Work Experience Reuse in Pattern Based Task Management

Ying Du[1], **Uwe V. Riss**[2], **Ernie Ong**[1][1], **Philip Taylor**[1]
([1] SAP Research CEC Belfast, UK
[2] SAP Research CEC Karlsruhe, Germany
{ying.du, uwe.riss, ernie.ong, philip.taylor}@sap.com)

Liming Chen, David Patterson, Hui Wang
(University of Ulster, UK
{l.chen,wd.patterson, h.wang} @ulster.ac.uk)

Abstract: Pattern based task management has been proposed as a promising approach to work experience reuse in knowledge intensive work environments. While initial work has focused on the conceptualization and development of a generic framework, the process and user interaction of the task pattern lifecycle has not been addressed. In this paper, we introduce task copy augmented by Abstraction Services as a novel approach to facilitate task pattern creation and maintenance in a semi-automatic fashion. Also, we develop the architecture to demonstrate the underlying ideas by leveraging the advantage of semantic technologies.

Keywords: Task Management, Experience Reuse, Task Pattern, Knowledge Work, Semantic Technologies
Categories: M.3, M.8, H.5.3, K.8.0

1 Introduction

With the emergence of the "knowledge economy" and "knowledge society" [David, 02] driven by Internet-based information and communication technologies, rapid transfer of knowledge is enabling organisations and enterprises to achieve objectives quicker and at lower cost, thus gaining competitive advantages. Traditional Knowledge Management approaches have been regarded as inadequate to support knowledge work efficiently, especially with respect to experience transfer in knowledge-intensive work [Schutt, 03]. To make knowledge transfer more efficient we have to take the respective work context explicitly into account and have to represent it properly. This consequently gives the rise to a growing demand for, and extensive research on explicit modelling and processing of work experience [Sun, 05].

A central challenge is that work experience is typically tacit, unstructured, and hard to capture and communicate. This determines the particular challenges in experience modelling, representation and reuse. To address this issue, attention has been paid to the capture and modelling of relationship between activity and knowledge as: successful action certainly requires specific knowledge, and work

[1] Current address: The British Library, UK. ernest.ong@bl.uk

experience is incrementally gained from the execution of work processes [Wiig, 04] [Riss, 05a].

Recently, the Pattern Based Task Management (PBTM) approach [Riss, 05b] has been proposed as a means to transfer and reuse valuable work experiences, especially for knowledge-intensive work. It introduced the task pattern as a key concept for modelling and representing work experiences. Due to their proximity to task execution, these task patterns are better suited to transfer task related knowledge since they guide knowledge workers in a concrete way through their work activities. In the NEPOMUK project, we have consequently followed this approach making extensive use of semantic technologies to bring the handling of information, tasks, and task patterns closely together. Approaches based on Business Process Modelling proved to be inadequate to support processes in knowledge work because of their contingent characters. The separation of process and information assumed in these approaches does not work since the processes crucially depend on the information gained during the execution. Therefore the PBTM considers information objects as a crucial point in task guidance, and leaves the actual execution decision to the user. Due to the entwinement of task management and task pattern creation and maintenance, PBTM can only be realized on efficient task management. Hence, effective technologies and easy-to-use tools to support task pattern lifecycle within task management are foundational to the PBTM approach [Jarodzka, 07].

The approach is currently used in the MATURE project with the goal to establish a global experience lifecycle as proposed in [Ong 07]. However, such a lifecycle requires the extensive participation of users and their contribution.

The remainder of the paper is organised as follows. Section 2 introduces the PBTM approach and discusses its challenges. Section 3 describes key concepts and interrelationships of the architecture for the proposed approach. Section 4 outlines concrete implementation details and demonstrates how each proposed concept can work together to facilitate work experience reuse. We discuss related work in Section 5 and conclude the paper in Section 6.

2 The PBTM Approach

PBTM was proposed under the observation that knowledge work is characterized by ad hoc processes and the proceeding of which centrally depends on the information gained in the course of execution. However, this does not mean knowledge work is completely irregular but only that regularly repeated process structures appear in certain isolation of as well as in strong dependence on available information. Even if the regular patterns that appear in these processes are not systematic to suffice for process modelling, it is nevertheless worth to record them and make them available to other users. These patterns mainly appear on task level so that this is the level on which the PBTM supports reuse. Consequently, we call them task patterns. If a particular process pattern has been repeated several times, it can be regarded as the best practice for a particular type of tasks and for a specific group of users. Typically, established work practice is formally described in process models and made available by workflow systems. However, workflow systems which depend on predefined models are expensive and rigid. Hence, they are inadequate to support flexible and

dynamic work processes such as knowledge work since these processes are subject to regular changes. The PBTM emerged with the aim to overcome this barrier.

