Exploiting Context Information for Identification of Relevant Experts in Collaborative Workplace-Embedded E-Learning Environments

Robert Lokaiczyk[1], Eicke Godehardt[2], Andreas Faatz[1], Manuel Goertz[1], Andrea Kienle[3], Martin Wessner[4]

[1] SAP Research CEC Darmstadt
Bleichstr. 8, 64283 Darmstadt, Germany
{robert.lokaiczyk, andreas.faatz, manuel.goertz} @sap.com

[2] Fraunhofer IGD
Fraunhoferstr. 5, 64283 Darmstadt, Germany
eicke.godehardt@igd.fraunhofer.com

[3] Fraunhofer IPSI
Dolivostr. 15, 64293 Darmstadt, Germany
andrea.kienle@ipsi.fraunhofer.com

[4] Ludwig-Maximilian-University
Leopoldstr. 13, 80802 Munich, Germany
martin.wessner@psy.lmu.de

Abstract. This work introduces an approach to discover collaboration partners and adequate advising experts in a workplace-embedded collaborative e-learning environment. Based on existing papers dealing with work task and user context modeling, we propose the following steps towards a sucessful collaboration initiation. In the beginning, the user's current process task needs to be identified (1). Taking into account the knowledge about the current process, availability of experts as well as organizational and social distance, relevant experts regarding the actual work task of the learner are pre-selected by the environment (2). Depending on the pre-selection and users' preferences, the potential collaboration partners are displayed in an expert list (3). That way, the learner is able to initiate beneficial collaborations, whose transcripts are used to enhance the existing knowledge base of learning documents (4).

1 Introduction

Frequently changing work contexts, transient processes, short product life-cycles and a rapidly changing world in a dynamic knowledge economy create the need for continuing and lifelong employees' training. Companies believe in workplace-embedded learning solutions to cope with the increasing complexity. Different from the traditional 'learn first, apply later'-approach, the required knowledge for solving the current work task is needed right in time during the work process. The user can immediately profit from the learning content that is provided by an embedded e-learning environment. Besides presenting learning resources, potential experts of the work task will be pre-selected by the environment for possible collaboration and discussion. The community of business process oriented knowledge management is convinced that users will benefit to a high degree from the integration of e-learning into business processes. Learning on the fly while performing a task can occur when collaborating, e.g. when users conjointly deal with a specific work task. Within this process, a user might play different roles. In a general role of a knowledge worker, the user is working in a knowledge and information intensive work process. As a learner, s/he acquires further know-how by reading documents or collaborating with experts. Last, as an expert, the user shares know-how with others and acts as a teacher. [12] points out that the borders between the mentioned roles are blurred. Depending on the current business process, users dynamically transfer between the different roles. As soon as a knowledge worker has to solve a subtask without having the necessary knowledge, s/he becomes a learner. Having knowledge in some further area, s/he supports others in performing a task as a teaching expert. In the following, we address the identification and selection problem of potential experts for a specific task. The example was added for illustration of our problem:

Example: It is the first time that Anna performs a specific business process. Until now, she did not gain necessary knowledge how to deal with the problem. Indeed, the existing knowledge database provides documents related to the problem. Nevertheless, this information is not sufficient to perform the task in an effective way. She depends on experts and teachers directing her through the process. But who are those experts and how can she find them in the company?

Both the current business process and the learner (Anna) influence the categorization of teachers' expertise within a specific topic. In our paper, two approaches for business process and context modeling will be introduced. Section 3 describes the contextualized initiation of cooperation between learners and teachers by means of the example above. More detailed, subsection 3.1 points out how the current process step (task) can be determined. The process of identification and ranking of potential experts is illustrated in 3.2 and 3.3. Subsection 3.4 introduces a way to integrate and reuse the identified collaboration knowledge into the learning solution. An implementation of the approach will be described in section 4. The paper ends with a short summary and future work in section 5.

2 Related Work

Recently, several approaches for knowledge and business process integration have been developed. Additionally to the processes, business-process oriented knowledge management has to consider and model the users of the knowledge system plus the context of use. Below, several works dealing with process and context models will be mentioned and described in a brief way. Hardly any of the authors in this area raises the issue of collaborative and workplace-embedded e-learning.

Furthermore, there is a lack of concepts how to integrate the arising knowledge during collaborations into the knowledge solutions.

