Knowledge engineering within the application-independent architecture SEASALT

Meike Reichle, Kerstin Bach* and Klaus-Dieter Althoff

Intelligent Information Systems,
University of Hildesheim,
Marienburger Platz 22,
Hildesheim, 31141, Germany
Fax: 49-5121-883755
E-mail: reichle@iis.uni-hildesheim.de
E-mail: bach@iis.uni-hildesheim.de
E-mail: althoff@iis.uni-hildesheim.de
*Corresponding author

Abstract: Sharing Experience using an Agent-based System Architecture Layout (SEASALT) presents an instantiation of the Collaborating Multi-expert Systems (CoMES) approach. It integrates techniques from software engineering and combines them with artificial intelligence methodologies. The approach offers an application-independent architecture that features knowledge acquisition from a web-community, knowledge modularisation and agent-based knowledge maintenance. The paper introduces a travel medicine as application domain which applies SEASALT and describes each part of the novel architecture for extracting, analysing, sharing and providing community experiences in an individualised way.

Keywords: knowledge engineering; distributed case-based reasoning; knowledge and experience management.

Reference to this paper should be made as follows: Reichle, M., Bach, K. and Althoff, K-D. (2011) 'Knowledge engineering within the application-independent architecture SEASALT', *Int. J. Knowledge Engineering and Data Mining*, Vol. 1, No. 3, pp.202–215.

Biographical notes: Meike Reichle studied International Information Management at the University of Hildesheim from 2001 to 2007, with a semester abroad at the University of Glasgow during 2003/2004. She graduated in 2007 and is since working in a PhD position at the University of Hildesheim's Intelligent Information Systems Lab. Her main research interests include the combination of case-based reasoning and machine learning techniques, multi-agent systems, knowledge and experience management and natural language processing.

Kerstin Bach studied Information Management and Information Technology at the University of Hildesheim from 2001 to 2007. She graduated in 2007 and is since working in a PhD position at the University of Hildesheim's Intelligent Information Systems Lab. Her main research interests include intelligent information systems, agent-based systems, travel medicine, case-based reasoning, textual case-based reasoning, knowledge discovery and similarity. She is the Project Manager of the docQuery Project, a joint project of the Intelligent Information Systems Lab and mediScon worldwide.

Klaus-Dieter Althoff finished his Doctoral dissertation at the University of Kaiserslautern in 1992, from where he also received his Habilitation in 1997. From 1997 to 2004, he was responsible for the Experience Management Systems and Processes at the Fraunhofer Institute for Experimental Software Engineering. Since 2005, he is a Full Professor at the University of Hildesheim and is directing a research group on intelligent information systems. His main interests include techniques, methods and tools for developing, operating, evaluating, and maintaining knowledge-based systems, with a focus on case-based reasoning, agent technology, experience management and machine learning.

1 Introduction

The development and expansion of Web 2.0 applications in the last years has resulted in the fact that formalised and structured documents have been largely replaced by individually structured and designed documents and experiences. Instead of using ready-made forms or templates to express their opinions, Web 2.0 participants present their experiences and ideas individually – for example via blog or forum posts, on mailing lists or in wikis. In order to keep up with the development towards more sophisticated social software applications, the techniques and approaches for intelligent information systems have to develop further as well. Traditional approaches like strictly structured monolithic databases or highly specialised text mining approaches cannot deal sufficiently with the wealth of experiences provided in today's Word Wide Web.

In this paper, we present a novel architecture for extracting, analysing, sharing and providing community experiences. Our architecture is geared to real world scenarios where certain people are experts in special domains and the knowledge of more than one expert as well as the composition of a combined solution are required in order to solve a complex problem. Based on the CoMES approach (Althoff et al., 2007), we use artificial intelligence (AI) technologies to identify relevant information in web communities, process the experiences and provide them via an interface using a collaborating multi-agent architecture.

The core methodology for the realisation of Sharing Experience using an Agent-based System Architecture Layout (SEASALT), is case-based reasoning (CBR) (Aamodt and Plaza, 1994). CBR is based on the hypothesis that similar problems have similar solutions. The cases in CBR systems are represented as sets of problem descriptions and their according solutions. As described in Aamodt and Plaza (1994), CBR can be divided into four processes: retrieve, reuse, revise and retain. All cases are stored in a case base on which the retrieval is executed. In contrary to common retrieval mechanisms, in CBR a similarity based *retrieval* produces partial matches ordered by relevance to the given problem descriptions. In the *reuse* phase the given solution is adapted to the actual problem, before the suggested solution is *revised* whether it can be applied or not. The new created case is *retained* in order to further develop the case base competence. CBR has already been successfully applied in many industrial and academical applications (Bergmann et al., 2003, 2009). Moreover, CBR is a technology for reusing experiences (Plaza, 2008) and the technologies used within a CBR system can be customised according to a given domain.

