessSector	Code (int)

Case content

The case content relates to information used to process a case, for example documents and links to external information sources. Table 5 provides content elements and their manifestation. The content element 'case folder' pools information of several types used within case execution, like

- Documents used, created and/or updated throughout the Titolo Unico process
- Reports/transcripts and/or notes about decisions - i.e. accepted or rejected application and explanation
- Lesson learned, i.e. missing documents, misinterpretation of law articles
- Involved organisational units, e.g. superintendency for the artistic, architectural, historical and environmental goods, port authorities, municipal offices (organisational units involved in the case and the reason for it)
- Declaration sections – as for example:
    1. Qualifying the intervention,
    2. Urban destination in force and in safe guardian and /or (only applied in those cases compliant to art.37, c.4 D.P.R. 380/01) in force at the time of implementation of the intervention,
    3. Current status of the property,
    4. Calculation of the construction tax,
    5. Architectural barriers,
    6. Security of systems,
    7. Health and hygiene regulations,
    8. Fire prevention,
    9. Cultural heritage,
    10. Archeological heritage,
    11. Landscape assets, etc.

Table 5: Case content

Content Element	Content Manifestation
Case Folder	Documents created or used
	Reports/notes about decisions
	Descriptions of lesson learned
	List of involved organisational units
	Notes on declaration sections
	Further information

4.2. Similarity Model

The representation of characterisations of cases in the Learn PAd ontology allows for inferring similar cases. Therefore a query case is compared to former/historical cases (see Figure. 7). This function is part of the CBR retrieval phase as depicted in Figure 2 (The CBR Cycle).

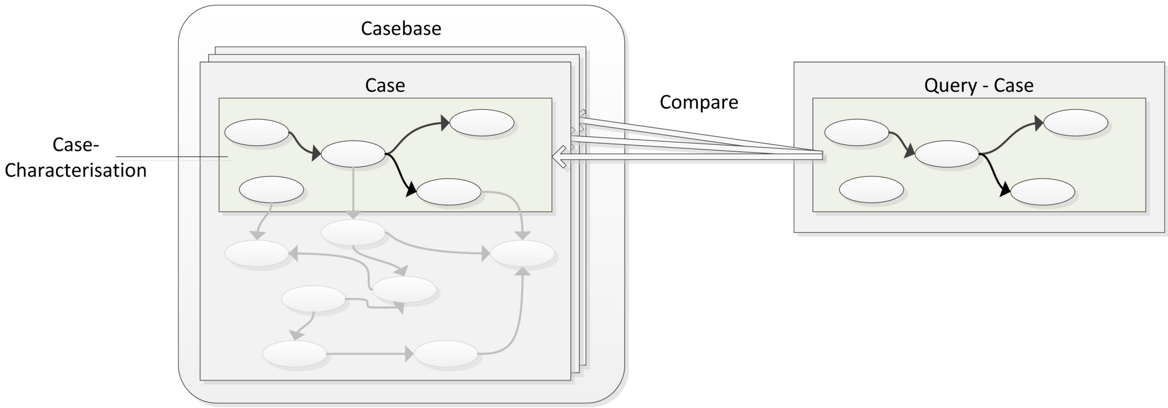

Figure 7: Comparison of a query case with historical cases

In order to retrieve appropriate cases similarity measures must be applied. In general, one distinguishes between global similarity measures, which are defined on the level of cases, and local similarity measures, which are defined on the level of attributes. In situations where cases consist of simple attribute-value pairs, local similarity is applied to the involved case attributes. In structural or ontology-based CBR (see section 2.3), cases can have complex objects as attributes and hence local similarity must also be applied recursively to the attributes of such objects.

The global similarity measure provides a way to aggregate all the local similarity values into one value that reflects the similarity between two cases based on the local similarity of their attributes.

For the SUAP scenario, case characterisations are mostly simple attribute-value pairs - hence the global similarity measure can be a simple weighted average of local similarities, i.e. to calculate the similarity measure (SM) we adopt the following formula:

$$SM = \sum_{i=1}^{n}(w_i * cf) \Big/ \sum_{i=1}^{n} w_i$$

where *n* is the number of characterisation concepts (attributes) considered, w_i is the weight of attribute i and *cf* is the value of the calculated local similarity that ranges between 0 and 1.

Dividing by the sum of weights ensures that if the inserted query elements perfectly match with the elements of an old case, the latter will result with a similarity value of 1 (1 is the highest while 0 is the lowest).

For other scenarios where case characterisations are more complex, more sophisticated functions can be used (Witschel et al. 2015).

Regarding local similarity, applied functions depend on the attribute type. For String attributes (i.e. free text to be entered by the user), one often adopts string similarity measures such as the *Levenshtein* string edit distance (which is the minimal number of edit operations when transforming one string to another) or JaroWinkler similarity (William W. Cohen n.d.). For categorical attributes where possible values are taken from a predefined list, but not structured in any particular way, one often uses either a similarity matrix (where similarity is defined for each combination of attribute values) or - in cases where such a matrix becomes too large - resorts to a simple equality (corresponding to a similarity of 1) or inequality (similarity 0) of attribute values. In our scenario, this is e.g. true for the “Applicant” attribute (see Table 7 below).

Our domain analysis revealed two additional relevant attribute types that need to be addressed by special local similarity measures:

- **Categorical attributes with taxonomic value range**: often, attribute values can be hierarchically structured (via a so-called taxonomy, i.e. a tree structure). In such cases, one often considers values that are located in the same branch of the tree as more similar than ones that reside in different branches, because they often share certain properties. As an example, consider the attribute “Application type” and its value range as depicted above in Figure 6 and represented as a tree below in Figure 8. Many of the other case attributes have a taxonomic structure, too, see Table 7 below.
- **Categorical attributes which can take more than one value**: this essentially means that there might be a 1:n relationship between a case and the attribute, i.e. the case can be associated with more than one value of the attribute. As an example (refer Figure 6), consider the attribute “Zone”: obviously, a business can be located in a beach area that is at the same time part of a national park or a regional park, e.g. the beach area of the city of Sirolo (in the Ancona province) belongs to the regional park

of Conero[7]. Hence, a case has to be associated with both the values “beach area” and “regional park” of the “Zone” attribute.

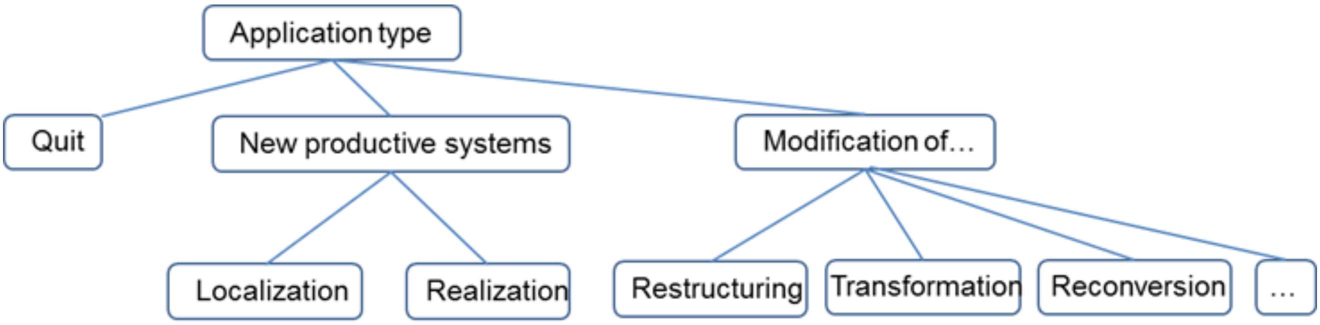

Figure 8: Taxonomic structure of the value range of attribute “application type”

Based on the findings of prior research, we adopt the following similarity functions to deal with these two cases:

- For taxonomic attributes, several approaches to defining local similarity measures exist, many of which are based on the length of the path between two nodes in the taxonomy tree. We follow the reasoning of Bergmann (Bergmann 2002) who argues that such approaches are not useful in many domains because the depth of tree branches may vary considerably. This results often because certain topics are modeled in more detail - but it does not mean that nodes in deeper tree branches should be more dissimilar to other branches (which happen when using path length). Instead, Bergmann proposes to manually assign a similarity value to each inner node of the tree based on expert experience. The similarity of two leaf nodes is then the value that is assigned to the lowest common parent node of the two leaves - or 1 if the values are equal. In our example, we may consider an assignment as depicted in Figure 9. The rationale is that e.g. the introduction of a new business - even if it is not the exact same sub-type - is more similar to another new business than to a modification of an existing business. Hence we define:
    - sim(Localization, Realization) = 0.8 > 0 = sim(Localization, Restructuring).

[7] https://it.wikipedia.org/wiki/Parco_regionale_del_Conero

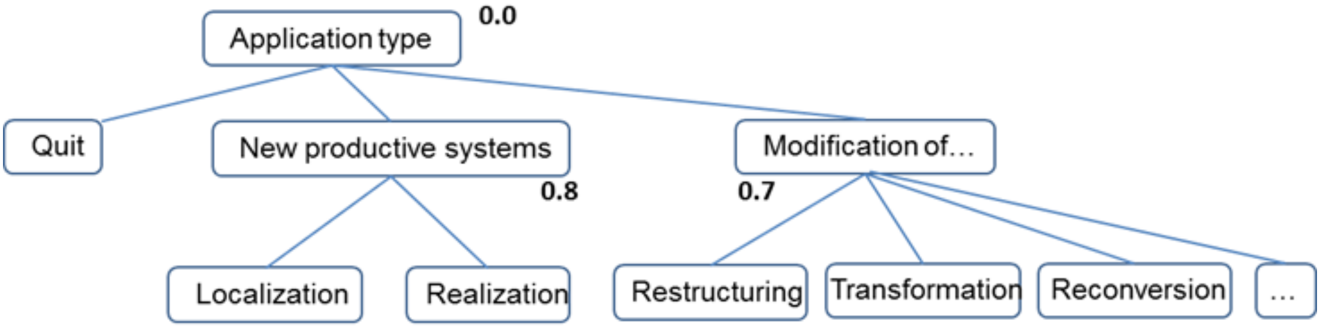

Figure 9: Taxonomy with similarity values assigned to inner nodes

- For 1:n relationships between cases and attributes, we rely on our own research (Witschel et al. 2015), which is inspired by retrieval functions in information retrieval. The main idea is that - as not guaranteed by most existing similarity measures - a historical case should not be “punished” for having attribute values that are not shared with the new case (which we call the “query case”). As long as the values of the historical case matche values of the query case, its additional values are neglected. For example, consider the two historical cases C1 and C2 and its attribute 'zone'. Value for 'zone' is in C1 ' beach area' and in C2 it is ' beach area and national park' (cf. Table 6). If a civil servants wants to find cases that are similar to a new business which is located in a 'beach area' (Q1), we argue that both C1 and C2 should be equally similar to Q1 because both share the value of the zone attribute ('beach area'. However, if in case a civil servant wants to find cases in which a new business is located in a 'beach area *and* in a national park (Q2) only C2 should be provided as it covers more relevant aspects as C1.