The task pattern is a key concept of the PBTM approach, which provides a template of abstract information and describes common properties of similar tasks. It can be used as a basis of creating new task instances. In real world applications, a task pattern is likely to become very complex and thus requires considerable user interactions during the task pattern creation and maintenance processes [Riss, 07]. A task pattern's lifecycle usually starts from manual creation by specifying generic views of similar tasks. Task patterns, therefore, are represented as a carrier of the user's work experiences. When the task pattern is applied as a template of creating new task, the embedded work experiences can be reused for the task to execute.

Additional efforts required in the task pattern maintenance have been addressed as the major challenge of enabling the PBTM approach. This is because knowledge workers are usually focusing on the tasks at hand and are likely to reject the abstraction of existing tasks, e.g., contribute to task patterns, as part of their activities. Such activities merely add more administrative overhead for which the immediate benefit is not always clear. Therefore, individual costs must be reduced to as low as possible, and one of our proposed solutions is to populate task patterns in a semi-automated fashion.

3 Realizing PBTM in a Semi-Automated Fashion

3.1 Task Copy and Task Reuse

Towards the aim of reusing past experience and work structures, knowledge workers are unlikely to consider abstracting previous tasks to task patterns as an initial step. Therefore, we hypothesize that knowledge workers will refer to similar ongoing or completed tasks whose goals or contexts match those of their tasks at hand. This provides a point of reference from which the knowledge workers may exploit opportunities to transfer and therefore reuse their efforts and experiences from earlier activities by directly copying information from previous tasks. This is called task-oriented experience reuse or task copy.

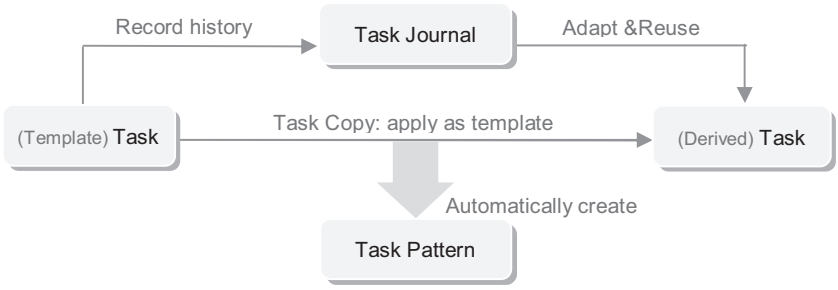

Figure 1: Task Copy based Experience Reuse processes

If we use task copy as the basis for work experience reuse, the next natural question is how such a copy operation will be realised. There are two options. First, entire task structures and details are duplicated, with the assumption that everything is implicitly relevant to the next task context; second, the user is responsible for explicitly selecting every detail to be copied. The former is likely to be useful for a small set of tasks. This is because it will likely result in information overload for a user. It results in a situation which requires the user to spend potentially more effort customising the duplicated task than to start a new one. The second option could also overwhelm users as it requires them to consider too many details from previous tasks, thus leading to a situation similar to that of the first. This reflection informs us that a more helpful position to consider the reuse of past work structures is somewhere in between these two extremes.

To that end, we proposed the concept of Task Journals [Riss, 08] as the basis on which records of past work experience can be made more comprehensible. Task Journals describe previous task activities and information objects used during task executions in the order of their usage. For example, it describes at which point in time a website x was attached as a bookmark to task y. Therefore, Task Journals can explicitly describe how a particular task is completed and in which order the respective steps are executed. Thus, it is possible to leverage the historical information to facilitate the creation of task pattern.

Since every task differs in its context and details, the user is unlikely to be able to reuse the details directly from a template task (from which the new task is copied), via its task journals. Instead, we expect that the user will be required to adapt the information captured in task journals from the template task to suit the context of the new task. For example, the user may select a different document template as the basis for a report in the new task. This final step of user adaptation is therefore necessary to enable the task journals to be contextualised in the new work context.