Within SEASALT, we are organising the tasks to be executed in layers according to the objectives they address: A variety of *knowledge sources* from the WWW, tasks that concentrate on *knowledge formalisation* and *knowledge provision* as well as a common *knowledge representation*. Knowledge formalisation from WWW sources is carried out by different approaches (Braun et al., 2007; Brunzel and Spiliopoulou, 2008; Van Damme et al., 2007). The knowledge provision using multi-agent systems has already been implemented (Plaza and McGinty, 2006; Leake and Sooriamurthi, 2003), however we propose a combination of information retrieved from heterogeneous knowledge sources.

2 Application of SEASALT

In this section, we will introduce docQuery, a travel medicine application based on the SEASALT approach. Examples from this application domain will be used to exemplify different aspects of the SEASALT architecture.

Travel medicine is the prevention, management and research of health problems associated with travel. Those aspects play a major role alongside with individual aspects concerning the health status of the traveller and the desired destination. Therefore, information about the traveller's home region as well as the destination region, the activities planned and additional conditions have to be considered when giving medical advice. Travel medicine starts when a person moves from one place to another by any kind of transportation and ends after returning home healthy. In case a traveller gets sick after a journey, a travel medicine consultation might also be required.

Nowadays, it is easier than ever to travel to different places, experience new cultures and get to know new people. In preparation for a healthy journey, it is important to get a high quality and reliable answer on travel medicine issues. Both laymen and experts should get information they need and, in particular, they understand. For that reason, we would like to introduce docQuery – a medical information system for travellers. Whether somebody travels frequently or occasionally, on business or for leisure, individually or with the whole family, docQuery should be able to provide individualised knowledge. The docQuery project focuses on high quality information that can be understood by everybody and maintained by a number of travel medicine experts, supported by intelligent methods executed by agents. Furthermore, the various and heterogeneous fields require independently organised knowledge sources. In comparison to traditional approaches that mostly rely on one monolithic knowledge resource, the docQuery system will adapt to the organisation of knowledge given by the expert. An analysis of the expert's tasks shows that the information gathered from different channels (mailing lists, web forums, literature) has to be organised, analysed and synthesised before it can be provided. docQuery concentrates on a web community in which experts exchange and provide qualified information. We decided on creating an expert community instead of a community of users because the medical aspects of our domain require reliable information. Nevertheless, the contributors are not only medical practitioners, they are also expatriates or frequent travellers. The information provided can be compared with information someone receives while asking friends instead of doing a survey without knowing who is answering certain questions. Users in our expert community share their experiences while explaining different aspects on a web forum. We further on can use these discussions as cases within our CBR system(s).

docQuery can be used by inserting the key data on a travellers journey (like travel period, destination, age(s) of traveller(s), activities, etc.) and the system will prepare an individual composed information leaflet right away. The traveller can take the information leaflet to a general practitioner to discuss the planned journey. The leaflet will contain all the information needed to be prepared and provide detailed information if they are required. In the event that docQuery cannot answer the travellers question, the request will be sent to the expert community who will answer it. The reference query for the remaining part of the article will be the following:

> "A German family consisting of a 34 year old man, a 32 year old woman, a 12 year old boy and a 8 year old girl are planning to travel to Alor to dive and afterwards they will travel around Bali by car from March 8th to March 20th. And they would like to know all relevant information that they need for their journey."