2.1 Process Modelling

Process modeling provides background information for determining in which working step experts and learners currently are and which tasks they have already worked on. Moreover, businesses applied process modeling as part of their workflow management systems over the last decade. This means that workplace-embedded learning has to consider process modeling as an essential part of a realistic application scenario. Van Welie defines a "task" as an necessary activity to achieve a specific goal [22]. Existing works in task modeling can be assigned either to event-based or state-based models. In the following, we will deepen one representative for each class.

The Business Process Modeling Notation (BPMN) [15] is a standardized graphical notation for drawing business processes in a workflow. BPMN is highly related to UML-modeling. Beyond a coverage of activities and their temporal and logical constraints, the language allows to group activities which are logically related to each other by swim lanes. Artifacts are mainly data objects. Data objects are typed and represent the input and output of activities. BPMN is a representative for event-based modeling languages defining events and activities as continuous elements.

A Petri net is one of several mathematical representations of discrete distributed systems. As a modeling language, it graphically depicts the structure of a distributed system as a directed bipartite graph with annotations. As such, a Petri net has place nodes, transition nodes, and directed arcs connecting places with transitions. Places may contain a number of tokens. Transitions act on input tokens by a process known as firing. A transition is enabled if it can fire, i.e., there is the defined number of tokens in every input place and the output places are able to store the new tokens. Typed events can be expressed by multi-colored tokens. Van der Aalst discusses Petri nets for work process modeling in [1]. In contrast to BPMNs, Petri nets are state based. Beyond events and activities, the current state is modeled in form of token assignment.

2.2 Work and Usercontext

Besides the formalized work process model, there are more indicators of the working users' context, which can be exploited for searching experts. The following paragraph presents related work in two different areas regarding context dependant expert identification for workplace-oriented collaborative learning. On the one hand, the task context of the learner has to be recognized since it highly influences the pre-selection of experts. On the other hand, the users' context is considered in a broader scope to show how it can influence to identification of suitable resources and experts.

Task Context CALVIN [2] is a system considering the task context. Bauer and Leake define the task context as a term-vector-description of the current document. Using a difference analysis, the Wordsieve system analyses sets of terms over time. Task switches can thus be recognized by considering a difference threshold over the term sets. The system exclusively performs document based using the content of a web browser window. [9] expands the definition of task context by the factors complexity, challenges and dependencies. Bayesian belief models indicate suitable moments for disruption of the work process. The structuring and categorization of the process into sub-tasks is done manually by experts. The Pinpoint system [3] provides task-specific document recommendations. Task recognition in an automated way is not intended. The task is manually selected in a list created by domain experts. In a nutshell, existing systems deal with user support in recommendation environments usually without automatic task recognition.

User Context Apparently, the user's context usually reaches far beyond the current task context. Regarding the context driven expert identification, facts like existing qualifications, experiences with the system and available tools or preferences influence the selection process. Certainly, those facts will be included during the expert selection beyond the common task context.

Existing systems designed for expert recommendations are currently based on application and domain specific heuristics. They compare personal profiles and discover similarities [13]. In the area of cooperative learning, [24] specifies context independent of the user in a first step. Here, the authors basically consider and define a didactic model, the goal, performance instructions, existing input materials and tools, learning methods for the group, time frame, and finally benchmarks. Subsequently, this definition will in case of an upcoming cooperation be extended with user dependant conditions and additional information. Those conditions include, amongst others, previous knowledge, personal preferences for cooperation partners, times and tools. Based on all those attributes, the best fitting partners for cooperation will be selected.

[13] motivates a flexible system architecture to benefit from application and domain specific heuristics while developing expert-recommender systems. Such systems require a profiling supervisor, an identification supervisor and a selection supervisor. The profiling supervisor creates and administers user profiles using configurable modules and diverse data sources. An identification supervisor selects applicable resources and persons consulting configurable heuristics. A selection supervisor filters the list according to dynamic strategies and preferences.

We take up this architecture in an adapted way to fulfill the specific requirements of expert identification in workplace-embedded collaborative e-learning environments.

3 Approach

The following approach for business-process oriented expert selection was designed and developed in the context of the APOSDLE[1] project. APOSDLE is an integrated project (IP) in the area of Technology Enhanced Learning (TEL) aiming at a conceptual and technical integration of the three roles knowledge worker, learner and teacher into a model of work-integrated learning. Here, the APOSDLE platform provides a fusion of learning solutions with the computer-based work environment of the users. The overview below is part of the project and was mainly developed in cooperation and discussions among project partners. Main design focus is the seamless fusion between performing a skill-intensive work process as a knowledge worker and a situation where the knowledge worker as a learner needs to consults one or more experts.

