Analysis and Comparison of Models for Individual and Organizational Knowledge Work

Ansgar Scherp, Daniel Eißing, and Steffen Staab

University of Koblenz-Landau, Germany, E-Mail: {scherp,eissing,staab}@uni-koblenz.de

Abstract

Making implicit procedural knowledge explicit by business processes is an established means for analyzing and improving efficiency of organizational knowledge work. Such an explicit definition of processes is only to a limited extend applicable to individual knowledge work due to its high complexity and dynamics. Nevertheless, there are structured and repeating activities in individual knowledge work that are worth to capture. Dynamic and ad-hoc workflows allow for capturing these activities. However, they do not allow for synchronizing them with business processes. This hampers communication of best practices and transfer of knowledge between the knowledge workers. Our analysis shows that none of the existing models fully provides the desired functionality. Thus, we have developed our own approach the pattern system strukt and applied it at the example scenario of a real architectural office. The strukt prototype helps knowledge workers to organize their activities, collaborate with others, and synchronize one's very individual activities with the business processes.

1 Introduction

For optimizing organizational knowledge work, implicit procedural knowledge is made explicit by defining processes [25], [26]. These processes are amenable for analysis, planning, and optimization. Besides optimizing business processes, the planning of an organizational IT-landscape benefits from explicit knowledge as it can be better adapted to the processes and requirements of the organization using, e.g., project management tools or service-oriented architectures [12]. In contrast, individual knowledge work considers the activities of knowledge management from the viewpoint of the knowledge workers. Unlike organizational business processes, activities of individual knowledge work are less amenable to analysis and planning. This is due to the high complexity and variability of individual knowledge work, which makes it harder to explicitly capture the activities. In addition, many activities of individual knowledge work occur only rarely. Thus, the high effort of defining explicit business processes is not justified. Although the activities of individual knowledge work are less amenable to analysis and planning, they often contain sub-activities that provide some structure and thus can be explicitly planned [34]. For example, in a large construction project it might be necessary to obtain approval before some further activities can be carried out.

Activities of individual knowledge work are also often part of organizational business processes and define process deadlines, communication partners, and documents [34]. Thus, it seems to be worthwhile to consider individual knowledge work from the perspective of business process optimization and to synchronize the activities of individual knowledge work with the processes of organizational knowledge work.

In this paper, we analyze and compare existing models and approaches for representing structured workflows of organizational knowledge management and activities in individual knowledge management, the weakly structured workflows. None of the existing models allow for representing both processes of organizational knowledge work and activities of individual knowledge work and integrating them. As solution to this problem, we have developed strukt, a system of ontology design patterns for the semantic integration of individual knowledge work and organizational knowledge work. The strukt application allows knowledge workers for collaboratively creating, modifying, and executing activities of individual knowledge work and synchronizing them with business processes.

2 Organizational Knowledge Work

In order to understand the challenges of providing a model for representing and integrating organizational knowledge work and individual knowledge work, we first introduce our notion of knowledge and the key concepts in organizational knowledge work, namely business processes and workflows. Subsequently, we discuss in Section 3 the characteristics of activities in individual knowledge work.

A frequent definition of knowledge [2] is its distinction from data and information [33], [32]. Data are the raw signals like digits and characters. Information enriches this data with some meaning, i.e., some metadata. Knowledge adds a purpose to information in order to achieve a specific goal within the organization. Knowledge is to be continuously revised within the organization. Based on this, organizational knowledge work considers knowledge from the perspective of the company or organization. A central role plays business process modeling. It aims at making implicit procedural knowledge of the organization explicit by defining processes and thus amenable to formal analysis and optimization [26]. A business process describes a set of organizational activities that uses one or more resources as input and produces a value for the customers of the organization [19].

Business processes consider the organizational activities in a top-down view and aim at improving the use of resources within the organization by an explicit analysis and planning of the organizational activities. Business process modeling differentiates between the *definition phase* and *execution phase* [38], [33]. In the definition phase, existing business processes are captured and orchestrated into a (semi-)formal business process model. In the execution phase, the previously created process model is implemented. To this end, a *workflow* is defined as supporting the operative execution of a business process in parts or total [38]. Workflows can be distinguished into *workflow models* and *workflow instances* [33]. The workflow model describes a generic definition of a workflow under consideration of activities, alternatives, and roles participating in the processes. Workflow instances are instantiations of a workflow model under consideration of current process parameters that define a concrete flow through the processes and capture the artifacts created while executing the processes. Adapting the workflow instance after its instantiation from the model is typically not foreseen [33]. If a workflow turns out to be inflexible or inappropriate at any time during the execution of a workflow instance, one needs to adapt the workflow model. This implies that all running workflow instances need to be adapted accordingly. Thus, a more flexible behavior of the workflow instances and less dependencies with the workflow model is desired.

