TaskNavigator：柔軟なワークフロー管理と適時情報配信

Agile Workflow Management and Proactive Information Delivery in TaskNavigator

鈴木　剛*　前田　薫*　金崎　克己**　ハラルド ホルツ***　オーレグ ロスタニン***
Takeshi SUZUKI　Kaoru MAEDA　Katsumi KANASAKI　Harald HOLZ　Oleg ROSTANIN
アンドレアス デンゲル***
Andreas DENGEL

要　旨

本稿ではドイツ人工知能研究所との共同研究の成果である，TaskNavigatorシステムについて報告する．TaskNavigatorは研究開発などの複雑な業務をサポートするシステムであり，柔軟なワークフロー管理においてタスクとその関連情報の再利用の有効性を向上させ，これらの業務に携わるユーザの生産性を高めることを目的としている．提案システムはタスクの作成時及び実行時に，過去の類似タスク，関連するタスクモデル，関連情報等を自動的に検索し，結果をユーザに対して適時提示することで，複雑な業務におけるタスク作成及び実行を支援する．更にグループで共有するべき情報や成果物をタスクと関連付けて管理し，グループ全体の生産性を向上させる．ケーススタディの結果により，本システムが複雑な環境下におけるタスクの効率と，その成果物の品質を向上させることを示す.

ABSTRACT

As one of the deliverables of a joint project "Virtual Office of the Future", with German Research Center for Artificial Intelligence, we report on the TaskNavigator system, which is capable of supporting complex tasks, for instance, research and development. In the agile workflow management, the proposed system enhances reuse of tasks and their relevant information, consequently aims at improving productivity of the users. The system proactively retrieves similar tasks, relevant task models, and relevant information of the tasks and presents the results when the user creates or enacts the tasks in complex situations. In addition, the system provides a collaborative documenting space, so as to improve productivity of the group by managing all the task-relevant information shared among group members. Case study results show that the proposed system improves time efficiency and quality of the output documents of the tasks in the complex situations.

*　ソフトウェア研究開発本部 研究統括センター
R&D Management and Planning, Software R&D Group
**　ソフトウェア研究開発本部
Software R&D Group
***　ドイツ人工知能研究所
German Research Center for Artificial Intelligence

1. Introduction

The recent emergence and popularity of several new desktop search engines such as Google Desktop Search*, etc. has clearly shown the need for tools that help users in managing their personal knowledge space (PKS). Typically, the documents needed by a knowledge worker for the task at hand are spread over various places such as e-mail folders, file system folders, or paper stacks on the desk. While the concept of a desktop-wide search certainly relieves the user from the burden of querying several different information sources (e-mail, local and network drives, etc.), current desktop search engines still follow the standard, passive query/retrieve model: the user has to explicitly 'pull' for information that might be relevant for a task he/she is currently trying to accomplish. Besides being inefficient, empirical studies have shown that such pull approaches typically lead to suboptimal reuse rates of available documents[1]. In order to address this issue, several business process-oriented knowledge management approaches have been developed for proactively providing process participants with information that is relevant with regard to their current tasks[2]. However, as most of these approaches rely on static workflow/process specifications, they typically are inadequate for weakly-structured processes such as knowledge-intensive office work processes. Currently, state-of-the-art workflow and document management systems offer valuable support only for routine activities in office work. In spite of such support, it has been claimed that knowledge-intensive office work has not reached satisfying increases in productivity in recent years (cf. 3). From our experience, knowledge work consists of both agile and strictly-structured processes that are often highly interleaved. Whereas recent project support systems aim at uniformly supporting users in both kinds of processes[4], an integrated approach for information support in the form of proactive information delivery seems to be still missing.

In this paper, we propose TaskNavigator, a novel workflow management system capable of supporting agile workflows. The system aims at improving productivity during knowledge-intensive work of both users and user groups, by enhancing process information reuse individually and among the group members. This paper is organized as follows: in Section 2, we provide an overview on the approach for lightweight process-oriented knowledge management that underlies the TaskNavigator prototype. Sections 3 and 4 describe TaskNavigator's functionalities and a usage example respectively. Evaluation results are presented in Section 5, followed by a conclusion in Section 6.