Figure 1 illustrates the approach taking up the example scenario mentioned in section 1. In the upper left corner, the user Anna has already performed some sub-tasks of the overall process. The subsequent tasks require knowledge that she has not learned until now. Anna has to acquire the necessary knowledge. To fasten the learning process, an expert guiding her through the learning process has to be selected. The APOSDLE platform is aware of Anna's current task context (1). Including the task context as well as Anna's and the experts' user context, the platform identifies (2) and displays (3) adequate teachers. In this example, the displayed experts are Michael and John. Michael is working in Anna's department. Since he has performed the process several times before and also edited a related learning document, he can easily guide her through the process. John and his colleagues have defined and established the process in the enterprise. Therefore, he is a well-known expert in this area.

Finally, the learner (Anna) makes a final decision about adequate cooperation partners in the list. In the example, Anna initiates collaboration with Michael. Ideally, relevant information of the cooperation will be extracted for re-use and stored in the APOSDLE platform. Later expert searches will consider this information in the selection process.

Following sections deepen the mentioned steps in detail. The numeration of the subsections relates to the numbering in figure 1.

3.1 Elicitation of Context

The goal of our research project is to enhance the productivity of knowledge workers by integrating learning, teaching, and working. In order to support this and many other aspects of an interweaved learning paradigm, the e-learning system needs to be aware of a user's current working task. This information can be seen as a prerequisite for finding suitable experts or resources and is to be

[1] APOSDLE (**A**dvanced **P**rocess-**O**riented **S**elf-**D**irected **L**earning **E**nvironment) is partially funded under the 6th framework programme (FP6) for R&D of the European Commission within the Information Society Technologies (IST) work program 2004 under contract no. IST-027023.

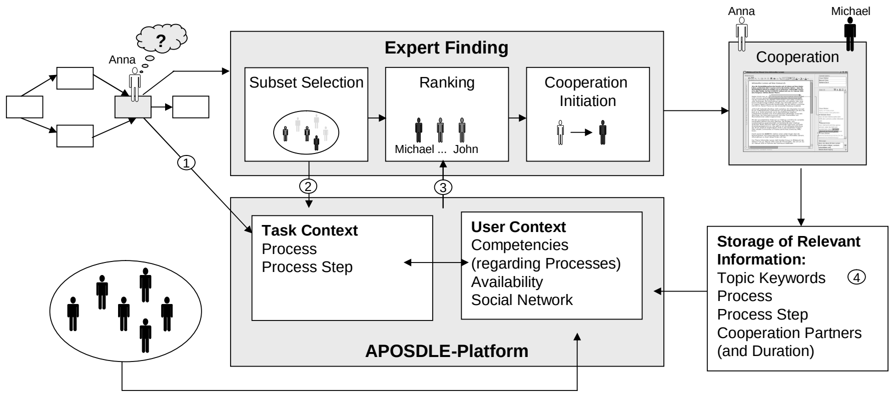

Fig. 1. Sidebar for displaying learning events and initiating collaborations

retrieved automatically and unobtrusively using low-level context information as indicators. The applicability of traditional machine learning (ML) algorithms to this problem is the subject of this section.

The goal of task prediction is to know the active task of the user at any point in time. Whereby task is a defined unit of work consisting of activities to reach a certain goal. The problem of task prediction is percepted as a machine learning task. When first using the system, it is untrained and the user needs to specify the task s/he works on from a predefined list of business tasks (manual selection). During the work process a context monitoring component logs any desktop events reflecting the user's actions. These include keyboard presses, application launches, document full texts, etc. That way, tagged training material of user's work streams with the task name as class label are collected and as soon as enough material is gathered the system trains a ML model of the user's work task in this business process. The optimal result is achieved when the user continues to work and s/he does not need to manually notify the system of task switches anymore. The task predictor automatically classifies the active tasks using continously recorded event streams (automated selection). Whenever classification detects a change in tasks, our e-learning environment displays a new list of associated learning resources and suitable experts regarding to the detected work task. The whole scenario is depicted in figure 2.