3 Individual Knowledge Work

In contrast to organizational knowledge work, individual knowledge work [11], [23], [33] describes knowledge work from the perspective of the individuals, i.e., the knowledge workers. Individual knowledge work is characterized by a high degree of variability in its execution, self-organization, interdisciplinary, and communication orientation [33]:

- **Variability**: Individual knowledge work typically comprises activities that are difficult to plan due to their high variability. A huge amount of input information and often changing communication partners require frequent revisions of one's procedures in conducting the required activities. New information and changing requirements often imply an adaptation of the procedure while conducting the activities. Many activities are also conducted rarely or the first time. Without some explicitly captured procedural knowledge, it is the responsibility of the knowledge worker to find a solution for a problem or task. In order to integrate the varying tasks into a busy and fragmented work schedule [20], the knowledge worker constantly needs to evaluate the different, concurrent activities and coordinate them in an integrated workflow.
- **Self-organization**: The high complexity and variability of individual knowledge work makes it impossible to define a-priori an exact procedure and the resources needed for a task. Thus, it is typically the knowledge worker's responsibility to plan and manage the efforts for conducting a task in terms of time and resources. In contrast to traditional organizational structures, organizational hierarchies are less present in individual knowledge work. Thus, besides expert knowledge also the ability for self-organization is required from the individual knowledge workers.
- **Interdisciplinary**: In contrast to the traditional skilled engineering worker, knowledge workers are often not specialized and typically cannot specialize to one particular field of business activity. In lieu, knowledge workers frequently have to be experts in many fields. They need to be able to quickly acquire new skills when required and play different roles within the organization.
- **Communication-orientation**: Knowledge work often requires an active communication with other participants. For example, there is a high need for communication with colleagues and the management. However, there is also an increased communication need with external participants such as customers, suppliers, and administrative office.

4 Requirements for Integrating Organizational and Individual Knowledge Work

The requirements are derived from the analysis and discussion of organizational and individual knowledge work (see Sections 2 and 3). In addition, we have conducted expert interviews with knowledge workers of an architectural office [30]. Each requirement is briefly described and a reference number REQ-<*x*> provided:

- **Weakly Structured Workflows (REQ-1)**: Individual knowledge work is characterized by a high complexity and variability [20]. Thus, the resources needed and activities to be conducted in a specific task are often not known *a priori*. Thus, a model support for representing weakly structured workflows is needed. This model shall be adaptable during execution time without violating the consistency of other workflow executions.
- **Support for Structured Workflows (REQ-2)**: In addition to representing the flexible activities of individual knowledge work, also the structured processes of organizational knowledge work need to be represented [19].
- **Integrating Weakly Structured Workflows and Structured Workflows (REQ-3)**: We need to integrate and synchronize the activities of weakly structured workflows and processes of structured workflows.
- **Workflow Models and Instances (REQ-4)**: A common feature of traditional business process models is to distinguish workflow models from actually executed workflow instances [33]. Such a distinction is often not made in individual knowledge work, as the activities in individual knowledge work are high in complexity and variability (cf. discussion in Section 3). We can also learn some generic procedural knowledge from the execution of weakly structured workflows. Thus, the distinction between instance and model should also be made for weakly structured workflows. It should be possible to modify workflow instances without affecting the model or any other currently executed workflow instance. Finally, it should be possible to abstract executed workflow instances to workflow models.
- **Descriptive Workflow Information (REQ-5)**: It should be possible to describe the resources involved in structured workflows and weakly structured workflows. These include the resources used, processed, or created in a workflow [26], the tools that are applied, the execution status of workflows, and information about scheduling like earliest begin, deadline, and others.

5 Comparison of Models for Knowledge Work

Based on the discussion of organizational and individual knowledge work in Sections 2 and 3 and the requirements in Section 4, we analyze and compare existing models for structured workflows and weakly structured workflows from the areas of traditional business process modeling, ad-hoc and dynamic workflows, as well as Semantic Web ontologies.