3.2 Task Evolution in Task Pattern

The task-oriented experience reuse described above is a low-cost approach of enabling experience reuse since descriptions of executed tasks can be used as they are. However, there are two major shortcomings. One is that this approach requires active user adaptation in each subsequent task, and consequently the results of each adaptation are lost. The other is the potential violation of the users' privacy since the task descriptions may also contain personal notes and other private information. This has motivated us to combine the strength of the task copy process with the task pattern concept which records user adaptations for task evolution. On the one hand, a task pattern can be automatically created whenever a task copy process happens, e.g., when a task is applied as a template task to another task, as shown in Fig.1. On the other hand, the public task pattern can be further enriched through collaborative efforts within an organisation, and therefore the adaptations of each subsequent task can be captured in this way.

Since a task pattern is automatically created from the task copy process, it is capable of capturing the evolution of a group of similar tasks. We define a collection of similar tasks as a task family which consists of a number of tasks derived from the same template task. This relationship can be captured in the task pattern model and represented in an ontology. For example, when a task pattern is automatically created,

a template task is associated to the task pattern. The semantic relations can explicitly describe that the template task is the basis of creating the task pattern. Moreover, tasks derived from a template task are represented as derived task in the task pattern model. In this case, a task pattern not only presents the generic knowledge of similar tasks but also records the evolution of a task family by describing the relationships between tasks. These task-task semantic relations can be further leveraged for task similarity computation.

3.3 Reflecting Task Knowledge in Abstraction Services

In the proposed task copy process, the reuse of task-oriented knowledge is enabled by transferring information captured in task journals from template task to derived task. Task journals play the role of experience carriers which describes the history of task execution, e.g., an email is attached to a task. Since information recorded in task journals could be rather concrete, it is insufficient to present a generic view of task knowledge. For this purpose, we propose the concept of Abstraction Services (see Figure 2) to, on the one hand, present generic task knowledge in task patterns, and on the other hand, provide salient details of a required task artefact, typically a resource such as a document or person. In short, the Abstraction Services must have the ability to describe generic information of a task pattern and visualize the generic knowledge into concrete examples which is so called *Candidate*. An instance of *Candidate* can be automatically extracted from task journals or manually generated. For example, in the task copy process, information resources embedded in a particular task journal can be automatically transferred to Abstraction Services with candidates pointing to information resources. In this way, past task knowledge can be explicitly visualized by the concept of *Candidate*.

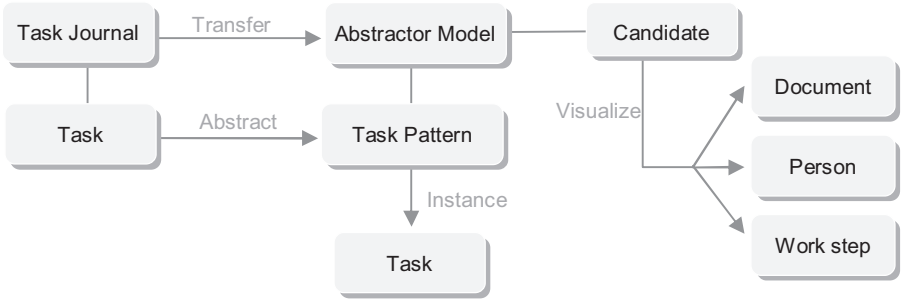

Figure 2: Models: Task Pattern & Abstraction Services

4 Implementation & User Scenarios

The Task Model Ontology (TMO) [NEPOMUK D3.1, 07] defines a number of schemas to describe general properties of task, task pattern (task pattern is modelled in the TMO extension) and other relevant concepts in supporting task management. We extend the TMO by introducing new attributes of task pattern and the new

concept of Abstraction Services into the model. The ideas described above had been implemented in the NEPOMUK Semantic Task Management Framework [Ong, 08]. In this section, we will describe a user scenario to demonstrate how task pattern and Abstraction Services work together to enable work experience reuse in the KASIMIR [Grebner, 08] task management system.

4.1 Automated Task Pattern Creation via Task Copy

In order to make the proceeding clearer we will give an example how task patterns are created and used. John Doe, a researcher in SAP Research, uses KASIMIR (see Figure 3: A), to manage his daily work. In KASIMIR, the upper panel is the task list which includes a number of completed and running tasks. The lower panel shows the details of the selected task. For example, the current selected task, *Write Nepomuk D3.3*, is the one John is working on, and task details such as task collaborators, related documents are showed in the lower panel.