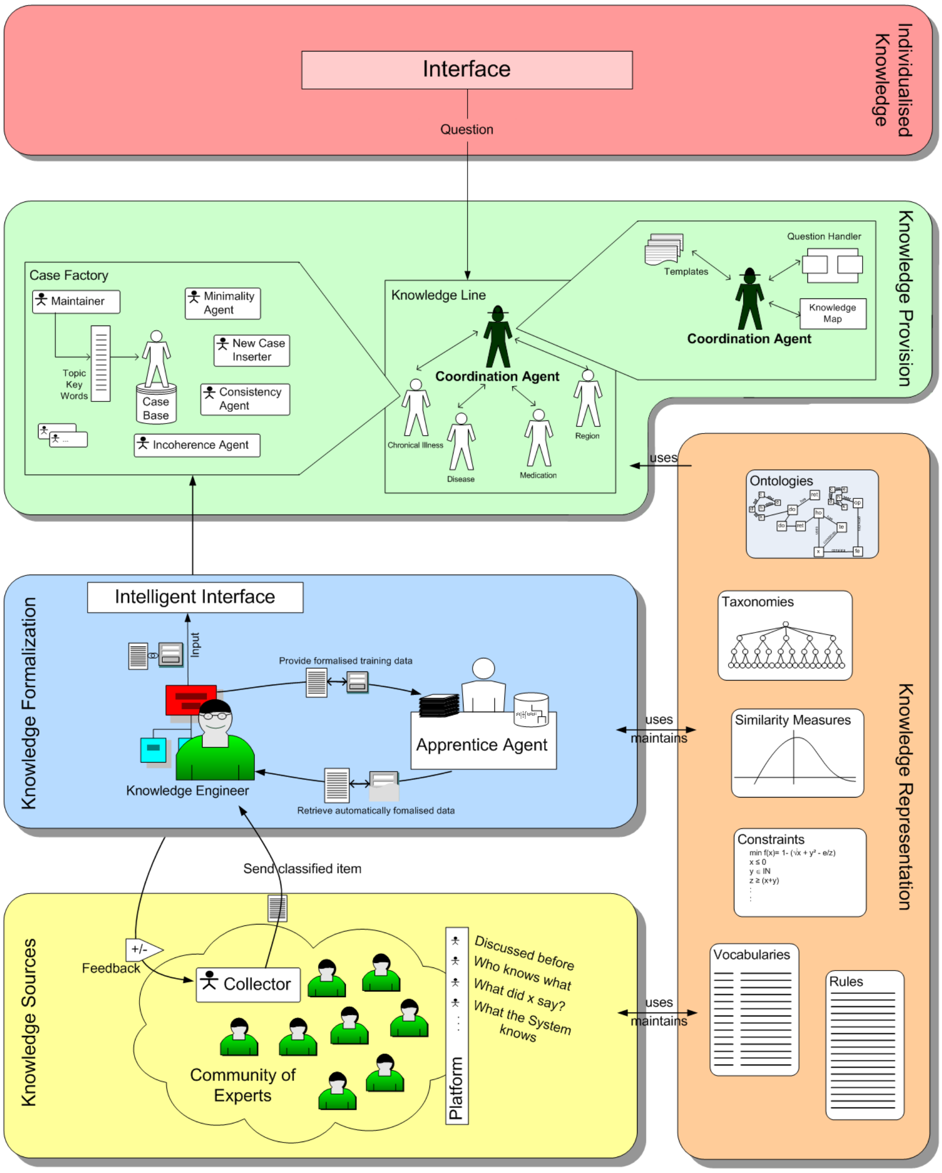
3 The SEASALT architecture

The SEASALT architecture provides an application-independent architecture that features knowledge acquisition from a web-community, knowledge modularisation and agent-based knowledge maintenance. It consists of several components which will be presented in the following sections, ordered by their role within general information management and exemplified using the docQuery project. Figure 1 shows the architecture and its five main components. The following subsections will each focus on one component and explain the ideas, applied methods and techniques behind them.

3.1 Knowledge sources

An interdisciplinary application domain such as travel medicine needs to draw information from numerous knowledge sources in order to keep up to date. Beyond traditional knowledge sources such as databases and static web pages the main focus of SEASALT are Web 2.0 platforms. The SEASALT architecture is especially suited for the acquisition, handling and provision of experiential knowledge as it is provided by communities of practice and represented within Web 2.0 platforms (Plaza, 2008). Within our implementation of SEASALT we used a web forum software that was enhanced with agents for several different purposes. We chose a forum since it is a broadly established WWW communication medium and provides a low entry barrier even to only occasional WWW users. Additionally, its contents can be easily accessed using the underlying database. The forum itself serves as a communication and collaboration platform to the travel medicine community, which consists of professionals such as scientists and physicians who specialise in travel medicine and local experts from the health sector and private persons such as frequent travellers and globetrotters. The community uses the platform for sharing experiences, asking questions and doing general networking. The forum is enhanced with agents that offer content-based services such as the identification of experts, similar discussion topics, etc., and communicate by posting relevant links directly into the respective threads such as in Feng et al. (2006).

Figure 1 The SEASALT architecture, the individual components are grouped into layers according to their function in knowledge management (see online version for colours)



The community platform is monitored by a second type of agents, the so called collector agents. These agents are individually assigned to a specific topic agent (see Section 3.2), their task is to collect all contributions that are relevant with regard to their assigned topic agent's topic. The collector agents pass these contributions on to the knowledge engineer and can in return receive feedback on the delivered contribution's relevance. Our collector agents currently use the information extraction tools TextMarker (Klügl et al., 2008) and GATE (Cunningham et al., 2002), to automatically judge the relevance of a contribution. The knowledge engineer reviews each collector agent's collected contributions and implements his or her feedback by directly adjusting the agents' rule base.

The SEASALT architecture is also able to include external knowledge sources by equipping individual collector agents with database or web service protocols or HTML crawling capabilities. This allows the inclusion of additional knowledge sources such as the web pages of the Department of Foreign Affairs or the WHO. Depending on the data interfaces of external resources we either treat them as a community or a topic agent. An external resource can be integrated as a topic agent if structured and with metadata enriched sources, for example well described web sources, should be included. If the data received from an external resource is unstructured or semi-structured text, we treat it as web community without collector agents and apply the regular SEASALT knowledge formalisation.