5.1 Traditional Business Process Models

Models for describing traditional business processes such as the Business Process Modelling Notation (BPMN) [28], extended Event-driven Process Chain (eEPC) [29], and Action Port Model (APM) [9] have a rigorously determined process execution flow and separate the business process modeling from its execution (see Section 2). Thus, they are less suitable with respect to the requirements stated in this work. Only APM suggests an adaptation of the business process model at runtime. However, the APM lacks of formality and low flexibility of the concepts it defines.

From the field of knowledge management, we find models like the Knowledge Business Process Improvement Framework (KBPI) [10] and Knowledge Modeling and Description Language (KMDL) [16]. Besides describing traditional business processes, these models provide approaches for planning of knowledge activities in the organizations. However, these models focus on capturing and planning organizational business processes for the purpose of management. Thus, they are comparable to the traditional business processes.

The Business Process Execution Language (BPEL) [4] allows for the technical interface specification of (automatable) web services. Planning and modeling individual knowledge work in form of weakly structured workflows is not part of BPEL. The extensions BPEL4People [24] and HumanTask [1] allow in principle for describing activities of individual knowledge work and model them as *black-boxes* within the organizational business processes. However, a specification of the black-boxes such as partitioning a task into sub-activities is conducted outside of the system. As such, the BPEL4People and HumanTask extensions are de-facto the same as the traditional business process models and less appropriate for representing individual knowledge work.

5.2 Models for Weakly Structured Workflows

Ad-hoc and weakly structured workflow models like the Process Meta-Model (PMM) [6] and the Task-Concept-Ontology (TCO) [34] do not require a strictly determined process flow like the traditional business process models and may be automatically extracted from natural language descriptions [17]. Such models are suitable to represent individual knowledge work. However, the lack of formal precision and missing integration with traditional business processes hinder their reuse.

5.3 Semantic Models

Semantic models employ technologies of the Semantic Web [3] to represent the processes and activities in knowledge work. Traditional business process models like BPMN [28] and eEPC [29] are available as semantic models in form of the sBPMN [21] and sEPC [21] ontologies. However, these ontologies are a mere mapping of the traditional models' features to a semantic model. Thus, sBPMN and sEPC still lack support for representing weakly structured workflows. Other models like the Toronto Virtual Enterprise Ontology (TOVE) [13], Enterprise Ontology (EO) [35], Knowledge Management Ontology (KMO) [22], Core Organizational Knowledge Entities Ontology (COKE) [18], as well as the W3C standard OWL-S [37] share the structural characteristics of traditional business process models such as a strictly determined business process flow and an explicit separation of modeling and executing the processes (see Section 2). Because of this, they are less suitable to model weakly structured workflows. Also the degree of formalization of the ontologies differs. Instead of providing formalization using description logics [5], often only a taxonomy in a specific domain is used. The DOLCE+DnS Plan Ontology (DDPO) [15] provides rich axiomatization and formal precision. This is obtained by using the foundational ontology DOLCE+DnS Ultralight (DUL) [7] as basis. DDPO specializes the ontology design pattern Descriptions and Situations (DnS) provided by DUL, which allows for defining different contextual situations by using roles. The concept $\mathrm{Description}$ defines the roles that are relevant in a specific situation. A $\mathrm{Situation}$ satisfies the $\mathrm{Description}$ when there is a real-world observable situation that matches the roles defined by the $\mathrm{Description}$. DDPO's Plan

concept is a specialized Description and defines at least one Task, one agentive role participating in the task, and a Goal. Thus, a Task is an activity within a plan and a Goal is a status that shall be achieved. A PlanExecution is an actual execution of a Plan. As such, a PlanExecution is a specialization of Situation and represents real-world situations that satisfy a Plan. In principle, the DDPO can be used to represent both traditional workflows as well as weakly structured workflows. However, DDPO does not distinguish structured workflows of organizational knowledge work and weakly structured workflows of individual knowledge work (REQ-3). In addition, the distinction between workflow models and instances (REQ-4) is not explicit in DDPO and it also does not provide for representing descriptive workflow information (REQ-5).

5.4 Summary

A summary of the analysis and comparison of the existing models for representing organizational and individual knowledge work is presented in Figure 1. For each model, the fulfillment with respect to the requirements stated in Section 4 is shown. The "+"-symbol means that the model fully supports the requirement. The "o"-symbol means that the requirement is partially fulfilled or with limitations and finally "-" is used when the model does not fulfill the requirement.