The machine learning algorithms we implemented and tested are of the types decision tree learning, rule learning, Naïve Bayes, and Support Vector Machines. Naïve Bayes (NB) was choosen due to its good overall performance [8, 6], even despite its assumption of class condition independence. Support Vector Machines (SVM) are machine learners that have been reported to perform well on text categorization problems [10]. One efficent training algorithm is *Sequential Minimal Optimization* (SMO) by Platt [16], which was implemented with a modification [21]. SVMs in general are assumed to find a good trade-off between overfitting und over-generalisation [4]. The well-known ID3 implementation by Ross Quinlan [19, 17] was choosen as concrete instance of decision tree learners, since it avoids overfitting by a pruning strategy [18]. One of the first successful learning techniques was rule learning [5]. Since it also generates human-readable classification rules and the efficiency and competitiveness was proven by its authors, the incremental reduced error pruning [7] (IREP) algorithm was also choosen for implementation.

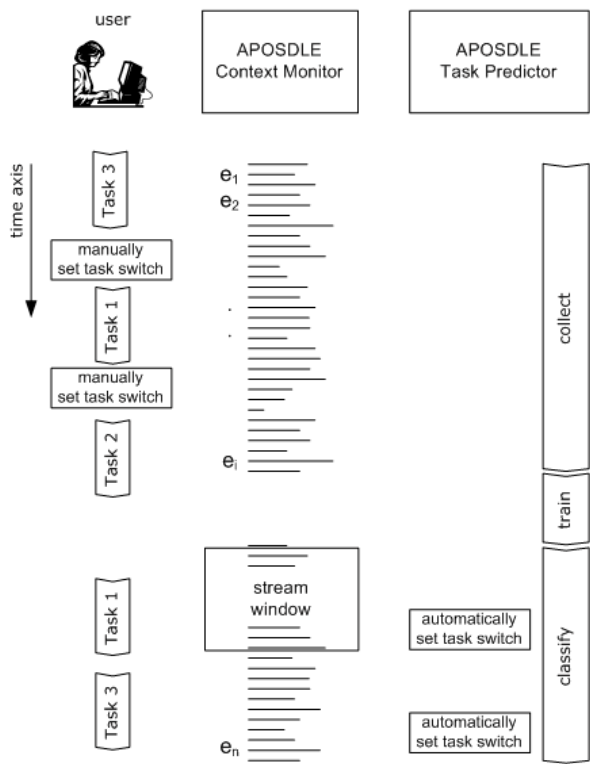

Fig. 2. Task Context Elicitation

In order to evaluate the task prediction in general and the four learning algorithms in particular, a scenario was created that resembles the real use cases well. As evaluation scenario for task prediction we used a modeled business process of a sample application domain. We decided to model business tasks like *market analysis*, *product design and specification*, *find and contact suppliers*, *contract placement* and *triggering production* in a sequential process model formulated in the process description language YAWL[1], which is based on Petri nets.

[1] Yet Another Workflow Language. See *http://yawlfoundation.org.*

An important requirement to our task prediction system is its suitability to situations where labeled training material is sparse. Therefore, the dependence of the implemented algorithms on data availability has been evaluated. Figure 3 shows the preliminary results of a first evaluation of prediction accuracy.

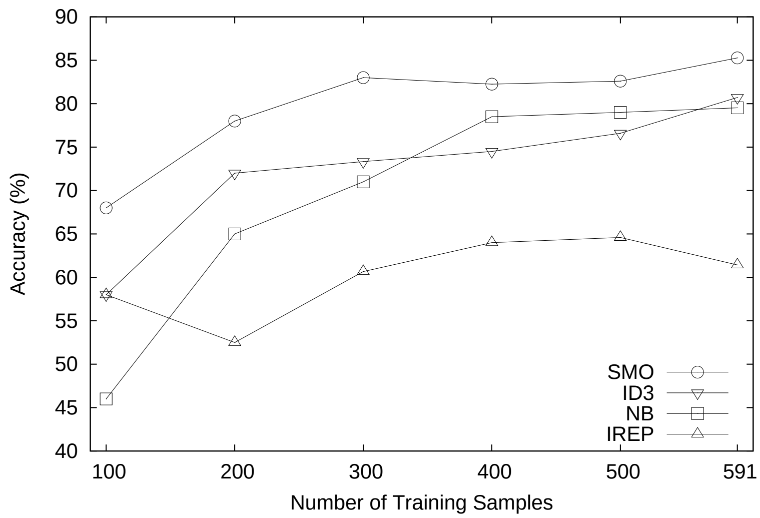

Fig. 3. Effect of Training Material Size on Accuracy

The highest gain of accuracy can be observed for Naïve Bayes. Euclid and IREP are influenced to the smallest degree by the training material availability. Starting at 200 samples, the relations between all algorithms are rather stable. SMO performs best in all scenarios. For the analyzed domain, the trade-off between classification accuracy and cost of collecting labeled data can be maximized with SMO and 300 training samples. This amount is as low as 20 minutes of recording per task (i.e. target label) and yields classification accuracies of 83%.