3.2 Knowledge formalisation

In order for the collected knowledge to be easily usable within the knowledge line (see Section 3.3) the collected contributions have to be formalised from their textual representation into a more modular, structured representation.

This task is mainly carried out by the knowledge engineer. In the docQuery project the role of the knowledge engineer is carried out by several human experts, who work together on the knowledge engineer's tasks. The knowledge engineer is the link between the community and the topic agents. He or she receives posts from the collectors that are relevant with regard to one of the fields, represented by the topic agents and formalises them for insertion into the topic agents' knowledge bases using the intelligent interface.

The intelligent interface serves as the knowledge engineer's case authoring work bench for formalising textual knowledge into structured CBR cases. It has been developed analogous to Bach (2007b) and offers a graphical user interface that presents options for searching, browsing and editing cases and a controlled vocabulary. The knowledge engineer's case authoring work bench uses the knowledge representations like ontologies or taxonomies to classify contributions. The according classification extraction suggestions are based on previous experiences and presented to the knowledge engineer along with the source contribution. Now it is the task of the knowledge engineer to proofread the automatic assignments and enhance the case with information gathered from the source data.

In the future, the knowledge engineer will be additionally supported by the apprentice agent. The apprentice agent is meant to support the knowledge engineer in formalising relevant posts for insertion in the topic agents' knowledge bases. It is trained by the knowledge engineer with community posts and their formalisations. The apprentice agent is currently being developed using GATE (Cunningham et al., 2002) and RapidMiner (Mierswa et al., 2006). We use a combined classification/extraction approach that first classifies the contributions with regard to the knowledge available within the individual contributions using term-doc-matrix representations of the contributions and RapidMiner, and then attempts to extract the included entities and their exact relations using GATE. Considering docQuery's sensitive medical application domain we only use the apprentice agent for preprocessing. All its formalisations will have to be reviewed by the knowledge engineer, but we still expect a significantly reduced workload for the knowledge engineer(s).

3.3 Knowledge provision

SEASALT's knowledge provision is realised using the knowledge line approach (Bach et al., 2008). The knowledge line's basic idea is a modularisation of knowledge analogous to the modularisation of software in the product line approach within software engineering (van der Linden et al., 2007). Within the SEASALT architecture, this knowledge modularisation happens with regard to individual topics that are represented within the respective knowledge domain. Within the docQuery application domain travel medicine, we identified the following topics: geography, diseases, pharmaceutics, constraints caused by chronic illnesses, vacation activities, local health facilities and local safety precautions. Modularising the system's knowledge base in such a way offers several advantages: modularisation in general serves to reduce redundancies and also increase system performance. A modularisation with respect to topics offers several additional advantages. Firstly, it is more easily comprehensible to human experts handling and co-designing a new system. Secondly, it facilitates maintenance since it offers the possibility to include experts on individual topics and have them maintain an individual knowledge base instead of the whole knowledge-based system. This is a particular advantage in interdisciplinary application domains which are usually highly specialised and only dealt with by a rather small number of experts.

Once the application domain has been modularised, the individual topics are represented by so-called topic agents. According to the SEASALT architecture, the topic agents can be any kind of information system or service including CBR systems, databases, web services, etc. External resources are also equipped with a software agent, so they can be integrated in the knowledge map and considered by the coordination agent during the distribution of retrieval queries.

Within the docQuery project, we used the empolis Information Access Suite e:IAS (empolis GmbH, 2005), an industrial-strength CBR system, for realising the individual agents (Althoff et al., 2008).

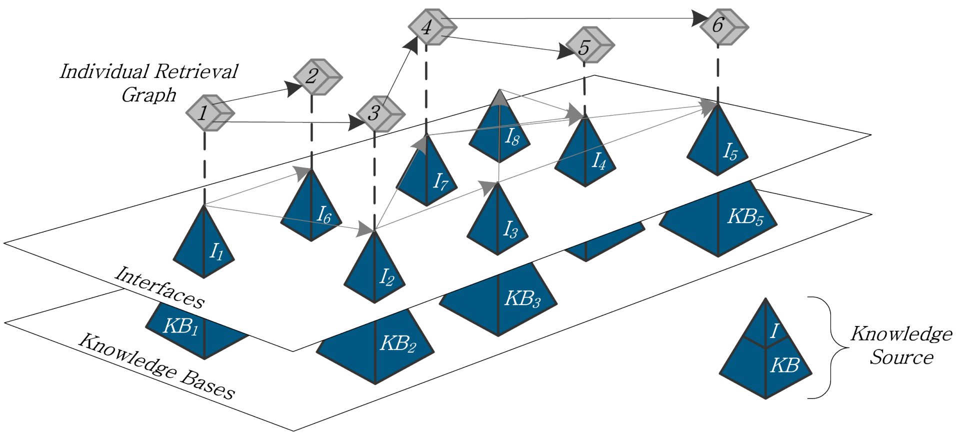
We additionally extended the topic agents' CBR systems with so-called case factories, which take care of the individual agents' knowledge maintenance. With respect to CBR systems this means mostly maintenance of the respective case bases. Further knowledge within these systems, such as for instance similarity measures and adaptation knowledge are currently still maintained manually. The case factory approach (Althoff et al., 2006) is a continuation of the experience factory (Basili et al., 1994), a reuse-oriented infrastructure for software engineering projects. It picks up on the underlying ideas of the experience factory and applies them to the maintenance of knowledge-based systems. The result is a distributed maintenance approach in which the complex maintenance task is split up into several simpler subtasks which are each carried out by individual agents (Bach, 2007a).

