Examining Adaptive Case Management to Support Processes for Enterprise Architecture Management

Matheus Hauder, Dominik Münch,
Felix Michel, Alexej Utz, Florian Matthes
Technische Universität München (TUM)
Chair for Informatics 19 (sebis)
Boltzmannstr. 3, 85748 Garching bei München, Germany
Email: {matheus.hauder, dominik.muench, felix.michel, alexej.utz, matthes}@tum.de

***Abstract*—Enterprise Architecture (EA) management is a well-defined practice for conducting coherent analysis, design, planning, and implementation for the successful execution of business strategy. In practice many organizations are struggling with EA frameworks since the processes to manage the EA are not adequately supported. In this paper we examine Adaptive Case Management (ACM) as an emerging paradigm to support agile, lean, and collaborative processes for EA management (EAM). In contrast to traditional workflow management, ACM empowers end users to adapt their processes at run-time. Based on the findings of an extensive literature review we derived requirements for ACM and developed a prototype based on these requirements to support processes for EAM. We applied our solution for the development of a planned state of the architecture in the EA department of a German insurance organization. The findings from expert interviews with three enterprise architects and two EA consultants of this insurance organization are very promising and provide valuable insights how more effective process support for EAM can be achieved in the future.**

I. Introduction

Organizations are challenged with increasing complexity, quickly changing business requirements, and globalization. Enterprise Architecture (EA) is promoted as a means to conquer these challenges and realize better alignment of business and IT using a holistic model of the organization's business processes, information systems, infrastructure components as well as relationships among them [1], [2], [3]. Being applied by an increasing number of organizations, the corresponding discipline EA management (EAM) fosters the mutual alignment of business and IT [3]. Depending on the concerns of involved stakeholders, e.g. decision makers, solution architects, and domain experts, several instruments are required to effectively manage the EA.

These instruments comprise an information model that captures required entities to answer stakeholder concerns, reports to analyze instances of this information model, and processes to develop, manage, and update the EA [4], [5], [6]. While documentation and analysis of EA information models is already widely supported in practice through existing tool solutions [7], support of processes for the management of the EA are missing to a large extent. In this paper we refer to processes that describe how management of the EA is performed and not processes placed at disposal through IT solutions to support the business.

Critical challenges in EAM are discussed based on a literature survey by Lucke et al. in [8] and validated with expert interviews in [9]. The authors identified a lack of governance structure in many EA projects as one critical EAM challenge. The main reason for this challenge is that *"no formal steps exist for defining, maintaining, and implementing EA and EA frameworks are not rigid enough in describing these steps"* [8]. The approach for case handling in EAM presented in this paper proposes a solution to support these formal steps in EAM processes.

Well-established EA frameworks only provide very generic process descriptions to implement an EAM function, e.g. TOGAF [4], [10]. These process definitions are on a very abstract level and provide solely an informal guidance that can be used as a starting point for further refinement in organizations. First steps towards more detailed EAM process descriptions in literature are described by Buckl in [4] and Moser et al. in [5], whereas neither of these approaches suggests proper tool support or implementation for their approach.

Similar to the support of processes in the IT service management domain through configuration management solutions [11], processes for EAM could be implemented in EA tool solutions to support and capture knowledge of enterprise architects in their daily operations. Processes for EAM deal with the development of a target architecture, documentation and maintenance of the current architecture, definition of roadmaps, EA strategy development, and control of the architecture to name a few [5]. Implementing these processes is challenging since they are very knowledge-intensive and exceptions during the process appear frequently.

Using workflow management models to capture all these possible exceptions is not feasible, because the solutions get too complex to manage and maintain [12]. These workflows are typically process-driven and highly structured. A main focus in workflow management systems is the improvement of efficiency for business processes in organizations. Business processes encompass a larger scope compared to workflows since they also include data-driven and unstructured processes, e.g., groupware solutions, which are outside the scope of workflows [13]. Adaptive Case Management (ACM) systems balance between these data- and process-driven solutions and can be either completely structured or variable so that deviations of the specified routes in the processes are possible [14].