In spite of this positive preliminary evaluation conducted here, the task prediction still has to prove its performance in a larger field study, since the training data this evaluation relied on is just a few hours of recorded desktop work of a student. Several application domains with users of different computer experience and varying numbers of work process tasks need to be considered. Since these data are not yet available from ongoing user studies, the evaluation in this paper was limited to one scenario. Consequently, one of the next tasks is evaluating whether varying application domains with their text domains yield performance differences in context-dependant task prediction. The next section shows how the detected task is used in our approach for finding recommendations of suitable experts.

3.2 Identification of Relevant Experts

Based on the task context (and other information about the user contained in the user profile) we can recommend appropriate tools (applications and templates), resources (documents and learning material) and collaboration partners (peers and experts). In this section we will describe how to identify relevant experts. Due to space limitations we can only sketch our approach and go into more detail for some facets. Whether a user B is suitable as a collaboration partner for user A is determined by the contexts of users A and B. Note that the various parts of a context are of different importance depending on a user's role (seeking advice or being a potential expert). For example, the current task of the person seeking advice is important for identifying relevant experts, the current task of the potential expert is less important. In order to decide whether a user B is a potential expert for a user A seeking advice we consider the following paramters:

- **Competency:** User B has performed the current task of user A successfully a couple of times, i.e. B possesses all competencies necessary to perform this task. For the calculation of most suitable technical expert regarding a certain task, the task and competency history for each user needs to be stored permanently and a preceding process modeling becomes crucial. The user, who has completed the determined task most often, is probably most suited to help the learner from the technical point of view. Consequently, we define a normalized task suitability as:

$$suitability_{task}(B) = \frac{number\ of\ executions\ of\ task\ by(B)}{max_{X\ in\ userlist}(number\ of\ executions\ of\ task\ by(X))}.$$

- **Availability:** User B is currently available for collaboration. This criterion is of special importance in cases where advice is needed urgently. Information about availability originates from two different sources:
  - Automatic detection of availability: Similar to other synchronous communication tools such as Instant Messenging availability is inferred from the login status of a user. If a user is not logged in she is not available as a potential expert.
  - Manual setting of non-availability: For various reasons a user might not want to seen as available even if he is logged in the system. Reasons include for example high workload or a high amount of advice requests. Therefore, the user needs a way to manually set his status to not available.

In our example, both Michael and John are available as experts for Anna. In future versions other sources for detecting availability might be included. For example, the system might use a calendar to check whether there is an imminent upcoming meeting involving the potential expert. In that case the expert is probably not willing to start a collaborative session at the moment.

- **Organizational distance:** Organizational distance can be derived e.g. from current or past department or project affiliations of A and B. An organizational model, such as *organizational charts*, can be used to compute this distance. To be a suitable cooperation partner the organizational distance between A and B must be below a defined threshold d_{org}^{max}. Whereas we define the normalized organizational distance as:

$$d_{org}(A,B) = \frac{d_{org}^{max} - abs(level\ in\ org\ chart(A) - level\ in\ org\ chart(B))}{d_{org}^{max}}.$$

That way you can guarantee that a student assistant will never bother the CEO of a company.

- **Social Distance:** The social distance between A and B must be lower than a defined threshold d_{soc}^{max}. Social distance can be derived among others from preferences or dislikes towards users and topics and from extent of and satisfaction with previous collaborations between A and B. A social network representing groups and their interaction patterns can be used to compute the social distance [23]. Such social networks visualise users as nodes and sender-receiver relations as edges between nodes. One example of a sender-receiver relation is joint participation in a collaborative session in the APOSDLE environment. Consequently, it is possible to define the normalized social distance as $d_{soc}(A,B) = \frac{number\ of\ collaborations(A,B)}{max_{X,Y\ in\ network}(number\ of\ collaborations(X,Y))}$. But on principle all other sources which can be evaluated automatically such as e-mail or Instant Messaging) can be used, too. In our example a strong relation is assessed between Michael and Anna due to previous collaborations, while John did not collaborate with Anna so far. We consider the social network to be very important as previous studies showed that knowledge about and familarity with the collaboration partner plays an decisive role for knowledge sharing [11].