These properties of similarity are useful especially in cases where initial case characterisations (queries) are incomplete (see below), but they are not ensured by most similarity measures that are traditionally used in CBR. Finally, they result in a similarity function that is asymmetric:

$$P(E|F) = \prod_{e \in E} P(e|F)$$

Here *P* is the asymmetric similarity function that is applied to two sets of values *E* (taken from the query case) and *F* (taken from the historical case) of a given attribute that can take more than one value. As an example, we may consider comparing Q1 to C2 for the “Zone” attribute. In such a case, we have E = {beach area} because Q1 involves only beach area and F = {beach area, national park} since the historical case C2 refers to a business that is located in a beach area inside a national park. In order to apply the formula above, we have to define $P(e|F)$:

$$P(e|F) = \begin{cases} \max_{f \in F} p(e|f) & \text{if } \exists f \in F : p(e|f) > 0 \\ 0.01 & \text{otherwise} \end{cases}$$

where $p(e|f)$ is given by some elementary local similarity function and 0.01 is a smoothing factor. In our example, we assume for simplicity that we use equality for the “Zone” attribute, i.e. $p(e|f) = equals(e, f)$. This results in:

$$P(E|F) = P(\{beach\ area\}|\{beach\ area, national\ park\})$$
$$= \max\big(equals(beach\ area, beach\ area), equals(beach\ area, national\ park)\big)$$
$$= \max(1,0) = 1$$

That is, C2 is a perfect match for Q1. On the other hand, if we compare Q2 and C1, we have to use E = {beach area, national park} and F = {beach area} and hence compute:

$$P(\{beach\ area, national\ park\}|\{beach\ area\})$$
$$= \max\big(equals(beach\ area, beach\ area)\big) \cdot 0.01$$
$$= 1 \cdot 0.01 = 0.01$$

and note that the measure is indeed asymmetric. But it makes sense since C1 is not a good match for Q2 as it does not cover the aspect of a national park zone.

Table 6: Some historical and new (query) cases with their values of the "Zone" attribute

Case	Zones	...
C1	beach area	...
C2	beach area, national park	...
Q1	beach area	...
Q2	beach area, national park	

In our approach, functions and weights are implemented in a so called similarity ontology. Functions (with a range between 0 and 1) and weights (with a range between 1 and 10) are applied on attributes of ontology classes via annotation as shown in table 7. As can be seen from the table, we use the local similarity functions SoftTFIDFJaroWinkler, equality and Bergmann's taxonomy measure as described above. SoftTFIDFJaroWinkler is a string similarity measure that works well with names or text fields that consist of several words, which might be syntactically arranged in different ways without impacting semantic similarity (see (William W. Cohen n.d.)). For the aggregation of attribute values for 1:n attributes, we use the probabilistic similarity function, see above.

Table 7: Weights and functions to determine similarity of concepts

Case Concept	Weight (from 1 to 10)	Function (from 0 to 1)
Applicant	5	Probabilistic / Equals
Physical Location (city, province, region)	7	Taxonomy
Zone	8	Probabilistic / Taxonomy
Public Administration	7	Taxonomy
Application type	6	Taxonomy
BusinessSector	7	Probabilistic / Equals
Business Activity	7	Probabilistic / Equals
Business Activity (ATECO codes)	7	Probabilistic / Taxonomy
Description of application purpose	5	softTFIDFJaroWinkler
Organisational Units	8	Probabilistic / Equals
Decisions	7	Probabilistic / Equals

For calculating the similarity measure those concepts are used that the PA officer provides while executing the process. This means that in the beginning of a process only few attributes might be available whereas in the end all might be given. For example, if a PA officer inserts in the beginning of his/her work on a case only provides the name of the city (e.g. Senigallia), the zone (e.g. Beach Area), and the application type (e.g. Realization) similarity measure will be calculated on three elements only. Hence, all the historical cases where a business activity of type realization that took place on the coast of Senigallia will be shown as the most similar ones by the Learn PAd system.

The similarity model, although simple, allows for retrieving similar historical data according to the current available information. With our approach we are able to meet the users' requirements as we can

- reduce the time for identifying the involved organizational units (former effort - effort with Learn PAd) / **CBR1,**
- reduce the errors in identifying the involved organizational units (former no of errors - errors with Learn PAd) / **CBR2,**
- help identifying personal contacts for organizational units (former effort - effort with Learn PAd) / **CBR3,**
- improve the acceptance rate of an activity request (former acceptance rate - acceptance rate with Learn PAd) / **CBR4,**
- help identifying the incorrect selection of declarations (former effort - effort with Learn PAd) / **CBR5.**

As a future work we could consider a more elaborated similarity model where viewpoints or mental models are applied on different activities.

4.3. Adaptation Model

In addition to the already implemented recommendations in the Learn PAd system (cf. D5.2) also similar cases are suggested. The PA officer can view the suggestions and select the most appropriate one. Furthermore, he/she can adapt it to the case he/she currently performs and thus enhance the case base with a new case. This refers to the CBR reuse phase depicted in Figure 2 (The CBR Cycle). The case content (above described) of the selected similar case is displayed through a case adaptation interface. Via this interface the PA officer can adapt the retrieved content to the new situation, i.e. a copy of the existing case is modified to create the new case. According to the CBR cycle, when the new case is created, it is stored in the case base and available for retrieval. The next section shows a mockup of how a case adaptation user interface might look like.

Figure 10 shows how similar problems require similar solutions and how case reuse profits from this fact: when a user describes a new problem and retrieves cases with similar characterisation, there is a chance that large parts of the solution of these cases can be re-used. However, since the new problem differs in some aspects from the previous one(s), there is almost always the need for adapting the old solution to the new aspects of the new problem.

One refers to *adaptation knowledge* when describing the knowledge that is needed to find out which differences between the old and the new case require which adaptations to the new case. Since such differences and the corresponding adaptations often repeat themselves, research has been conducted that studies attempts to automatically learn adaptation rules from manual case adaptations (a discipline called “adaptation knowledge acquisition” or “AKA”, see e.g. (d’Aquin et al. 2007) (Kathleen Hanney 1997)).
The discovered adaptation rules are - roughly speaking - in the form:

IF <certain difference between old and new problem>
THEN <apply change to old solution>

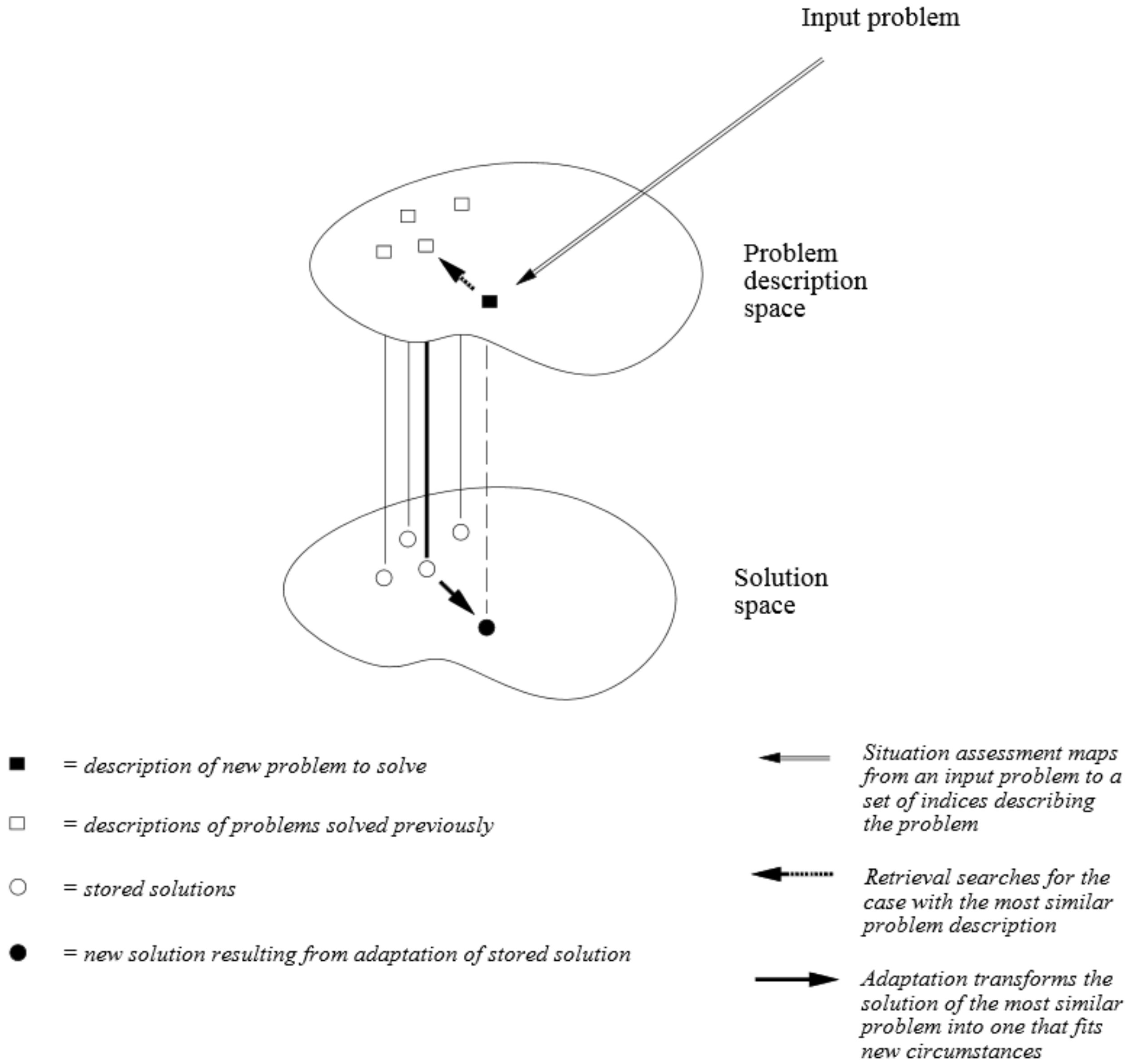

Figure 10: The adaptation process in CBR (Leake 1996)

Approaches that attempt to learn adaptation rules fully automatically are based on observing patterns in how humans adapt solutions when certain variations in the problem are observed. To discover such patterns reliably, a substantial amount of adaptation data is needed. In practical environments, the available data is almost never large enough to make such approaches work well.