2. The PERL Cycle

Our work aims at developing a light-weight approach for business process-oriented knowledge management (BPOKM) that can be applied for flexible knowledge work processes such as R&D or consulting processes. Since such processes cannot be defined and modeled in advance in sufficient detail to all for workflow-like enactment support, standard approaches to BPOKM (see e.g. 5) that are based on formal process models and formally specified information needs cannot be applied. Moreover, companies are not often willing to make high upfront investments into process and knowledge modeling activities when it is unclear if and when these investments are going to pay off.

Therefore, we propose the following bottom-up approach, called PERL cycle, for introducing BPOKM into a company, in order to support knowledge workers in their daily activities without requiring upfront process or ontology modeling. Fig.1 depicts the PERL cycle graphically: the approach is based upon an integrated tool support for flexible task management and proactive information delivery. The cycle builds upon the assumption so that information that is relevant for the knowledge workers' tasks is available on their desktops in form of electronic documents, e.g. files, emails, Wiki pages, etc.

* http://desktop.google.com/

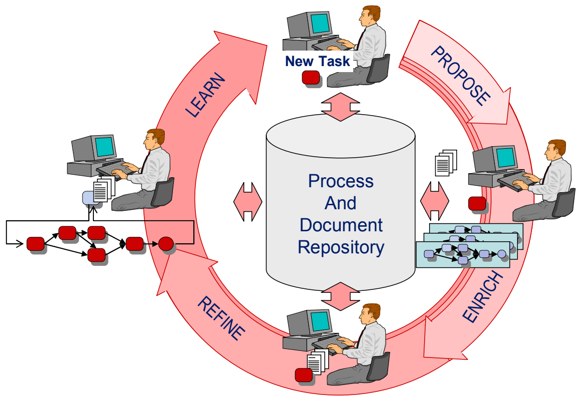

Fig.1 PERL cycle

Fig.1 depicts a (logical) central repository where the knowledge workers' tasks are stored, as well as the documents stored within the company. Four phases, called "Propose", "Enrich", "Refine", and "Learn" are grouped around this repository, and need to be supported by an appropriate tool environment.

Propose: For a newly created task, the system should proactively provide the knowledge worker with both available documents and process know-how that might be relevant in order to successfully complete the task.

Enrich: In order to receive more relevant document suggestions, the system should support the knowledge worker in easily enriching a task's description informally, e.g. by associating documents created or handled in the context of the task.

Refine: The system should support the knowledge worker in reusing process know-how by both proactively suggesting similar former tasks and their decomposition into subtasks from the repository and process guidelines, either textual or formal process definitions for the current task. Reuse of similar tasks should be supported by enabling the knowledge worker to create corresponding copies of a similar task's subtasks; reuse of process models should be supported by enabling the knowledge worker to instantiate subtasks according to the decomposition specified in the process model. In both cases, new (sub-)tasks are being created, for which the cycle will start again with phase "Propose".

Learn: The system should continuously learn by storing the actual decomposition of each task into subtasks, store new documents in the repository, support generalization of tasks into formal process models, learn how to categorize tasks to corresponding process models, and improve its proactive document suggestions over time.

In order to analyze the feasibility of such an approach, we developed a prototypical system, called TaskNavigator, that implements most parts of the PERL cycle. In the following, we will describe TaskNavigator's functionalities more detailed.

3. TaskNavigator

In accordance to the PERL approach, TaskNavigator provides tool support for:

- Agile task management (ATM)
- Proactive information delivery (PID)
- Process know-how reuse

We will present these functionalities in the following sections in more detail.

3-1 Agile Task Management

In addition to the standard functionalities of common task management tools, TaskNavigator allows users to maintain task-specific, hierarchical bookmark lists by attaching documents or URLs to a task and organizing them in task-specific folder structures.

Moreover, users can define simple ad hoc workflows by specifying predecessor relationships. Unlike rigid workflow management systems, the order of working with tasks in TaskNavigator is not strictly predefined. Task states have a recommending character, and users may start working on waiting tasks before predecessor tasks are completed.