	Weakly Structured Workflows	Structured Workflows	Integration SsW & SW	Workflow Model	Workflow Instances	Resources, Tools, Status, Scheduling
BPMN [OMG '10]	-	+	-	+	-	o
(e)EPC [Scheer '01]	-	+	-	+	-	o
APM [Carlsen '98]	o	+	-	+	-	o
BPEL [Alves et al. '07]	-	+	-	+	-	+
KBPI [Dalmaris et al. 2007]	-	+	-	+	-	o
KMDL [Gronau et al. 2008]	-	+	-	+	-	o
PMM [Bolinger et al. '09]	+	+	-	o	+	+
TCO [Schwarz '03]	+	-	+	o	+	-
sBPMN [Hepp et al. '06]	-	+	-	+	-	o
sEPC [Hepp et al. '06]	-	+	-	+	-	o
TOVE [Fox et al. '98]	-	o	-	+	+	o
EO [Uschold et al. '98]	-	+	-	+	+	+
KMO [Holsapple & Joshi '04]	-	+	-	+	+	o
COKE [Gualtieri & Ruff. '05]	-	+	-	+	+	o
OWL-S [Martin et al. '04]	-	+	-	+	+	-
DDPO [Gangemi et al. '04]	+	o	o	+	+	-

Figure 1: Comparison of Models for Individual and Organizational Knowledge Work

None of the existing models fulfill all requirements as introduced in Section 3. Traditional business process models like BPMN and (e)EPC miss representing weakly structured workflows of individual knowledge work. On contrary, weakly structured workflows such as PMM are in principle enabled to represent the activities of individual knowledge work. However, they lack formal precision and do not allow for an integration with traditional business process models. TCO supports in principle the integration but does not consider defining structured workflows. DDPO differs from the other models insofar as it in principle allows for modeling both organizational business processes and activities of individual knowledge work. In addition, DDPO makes use of the foundational ontology DUL, which has proven in the past to serve as solid modeling framework for formal ontologies [31]. DDPO inherits the formal nature of DUL and thus is well suited for describing and integrating organizational and individual knowledge work.

6 Pattern System *strukt* and its Application

In order to represent and integrate organizational knowledge work and individual knowledge work, we have developed the pattern system *strukt* (from the German word Struktur). It consists of different ontology design patterns and is introduced in Section 6.1. The strukt application described in Sections 6.2 and 6.3 uses the pattern system. It allows for conducting individual activities and synchronizing them with business processes.

6.1 Integrating Organizational and Individual Knowledge Work with strukt

The foundational ontology DOLCE+DnS Ultralight (DUL) [7] serves as basis for our ontology strukt as shown in Figure 2. Foundational ontologies define the very basic and general concepts and relations that make up the world [27] like objects and events. They provide a high axiomatization of the concepts defined and are applicable to a wide variety of different fields. As foundational ontologies like DUL follow a pattern-oriented design, also strukt provides ontology design patterns. Similar to design patterns in software engineering [14], ontology design pattern describe a specific, recurring modeling problem that occurs in a specific modeling context and provide a proven, generic solution to this modeling problem [8], [14]. The pattern system strukt [30] is based on DDPO (see Section 5.3) and specializes different design patterns provided by DOLCE+DnS Ultralight (DUL). Most importantly, it re-uses the Descriptions and Situations (DnS) pattern that provides a formal specification of context [27]. We briefly describe the pattern system strukt. For a detailed description, we refer to [30]. As shown in Figure 2, the foundational ontology DUL provides the Workflow Pattern, which reuses and specializes the DnS Pattern as indicated by the dotted arrow. The Workflow Pattern provides a formalization of the planning of processes. Further patterns reused by the Workflow Pattern are the Task Execution Pattern that defines processing of tasks in activities, the Role Task Pattern that associates roles with tasks, and the Part-of Pattern for (de-)composing entities [36]. The Sequence Pattern describes the order of entities such as tasks and activities. Foundational ontologies like DUL can be refined by core ontologies towards a particular field by adding detailed concepts and relations [27].