Therefore, we propose a semi-automatic approach to acquiring adaptation knowledge - somewhat similar in spirit to the work of Cordier et al. (2008). The idea is that, when a user manually adapts a solution, it is possible to generate several plausible hypotheses that explain why this adaptation is needed (i.e. hypotheses that refer to a difference between the new and the old case that made the adaptation necessary). When presenting all reasonable hypotheses to the user and obtaining her feedback - i.e. asking her to pick the correct hypothesis - it becomes possible to infer reliable adaptation rules based on a single manual adaptation.

We illustrate this with an example: assume that a civil servant has entered some data about a new case as follows:

Table 8: Example of characteristics of a new case

Attribute	Value
Application type	Realization
Zone	Beach area
Public administration	Senigallia

The civil servant retrieves the following similar case (where the organisational units to be involved are the solution).

Table 9: Characteristics of a retrieved (similar) case

Attribute	Value
CaseId	123
Application type	Localization
Zone	Urban area
Public Administration	Senigallia
Organisational units	commercial office, urban office

The user then decides to copy the solution of the old case and then adapts the solution by replacing “urban area” with “port authorities” and “office of maritime domain”.

Figure 11 shows an example of a user dialogue that could be used to test adaptation hypotheses.

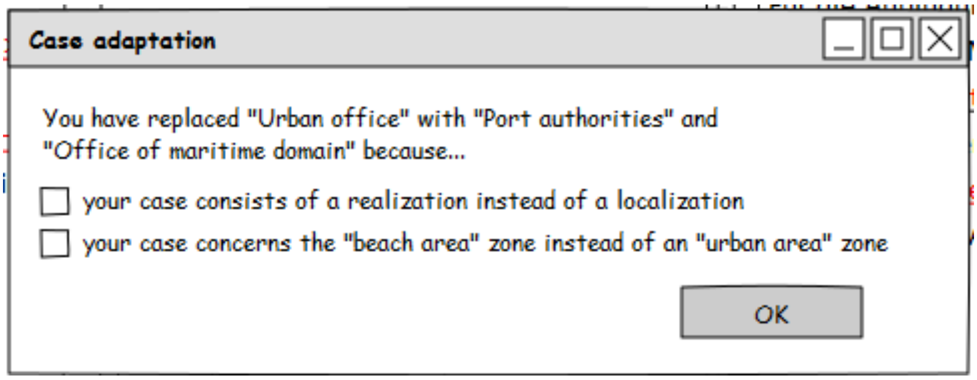

Figure 11: User dialogue for testing adaptation hypotheses

Assuming that the user selects the second option (which is intuitively more plausible), the system can record the following adaptation rule that can be used to support future case adaptations:

IF (problem contains zone=”beach area” instead of zone=”urban area”) THEN replace “urban office” with “port authorities” and “office of maritime domain” in the solution.

4.4. User Interface

Figure 12 shows the mock-up of an user interface of the Learn PAd system displaying the case characteristics (i.e. attributes of case and their values) of a case a PA is about to handle. We assume the PA officer is in the “Assess Application” task and keyed in all application data relevant for the case (when Learn PAd is productively implemented the data can be inserted automatically from PAs’ legal systems). All the case concepts with a probabilistic function (see Table 7) can contain multiple elements. That explains the symbol “plus” in the respective textboxes. Elements that are already known such as specific PAs, application types, business activities etc, are depicted in Figure 12 via drop boxes, i.e. those boxes with a little upside-down “arrow”.
As soon as he/she clicks on the “similarity measure” button similar cases are retrieved and displayed in the ‘recommender panel’ at the right hand side of the user interface. In Figure 12, two cases were retrieved.

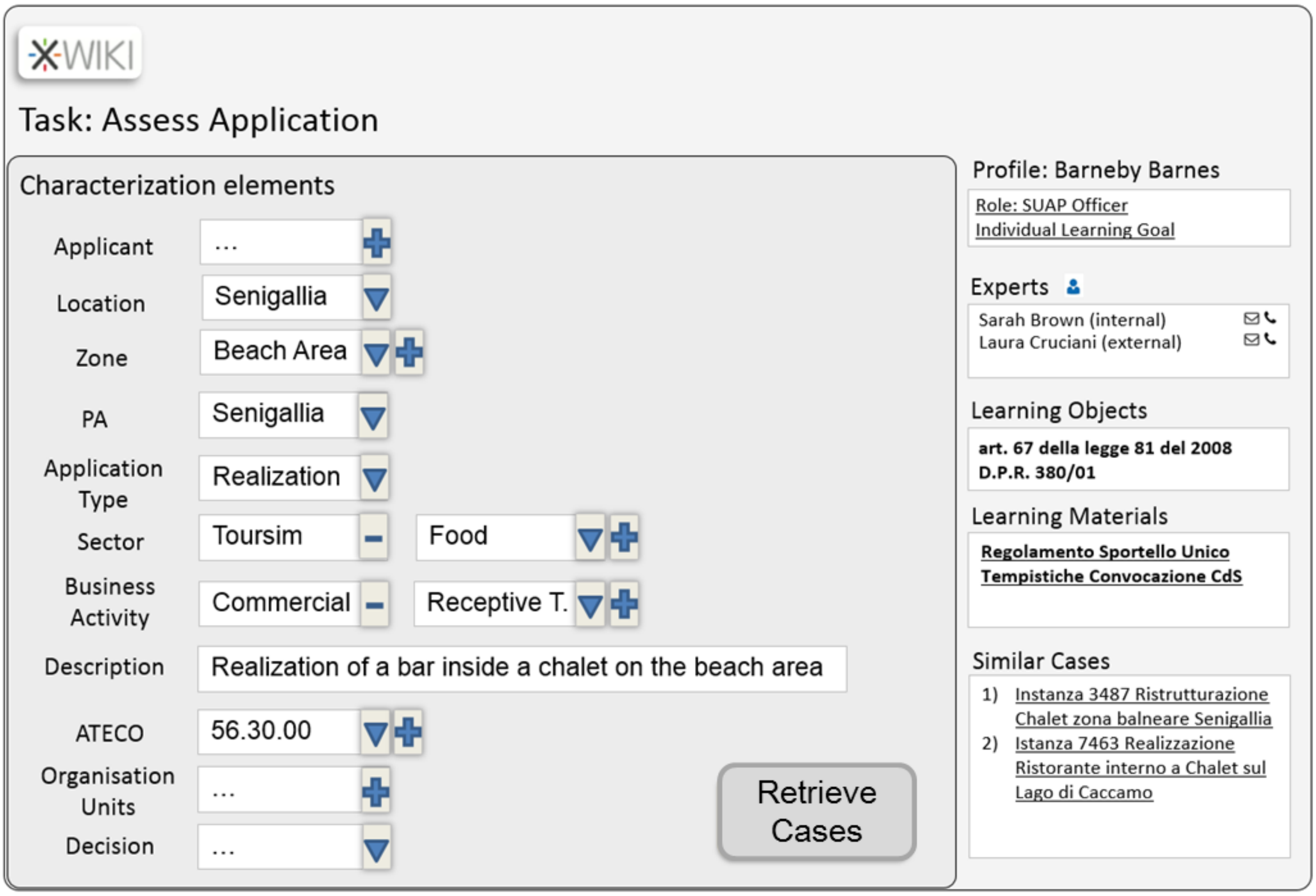

Figure 12: Characterisation elements user interface

If a similar case is selected (in the mock-up it is case 1 - Instanza 3487 etc.), the user interface depicted in Figure 17 will pop up. The interface shows the case characteristics of the new case on the left hand side and the case content of the selected, former case on the right hand side.

5. Technical Architecture and Implementation

The technical architecture describes the implementation of the CBR module based on the ICEBERG Toolkit (section 5.1) and its integration in the Learn Pad system (section 5.2).

5.1. Implementation of CBR

The Information and Knowledge Management Research Group (IKM-Group) of the FHNW developed the interlinked case-based reasoning toolkit or, in short, the ICEBERG Toolkit.

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 actions. Rated to the ICBERG Toolkit, 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.

The ICEBERG Toolkit is a CBR application including a set of ontologies and a JavaFX based user interface. **Errore. L'origine riferimento non è stata trovata.** shows the main concepts of the ICEBERG approach.
The toolkit consists of a persistence & business logic, which provides a case base with reasoning- and similarity functions. In addition, the toolkit consists of a generic (driven by the ontology) graphical user interface.

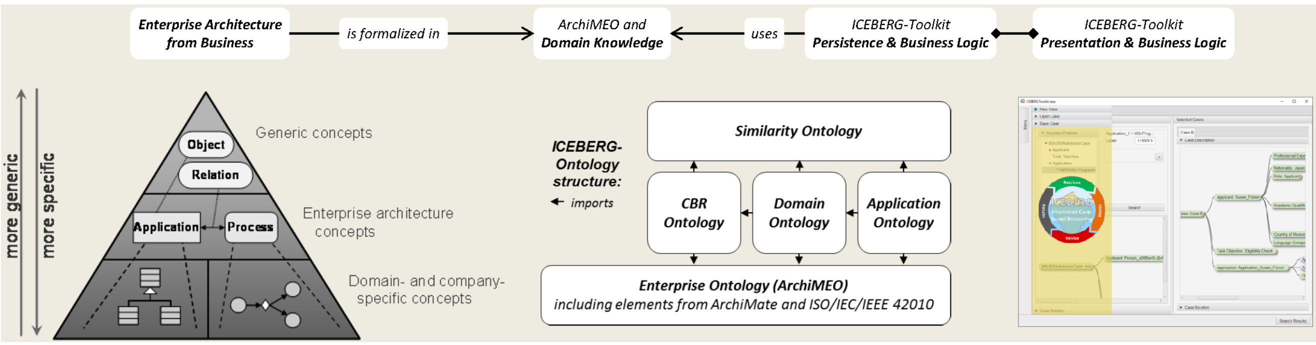

Figure 13: ICEBERG implementation (adapted from Martin et al. (2013))

For the Learn PAd project the core CBR component, which is part of the persistence & business logic, is adapted to meet the specific requirements and to fit into the Learn PAd platform.