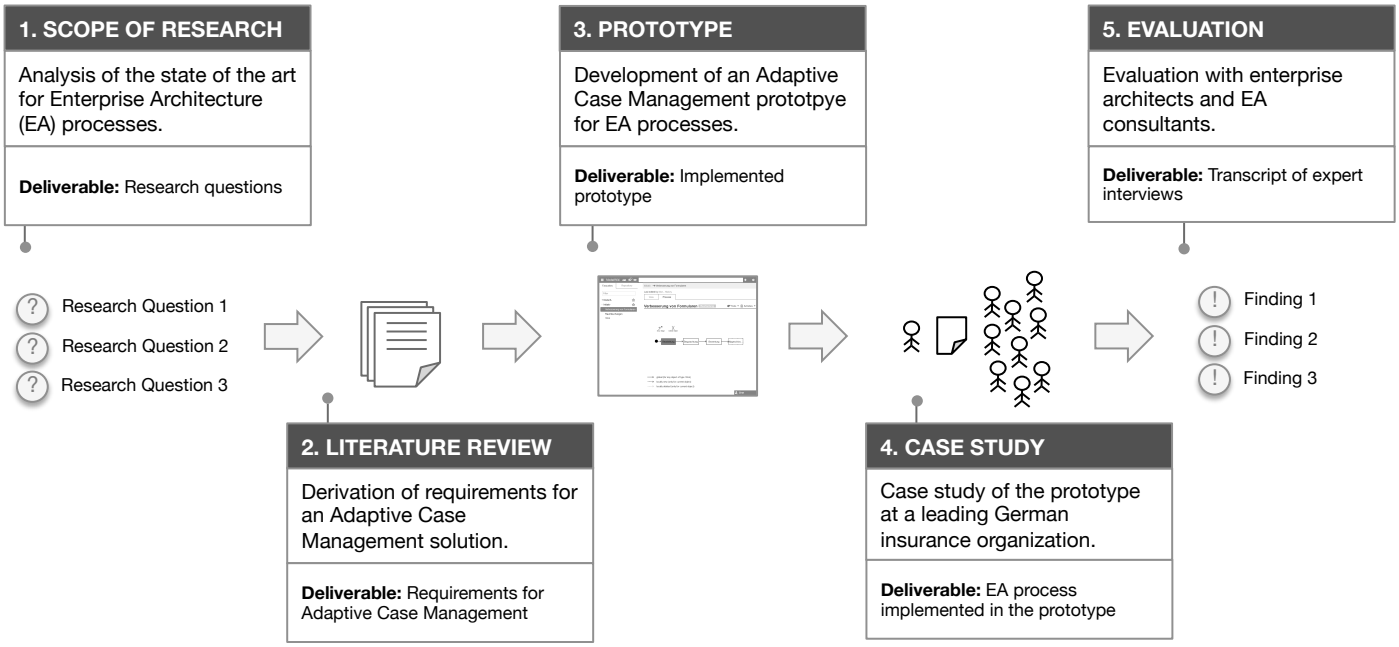

Fig. 1. Individual steps and their deliverables applied within the research method in this paper to examine the applicability of ACM for EAM processes

Given the limitations of existing workflow management solutions, which are too restrictive and struggle with changes during process execution [15], we propose the adoption of the ACM paradigm to implement EAM processes in organizations. The ACM paradigm is considerably different from workflow management since it is not focusing on what should be done rather than what can be done [14]. Instead of a strict separation between design-time and run-time of a process in workflow management, the ACM paradigm follows the idea of templates that can be adapted by end users at run-time [16]. In this sense templates are not implemented using a programming language by an IT expert, but end users are empowered to adapt and incrementally improve these templates.