For each of these criteria the degree of fulfilment is determined and ranges from 0 to 1. In addition, for each criterion a threshold is defined above which the criterion is seen as fulfilled. All users who meet the above mentioned criteria are treated as potential collaboration partners. Depending on the degree of fulfilment for each criterion and depending on user preferences the list of experts is sorted and presented to the user in order to select and invite one or more experts as collaboration partner(s).

3.3 Prioritization of the list of potential experts

After identifying potential experts this step deals with an appropriate prioritisation of the candidates. The aim of this step is the presentation of a list of potential cooperation partners that is ordered by descending appropriateness. From this list the learner can choose manually a collaboration partner. The prioritisation of the list of potential experts is determined by the compliance of the above mentioned criteria (competence, availability, organizational and social distance) as well as by the preferences of the learner. The preferences of the learner specify the *individual* importance of a criterion (scale 0 to 1), where the sum of all weights of the criteria must be 1. They are for example defined by the user as part of his user profile. Furthermore they could be specified interactively in the APOSDLDE environment by competing sliders. For the sorting of the list of potential experts: a user can for example define that the criterion social distance is absolutely important (scale: 1) and the criterion organisational distance absolutely unimportant (scale: 0). Now a level of suitability of an expert to a defined user in his context can be defined:

$$suitability_{user}(expert) = \frac{\sum_{i=1..n} compliance\ of\ criterion_i(expert) * weight_{user}(i)}{n}$$

This results in a scale between 0 and 1 for the suitability level of a potential expert, where 1 means perfectly suitable and 0 means absolutely not suitable. For the user the list of experts is presented by descending level of suitability.

3.4 Return of relevant information

After finding and presenting a sorted list of context-related experts by the technical system the learner chooses one or more experts from the presented list. With these experts the learner wants to initiate a cooperation step. The APOSDLE-Platform offers a tool that integrates synchronous cooperation, e.g. on a whiteboard, and text-based communication in form of a chat [14]. For this paper the cooperation step itself is not relevant; relevant for this paper is the question which data of such a cooperation step should return to the knowledge base and be available for further queries. Concerning the content of the cooperation situation a transcript can be stored that contains amongst others the communicative contributions. This transcript can be linked to other context information concerning the task or the user in order to find it during a later (expert-)search. Further items of context information concerning the task are:

- **Task/Process:** If the cooperation was initiated with respect to an identified task context (see Figure 1) the information about the concrete task/process should be stored on the platform. A user which has a problem with the same task or process later on could maybe solve his problem by reading the corresponding cooperation transcript. That way, no further cooperation with experts is necessary.
- **Topic of the cooperation artefacts:** In order to relate a cooperation artefact on a content level we follow two paths: The platform offers an automatism to relate the cooperation transcript to topics of an existing list of keywords [20]. Additionally, the participants can add further, manually defined keywords after finishing the cooperation.

Further items concerning the context of the user are:

- **Participants:** The storage of participants has two functions. On the one hand it relates persons and tasks as well as persons and dedicated competencies. For further searches of expert concerning the corresponding task these persons are more probably experts. On the other hand a social network can be built on joint participation in the cooperation. This social network has influence on the choice and presentation of appropriate experts for the person (see “social distance” as described above).

– **Length of a cooperation:** From the length of a cooperation one can derive the intensity of knowledge exchange (at least in some cases). Especially very short cooperations are often less helpful for further situations because they are less detailed or explicit and therefore not comprehensible for others.

4 Realization

During the first year of the project an integrated prototype was created, which supports workplace-embedded, individual and cooperative learning. This prototype was realized in a client server architecture and developed in the programming languages Java and C# . The user interacts with a sidebar on the client part of the prototype (see Figure 4). This sidebar displays learning resources and experts suitable to the actual task and necessary competencies.