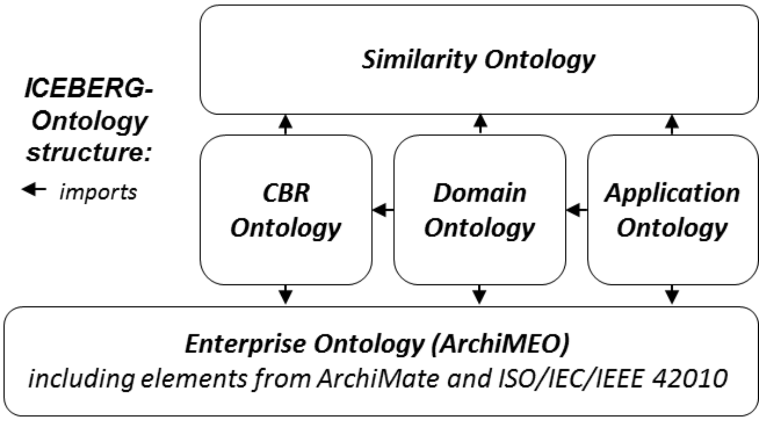

Figure 14: ICEBERG ontology structure (adapted from Martin et al. (2013))

This approach relies on three main ontologies: similarity ontology, CBR ontology and ArchiMEO. **Errore. L'origine riferimento non è stata trovata.** (ICEBERG ontology structure) shows the dependencies (imports) of the ontologies. These ontologies are formalised using RDFS-Plus and extended with certain resources in the namespace of OWL. The depicted domain ontology and application ontology are placeholders for various ontologies developed in the Learn PAd project. The enterprise ontology (ArchiMEO) is the foundation of the ICEBERG approach. The retrieval and configuration of the CBR system is done in the CBR ontology using the similarity ontology. The similarity ontology contains the main concepts for modelling and defining ontology-based similarity models.

5.1.1. Formalization of Case Characterisation and Content

The conceptual similarity model was elicited in interviews with Marche region end users as described in Chapter 3. To formalize the model we utilized and adapted the CBR ontologies from ICEBERG Toolkit described above.

The similarity ontology provides annotations properties for classes, relations (object properties) and literal properties (data type - and annotation properties). The annotations properties encapsulate a set of similarity properties to select mainly the similarity function as well as the weight. For simple properties a set of literal value similarity functions is provided. For the aggregation of similarity values a set of aggregation functions, like cosine, average, probabilistic, etc. is provided. The aggregation functions can be applied to a class to aggregate the class instances property values or to a relational property to aggregate the similarity values of multiple class instances. **Errore. L'origine riferimento non è stata trovata.** reflects the characterisation space for the default case in Learn PAd.

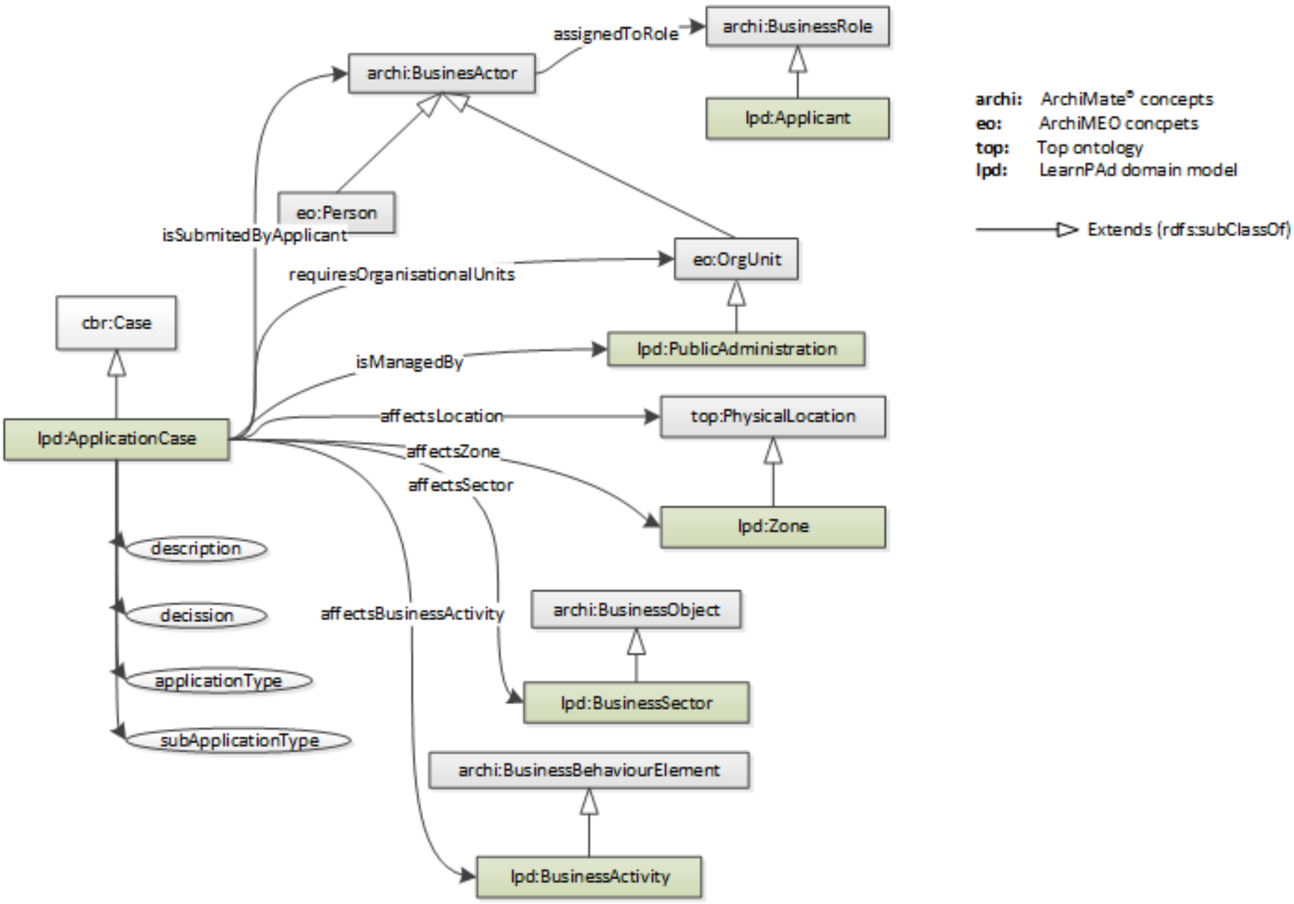

Figure 15: Application Case Characterisation

5.1.2. CBR Lifecycle Management

In Learn PAd we differ between the functionalities integrated in the platform and the lifecycle management of the cases using the Java FX frontend. Both the platform and the management frontend use the same case base. The frontend as depicted in Figure 16 supports the full CBR cycle, where new cases can be created, similar cases can be retrieved, selected cases can be adapted automaticly and manually and finally cases can be released as learned cases to the case base.

The form to enter the new case characterisation as shown in the upper part of Figure 16 is dynamically generated based on the Learn PAd similarity ontology. Once the new case is described (characterisation entered), the retrievieal of similar cases can be executed. The new case characterisation is compared with the characterisations of past cases in the case base. Finally a ranked list as depicted in the lower part of Figure 16 is shown.