Among TaskNavigator users, tasks including subtasks and attached bookmarks can be delegated to one another. Automatic email notifications are sent to delegatees who find the delegated task on their task lists. The delegator is informed via email and can track the task's current state on his/her list of delegated tasks.

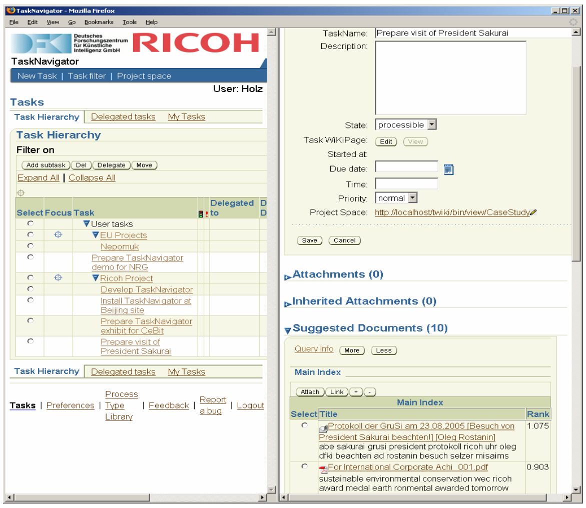

Fig.2 Hierarchical view on the user's tasks

TaskNavigator allows users to decompose a task into subtasks, resulting in hierarchical work breakdown structures. Fig.2 depicts an example screenshot from TaskNavigator: the left-hand frame shows an expanded task/subtask hierarchy, while the right-hand side shows details for the currently selected.

3-2 Building Personal Knowledge Spaces

In order to build a relevant information repository, we start with the contents of the company's shared network drives and users' local file systems. In order to exploit the users' native structures their desktops as well as the structures from the company's shared network drives, we make use of BrainFiler[7], a commercial system which realizes a personalized document management environment allowing multi-criterial classification of documents, search functionalities such as Boolean search and document similarity evaluation, as well as incorporation of remote BrainFiler instances. BrainFiler enables a user to build a personal information model by allowing to import and synchronize native structures such as e-mail folders, bookmarks, and file directories together with contained e-mails and documents. The imported structures are shown as trees and can be arranged in different views. Nodes in the imported structures get their meaning by a document term-similarity vector determined statistically from the assigned documents. These structures then can be used for a conceptual search as well as a combination with the keyword-based search.

3-3 Proactive Information Delivery

In addition to task-centered information structuring, TaskNavigator realizes task-oriented PID, i.e. the system automatically retrieves potentially relevant documents from various different information sources and suggests these documents to the user (Fig.2). For document retrieval, TaskNavigator transparently triggers a query to BrainFiler. The query for a task is determined by the task name, task description, and the attached bookmarked documents.

The component realizing the PID feature is used by TaskNavigator to proactively suggest process know-how in the form of similar tasks or available Process Types (PT), which is explained in the next section. Since we support hierarchical task decomposition, different information types can be of different relevancy for tasks, depending on their granularity level: high-level tasks have to be decomposed into subtasks, so information about similar tasks and PTs often is of greater importance for such tasks than domain-related documents; for low-level tasks, i.e. concrete steps, documents containing information about the concrete topics of such tasks are often more important.

In parallel to the retrieval of documents shared among group members, TaskNavigator integrates a local desktop search by sending requests to local BrainFiler installations. BrainFiler allows indexing and retrieval of user's e-mails that can be also highly relevant to the user current tasks. If the user considers a suggested document interesting and relevant for the current task, he/she has two alternatives to associate the task with the document, which are making the user's own copy of the document and making a link to the document.

3-4 Process Know-How Reuse

Knowledge intensive work is characterized by interleaving both rigid and agile processes. While modeling rigid processes is supported by state-of-the-art process modeling tools, agile processes are a subject of discussion in current scientific and technological research. Despite the flexible and unpredictable nature of agile processes, it is highly desireable to be able to reuse them or parts thereof, especially in the knowledge intensive work.

The easiest case of process know-how reuse without requiring process modeling efforts is so called instance-based task reuse. In case of task instance reuse, users are provided with the functionality for retrieving similar tasks. If another task has category that is similar to the category of the current task, it is regarded similar.