The ACM paradigm is not dictating end users a predefined course of action, but provides them with adequate knowledge about the case empowering them to make decisions on their own to facilitate an emergent design of these processes. Support for these knowledge-intensive processes will be a decisive factor for many organizations [17], [18]. In this paper, we examine the ACM paradigm to support EAM processes which are considered as very knowledge intensive. A comprehensive overview of requirements that a solution needs to fulfill for EAM processes is presented by Buckl in [4].

II. Research Methodology

We examined ACM to support EAM processes using a design science approach [19] with the major steps and deliverables that are illustrated in Figure 1. In the first step we analyzed the state of the art for EAM processes (cf. Section III) and developed three research questions. In the second step we derived requirements for an ACM solution from a literature review (cf. Section IV) to answer our first research question. Based on the terms ”Case Handling”, ”Case Management”, and ”Adaptive Case Management” we conducted an extensive literature review of conference proceedings, journals and books with the Web of Science, Google Scholar, IEEExplore, CiteSeerX, SpringerLink and the library of our research institution using the structured approach according to Webster et al. [20] to derive ACM requirements for the first research question:

- *Research Question 1: What are the requirements that need to be fulfilled for an ACM solution?*

Based on these requirements we developed an ACM prototype for EAM processes. The solution is a web application that allows end users to adapt process templates at run-time. These templates can describe EAM processes from a predefined library. This prototype is the basis for a case study that we conducted in a leading German insurance organization (cf. Section V) to evaluate the second research question:

- *Research Question 2: How can an ACM solution be applied to support EAM processes?*

The organization already has a process for the development of a planned state of the EA defined that is currently not supported through an EA tool. The process is very similar to the definition in the TOGAF Architecture Development Method (ADM) [10]. Roughly speaking, it defines an architecture vision and analyzes the business, information systems, and technology architecture in order to describe a future state of the EA. We evaluated the prototype for the existing process to develop a planned state of the EA in this organization to answer the third research question:

- *Research Question 3: To which extent is an ACM solution useful for EAM processes in practice?*

Finally, we conducted interviews with all members of the EA team (three enterprise architects and two EA consultants) from this organization. Section VI presents our findings from these five expert interviews and discusses the feasibility of our approach to support EAM processes.

III. Related Work

In this section we summarize relevant related work on processes and coordination in EAM, workflow management, and case management. To the best of our knowledge this paper is the first approach towards better tool support of EAM processes through the application of the ACM paradigm.

A. Processes and Coordination

Various process patterns for EAM are presented by Moser et al. in [5]. In this paper the authors illustrate EAM process patterns for centralized manual data maintenance, decentralized manual data maintenance, automatic data maintenance, architecture controls by applying a release workflow, lifecycle management as well as verification and audit. These process patterns are embedded within a high level generic EAM process landscape. All patterns are based on the methodology patterns that can be found in the EAM pattern catalog [21]. This catalog is a collection of patterns for EAM that were observed in practice. Originating from these patterns the method building blocks consist of several functions for EAM processes that can be used to extend existing EA frameworks and develop an organization specific method [4].

A process for automated maintenance of EA models is presented by Farwick et al. in [22]. This process describes integration of federated information sources with a central EA repository. Although the presented process is more detailed, it provides no means for the resolution of conflicts during integration of different data sources and a concrete tool implementation is missing. In our previous work the conflict resolution process presented by Roth et al. in [23] shows a solution as well as tool implementation for this problem. In this solution tasks are used to collaboratively resolve conflicts that appear during automated model maintenance. Nevertheless, these tasks are limited to the resolution of conflicts and it is not possible to adapt these tasks as well as the overarching process by end users. The solution in this paper is an extension of our previous work and could be integrated with these tasks.