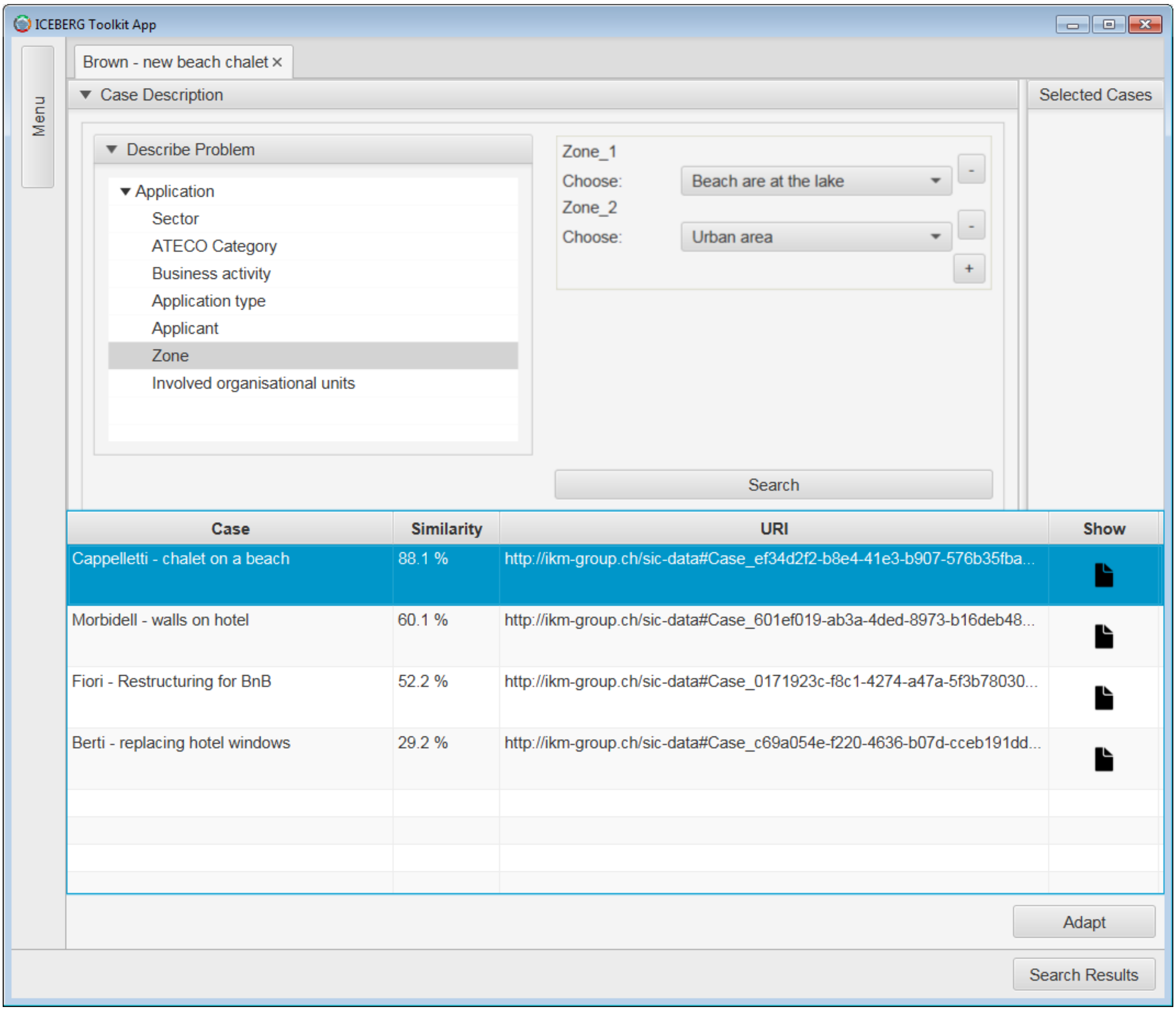

Figure 16: Case lifecycle management frontend

The best matching cases can be chosen to learn from and to adapt the new case content accordingly. In the case adaption view (**Errore. L'origine riferimento non è stata trovata.**), the content of the selected similar case is shown on the right side and can be assigned automaticly or manually to the new. The possible case content elements of the Learn PAd default cases are listed and described in Section 4.1. Additional or new content, like relevant documents for the new case can be added to the table of document through drag and drop functionality.

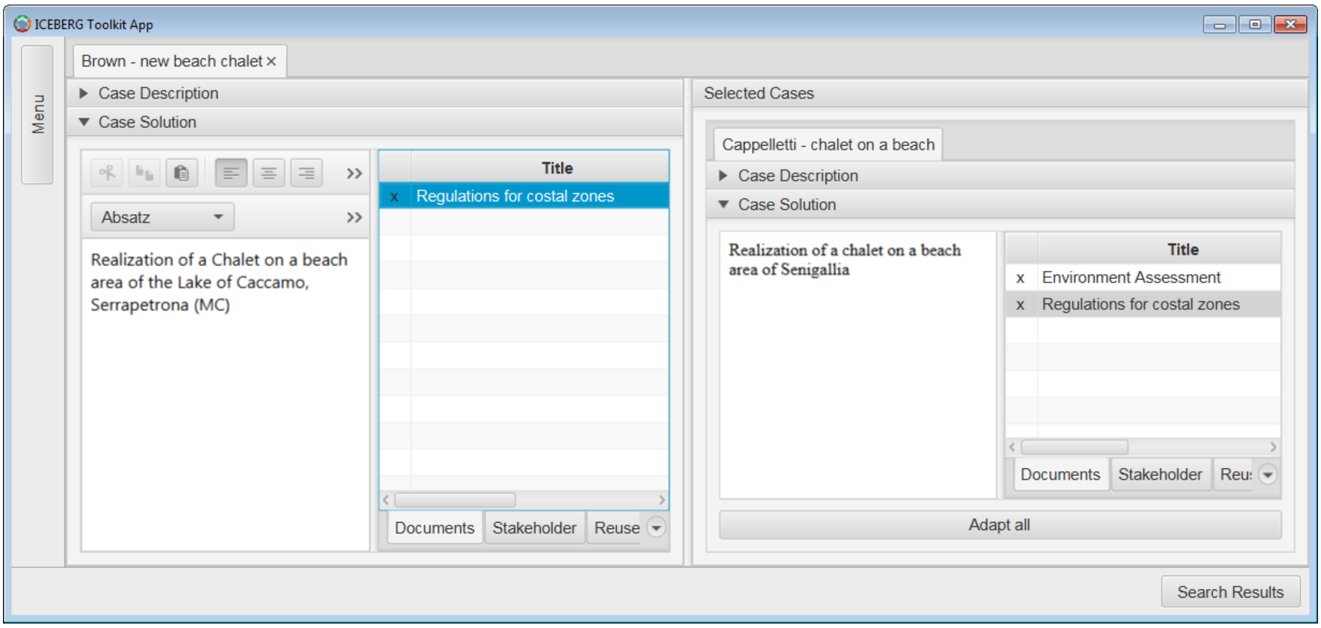

Figure 17: Case adaption

5.2. CBR Integration into Learn PAd System

The integration of the adpated CBR module of the FHNW ICEBERG Toolkit in the Learn PAd ontology and recommender (OR) component is realized as depicted in **Errore. L'origine riferimento non è stata trovata.**.

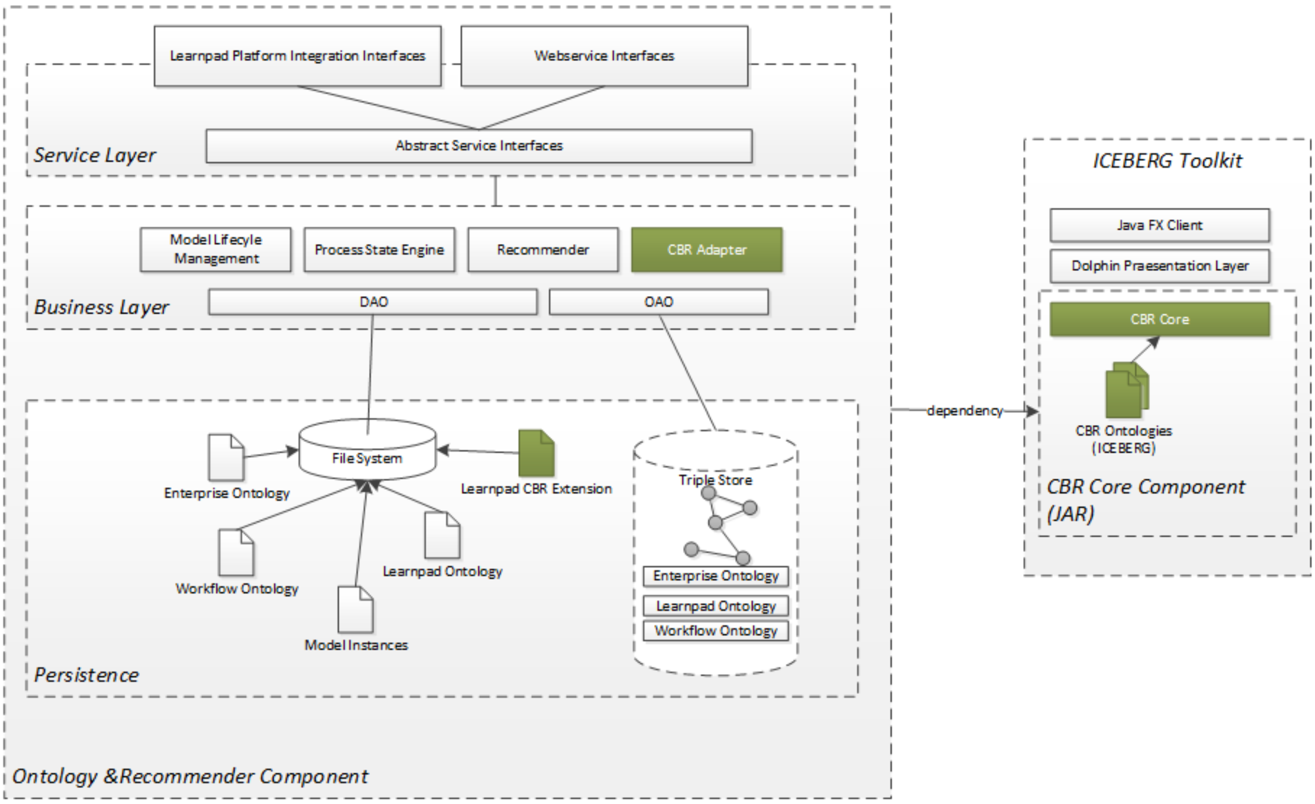

Figure 18: CBR component integration

The ontologies of the OR component will be extended by the ICEBERG CBR ontologies (CBR -, project - and similarity - ontology). These ontologies must be imported by the Learn PAd domain ontology. The functionality to search and retrieve similar cases is provided via the exposed interfaces of the OR component. Mainly the Collaborative Workspace (CW) component will use these services and provides the possibility to search for cases matching the concerns of the learning user. The characterisation of a new case can be described in a form according to **Errore. L'origine riferimento non è stata trovata.**. With this characterisation of the new case, the OR component uses the functionalities of the integrated CBR module and searchs in the case base for cases with a similar characterisation. The top ranked matching cases will be shown to the user and he can chose one and lookup in the management tool the conent elements (documents used for the case, involved organisational units, etc.).

The CBR funtionality is available in all of the three learning modes, i.e. browsing, simulation and execution.

In execution and simulation mode a new default case folder is created as soon as a process is started. All case relevant information is gathered during the process. The new case is closed and implicitly released as a learned case to the case base, if the process ends. For recommendation at any time in the process the default case - and the values gathered up to then - can be used to query the case base for similar cases. In the following section integration of CBR into the Learn PAd simulation component is described in more detail.

Technical integration with the simulation component

When a user starts the simulation the above mentioned default case folder is created and he/she is asked to describe the new case in form according **Errore. L'origine riferimento non è stata trovata.** . During the simulation process the case file content as well as the case characterisation can be completed. The data is stored in the case folder and as case instances of the CBR-concepts of the Learn PAd ontology.

The simulation component is integrated via the Learn PAd core platform via RESTful service interfaces as depicted in **Errore. L'origine riferimento non è stata trovata.**.