The information about retrieved similar tasks can be adapted and reused during the enactment of the current process. The newly created task instances are retained in the repository of task instances and can be reused later by the user himself or by others. By analyzing the rate of different tasks reuse, the most frequently reused tasks (typical tasks) can be found. If needed, typical enterprise-wide tasks can be modeled more formally using PT.

A PT is a semi-formal abstract description of a process activity that contains textual description of the activity, pre- and post- conditions for the activity, as well as possible approaches for executing the activity (PT decomposition) and activity-related documents. Tasks created as instances of a certain PT inherit the properties of the PT, e.g., related documents and decompositions into sub-activities. During PT instantiation, users can choose the necessary decomposition of the PT and tailor the task structure according to their situation.

Our suggestion is that by combining agile process modeling, task instance reuse and proactive information delivery, we provide a basis for effective sharing of process know-how between participants of agile knowledge intensive processes.

3-5 Collaborative Documenting Space

Wiki has become a widely used collaboration platform in both small and large institutions. Its flexibility and lightweight characteristics match best with TaskNavigator concepts. In TaskNavigator, Wiki helps users collaborate on tasks and their related information.

Wiki adds yet another type of document repository that enables embedded cross-reference links and collaborative authoring to TaskNavigator. Task-subtask relationships can be easily mapped to document-subdocument relationships by representing each section in a document as a Wiki page. Wiki pages are indexed by the DMS BrainFiler and become candidates for PID suggestions just as documents in other repositories.

Wiki provides means for informal process descriptions. Corresponding to each PT, users can associate a Wiki page to have informal process descriptions. Users who are about to instantiate a process type can find its descriptions and how-to's. They can also add comments, new suggestions, and additional references. This encourages building and sharing best practices in PT Library.

Thirdly, a platform for task-related discussions is supported by Wiki. When task is delegated to a user, discussions between the delegator and the delegatee are facilitated with Wiki. TaskNavigator provides an easy way to create a Wiki page that corresponds to a task. This "task-specific Wiki page" can be used as a communication place for all users involving that task (or just a scratch pad for a solo worker). When the task is finished, the page remains as a record how the user performed that task.

We extended the PID functionality for Wiki pages. PID now suggests related documents and document folders when users are viewing/editing Wiki pages. TaskNavigator performs an information retrieval using the page name and its contents as query string. This is our first approach to provide document-oriented PID.

Wiki page categorization is also extended with PID. Wiki categories are now unified with document classification categories in the DMS BrainFiler. PID recommends possible categorizations for the current Wiki page based on the similarity of its contents to Wiki pages that are already categorized. This Wiki page categorization is also visible from task-oriented PID. User groups can share and evolve on information categorization hierarchies, uniformly from both task- and document-oriented views.

Fig.3 illustrates how Wiki based PID helps the user. In this example, the user is working on a Wiki page that describes different software testing tools. Also, Wiki categories related to testing and tools are suggested as a possible categorization.