There are two main advantages to this approach. On the one hand, the splitting into simpler subtasks allows for an easier automation of the individual tasks. On the other hand, some of these subtasks represent conflicting aims (for instance completeness vs. minimality). Configuring the priority of the representing agents the overall maintenance is very flexible and can easily be adjusted according the individual needs of the respective topic agents.

However, decentralised knowledge also brings some challenges. Especially contradicting information and various ways of distributing knowledge. Contradicting information might occur while retrieving information from different sources and combining them. SEASALT uses so called inter and intra case base dependencies for defining constraints and therewith resolving conflicts. Different possibilities for structuring knowledge might lead to a modularisation driven by the knowledge engineers point of view or information that can be stored in more than one knowledge base. Our experiences are that the more clear and documented each attribute is, the easier the according case base for new information can be assigned.

The topic agents are orchestrated by a central coordination agent. The coordination agent receives a semi-structured natural language query from the user, analyses it using a rule-based question handler and subsequently queries the respective topic agents using incremental reasoning, that is using one agent's output as the next agent's input. In doing so, the coordination agent's course of queries resembles the approach of a human amateur trying to answer a complex travel medical question. Confronted, for instance, with the question *"Which safety precaution should I take if I want to go diving in Alor for two weeks around Easter?"* and being no expert on the field an average person would first consult someone or something in order to find out that Alor is an Indonesian island. Reading up on Indonesia, the person would then find out that the rainy season in Indonesia ends around Easter and that there is a heightened risk of malaria during that time. The person would then look up information on malaria and find out that the risk of contracting malaria can be significantly reduced using prophylactic drugs. Knowing this, he/she would then go on and acquire information on malaria preventions and so on and so forth. This approach is mimicked by the coordination agent's incremental reasoning approach.

Figure 2 The knowledge map (see online version for colours)



Note: Bottom to top: the knowledge sources with their interfaces, possible routes through the knowledge map and the route chosen by the coordination agent.

The knowledge needed in order to carry out this incremental reasoning process is represented within the so called knowledge map, which provides formal representations of all topic agents and possible output/input connections encoded in a graph-like structure. As depicted in Figure 2, the graph is constructed in such a way that every node represents a topic agent consisting of a knowledge source. Each knowledge source contains the knowledge base and an interface that interacts with the other knowledge sources and the coordination agent. A knowledge source can be a CBR system as well as a web service underneath a software agent that realises the collaboration between the agents.

The directed edges connecting the nodes denote input/output relations, signifying that one topic agent's output can serve as another agent's input. Thus a complete retrieval run is a path through that graph whereas the actual route is chosen with respect to different optimisation criteria. Based on the knowledge map we then use a modified Dijkstra (1959) algorithm to determine an optimal route over the graph. If confronted with redundant knowledge sources, the algorithm tries to optimise the path according to the knowledge sources respective properties. To that end, the algorithm is modified in such a way that it optimises its route by trying to maximise the arithmetic mean of all queried nodes. In the case of a tie between two possible routes, the one with the lesser variance is chosen.

The coordination agent's implementation is described in detail in Bach et al. (2008), its theoretic foundations are described in Reichle et al. (2009). Finally, the coordination agent uses the query results and prefabricated templates to compose an information leaflet to be given to the user. This leaflet comes as a simple PDF file and can be printed out and taken along to a consultation with a general practitioner. The leaflet is not meant to replace a professional consultation by a trained medical doctor, instead it is meant to be a reliable information resource that can be taken to the travellers medical practitioner, because the vaccinations and medication suggestions have to fit the patient's medical history.

3.4 Knowledge representation

Since the miscellaneous agents operating on the community platform (see Section 3.1), the knowledge engineer's tools (see Section 3.2 and the CBR systems of the individual topic agents (see Section 3.3) deal with the same knowledge domain(s), it makes sense to join their underlying knowledge models. This does not only greatly facilitate knowledge model maintenance but also allows for an easier interoperability between the individual components.