Core ontologies specialize the generic concept definitions of the foundational ontology but are still applicable to a large variety of different domains. The ontology strukt is such a specialization of DUL and reuses different ontology design patterns that DUL offers. Central patterns of strukt are the Weakly Structured Workflow Pattern and the Structured Workflow Pattern in combination with the Transition Pattern as shown in Figure 2. They fulfill requirements REQ-1 and REQ-2 as described in Section 4. The Workflow Integration Pattern allows for integrating weakly structured workflows and structured workflows and thus serves REQ-3. The Workflow Model Pattern differentiates workflow models and workflow instances as requested by REQ-4. Finally, the Condition Pattern, Resource Pattern, Status Pattern, and Scheduling Pattern allow for further describing weakly structured workflows and structured workflows and thus fulfill REQ-5. Each pattern is designed to solve a specific modeling problem towards the overall goal of representing and integrating organizational and individual knowledge work. Thus, the patterns of strukt are not a set of independent ontology design patterns but define relations to each other and are designed to be applied in combination. Such a set of related ontology design patterns is called a pattern system [8]. Finally, the pattern system strukt can be applied in various domains that need to represent knowledge work. To this end, it is specialized and integrated with, e.g., an architectural ontology or financial administration ontology as shown in Figure 2.

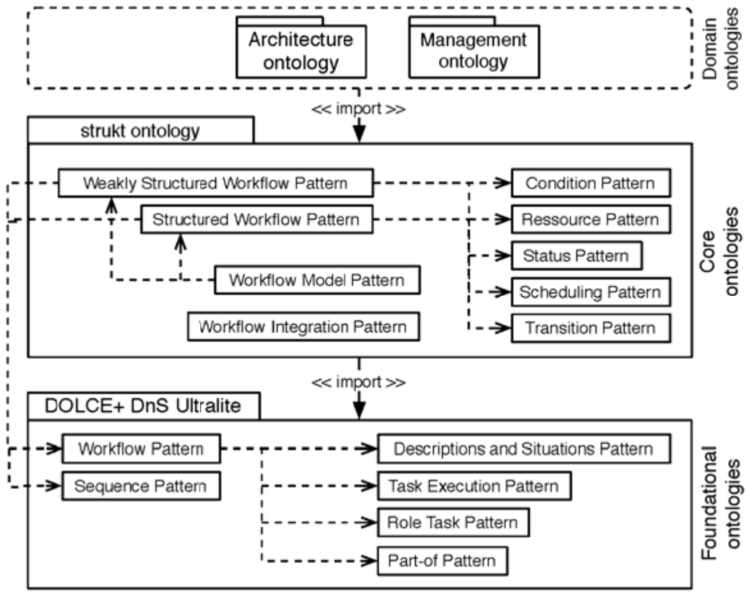

Figure 2: Overview of the strukt Ontology in UML Notation

6.2 Web-based Prototype Application for the Individual Knowledge Worker

The strukt prototype allows knowledge workers to organize their activities. This is shown at the example of an architectural office conducting different kinds of construction projects depicted by the screenshot in Figure 3. The web-based prototype provides a *task space* for managing the weakly structured workflows with their tasks and sub-tasks. The knowledge workers can view the details of tasks, create and modify tasks, extract workflow models from workflow instances and instantiate workflow models. To display the tasks and subtasks of a weakly structured workflow, the small triangle symbol next to the task is clicked as shown in Figure 3 at the example *Building application Mornhinweg Inc*. Deadlines, appointments, involved agents, task status, and other important details are shown on the right hand side. The checkbox left to a task name is clicked to mark it as finished. A lock symbol in the checkbox (indicated as small box) shows that a task has unfulfilled dependencies (indicated by the arrows) and thus cannot be accomplished. Optional tasks are indicated with the keyword *(opt)*. The order of tasks can be changed by drag and drop interaction. The *Properties* tab shows details of a task such as its status and time, contributors, location, deadline, and project. Here, further contributors for a task can be invited as well as tasks assigned to colleagues like other knowledge workers. The tools used to process a task can be investigated under the *Tools* tab and the conditions associated with a task like a role that needs to sign a document are shown with the *Conditions* tab (not shown in the figure).

The strukt prototype provides the *workflow transformation menu* depicted in Figure 4 to abstract a workflow model from an executed workflow instance. The tasks and subtasks of the executed workflow are the table rows. The columns represent the task names, conditions, roles, involved documents and tools, and whether the task is optional. For creating the workflow model, tasks can be removed and reordered by drag and drop interaction.