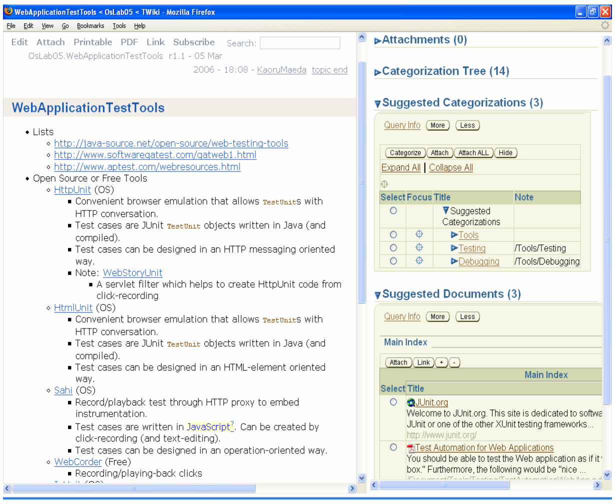

Fig.3 Collaborative authoring platform built upon Wiki and PID functionality

4. Usage Example: Preparing a visit of foreign delegation

Let us assume the following situation: a TaskNavigator user in company A has got a task to prepare a visit of the president of foreign partner company B. The user creates a new subtask "Prepare visit of president of company B" in the task "Joint project with company B", clicks on it to see task details in the right frame. The TaskNavigator system retrieves documents from the document repository that might be relevant for preparing the visit. For the current task the system has found 10 documents, top 2 of them are an e-mail containing details about the president's visit and a scanned news article about the president. The user decides that these 2 documents are interesting for him and he can download them or attach them to the task. Since the user has not prepared such visits yet, he would like to know whether other colleagues did it before. By clicking "check" button in the "Similar tasks" pane the user receives a list of tasks that are likely to be similar to the current one. The task "Prepare visit of department head of company C" once performed by a colleague looks very similar. The colleague decomposed this task into three subtasks:

- Reserve meeting room
- Prepare presentations
- Find suitable gifts for delegation.

Because all subtasks are relevant for the current visit preparation as well, the user copies these subtasks to his own task list via TaskNavigator. Moreover, for the task "Find suitable present", the user notices that his colleague attached some documents on foreign business culture that he found useful for his task. Seemingly he had searched the Internet for respective information. The two documents he found and attached are now also available to the current user.

5. Evaluation

In this section, we describe the experiments we conducted to evaluate TaskNavigator's performance. While pilot tests were conducted with students, the feasibility test and the case study were done with office workers.

5-1 Feasibility test

The feasibility test was conducted with computer science researchers at the Knowledge Management department at the DFKI. Around 20 researchers used the TaskNavigator system for their daily task management. The duration of the test was two months; the first half was for test usage, to get accustomed with the system, and in the second half, we evaluated the subjects' activities on the system.

We collected several metrics to evaluate the usage of the system functionalities. This enables us to validate the qualitative feedback from the viewpoint of system usage. By combining basic metrics, e.g. a number of tasks and total time spent on the system, we can derive useful measures to evaluate the system, e.g. a number of suggested information items per unit time or per task. Summary of results from the DFKI feasibility test is shown in Table 1.

Table 1 Excerpt from DFKI feasibility test

Created tasks	518
Delegated tasks	10
Attached information items	151
Access to the attached information items	163
Suggested information items	30198
Access to the suggested information items	271

Considering the amount of attached (151) and accessed (163) information items, some information items were used more than once. In addition, the number of created tasks (518) and the accumulative number of suggested information items (30198), indicate that on the average, 58 information items were suggested for a task.

The number of accesses to the suggested information items (271) is considerably smaller than the number of suggestions. This is due to the system configuration which upon access to the task, the system provides a newly calculated set of suggestions at that point of the time. Consequently, the accumulative number of the suggested items became large. Thus here we just point out that in comparison between the number of accesses to attached information items (163) and one to suggested items (271), we could say the suggestions were regarded as meaningful as the attached information items.

The difference between the number of created tasks (518) and delegated tasks (10) shows that the system was mainly used for organizing one's own work. Since the researchers are working in a similar context, and have been in touch with each other, they might not have needed the task delegation function during the short test period.

5-2 Case study

The TaskNavigator case study was conducted at the Ricoh Software Research Center in Beijing (SRCB) with experienced researchers as subjects. Here, we tested the system in the software requirements analysis. Two researchers at SRCB took a role of requirements engineers, analyzing requirements of a software product. Three other researchers at SRCB were domain experts to help the two researchers by providing product information. The main two researchers communicated with researchers in Ricoh Software R&D Group in Tokyo, who were customers. The duration of the case study was three months, the first month was used to get accustomed to the system, and during the latter two months, we tested the system. The two requirements engineers provided output documents about use cases and domain models for the analysis.

Table 2 Excerpt of metrics from SRCB case study

Created tasks	95
Delegated tasks	20
Attached information items	56
Access to the attached information items	174
Suggested information items	3420
Access to the suggested information items	53

The summary shown in Table 2 shows is of the latter months, and of the two main researchers who took roles of requirements engineer. We used the questionnaire to gather feedback from the subjects.