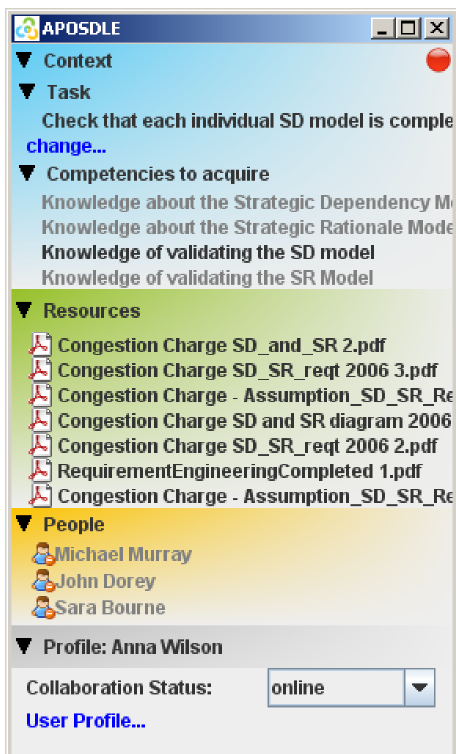

Fig. 4. Sidebar for displaying learning events and initiating collaborations

The selection of matching collaboration partner and learning resources is calculated in a server component, the APOSDLE platform. The platform is also used to store extensive user profiles, which contain user history, task dealt with and competencies aquired. But also the availability of potential experts and the current work situation is kept there. That way, the platform can consider all necessary pieces of information mentioned in Section 3.2 to find suitable resources and experts.

Directly from the sidebar, the user is able to initiate a collaboration with the desired expert. Both collaboration partners join a common collaboration room, where they can exchange text messages und collaboratively work on or discuss about certain documents and presentations. In Figure 5 a collaboration room is depicted, which additionally shares a defined context of the collaboration initiator. Consequently, the invited experts is able to get quickly an idea of the learners problem and can provide help uncomplicatedly.

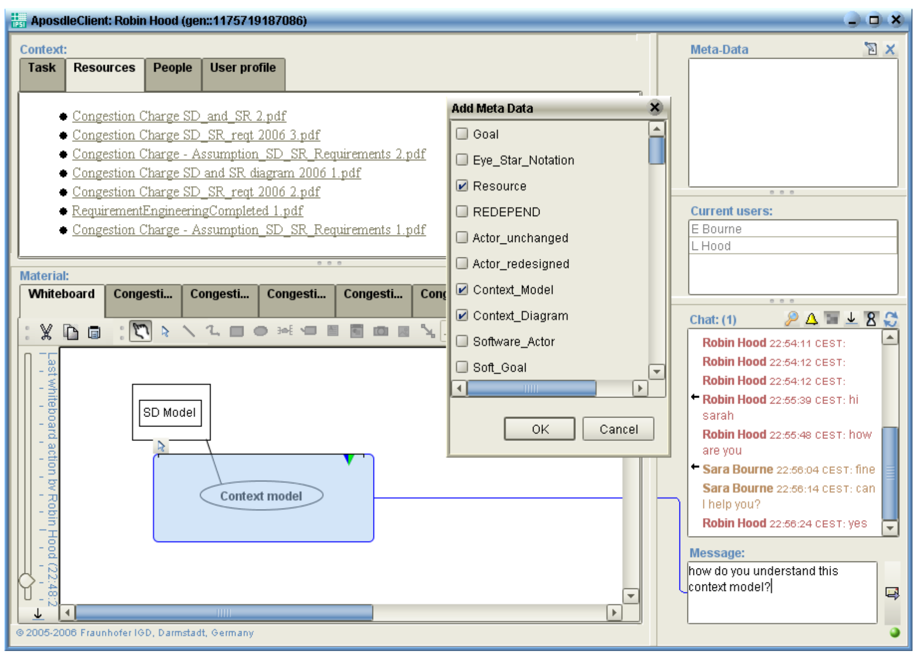

Fig. 5. Collaboration tool with shared context and referenced knowledge artefacts

5 Summary and Outlook

This work introduces a context-aware approach to discover collaboration partners and adequate experts in a workplace-embedded e-learning environment. The approach fuses the area of process integrated e-learning with on the fly knowledge transfer. In a first step, Business Process Modeling Notation (BPMN) and Petri Nets are introduced as promising ways for process modeling. Then, a machine-learning-approach for task context elicitation is introduced and its preliminary results are presented. Particularly, this step is a foundation for the main section of identification and prioritization of experts. Whereas, the identification of relevant experts here is mainly based on competency regarding a certain task, availability and organizational and social distance between learner and teacher. Within a list of potential experts, the user finally selects adequate collaboration partners herself. At the end, the learning process completes with the extraction and storage of emerging collaboration information in the knowledge platform. This information both includes the task context like process, topic and the user context (collaboration partners, competencies, session length). The whole approach is illustrated following an example scenario. Based on this approach, the APOSDLE prototype was designed and developed. Currently, the system is evaluated in the field in cooperation with project partners. The evaluation will in principal analyze the capability of the approach under realistic circumstances. Future work will include a detailed analysis of the study results.