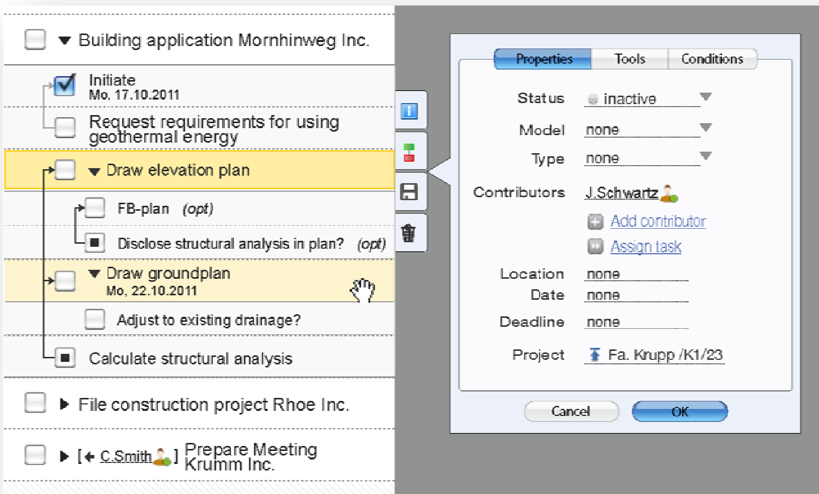

Figure 3: User Interface of the strukt Prototype for the Individual Knowledge Worker

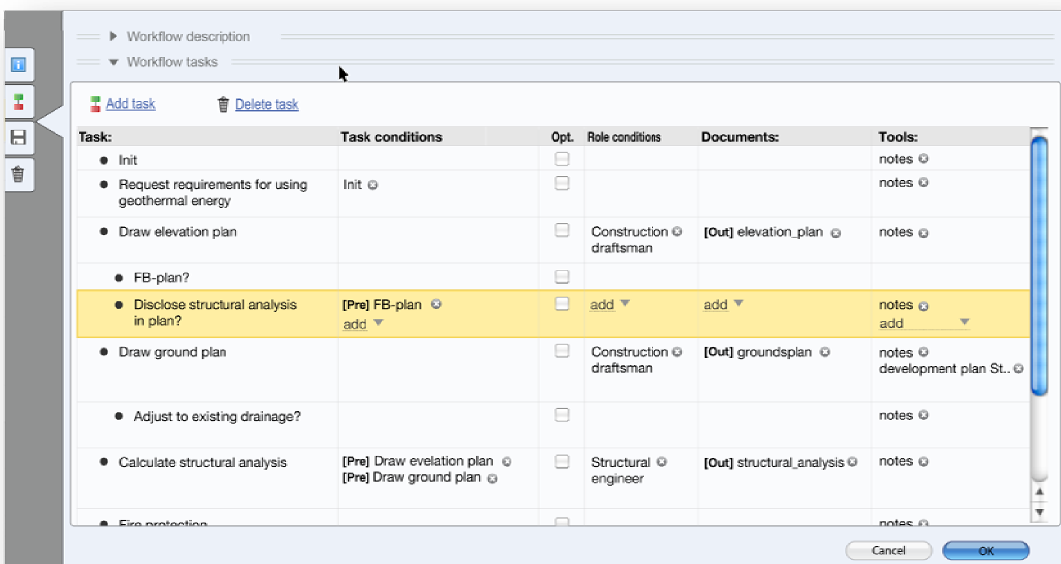

Figure 4: Creating a Workflow Model from an Executed Workflow Instance

6.3 Prototype Integration with Structured Workflows

The strukt prototype provides a simple workflow management system as test environment for synchronizing weakly structured workflow activities and pre-defined business processes. It serves as proof of concept that weakly structured workflows can be connected with structured ones, i.e., that individual knowledge work can be integrated with organizational knowledge work using the pattern system strukt. We have implemented different test cases: For example, a task is assigned from a business process to an agent of a weakly structured workflow. In another example, the status of a task in a weakly structured workflow is observed by a business process. The business process of the structured workflow is only finished when the task in the weakly structured workflow is marked as accomplished. A user interface for the workflow management system is not provided. It is assumed that the strukt prototype is integrated in an existing business process engine providing its own interface.

7 Lessons Learned

The prototype application demonstrates that the pattern system strukt can actually be used to represent structured workflows and weakly structured workflows and that structured processes and weakly structured activities can be integrated and synchronized. Early feedback from knowledge workers is promising. However, it also shows that it is difficult to develop a user interface at the right level of abstraction that is intuitive to use while at the same time provides powerful features for managing one's activities, assigning tasks to others, and collaborating with others.