The basic knowledge representation has been created based on general resources like WordNet (http://wordnet.princeton.edu/) and further developed during the runtime of the application. SEASALT's knowledge representation includes rules, vocabulary, ontologies and taxonomies. As described in Roth-Berghofer (2004) and Weber and Wu (2004) within the field of CBR the so-called knowledge container of Richter (1998) can be filled using these representations. The *vocabulary* of an application domain is described as system defining knowledge container that covers the general knowledge like attributes, entities, etc., of an application domain. The vocabulary has to be created either using already existing ontologies by extracting and providing terms or written by hand in very specialised application domains. Further on, the vocabulary can be (semi-)automatically extended using web mining and ontology mining technologies enabling a knowledge-based system extending the coverage of its application domain. Mika (2005), for example, describes how vocabulary derived from folksonomies that were created based social tagging communities can be extracted.

Similarity measures, which contain knowledge about the degree of difference between cases, concepts or occurring terms of cases, can be derived from ontologies as well. The relations between terms can be interpreted as similarity between them, because they, for instance, describe is-a-relations, subclasses, super-classes or synonyms. Based on this, knowledge selection preferences can be assessed at retrieval time. Further on, user-generated content (UGC) also provides knowledge that can be used to create similarity measures: Milne et al. (2009) and Plaza and Baccigalupo (2009), for example, use user tags and the resulting folksonomies to develop similarity measures between cases and/or concepts.

Within our current implementation, we use hand-crafted case representations that are automatically instantiated. As described by Smyth et al. (2009), UGC can be used to create cases. Currently, we prefer a semi-automatic process that integrates the knowledge engineer who checks the cases in order to ensure a certain quality of the extracted cases.

Depending on application domain and topic (of the topic agents), *completion and adaptation rules* either have to be hand crafted (like medications and vaccinations in the docQuery application domain or can be extracted from a web community as well (e.g., Ihle et al., 2009).

Parts of the knowledge representation for the purpose of the docQuery project were handmade, some are external, such as for instance WordNet or ICD10 (http://www.who.int/classifications/icd/en/). The knowledge representations grow incrementally while new cases are extracted from the web community and unknown terms are collected and modelled using the knowledge engineer's tools (see Section 3.2). Currently, we focus on combining CBR systems as underlying knowledge sources for the topic agents. Since each topic agent has its own case factory, the knowledge representations also develop while case factory agents evolve each case base by inserting new cases and transferring knowledge between the knowledge containers according to (Richter, 2000).

3.5 Individualised knowledge

The user interacts with an application that is built based on SEASALT via a web-based interface. The web-based interface offers a semi-structured input in the form of different text fields used for entering information on the destination, the traveller, the time of travel and so on. Depending on how many information has to be inserted, also a dialogue-based interface can be implemented. A web interface that is based on dialogues for gathering information is especially useful, because interdependencies between information can be obtained. Further, the user is guided through the dialogue and contradicting information can be prevented. Nevertheless, a dialogue always restricts the scope of information which might lead to misunderstandings. Within docQuery, a traveller only has to provide five different kinds of information, so we decided to use form-like mask.

After the requested information has been produced by the system it has to be presented to the user. The docQuery system provides individualised knowledge to its users by generating information leaflets as PDFs that only include information which is relevant to the respective user and its journey and can be used by the traveller to consult a physician for final advice and prescriptions. We decided to generate PDFs because it is similar to the information travellers are used to. But if the SEASALT-based application runs only on the WWW an appropriate and interactive presentation of the results is also possible. The PDFs are created using previously defined templates and the templates are filled with the individual information retrieved from the coordination agent.

4 Evaluation

A comparative evaluation of the SEASALT architecture in general is difficult, since the tasks within the docQuery project were executed manually until we started to introduce the system. Also, we think that a purely local evaluation with regard to performance and runtime would be of little value to fellow researchers. Because of this, we chose to do a practical evaluation within our first application domain travel medicine. The domain required a modularisation of knowledge sources because the practitioners do not only use medical knowledge, but also need for instance regional and political information. This requirement is met by the concept of the knowledge line, because the topic agents represent an expert and their collaboration for the composition of information leaflets. Especially, the regional and political information have to be up-to-date and therefore we are able to extract knowledge about such topics from web communities and provide them within docQuery. The developed knowledge acquisition process optimises the time until this information is available. Our application partner's current best practice is the manual assembling of information leaflets, mostly copy-pasting recurrent texts (like general information and warnings) from prepared templates and external sources. The application partner has been compiling these information leaflets for several years and has in the meantime optimised the process as far as possible. Using this approach, a trained medical practitioner needs about an hour to create a complete leaflet. First tests have shown that the docQuery system offers a significant time saving and takes a lot of repetitive tasks from the medical practitioner. Even when counterchecking every generated leaflet and, if necessary, adding corrections or additional information the process of composition of information leaflets is significantly accelerated using docQuery.