Progress of EAM processes depends on the availability of information to a large extent, e.g. information about the EA or ongoing projects from stakeholders in the organization [4]. This makes data to the primary driver of EAM processes and not so much a predefined course of action. Reacting to exceptional changes is an important capability for EAM process support, since many enterprise architects are faced with ad-hoc EA demands from decision makers or stakeholders in their organization [31]. Existing frameworks for EAM only provide process descriptions on a very coarse-grained level, e.g. the architecture change management steps defined in TOGAF [10]. Example steps are for instance establish value realization process, manage risks, or provide analysis. The ACM paradigm is able to support these steps through its goal oriented approach [16], [30]. Additional requirements and a comparison of processes for EAM in various frameworks is presented by Buckl in [4]. Considering existing literature on EAM processes presented in the previous section, we expect that many other EA frameworks have very similar requirements and the solution proposed in this paper can be applied for them likewise.

A coordination model for enterprise architecting effectiveness has been presented by Espinosa et al. [32]. In their paper the authors explain how coordination processes, i.e. organic and mechanistic coordination, and cognitive coordination influence effectiveness of the architecting process. The coordination processes and cognitive coordination directly impact architecting effectiveness, whereas cognitive coordination additionally moderates impact of the coordination processes. Furthermore, the authors describe dynamic repercussions within this model as the self-fueling effect. The case handling approach presented in this solution is targeting coordination processes and therefore aims at improving the overall architecting effectiveness. In a paper presented by Espinosa et al. in [33] the authors discuss team knowledge as a primary cognitive coordination mechanism promoting the self-fueling effect. It is further emphasized that coordination processes purposely aimed at building up team knowledge lead to successful architecting. Our research to investigate case handling for EAM aims at improving these coordination processes by making them explicit and shareable in organizations.

B. Workflow Management

Workflow management technology enables organizations to handle complex and long-running processes more efficiently. The downside of this approach is that defined operational processes are very rigid and hard to modify. This makes it difficult to incorporate exceptional and evolutionary changes of a process [26]. Case handling is promoted as a paradigm to support flexible and knowledge-intensive business processes [25], [34], [35]. The main differences between workflow management and case handling found in literature are summarized by Van der Aalst et al. in [14]. Cases handling minimizes focus on control flow, i.e. the precedence relations among tasks, and determines the logical state of a case based on the presence of data objects. Case handling also separates between authorization and distribution through a variety of roles associated with a case or task. Next to the execution roles in workflow management, case handling provides a redo role to delegate activities and a skip role to jump over tasks [14], [35]. This provides a higher degree of flexibility during execution of the case and simplifies handling of exceptional changes.

C. Case Management

Case management can be further distinguished into Production Case Management (PCM) and Adaptive Case Management (ACM) [36]. In PCM all required tasks are defined ahead of time by a case designer, while ACM empowers end users to additionally adapt cases at run-time [37], [16], [36]. This makes ACM suitable for processes that are unpredictable in which so called knowledge workers are not controlled but responsible to perform tasks and make decisions on their own [38]. Minimum effort for setting up a process in an ACM system are data objects and goals. The ACM paradigm provides knowledge workers in the simplest case with goals and required information to achieve these goals. As a result processes with high control flow complexity and high variation in outputs might be easier to implement in ACM compared to traditional workflow management [30]. A detailed summary on the requirements for ACM are presented in the following section.