In contrast to prior work, the strukt prototype allows for contextualized views onto the same workflow. This is achieved by using the DDPO and the ontology design pattern DnS (see Sections 5.3 and 6.1). For example, a knowledge worker like the architect Jim is responsible for filing the building application and instantiates the associated weakly structured workflow. As individual knowledge workers valuate and conduct tasks differently, the strukt prototype allows Jim to modify the weakly structured workflow during its execution. Also concrete construction projects and specific stakeholder requirements may imply adaptations of the weakly structured workflow execution. By design of the strukt ontology and its prototype, these workflow modifications do not require changing the actual model of the weakly structured workflow. In addition, the strukt prototype also does not require adapting any existing executions of the same weakly structured workflow. Finally, it is still possible to abstract the recent execution of a weakly structured workflow (as shown in Figure 4) and create a workflow model from it for future reuse.

8 Summary

Business process management is an established means for improving efficiency in organizational knowledge work by capturing implicit procedural knowledge in the organization and making it explicit through the specification of processes. On the contrary, individual knowledge work is characterized by a high degree of complexity and variability, which is difficult to capture in an information system. This hinders adoption and communication of best practices in individual knowledge work within the organization and its synchronization with organizational knowledge work. Nevertheless, it seems to be worthwhile to consider individual knowledge work from the perspective of business process optimization as it is often embedded in organizational businesses processes and provides some structure like sub-tasks that can be explicitly captured. Thus, we have analyzed different existing models and approaches for representing individual knowledge work and organizational knowledge work. As none of the existing solutions allow for synchronizing the activities of individual knowledge work and processes of organizational knowledge work, we have developed the pattern system strukt that is based on the foundational ontology DOLCE+DnS Ultralight and makes use of ontology design patterns. A prototype application shows the general applicability of our pattern system approach.

Acknowledgement: This work has been co-funded by the EU in FP7 in the ROBUST project (257859).

9 References

[1] Agrawal A; Amend M; et al. (2007): Web Services Human Task. http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-bpel4people/WS-HumanTask_v1.pdf. Last retrieved 07.09.2011.

[2] Alavi, M; Leidner, DE (2001): Review: Knowledge Management and Knowledge Management Systems, MIS quarterly, 25(1):107-136.

[3] Allemang, D; Hendler, J (2008): Semantic Web for the Working Ontologist, Morgan Kaufmann.

[4] Alves, A; Arkin, A; et al. (2007): Web Services Business Process Execution Language, http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html. Last retrieved 07.09.2011.

[5] Baader, F; Calvanese, D; McGuinness, D; Nardi, D; Patel-Schneider, P (2004): The description logic handbook, Cambridge.

[6] Bolinger, J; Horvath, G; Ramanathan, J; Ramnath, R (2009): Collaborative workflow assistant for organizational effectiveness, In: *Applied Computing*, 273-280, ACM.

[7] Borgo, S; Claudio, M (2009): Foundational choices in DOLCE, In: *Handbook on Ontologies*, Springer.

[8] Buschmann, F; Meunier, R; Rohnert, H; Sommerlad, P; Stal, M (1996): Pattern-Oriented Software Architecture, Volume 1: A System of Patterns, Wiley.

[9] Carlsen, S (1998): Action Port Model: A Mixed Paradigm Conceptual Workflow Modeling Language, In: *Cooperative Information Systems*, 300-309, IEEE.

[10] Dalmaris, P; Tsui, E; Hall, B; Smith, B (2007): A Framework for the Improvement of Knowledge-Intensive Business Processes, Business Process Management Journal, 13(2):279-305, Emerald, UK.

[11] Davis, GB (2002): Anytime/Anyplace Computing and the Future of Knowledge Work, Communications ACM, 45(12):67-73, ACM.

[12] Erl, T (2005): Service-Oriented Architecture: Concepts, Technology, and Design, Prentice Hall.

[13] Fox, MS; Barbuceanu, M; Grüninger, M; Lin, J (1998): An Organization Ontology for Enterprise Modelling, In: *Simulating Organizations: Computational Models of Institutions and Groups*, 131-152, MIT Press.

[14] Gamma, E; Helm, R; Johnson, R; Vlissides, J. (2004): Design patterns: elements of reusable object-oriented software, Addison-Wesley.