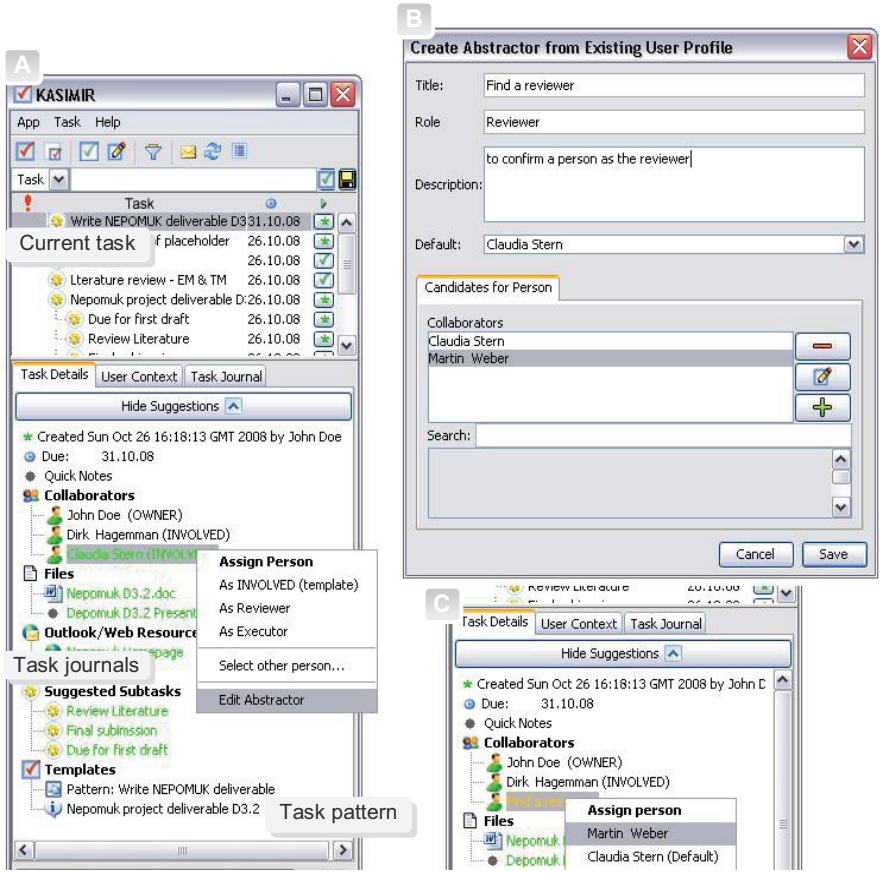

Figure 3: The prototype system of Task Pattern and Abstraction Services application.

In the task list, the completed task: *Nepomuk project deliverable D3.2* records all task details, such as collaborators, documents, sub-tasks and other information resources, in its task journals. John selects the completed task as a template to his current task: *Write Nepomuk D3.3*. Here, a task pattern is automatically generated and shown in the bottom of the lower panel. Meanwhile, task details captured in task journals have been displayed with green colour as recommendations. For example, Claudia Stern was a project reviewer in the completed task, and she has been recommended by the system as a participant in the current task. Therefore, previous task knowledge can be captured by and transferred via task journals, i.e., the items in figure 3 highlighted by green colour. This process makes past work experiences reusable in planning and executing new tasks.

4.2 Task Pattern and Abstraction Services

A user can manually modify an automatically created task pattern by specifying the task pattern's name and description from a more generic level. Besides that, a task pattern can be associated with *Abstractor*s provided by the abstraction services. For example, manually add a contact as the *Person Candidate* to an abstractor as shown in Figure 3:B. We also introduce an alternative way to support Abstractor Services, which is recommend potential candidates mined from task journals. In the user scenario, Claudia is recommended as a possible reviewer in the new task but she is busy in another project and won't contribute to the new task. Hence, John selects 'edit abstractor' menu and the system automatically generates Abstractor Services with adding Claudia as a possible collaborator. When John goes back to the KASIMIR (Figure: C), the lower panel shows the Abstractor Services named with 'Find a reviewer'. This allows the Abstractor Services to suggest a person who can work as a reviewer for the new task. By right clicking the 'Find a reviewer' icon, a task collaborator can be directly assigned from the candidate list, e.g., Claudia or Martin.