5 Summary and outlook

In this paper, we presented the SEASALT architecture and described and exemplified its individual components using SEASALT's first instantiation, the docQuery project. The SEASALT architecture offers several features, namely knowledge acquisition from Web 2.0 communities, modularised knowledge storage and processing and agent-based knowledge maintenance. SEASALT's first application within the docQuery project yielded very satisfactory results, however, in order to further develop the architecture, we are planning to improve it in several areas. One of these are the collector agents working on the community platform, which we plan to advance from a rule-based approach to a classification method that is able to learn from feedback, such as for instance CBR, so more workload is taken off the knowledge engineer. Also to this end, more work will go into the apprentice agent, which is currently being developed. Another area of research that we currently look into are trust and provenance of information. SEASALT incorporates information from a large number of sources and we are currently looking into methods for making the source of the individual pieces of information more transparent to docQuery's users and thus improve the system's acceptance and trustworthiness. Finally, we are planning to also apply the architecture in other application scenarios, e.g., supporting psychological scientists in cognitive modelling (Newo and Althoff, 2008) or decision support in the nutrition domain (Ihle et al., 2009), in order to further develop it and also ensure its general applicability in different application scenarios.

References

Aamodt, A. and Plaza, E. (1994) 'Case-based reasoning: foundational issues, methodological variations and system approaches', *AI Communications*, Vol. 1, No. 7.

Althoff, K-D., Bach, K., Deutsch, J-O., Hanft, A., Mänz, J., Müller, T., Newo, R., Reichle, M., Schaaf, M. and Weis, K-H. (2007) 'Collaborative multi-expert-systems – realizing knowledge-product-lines with case factories and distributed learning systems', in J. Baumeister and D. Seipel (Eds.): *KESE @ KI 2007*, Osnabrück.

Althoff, K-D., Hanft, A. and Schaaf, M. (2006) 'Case factory – maintaining experience to learn', in M. Göker and T. Roth-Berghofer (Eds.): *Proc. 8th European Conference on Case-Based Reasoning (ECCBR'06)*, Ölüdeniz/Fethiye, Turkey, Springer Verlag, Berlin, Heidelberg, Paris.

Althoff, K-D., Reichle, M. and Bach, K. (2008) 'Realizing modularized knowledge models for heterogeneous application domains', in P. Perner (Ed.): *Advances in Data Mining, Proceedings of the 8th Industrial Conference on Data Mining (ICDM'08), LNAI*, Vol. 5077, pp.114–128, Springer.

Bach, K. (2007a) 'docQuery – a medical information system for travellers', Internal project report.

Bach, K. (2007b) 'Domänenmodellierung im textuellen fallbasierten schließen', Masters thesis, Institute of Computer Science, University of Hildesheim.

Bach, K., Reichle, M., Reichle-Schmehl, A. and Althoff, K-D. (2008) 'Implementing a coordination agent for modularised case bases', in M. Petridis and N. Wiratunga (Eds.): *Proc. of 13th UKCBR @ AI'2008*, pp.1–12.

Basili, V.R. Caldiera, G. and Rombach, H.D. (1994) 'Experience factory', in J.J. Marciniak (Ed.): *Encyclopedia of SE*, Vol. 1, John Wiley & Sons.

Bergmann, R., Althoff, K-D., Breen, S., Göker, M.H., Manago, M., Traphöner, R. and Wess, S. (2003) 'Developing industrial case-based reasoning applications: the INRECA-methodology', *LNCS Springer chapter Selected Applications of the Structural Case-Based Reasoning Approach*, Vol. 1612, pp.35–70.

Bergmann, R., Althoff, K-D., Minor, M., Reichle, M. and Bach, K. (2009) 'Case-based reasoning – introduction and recent developments', *Künstliche Intelligenz: Special Issue on Case-Based Reasoning*, Vol. 23, No. 1, pp.5–11.

Braun, S., Schmidt, A., Walter, A., Nagypal, G. and Zacharias, V. (2007) 'Ontology maturing: a collaborative Web 2.0 approach to ontology engineering', in N. Noy, H. Alani, G. Stumme, P. Mika, Y. Sure and D. Vrandecic (Eds.): *Proc. of the Workshop on Social and Collaborative Construction of Structured Knowledge*.

Brunzel, M. and Spiliopoulou, M. (2008) 'Discovering groups of sibling terms from web documents with XTREEM-SG', *J. Data Semantics*, Vol. 11, pp.126–155.

Cunningham, H., Maynard, D., Bontcheva, K. and Tablan, V. (2002) 'GATE: a framework and graphical development environment for robust NLP tools and applications', in *Proc. of the 40th Meeting of the Association for Computational Linguistics*.

Dijkstra, E.W. (1959) 'A note on two problems in connexion with graphs', *Numerische Mathematik*, Vol. 1, pp.269–271.

empolis GmbH (2005) *Technical White Paper e:Information Access Suite*, Technical report, empolis GmbH.

Feng, D., Shaw, E., Kim, J. and Hovy, E. (2006) 'An intelligent discussion-bot for answering student queries in threaded discussions', in *IUI '06: Proc. of the 11th Intl. Conference on Intelligent User Interfaces*, pp.171–177, ACM Press.