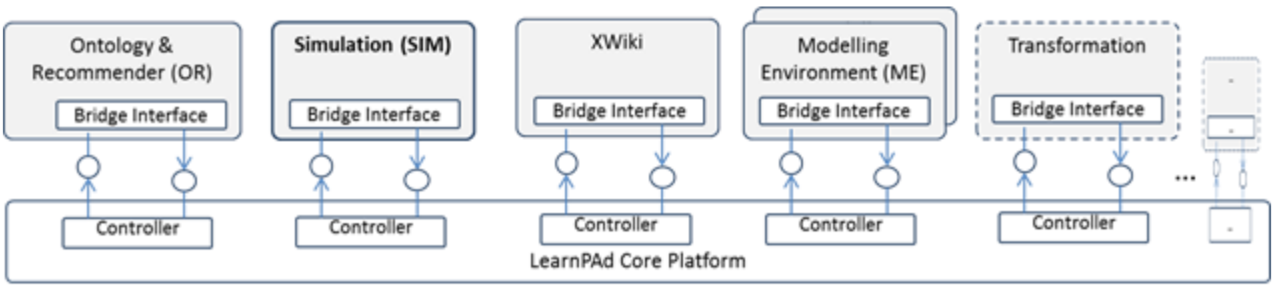

Figure 19: Learn PAd simulation component integration

Technical interfaces of the simulation component

The OR component listens to a couple of technical interfaces (REST services) provided by the simulation component (SIM) and receives simulation data encapsulated in events as depicted in **Errore. L'origine riferimento non è stata trovata.**.

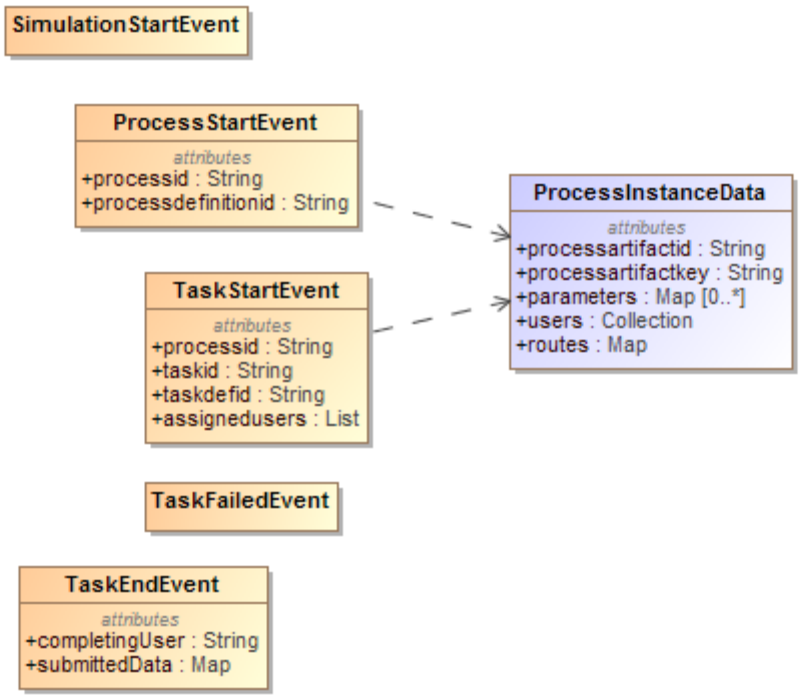

Figure 20: Event paramters of the simulation component

The data attribute in some event parameters encapsulates the case description resp. characterisation in the form of a simple key/value map. An adapter in the OR component maps this data into instances of the Learn PAd case characterisation ontology. The retrieval of similar cases during the simulation of a process is made available via the sidebar of the collaborative workspace. The evaluation of the CBR integration in the Learn PAd platform is described in chapter 6.2.

6. Evaluation

Evaluation of the approach is done in three ways:

1. Early evaluation, focussing on the achieved quality of recommended cases;
2. Evaluation via simulation, focussing on the user interface and case handling (retrieval, adaptation);
3. Evaluation within demonstrator assessment (performed as part of task 8.6).

For D5.3, we can provide results of the early evaluation whereas results of 2) and 3) will be provided as part of D8.4.

6.1. Early Evaluation

Early evaluation is done with two representatives of Marche Region. The evaluation procedure is set-up as follows:

- In the case base of the Learn PAd system 12 difficult former cases are captured;
- An expert extracts aspects from these cases to create a fictitious new one;
- The expert identifies the three most similar cases to the fictitious one and determines their ranking;
- The fictitious case is captured and used for CBR according to the similarity model described in Section 4.2; similar cases are suggested and ranked;
- The expert compares the suggestion with his/her manual selection to answer the following questions:
    - Are the three cases he/she considered relevant suggested (at all)?
    - Are the three cases ranked on top?
    - Are they ranked similarly?
    - If other case(s) are ranked higher are these cases relevant (i.e. did the expert overlook an aspect when determine the similar cases manually).

The 12 cases:

The 12 captured cases were the following:

- **515.2015** Restructuring of a chalet for the realization of an internal bar – Senigallia.
- **829.2015** Restructuring of a chalet and adjustment of the beach area (maintenance purposes) – Senigallia.
- **859.2015** Expansion of an internal surface of a craft building used as a laundry business activity - Senigallia.
- **889.2015** Expansion – Installation of removable covers for outdoor dining in a restaurant business activity – Senigallia.
- **1118.2015** Realization of masonry walls on hotel business – Senigallia.
- **1267.2015** Transformation – Replacing of windows fixtures in a hotel business activity – Senigallia.
- **655.2015** Realization – Installation radioelectric antenna for WiFi data transmission in protected area – Sarnano (Monti Azzurri).
- **22294.2013** Realization of a petrol station – Ascoli Piceno.
- **27409.2014** Realizazion of a residencial and protected facility for elderly people – Ascoli Piceno.
- **64682.2014** Realization of a recovery/waste disposal plant - Ascoli Piceno
- 195.2015 Realization of a petrol station and a building crafts - Cartoceto (Valle del Metauro).

- **431.2014** Restructuring of a civil building to allocate as a B&B– Urbania (Valle del Metauro).

The fictitious case:

The fictitious case is based on some aspects that take place in the above listed cases. This case is comprehensible described in the deliverable D5.2 and it is about realizing a chalet in a beach area of the lake of Caccamo, in the town of Serrapetrona (province of Macerata). Table 8 shows the characterisation elements of the fictitious case.

Table 10: Characterisation elements of the fictitious case

Characterization space	**Elements**	
Applicant:	Susan Brown	
Phisical Location:	Serrapetrona **(city)**	
	Macerata **(province)**	
	Marche **(region)**	
Zone:	Beach area at the lake (**Costal_Zone**), Unione Montana Monti Azzurri **(Regional_Protected_Area_Zone)**	
Application type:	Realization	New productive systems
Public Administration:	Monti Azzurri **(aggregated PA)**	
Sector:	Building, Environment, Public Land, Tourism	
Business Activity:	Receptive Tourism	

ATECO:	***(Category)*** Code: **55** - Accomodation **43** – Specialized Construction activities **41**- Contruction of buildings	***(Subcategory)*** Code:**55.20.51** - Boutique hotels for short staying, houses and apatments for holidays, bed&breakfast, residence. Code:**55.20.40** - Marine and Mountaine Summer camps. Code: **43.21.01** Installation of electrical systems Code:**43.22.01** Installation of idraulic systems Code: **43.22.03** Installation of antifire systems Code:**43.31.00** - Plaster and stucco Code:43.39.01 – Non-specialized construction (masons) Code: **41.20.00** – Construction of residential and not residential buildings
Description of application purpose:	Realization of a Chalet on a beach area of the Lake of Caccamo, Serrapetrona (MC);	
Organisational Units:	(not considered)	
Decision:	(not considered)	

The most similar cases according to RM

Following we list the three similar cases according to RM, starting from the most similar. Additionally, the rank of each case is motivated.

1. **829.2015:** Restructuring of a chalet and adjustment of the beach area (maintenance purposes) – Senigallia.

   Motivation: Description of the application case is very similar. The sectors (Building, Environment, Public Land and Tourism), the business activity (Receptive Toursim), the ATECO code (55-Accomodation (sub code 55.20.40 – Marine and Mountine Summer camps) and 43 – Specialized Construction Activities (sub code 43.39.01 – Non-specialized construction) and the zone (Costal_Zone) overlap.

2. **1118.2015***:* Realization of masonry walls on hotel business – Senigallia.

   Motivation: The application type (Realization), the sectors (Building, Environment and Tourism), the business activity (Receptive Toursim) and the ATECO code (55 - Accomodation and 43 – Specialized Construction activities (sub code 43.39.01 – Non-specialized construction) overlap.

3. **431.2014**: Restructuring of a civil building to allocate as a B&B– Urbania (Valle del Metauro).

   Motivation: The sectors (Environment and Tourism), the business activity (Receptive Toursim), the zone (both belong to a Regional Protected Area Zone) and the ATECO code (55 – Accomodation (sub code: 55.20.51 Boutique hotels for short staying, B&B etc.) and 43–Specialized Construction Activities (sub codes: 43.22.01- Installation of idraulic systems, 43.22.03 – Plaster and Stucco, 43.31.00 Non-specialized construction) overlap. Additionally, likwise the the PA Monti Azzurri, PA Valle del Metauro is an aggregated PA.

Retrieval result

The eight highest-ranked retrieval results of two retrieval runs are shown in Table 11, the cases considered relevant by the expert are marked in bold. The first run was done with the initial setting of weights for the combination of partial similarities as estimated by the experts and defined in

Table 7.
The second run is the result of some experimentation with weights that was done to achieve a better rank for the cases identified as relevant by the expert. As will be discussed later, such tuning of course carries the risk of *overtraining*, i.e. fitting the weights too closely to the particular small sample of cases studied in this evaluation. The newly derived weight vector therefore needs subsequent plausibility check with the expert.