5 Related Work

The UAM project [Moran, 05] [Moody, 06] [Cozzi, 06] employs a pattern based approach to managing and reusing valuable work experiences. It proposes the *activity pattern* as the abstract representation of work experience and exploits semantic technologies to enrich tasks and patterns in a semantic way. In UAM, an activity pattern is simply represented as a checklist with a number of work steps, along with actors and resources. Activity patterns are controlled by a group of experts who are responsible for the creation and maintenance of activity patterns. This requires efforts made from both the activity pattern experts and providers [Convertino, 07]. In PBTM we follow a flexible and low-cost approach to task pattern management, e.g., a task pattern can be generated in a semi-automatically fashion by leveraging the past task knowledge captured in task journals. We also introduce the Abstraction Services into task pattern model, which helps users to identify resources required in the task pattern.

Another related approach is the Collaborative Task Management (CTM) that presents a conceptual framework for unobtrusive support of agile business processes. [Stoitsev, 07]. The framework enables the modelling, exchange, and reuse of light-

weight task structures defined by the users. In addition to the person-to-person exchange of best-practices, it enables 'outsourcing' of dynamic task structures and resources in personal workspaces and organizational units. Thus, they can be managed according to local domain knowledge and made available for reuse in shared repositories. The delegation of tasks generates enterprise process chains that go beyond the user's personal workspace. The framework facilitates knowledge management and supports proactive tailoring of underspecified business processes by the users.

To facilitate knowledge reuse, the TaskNavigator [Holz, 06] follows a different approach which is so called instance-based task reuse. It makes the use of previously executed tasks as the cases for planning and organizing future similar tasks. This is similar to our task copy approach. However, the two approaches have different focuses. The task copy aims to providing an easier way to task pattern creation and maintains. Moreover, in the task copy process, not only task information is reusable but also the history information captured in task journals is visible to the user. This is valuable to task executors in terms of providing enriched background information of the task execution history.

6 Conclusion and Future Work

This paper described a semantic-enabled task pattern based approach to work experience reuse. The approach brings together a number of interesting concepts, i.e. task copy, task journal, task pattern, and abstraction services. We described these concepts and their roles in the realisation of the proposed approach. By recording the history of past tasks, task journals are used as a means of capturing past work experiences. It is particularly the temporal sequence that makes the proceeding in the task execution more transparent since otherwise in complex tasks the user might be overwhelm by the masses of offered resources. The task copy process is used to transfer these captured experiences to new tasks. Task pattern serves as a knowledge model to represent the structural relations at abstract level of task knowledge, which but can be leveraged to facilitate task similarity comparison. The work presented here lays a solid foundation for task pattern applications.

For the future work, we plan to focus on three aspects. The first one is to extend the Abstractor Services beyond the described implementation. Currently, the described services do not yet consider context information of the current task executor. For example, in a travel task the destination of the user who applies a task pattern should be considered for the provision of travel information. The same holds for suggested experts which might depend on the organizational unit to which the executor belongs. The second one is improving the user interaction for task patterns and task similarity computation for the identification of suitable templates. The third aspect we considering for the future work is algorithm design and development in order to compute similarity between tasks and task patterns with the ultimate purpose of facilitating knowledge reuse for novice users. The general direction of these development can be found in [Schmidt, 09].

Acknowledgements

This work was supported by the European Union IST fund (Grant FP6-027705, project NEPOMUK and Grant FP7- 216346, project MATURE).

References

[Convertino, 07] Convertino, G., Moran, T.P., Smith, B. A.: Studying Activity Patterns in CSCW, In Proc. CHI'07, ACM, CA, 2339-2344, 2007.

[Cozzi, 06] Cozzi, A., Farrell, S., Lau, T., Smith, B. A., Drews, C., Lin, J., Stachel, B., Moran, T. P.: Activity management as a web service, IBM System Journal, vol. 45, 695-711, 2006.

[David, 02] David, P. A., Foray, D.: An introduction to the economy of the knowledge society, International Social Science Journal, vol. 54, 9-23, 2002.

[Grebner, 08] Grebner, O., Ong, E., Riss, U. V.: KASIMIR – Work process embedded task management leveraging the Semantic Desktop, In Proc. Multikonferenz Wirtschaftsinformatik, 715-726, 2008.