Ihle, N., Newo, R., Hanft, A., Bach, K. and Reichle, M. (2009) 'CookIIS – a case-based recipe advisor', in S.J. Delany (Ed.): *Workshop Proceedings of the 8th International Conference on Case-Based Reasoning*, pp.269–278, Seattle, WA, USA.

Klügl, P., Atzmüller, M. and Puppe, F. (2008) 'Test-driven development of complex information extraction systems using TextMarker', in G.J. Nalepa and J. Baumeister (Eds.): *KESE, CEUR Workshop Proceedings CEUR-WS.org*, Vol. 425.

Leake, D.B. and Sooriamurthi, R. (2003) 'Dispatching cases versus merging case-bases: when MCBR matters', in *Proceedings of the Sixteenth International Florida Artificial Intelligence Research Society Conference, FLAIRS-2003*, pp.129–133.

Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M. and Euler, T. (2006) 'YALE: rapid prototyping for complex data mining tasks', in L. Ungar, M. Craven, D. Gunopulos and T. Eliassi-Rad (Eds.): *KDD'06: Proc. of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.935–940, ACM.

Mika, P. (2005) 'Flink: Semantic Web technology for the extraction and analysis of social networks', *Web Semantics: Science, Services and Agents on the World Wide Web*, Vol. 3, Nos. 2–3, pp.211–223, selected papers from the *International Semantic Web Conference 2004 – ISWC 2004*.

Milne, P., Wiratunga, N., Lothian, R. and Song, D. (2009) 'Reuse of search experience for resource transformation', in S.J. Delany (Ed.): *Workshop Proceedings of the 8th International Conference on Case-Based Reasoning*, pp.45–54, Seattle, WA, USA.

Newo, R. and Althoff, K-D. (2008) 'Learning to cope with critical situations – an agent based approach', in G.J. Nalepa and J.B. Kaiserslautern (Eds.): *Proceedings of the 4th Workshop on Knowledge Engineering and Software Engineering (KESE2008) at the 31st German Conference on Artificial Intelligence (KI 2008)*, pp.55–60, Germany.

Plaza, E. (2008) 'Semantics and experience in the future web', in *ECCBR '08: Proceedings of the 9th European conference on Advances in Case-Based Reasoning*, pp.44–58, Springer-Verlag, Berlin, Heidelberg.

Plaza, E. and Baccigalupo, C. (2009) 'Principle and praxis in the experience web: a case study in social music', in S.J. Delany (Ed.): *Workshop Proceedings of the 8th International Conference on Case-Based Reasoning*, pp.45–54, Seattle, WA, USA.

Plaza, E. and McGinty, L. (2006) 'Distributed case-based reasoning', *The Knowledge Engineering Review*, Vol. 20,No. 3, pp.261–265.

Reichle, M., Bach, K., Reichle-Schmehl, A. and Althoff, K-D. (2009) 'Management of distributed knowledge sources for complex application domains', in K. Hinkelmann and H. Wache (Eds.): *Proc. 5th Conference on Professional Knowledge Manegement – Experiences and Visions (WM2009)*.

Richter, M.M. (1998) 'Introduction', in M. Lenz, B. Bartsch-Spörl, H-D. Burkhard and S. Wess (Eds.): *Case-Based Reasoning Technology – From Foundations to Applications, LNAI*, Vol. 1400, Springer-Verlag, Berlin.

Richter, M.M. (2000) 'Fallbasiertes schliessen', in J.S.G. Görz and C-R. Rollinger (Eds.): *Handbuch der Künstlichen Intelligenz*, pp.407–430, Oldenbourg Wissenschaftsverlag, Oldenbourg.

Roth-Berghofer, T.R. (2004) 'Explanations and case-based reasoning: foundational issues', in P. Funk and P.A. González Calero (Eds.): *Advances in Case-Based Reasoning*, pp.389–403, Springer-Verlag, Berlin, Heidelberg, Paris.

Smyth, B., Champin, P-A., Briggs, P. and Coyle, M. (2009) 'The case-based experience web', in S.J. Delany (Ed.): *ICCBR 2009 Workshop Proc., Workshop Reasoning from Experiences on the Web*, to appear.

Van Damme, C., Hepp, M. and Siorpaes, K. (2007) 'FolksOntology: an integrated approach for turning folksonomies into ontologies', in *Bridging the Gep between Semantic Web and Web 2.0 (SemNet 2007)*, pp.57–70.

van der Linden, F., Schmid, K. and Rommes, E. (2007) *Software Product Lines in Action – The Best Industrial Practice in Product Line Engineering*, Springer, Berlin, Heidelberg, Paris.

Weber, R. and Wu, D. (2004) 'Knowledge management for computational intelligence systems', *IEEE International Symposium on High-Assurance Systems Engineering*, pp.116–125.