References

1. W. Aalst. Three Good reasons for Using a Petri-net-based Workflow Management System. In S. Navathe and T. Wakayama, editors, *Proceedings of the International Working Conference on Information and Process Integration in Enterprises (IPIC'96)*, pages 179–201, Camebridge, Massachusetts, 1996.
2. T. Bauer and D. Leake. A Research Agent Architecture for Real Time Data Collection and Analysis. In *Proceedings of the Workshop on Infrastructure for Agents, MAS, and Scalable MAS*, 2001.
3. L. Birnbaum, W. J. Hopp, S. Iravani, K. Livingston, B. Shou, and T. Tirpak. Task aware information access for diagnosis of manufacturing problems. In *IUI*, pages 308–310, 2005.
4. C. J. C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.
5. P. Clark and R. Boswell. Rule Induction with CN2: Some Recent Improvements. *Proceedings of the Fifth European Working Session on Learning*, 482:151–163, 1991.
6. P. Domingos and M. J. Pazzani. Beyond Independence: Conditions for the Optimality of the Simple Bayesian Classifier. In *International Conference on Machine Learning*, pages 105–112, 1996.
7. J. Fürnkranz and G. Widmer. Incremental Reduced Error Pruning. In *Proceedings the Eleventh International Conference on Machine Learning*, pages 70–77, New Brunswick, NJ, 1994.
8. J. Han and M. Kamber. *Data Mining. Concepts and Techniques.* Morgan Kaufmann Publishers, 2001.
9. E. Horvitz, J. Breese, D. Heckerman, D. Hovel, and K. Rommelse. The Lumiere project: Bayesian user modeling for inferring the goals and needs of software users. In *In Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, pages 256–265, Madison, WI, July 1998.

10. T. Joachims. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. *Proceedings of the 10th European Conference on Machine Learning*, pages 137–142, 1998.
11. A. Kienle, N. Menold, and T. Herrmann. *Wissensgenese, Wissensverteilung und Wissensorganisation der der Arbeitspraxis*, chapter Technische und organisatorische gestaltungsoptionen für unternehmensinterne Wissensmanagementprojekte, pages 109–153. Westdeutscher Verlag, 2003.
12. J. Lave and E. Wenger. *Situated Learning : Legitimate Peripheral Participation*. Cambridge University Press, September 1991.
13. D. W. Mcdonald. *Supporting nuance in groupware design: moving from naturalistic expertise location to expertise recommendation*. PhD thesis, 2000. Chair-Mark S. Ackerman.
14. M. Mühlpfordt. Dual Interaction Spaces: Integration synchroner Kommunikation und Kooperation. In M. Mühlhäuser, G. Rößling, and R. Steinmetz, editors, *DeLFI*, volume 87 of *LNI*, pages 99–110. GI, 2006.
15. Object Management Group. OMG BPMN Final Adopted Specification. http://www.omg.org/docs/dtc/06-02-01.pdf, 2006.
16. J. Platt. *Fast Training of Support Vector Machines using Sequential Minimal Optimization*. MIT Press, 1998.
17. J. R. Quinlan. Induction of Decision Trees. *Machine Learning*, 1(1):81–106, 1986.
18. J. R. Quinlan. Simplifying Decision Trees. *International Journal of Man-Machine Studies*, 27(3):221–234, 1987.
19. J. R. Quinlan. Learning decision tree classifiers. *ACM Computing Surveys (CSUR)*, 28(1):71–72, 1996.
20. P. Scheir, P. Hofmair, M. Granitzer, and S. Lindstaedt. The OntologyMapper plug-in: Supporting Semantic Annotation of Text-Documents by Classification. In *Semantic Systems From Vision to Applications - Proceedings of the SEMANTICS 2006*, pages 291–301. Österreichische Computer Gesellschaft, 2006.
21. B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, 2002.
22. M. van Welie, G. C. van der Veer, and A. Eliëns. An Ontology for Task World Models. In P. Markopoulos and P. Johnson, editors, *Design, Specification and Verification of Interactive Systems '98*, pages 57–70, Wien, 1998. Springer-Verlag.
23. S. Wasserman, K. Faust, and D. Iacobucci. *Social Network Analysis : Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge University Press, November 1994.
24. M. Wessner. *Kontextuelle Kooperation in virtuellen Lernumgebungen*. Eul, Lohmar, 2005.