[Holz, 06] Holz, H., Rostanin, O., Dengel, A., Suzuki, T., Maeda, K., Kanasaki, K.: Task-based process know-how reuse and proactive information delivery in TaskNavigator. In: Proceedings of the 15th ACM International Conference on Information and Knowledge Management, ACM Press, New York, 522-531, 2006.

[Jarodzka, 07] Jarodzka, H.M., Grenber, O., Riss, U.V., Ong, E.: Motivate users to construct collective knowledge via IT - A psychological view on pattern-based task management, In Proc. Gronau, N. (eds.) 4th Conference on Professional Knowledge Management -Experiences and Visions, GITO, Verlag, Berlin, 372-381, 2007.

[Moody, 06] Moody, P., Gruen, D., Muller, M. J., Tang J., Moran, T. P.: Business activity patterns: A new model for collaborative business applications, IBM System Journal, vol. 45, no. 4, 683-694, 2006.

[Moran, 05] Moran, T. P.: Unified Activity Management: Explicitly Representing Activity in Work-Support Systems, In Proc. Workshop on Activity: From a Theoretical to a Computational Construct, ECCSCW 2005, Paris, 2005.

[NEPOMUK D3.1, 07] NEPOMUK Deliverable D3.1 Task Management Model, 2007, http://nepomuk.semanticdesktop.org/xwiki/bin/download/Main1/D31/D3.1_v10_NEPOMUK_Task_Man agement_Model.pdf.

[Ong, 07] Ong, E., Grebner, O., Riss, U.V.: Pattern-Based Task Management: Pattern Lifecycle and Knowledge Management. In: Proceedings of the 4rd Conference Professional Knowledge Management, Vol. 2. IKMS 2007 Workshop. Potsdam, Germany, 357-364, 2007.

[Ong, 08] Ong, E., Riss, U.V., Grebner, O., Du, Y.: Semantic Task Management Framework, In Proc. I-KNOW'08 and I-MEDIA'08, Graz, Austria, 387-394, 2008.

[Riss, 05a] Riss, U.V.: Knowledge, Action, and Context: Impact on Knowledge Management, In: K.-D. Althoff, A. Dengel, R. Bergmann, M. Nick, T. Roth-Berghofer (Eds.): Professional Knowledge Management, Lect, Notes in Artificial Intelligence, vol. 3782, 598-608, 2005.

[Riss, 05b] Riss, U.V., Maus, H., Aalst, W.: Challenges for Business Process and Task Management, Journal of Universal Knowledge Management, vol. 0, 77-100, 2005.

[Riss, 07] Riss, U.V., Cress, U., Kimmlerle, J., Martin, S.: Knowledge transfer by sharing task templates: two approaches and their psychological requirements, Knowledge Management Research & Practice, vol. 5, 287-296, 2007.

[Riss, 08] Riss, U.V., Grebner, O., Du, Y.: Task Journals as Means to Describe Temporal Task Aspect for Reuse in Task Patterns, In Proc. The ninth European Conference on Knowledge Management, UK, 721-729, 2008.

[Schmidt, 09] Schmidt, B., Riss, U.V., Reinhardt, W.: Task Patterns as Means to Experience Sharing. In: ICWL'09 Proceedings of the 8th International Conference of Web-based Learning. Aachen, Germany, 2009, forthcoming.

[Schutt, 03] Schutt, P.: The post-Nonaka Knowledge Management, Journal of Universal Computer Science, vol. 9, 451-462, 2003.

[Stoitsev, 07] Stoitsev, T., Scheidl, S., Spahn, M.: A Framework for Light-Weight Composition and Management of Ad-Hoc Business Processes. In: Winckler, M., Johnson, H., Palanque, P. (eds.) TAMODIA 2007. LNCS, vol. 4849, Springer, Heidelberg, 213–226, 2007.

[Sun, 05] Sun, Z., Finnie, G.: Experience Management In Knowledge Management, In: Khosla, R., Howlett, R.J., Jain, L.C. (eds.) Knowledge-Based Intelligent Information and Engineering Systems, LNCS, vol. 3683, Springer, Heidelberg, 979- 986, 2005.

[Wiig, 04] Wiig, K.: People-Focused Knowledge Management, Elsevier, Burlington, MA, 2004