[15] Gangemi, A; Borgo, S; Catenacci, C; Lehmann, J (2004): Task Taxonomies for Knowledge Content, METOKIS Deliverable D7, 20-42. http://www.loa-cnr.it/Papers/D07_v21a.pdf. Last retrieved 07.09.2011.

[16] Gronau, N; Palmer, U; Schulte, K; Winkler, T (2003): Modellierung von wissensintensiven Geschäftsprozessen mit der Beschreibungssprache K-Modeler, Professionelles Wissesmanagement - Erfahrungen und Visionen, 315-322.

[17] Groth, PT; Gil, Y (2009): A scientific workflow construction command line, In: *Intelligent user interfaces*, ACM.

[18] Gualtieri, A; Ruffolo, M (2005): An Ontology-based Framework for Representing Organizational Knowledge, In: *Knowledge Management*, Graz, Austria.

[19] Hammer, M; Champy, J (2003): Reengineering the Corporation: A Manifesto for Business Revolution, Harper Paperbacks.

[20] Hart-Davidson, W; Spinuzzi, C; Zachry, M (2007): Capturing & Visualizing Knowledge Work: Results & Implications of a Pilot Study of Proposal Writing Activity, In: *Design of Communication*, 113-119, ACM.

[21] Hepp, M; Belecheanu, R; et al. (2006): Business Process Modelling Ontology and Mapping to WSMO, SUPER Project IST-026850.

[22] Holsapple, CW; Joshi, KD (2004): A Formal Knowledge Management Ontology: Conduct, Activities, Resources, and Influences: Research Articles, Journal of the American Society for Information Science and Technology, 55(7):612, John Wiley.

[23] Kidd, A (1994): The Marks are on the Knowledge Worker, In: *Human Factors in Computing Systems*, 186-191, ACM.

[24] Kloppmann, M; König, D; et al. (2005): WS-BPEL Extension for People-BPEL4People, http://www.sdn.sap.com/irj/servlet/prt/portal/prtroot/docs/library/uuid/cfab6fdd-0501-0010-bc82-f5c2414080ed. Last retrieved 07.09.2011.

[25] Leistner, F (2010): Mastering Organizational Knowledge Flow: How to Make Knowledge Sharing Work. Wiley and SAS Business Series, Wiley & Sons.

[26] Nonaka, I (1991): The Knowledge-Creating Company, Harvard Business School Publication Corp., 69(6):96-104.

[27] Oberle, D (2006): Semantic Management of Middleware, Springer.

[28] Object Management Group (2010): Business Process Model and Notation (BPMN), http://www.omg.org/cgi-bin/doc?dtc/10-06-04.pdf. Last retrieved 2011.

[29] Scheer, AW (1998): Aris-Business Process Frameworks, Springer.

[30] Scherp, A; Eißing, D; Staab, S (2011): strukt - A Pattern System for Integrating Individual and Organizational Knowledge Work, Int. Semantic Web Conf., Bonn, Germany, 2011.

[31] Scherp, A; Saathoff, C; Franz, T; Staab, S (2011): Designing core ontologies, Applied Ontology, IOS Press, 6(3):177-221.

[32] Schreiber, G (2000): Knowledge engineering and management: the CommonKADS methodology, MIT Press.

[33] Schwarz, S; Abecker, A; Maus, H; Sintek, M (2001): Anforderungen an die Workflow-Unterstützung für Wissensintensive Geschäftsprozesse, Professionelles Wissens-management. Baden-Baden, Germany.

[34] Schwarz, S (2003): Task-Konzepte: Struktur und Semantik für Workflows, Professionelles Wissesmanagement, Luzern, Switzerland, 28:351-356.

[35] Uschold, M; King, M; Moralee, S; Zorgios, Y (1998): The Enterprise Ontology, The Knowledge Engineering Review, 13(1):31-89, Cambridge University Press.

[36] Varzi, AC (1996): Parts, Wholes, and Part-Whole Relations: The Prospects of Mereotopology, Data & Knowledge Engineering, 20(3):259-286, Elsevier.

[37] W3C (2004): OWL-S. http://www.w3.org/Submission/OWL-S. Last retrieved: 07.09.2011.

[38] Workflow Management Coalition (1999): WFMC-TC-1011: Terminology & Glossary. http://www.wfmc.org/. Last retrieved 07.09.2011.