Most of the delegations were for reviewing requirements. The ratio of the numbers of items and accesses shows that the researchers accessed the attached information items 3.1 times on the average. Almost the same number of accesses was realized regarding the suggested information items. On the average 36 information items were suggested to a task by PID. The higher number of the access to the attached items could mean that the requirements analysts attached very valuable information items which are worth while reusing, to their tasks. Compared to that, there is still room for improving the accuracy of PID. In the questionnaire, one of the two analysts mentioned that the PID functionality helped him during the requirements analysis activities, especially for understanding the problem domain, while the other one suggested that information suggested by PID (predefined process information of requirements analysis) was not adequate. This shows that the usefulness of the PID functionality depends on the information repository and on the expertise of the user. We assume the PID functionality is useful especially for new employees and individuals who are newly assigned to a project. Moreover, since the task information can be reused with its related documents and process information in the TaskNavigator system, the PID could be helpful to support business take over.

In the questionnaire, the subjects indicated that the ATM enables them to organize tasks faster and to improve the efficiency as well as productivity in planning and organizing tasks. In addition, they also stated that the ATM functionality is intuitive and easy to learn. They also mentioned that the TaskNavigator system is easy to use in general, including the learning of operations. However, they "slightly disagreed" to both efficiency improvement and productivity increase by applying TaskNavigator application. This could be caused by the quality of the stored information in the repository, which was also figured out by one of the requirements engineers.

Aside from the difficulties with the information repository preparation, the case study results show that the subjects gave a positive feedback on the system functionalities, the ATM and the PID. This supports our idea that, reusing tasks or task models with relevant information is effective in the situations where the integrity of prior information is doubtful.

6. Conclusion

In this paper we presented TaskNavigator, a prototype that realizes a light-weight approach to task-specific, proactive document delivery. Although we used a flexible workflow management system as a basis for the prototype, the presented approach is also applicable to simpler to-do list applications as found in the personal information management tools (e.g., PDAs) of today's office workers.

Based on the positive qualitative results collected so far concerning TaskNavigator usage, and taking into account the positive evaluation results already obtained for our process-embedded information support[8], we believe that an efficiency gain can also be achieved in an everyday office setting with the approach presented here, by making documents more easily available during the office worker's tasks, and helping to prevent that relevant documents might be overlooked.

Acknowledgements

Work funded in part by "Stiftung Rheinland-Pfalz für Innovation" (InnoWiss, TEAL).

References

1) Mahe S, Rieu C, "Toward a Pull-Approach of KM For Improving Enterprise Flexibility", 1st Int. Symp. on Management of Industrial and Corporate Knowledge (ISMICK'97), Compiègne, France, October 1997

2) Abecker, A., Hinkelmann, K., Maus, H., and Müller, H.-J., editors (2002). "Geschäftsprozessorientiertes Wissensmanagement". xpert.press Springer.

3) Schuett P, "Wissensmanagement im E-Learning", J.UCS, the Journal of Universal Computer Science, as part of a special issue on Knowledge Management in 2003

4) Riss, U, Rickayzen, A, Maus, H, and Aalst, Wv (2005). "Challenges for Business Process and Task Management Journal of Universal Knowledge Management". Special Issue on Knowledge Infrastructures for the Support of Knowledge Intensive Business Processes 0(2):77-100.

5) Abecker A., Papavassiliou G., Ntioudis S., Mentzas G., Müller S. "Methods and Tools for Business-Process Oriented Knowledge Management - Experiences from Three Case Studies". ICE 2003 - 9th International Conference of Concurrent Enterprising, Espoo, Finland, 16-18 June 2003. pp. 245-254.

6) Wiig, K.M. "Perspectives on Introducing Enterprise Knowledge Management". Proc. Of the 2nd Int. Conf. on Practical Aspects of Knowledge Management (PAKM98)

7) BrainFiler is a software product of the brainbot technologies AG, Germany. brainFiler home page. http://brainbot.com/site3/produkte/brainfiler.

8) Elst, L. v., Aschoff, F.-R., Bernardi, A., Maus, H., and Schwarz, S. (2003). "Weakly-structured workflows for knowledge-intensive tasks: An experimental evaluation". IEEE WETICE Workshop on Knowledge Management for Distributed Agile Processes (KMDAP03). IEEE Computer Press.

注1) Google Desktop SearchはGoogle社の商標です.