Table 11: Results of two retrieval runs with the Learn PAd CBR system

Rank	Run 1	Similarity	Run 2	Similarity
1	655.2015 Realization – Installation radioelectric antenna for WiFi data transmission in protected area	.046	**829.2015: Restructuring of a chalet and adjustment of the beach area**	0.44
2	**829.2015: Restructuring of a chalet and adjustment of the beach area**	0.42	**1118.2015: Realization of masonry walls on hotel business – Senigallia.**	0.36
3	195.2015 Realization of a petrol station and a building crafts	0.40	**431.2014: Restructuring of a civil building to allocate as a B&B**	0.36
4	**1118.2015: Realization of masonry walls on hotel business – Senigallia.**	0.40	515.2015 Restructuring of a chalet for the realization of an internal bar	0.34
5	**431.2014: Restructuring of a civil building to allocate as a B&B**	0.39	655.2015 Realization – Installation radioelectric antenna for WiFi data transmission in protected area	0.33
6	515.2015 Restructuring of a chalet for the realization of an internal bar	0.33	195.2015 Realization of a petrol station and a building crafts	0.31
7	22294.2013 Realization of a petrol station	0.30	1267.2015 Transformation – Replacing of windows fixtures in a hotel business activity	0.26
8	64682.2014 Realization of a recovery/waste disposal plant	0.30	889.2015 Expansion – Installation of removable covers for outdoor dining in a restaurant business activity	0.26

Discussion and conclusion

It turned out that the following weights were useful to achieve a good ranking – the previous weight is given in brackets in each case, cf.

Table 7: (Applicant=5 (5), Location=6 (7), Zone=8 (8), Application Type=3 (6), Public Administration=3 (7), Sector=8 (7), Business Activity=7 (7), Description = 6 (5), ATECO=12 (7)). The comparison reveals that changes were made in three major areas:

- The importance of the ATECO code for business activity was strongly increased. A likely reason is that the codes give a rather detailed impression of planned activities which is relevant for deriving parties to involve and decisions to be made.
- The weight for Public Administration was strongly decreased. Since all cases are from Marche Region, the local differences might not play such a big role.
- Similarly, the weight for the application type was decreased. Here, one might need to be cautious since the small case base did not involve any cases where a business was quit. It seems that for the other two types of applications, the differences are not so relevant.

Furthermore RM representatives became aware of the similarity of the case “515.2015 Restructuring of a chalet for the realization of an internal bar” after looking at our CBR system result, i.e. considering the 4th highest-ranked case of “Run 2” in Table 11. Subsequently, the RM representatives positioned this case at the 3rd place of the similar ranking. This led us again to slightly readjust the weight vector in order to reflect the expert’s favorite final result.

Obviously, with the tuned weights it is possible to achieve the ranking that the expert expected / recommended, which confirms the suitability of our similarity model and similarity functions.

6.2. Simulation and Collaborative Workspace

For the evaluation of the integrated CBR component we relay on the former cases gathered during the early evaluation. The former cases are entered to the case base with the case livecycle management frontent of the ICEBERG toolkit. Whereas in the early evaluation we focus on quality of the recommended cases within simulation we concentrate on usability.

The same fictive case as used for the early evaluation is entered during simulation as a description of a new application case. This description is stored implicitly in the case base as a new simulation case. The funktinonality of the collaborative workspace (sidebar with recommendations) is used to retrieve similar cases based on the fictive case description.

As a measure of quality it is expected that first the same three top ranked similar cases as in the early evaluation are recommend. And second the similarity value of the top ranked case shall be equal to the early evaluation result within a margin of 10 percent.

7. Conclusions & Future Work

In Task 5.3 we focused on the development of a case repository as a reference for PA officers giving access to descriptions of previous similar situations. Therefore we built on results achieved in the MATURE project regarding the agility of processes, respectively the variety of cases. Furthermore, we draw upon research conducted in a Swiss national project (Martin et al. 2013). As learned in several workshops with representatives of the Marche Region retrieving former cases for advice on how to handle current ones is an often-applied strategy for learning. We support this by using a semantic similarity measure that exploits the ontology in order to identify the cases that are the most similar to the current case. We could show that the approach is supported not only by reusing experiences from previous cases for learning, but furthermore, by reusing the solution from a previous cases the employee can provide feedback about the usefulness for the current situation. This is done adding annotations to the learning objects in the wiki applying the methods developed in Task 5.2. The information about the context in which such feedback was given can be exploited in future case adaptations: when a user adapts a historical case, she will be provided with ratings of solution elements that other users made when they were in a similar situation. This will allow users to quickly find the most useful solution elements to take over.

As a next step, we plan to provide so-called case viewpoints that reflect different information needs in the retrieval phase. In the case of Learn PAd, viewpoints are mainly given by the progress of process execution: in an early stage of process execution, fewer details about the case are known; later, e.g. the organisational units who have been involved are known and can be part of a query - i.e. one might want to retrieve cases where similar organisational units were involved. This means that in one viewpoint this information is part of the solution and in the next viewpoint, it becomes part of the characterisation and is thus used for retrieval. This implies that:

- elements can be both part of a case characterisation and a case content / solution,
- the viewpoint concept is applied not only to the case characterisation, but also to the content.

Another challenge for future work is how to adapt the similarity model to specific information needs of a user. That means that users are empowered (via some suitable GUI elements) to influence the weights of the similarity function. This might be required because, depending on the concrete application case, certain aspects may play a bigger role than in other situations. The specific challenge for ontology-based CBR is how to find a cognitively adequate representation of how weight changes impact similarity, especially in the face of complex case characterisation with nested relationships.

Bibliography

van der Aalst, W.M.P., Weske, M. & Grünbauer, D., 2005. Case Handling: A New Paradigm for Business Process Support. *Data and Knowledge Engineering*, 53, pp.162–192.

Aamodt, A., 1991. *A Knowledge-Intensive, Integrated Approach to Problem Solving and Sustained Learning*. Trondheim, Norway, Norway: Universitetet I Trondheim.

Aamodt, A., 1994. Case-Based Reasoning : Foundational Issues, Methodological Variations, and System Approaches. *Artificial Intelligence Communications*, 7(1), pp.39–59.

Aamodt, A., 2004. Knowledge-Intensive Case-Based Reasoning in CREEK. In P. Funk & P. González Calero, eds. *7th European Conference, ECCBR 2004, Madrid, Spain*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 1–15.

Aha, D. & Breslow, L., 1997. Refining conversational case libraries. In D. Leake & E. Plaza, eds. *Case-Based Reasoning Research and Development SE - 26*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 267–278.

Aha, D., Maney, T. & Breslow, L., 1998. Supporting dialogue inferencing in conversational case-based reasoning. In B. Smyth & P. Cunningham, eds. *Advances in Case-Based Reasoning SE - 23*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 262–273.

Aha, D.W., Breslow, L.A. & Muñoz-Avila, H., 2001. Conversational Case-Based Reasoning. *Applied Intelligence*, 14(1), pp.9–32.

Baccigalupo, C. & Plaza, E., 2007. A Case-Based Song Scheduler for Group Customised Radio. In R. Weber & M. Richter, eds. *Case-Based Reasoning Research and Development SE - 30*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 433–448.

Bach, K. & Althoff, K.-D., 2012. Developing Case-Based Reasoning Applications Using myCBR 3. In B. Agudo & I. Watson, eds. *Case-Based Reasoning Research and Development SE - 4*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 17–31.

Bello-Tomás, J.J., González-Calero, P.A. & Díaz-Agudo, B., 2004. JColibri: An Object-Oriented Framework for Building CBR Systems. In P. Funk & P. González Calero, eds. *Advances in Case-Based Reasoning SE - 4*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 32–46.

Bergmann, R., Freßmann, A., et al., 2006. Case-Based Support for Collaborative Business. In T. Roth-Berghofer, M. Göker, & H. A. Güvenir, eds. *8th European Conference, ECCBR 2006 Fethiye, Turkey, September 4-7, 2006 Proceedings*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 519–533.

Bergmann, R. et al., 2003. *Developing Industrial Case-Based Reasoning Applications*, Berlin, Heidelberg: Springer Berlin Heidelberg.

Bergmann, R., 2002. *Experience Management: Foundations, Development Methodology, and Internet-Based Applications*, Berlin, Heidelberg: Springer.

Bergmann, R., 1998. On the Use of Taxonomies for Representing Case Features and Local Similarity Measures. In L. Gierl & M. Lenz (Eds.), *Proceedings of the Sixth German Workshop on CBR*.

Bergmann, R. & Gil, Y., 2011. Retrieval of Semantic Workflows with Knowledge Intensive Similarity Measures. In A. Ram & N. Wiratunga, eds. *19th International Conference on Case-Based Reasoning, ICCBR 2011*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 17–31.

Bergmann, R., Kolodner, J.L. & Plaza, E., 2006. Representation in case-based reasoning. *The Knowledge Engineering Review*, 20(03), pp.209–213.

Bergmann, R. & Mougouie, B., 2006. Finding Similar Deductive Consequences – A New Search-Based Framework for Unified Reasoning from Cases and General Knowledge.

In T. Roth-Berghofer, M. Göker, & H. A. Güvenir, eds. *Advances in Case-Based Reasoning SE - 21*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 271–285.
Bergmann, R. & Schaaf, M., 2003. Structural Case-Based Reasoning and Ontology-Based Knowledge Management: A Perfect Match? *Journal of Universal Computer Science*, 9(7), pp.608–626.
Bichindaritz, I., 2004. Mémoire: Case Based Reasoning Meets the Semantic Web in Biology and Medicine. In P. Funk & P. González Calero, eds. *Advances in Case-Based Reasoning SE - 5*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 47–61.
Cheetham, W. & Price, J., 2004. Measures of Solution Accuracy in Case-Based Reasoning Systems. In P. Funk & P. González Calero, eds. *Advances in Case-Based Reasoning SE - 9*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 106–118.
Chen, H. & Wu, Z., 2003. On case-based knowledge sharing in semantic Web. In *Proceedings. 15th IEEE International Conference on Tools with Artificial Intelligence*. IEEE Comput. Soc, pp. 200–207.
Cognini, R., Hinkelmann, K. & Martin, A., 2015. A Case Modelling Language for Process Variant Management in Case-based Reasoning. In *AdaptiveCM 2015 – 4th International Workshop on Adaptive Case Management and other non-workflow approaches to BPM*. p. 12.
Cordier, A. et al., 2008. Opportunistic Acquisition of Adaptation Knowledge and Cases - The IakA Approach. In *European Conference on Case-Based Reasoning*. Trier, Germany: Springer, pp. 150–164.
d'Aquin, M. et al., 2007. Case Base Mining for Adaptation Knowledge Acquisition. *IJCAI*, 7, pp.750–755.
Dıaz-Agudo, B. et al., 2005. On Developing a Distributed CBR Framework through Semantic Web Services. In *Workshop OWL: Experiences and Directions, at International Conference on Rules and Rule Markup Languages for the Semantic Web*.
Dıaz-Agudo, B. & González-Calero, P., 2001. Knowledge Intensive CBR Made Affordable. In *Proceedings of the Workshop Program at the Fourth International Conference on Case-Based Reasoning*.
Ehrig, M. et al., 2004. Similarity for Ontologies - a Comprehensive Framework. In *In Workshop Enterprise Modelling and Ontology: Ingredients for Interoperability, at PAKM 2004*.
van Elst, L. et al., 2003. Weakly-structured workflows for knowledge-intensive tasks: an experimental evaluation. In *WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003.* IEEE Comput. Soc, pp. 340–345.
Fidjeland, M.K., 2006. *Distributed Knowledge in Case-Based Reasoning : Knowledge Sharing and Reuse within the Semantic Web*. Norwegian University of Science and Technology.
Gómez-Martín, P. et al., 2005. Opportunities for CBR in Learning by Doing. In H. Muñoz-Ávila & F. Ricci, eds. *Case-Based Reasoning Research and Development SE - 22*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 267–281.
Greene, D. et al., 2010. An analysis of current trends in CBR research using multi-view clustering. *AI Magazine*, 31(2), pp.45–62.
Greene, D. et al., 2008. An Analysis of Research Themes in the CBR Conference Literature. In K.-D. Althoff et al., eds. *Advances in Case-Based Reasoning*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 18–43.
Hefke, M., 2004. A framework for the successful introduction of KM using CBR and semantic web technologies. *Journal of Universal Computer Science*, 10(6), pp.731–739.

Hefke, M. et al., 2006. An extendable java framework for instance similarities in ontologies. In *ICEIS*. pp. 23–27.
Hefke, M., 2008. *Ontologiebasierte Werkzeuge zur Unterstützung von Organisationen bei der Einführung und Durchführung von Wissensmanagement*. Universität Fridericiana zu Karlsruhe.
Hefke, M. & Abecker, A., 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*. pp. 534–548.
Hefke, M. & Abecker, A., 2006b. KMIR - a Knowledge Management Implementation and Recommendation Framework using CBR and Semantic Web Technologies. In *CBR and Semantic Web Technologies, Proc. MKWI-2006*.
Hevner, A.R. et al., 2004. Design Science in Information Systems Research. *MIS Q*, 28(1), pp.75–105.
Kapetanakis, S. et al., 2010. A Case Based Reasoning Approach for the Monitoring of Business Workflows. In I. Bichindaritz & S. Montani, eds. *18th International Conference on Case-Based Reasoning, ICCBR 2010, Alessandria, Italy, July 19-22, 2010. Proceedings*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 390–405.
Kaster, D.S., Medeiros, C.B. & Rocha, H. V., 2005. Supporting modeling and problem solving from precedent experiences: the role of workflows and case-based reasoning. *Environmental Modelling & Software*, 20(6), pp.689–704.
Kathleen Hanney, M.T.K., 1997. The Adaptation Knowledge Bottleneck: How to Unblock it By Learning from Cases. *Proceedings of the 2nd International Conference on CBR*, pp. 359–370
Kim, J., Suh, W. & Lee, H., 2002. Document-based workflow modeling: a case-based reasoning approach. *Expert Systems with Applications*, 23(2), pp.77–93.
Kolodner, J., 1993. *Case-based Reasoning*, Morgan Kaufmann.
Leake, D. & Kendall-Morwick, J., 2008. Towards Case-Based Support for e-Science Workflow Generation by Mining Provenance. In K.-D. Althoff et al., eds. *9th European Conference, ECCBR 2008*. Trier, Germany: Springer Berlin Heidelberg, pp. 269–283.
Leake, D.B., 1996. CBR in Context: The Present and Future. In D. B. Leake, ed. *Case-Based Reasoning: Experiences, Lessons, and Future Directions*. Menlo Park: AAAI Press/MIT Press, pp. 1–35.
Lopez de Mantaras, R. et al., 2006. Retrieval, reuse, revision and retention in case-based reasoning. *The Knowledge Engineering Review*, 20(03), p.215.
Madhusudan, T., Zhao, J.L. & Marshall, B., 2004. A case-based reasoning framework for workflow model management. *Data & Knowledge Engineering*, 50(1), pp.87–115.
Marling, C. & Whitehouse, P., 2001. Case-Based Reasoning in the Care of Alzheimer's Disease Patients. In D. Aha & I. Watson, eds. *Case-Based Reasoning Research and Development SE - 50*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 702–715.
Martin, A., Emmenegger, S. & Wilke, G., 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*. IEEE, pp. 1–12.
Maximini, K. & Maximini, R., 2007. *Collaborative Agent-Based Knowledge Engine*,
McBride, B., 2001. Jena: Implementing the RDF model and syntax specification. In *Proceedings of the 2nd International Workshop on the Semantic Web.* Hongkong.
Minor, M. et al., 2010. Agile Workflow Technology for Long-Term Processes: Enhanced by Case-Based Change Reuse. In V. Sugumaran, ed. *Methodological Advancements in Intelligent Information Technologies*. IGI Global, pp. 279–298.

Mukkamalla, S. & Muñoz-Avila, H., 2002. Case Acquisition in a Project Planning Environment. In S. Craw & A. Preece, eds. *Advances in Case-Based Reasoning SE - 20*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 264–277.
Recio-García, J.A., Díaz-Agudo, B. & González-Calero, P., 2008. jCOLIBRI2 Tutorial. , pp.1–110.
Recio-García, J.A., González-Calero, P.A. & Díaz-Agudo, B., 2012. jcolibri2: A framework for building Case-based reasoning systems. *Science of Computer Programming*, (Group 910494).
Richter, M.M., 1998. Introduction. In M. Lenz et al., eds. *Case-Based Reasoning Technology SE - 1*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 1–15.
Richter, M.M., 1995. The Knowledge Contained in Similarity Measures.
Richter, M.M. & Weber, R.O., 2013. *Case-Based Reasoning: A Textbook*, Berlin, Heidelberg: Springer Berlin Heidelberg.
Riesbeck, R.C. & Schank, C.K., 1989. *Inside case-based reasoning*, Hillsdale, New Jersey: Lawrence Erlbaum Associates.
Rinderle, S. et al., 2005. Integrating process learning and process evolution–a semantics based approach. *Business Process Management*, pp.252–267.
Rissland, E., 2006. The Fun Begins with Retrieval: Explanation and CBR. In T. Roth-Berghofer, M. Göker, & H. A. Güvenir, eds. *Advances in Case-Based Reasoning SE - 1*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 1–8.
Roth-Berghofer, T. & Bahls, D., 2008. Explanation Capabilities of the Open Source Case-Based Reasoning Tool myCBR. In *UK Workshop on Case-Based Reasoning UKCBR 2008*. pp. 23–34.
Schank, R.C., 1999. *Dynamic Memory Revisited*, Cambridge: Cambridge University Press.
Schank, R.C., 1982. *Dynamic memory: a theory of reminding and learning in computers and people*, New York: Cambridge University Press.
Schank, R.C. & Abelson, R.P., 1977. *Scripts, plans, goals, and understanding: an inquiry into human knowledge structures*, New York: Hillsdale, N.J. : L. Erlbaum Associates.
Secchi, P., Ciaschi, R. & Spence, D., 1999. A Concept for an ESA Lessons Learned System. In P. Secchi, ed. *Proceedings of alerts and lessons learned: An effective way to prevent failures and problems (Tech. Rep. WPP-167)*. Noordwijk, Netherlands, pp. 57– 61.
Sene, A., Kamsu-Foguem, B. & Rumeau, P., 2015. Telemedicine framework using case-based reasoning with evidences. *Computer Methods and Programs in Biomedicine*, 121(1), pp.21–35.
Shimazu, H., 1998. A textual case-based reasoning system using XML on the world-wide web. In B. Smyth & P. Cunningham, eds. *Advances in Case-Based Reasoning SE - 24*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 274–285.
The Open Group, 2012. ArchiMate 2.1 Specification.
Wang, Y., Hu, T. & Zhang, S., 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)*. IEEE, pp. 4878–4883.
Wargitsch, C., Wewers, T. & Theisinger, F., 1997. WorkBrain: Merging Organizational Memory and Workflow Management Systems. In *Proceedings of KI'97 Workshop on Knowledge-Based Systems for Knowledge Management in Enterprises*.
Watson, I., 1999. Case-based reasoning is a methodology not a technology. *Knowledge-Based Systems*, 12(5-6), pp.303–308.
Watson, I., 2003. Understanding case-based reasoning. In I. Watson, ed. *Applying Knowledge Management: Techniques for Building Corporate Memories*. San Francisco: Elsevier, pp. 23–46.
Weber, B. et al., 2009. Providing Integrated Life Cycle Support in Process-Aware Information

Systems. *International Journal of Cooperative Information Systems*, 18(01), pp.115–165.
Weber, B. & Wild, W., 2005. Conversational case-based reasoning support for business process management. *Architecture*.
Weber, B., Wild, W. & Breu, R., 2004. CBRFlow: Enabling Adaptive Workflow Management Through Conversational Case-Based Reasoning. In P. Funk, G. Calero, & P. A, eds. *Advances in CaseBased Reasoning*. Springer, pp. 434–448.
Weber, R., Aha, D.W. & Becerra-Fernandez, I., 2000. Categorizing Intelligent Lessons Learned Systems. In D. W. Aha & R. Weber, eds. *AAAI Technical Report WS-00-03*. Menlo Park, California: AAAI Press, pp. 63–67.
Weber, R.O., Ashley, K.D. & Bruninghaus, S., 2006. Textual case-based reasoning. *The Knowledge Engineering Review*, 20(03), p.255.
Wilke, W. et al., 1997. A Framework for Learning Adaptation Knowledge Based on Knowledge Light Approaches. In *Proceedings of the Fifth German Workshop on Case-Based Reasoning*. pp. 235–242.
William W. Cohen, P.R.S.E.F., A comparison of string distance metrics for name-matching tasks.
Witschel, H.F. et al., 2015. A new Retrieval Function for Ontology-Based Complex Case Descriptions.
Yokoyama, T., 1990. An object-oriented and constraint-based knowledge representation system for design object modeling. In *Sixth Conference on Artificial Intelligence for Applications*. IEEE Comput. Soc. Press, pp. 146–152.