TABLE I. REQUIREMENTS FOR ADAPTIVE CASE MANAGEMENT DERIVED FROM THE LITERATURE REVIEW

ID	Requirement	Description	Reference
R1	Flexibility at run-time	Cases should not only be defined at design-time, but it also has to be possible that end users without any programming expertise can adapt them at run-time. This is necessary to handle unpredictable situations and react on exceptional changes during the execution of the process.	[24], [16], [25], [26]
R2	Evolving cases and templates	This requirement deals with the continuous collaborative improvement for future processes and not only the currently executed ones. Best practices need to be identified and generalized to process templates. Users also have to be supported through recommendations during the process execution that are derived from other similar cases.	[24], [16], [27], [28]
R3	Transparent responsibilities	Responsibilities for various parts in the case must be visible to all case workers. This is necessary to facilitate collaboration among the involved knowledge workers.	[29], [24], [16]
R4	Visible progress of the case	Case workers must be able to identify which parts of the case are work in progress and what the overall state of the case is. Open tasks as well as obstacles that pretend the case from being solved have to be visible to the case workers in the system.	[24], [16]
R5	Flexible assignment of roles	During execution of the case it should be possible that new case workers join the case. However, these roles should not always be mandatory and allow for enough flexibility during the execution of the case. In contrast to traditional workflow management, tasks should not only have an execute roles but also a skip and redo role.	[29], [24], [16], [14]
R6	Definition of case objectives	It has to be possible to express objectives and goals of the case that describe what should be achieved as final result. While the individual process steps to achieve this result might be unknown or highly volatile the case objectives will remain more stable over time.	[29], [24], [16], [30]
R7	Hierarchical structure of tasks	Structuring tasks in hierarchies makes it easier for case workers to add new tasks compared to process networks consisting of many interlinked elements. Tasks can be defined on an abstract level in the beginning and refined incrementally with increasing maturity or experience.	[24], [16]
R8	Integration of data in the case	Processes and data are tightly integrated in case management due to the document-centric nature of knowledge work. In contrast to traditional activity-centric workflow management, data is the primary driver for the progress of the case.	[24], [29], [16], [14]
R9	Definition of logical dependencies between tasks	Temporal-logical dependencies are required in the solution since not all tasks might be executable at any time in a process. Some tasks might have pre-conditions that have to be fulfilled before they can be executed correctly. The definition of these dependencies should be performed by the users.	[29], [16]
R10	Understandable and adaptable for business users	Business users without any technical programming expertise or modeling knowledge have to be able to comprehend the definition of the cases. They also have to be empowered to make decisions on their own and plan upcoming tasks in the system.	[29], [16]

IV. REQUIREMENTS FOR ADAPTIVE CASE MANAGEMENT

In order to answer our first research question (cf. Section II), we derived ten requirements for ACM from current literature. Table I summarizes these requirements with the identified scientific references. In the following we will briefly describe the requirements for ACM.

The requirement for flexibility at run-time is mentioned by various different authors in literature [24], [16], [25], [26]. This flexibility is required due to the unpredictable nature of knowledge work compared to routine processes in traditional workflow management, i.e., some steps in the case might not be foreseeable at design-time. While this requirement only deals with the execution of case at one point in time, support for evolving cases and templates fosters the continuous improvement for future cases [24], [16], [27], [28]. Best practices need to be generalized to templates and case workers have to be supported through recommendations that are retrieved from similar cases that were executed previously by other users. There might be many different responsibilities for the various aspects within a case. Case workers have to be aware of these responsibilities for the various parts in a case [29], [24], [16].

The case consists of a case file that captures all relevant information required to achieve the goals. Monitoring the progress of the overall case and ongoing tasks is crucial to keep track of the case [24], [16]. During execution of a case it must be possible for new case worker to join the case [29], [24], [16]. These roles need to provide enough flexibility if tasks cannot be completed by one role. Van der Aalst et al. [14] introduce an additional skip and redo role to provide this flexibility. The skip role defined for a task enables this role to omit the task and continue with remaining tasks in the case. The redo role for a task is able to revert all changes related with this task.

ACM focuses on what can be done rather on what should be done compared to workflow management. As a result goals and objectives become very important concepts that have to be expressed in the ACM solution [29], [24], [16], [30]. They are the starting point for a case and particular steps to achieve them might be defined subsequently during execution of the case. Similar to goals and objectives an ACM solution needs to support an hierarchical structure of tasks [24], [16]. Hierarchical structures are easier to handle for case workers compared to complex process networks and they can be incrementally refined during execution of the case.

Integration of data in the case is described as a requirement by several authors [24], [29], [16], [14]. Knowledge work is very data-driven and difficult to support through business

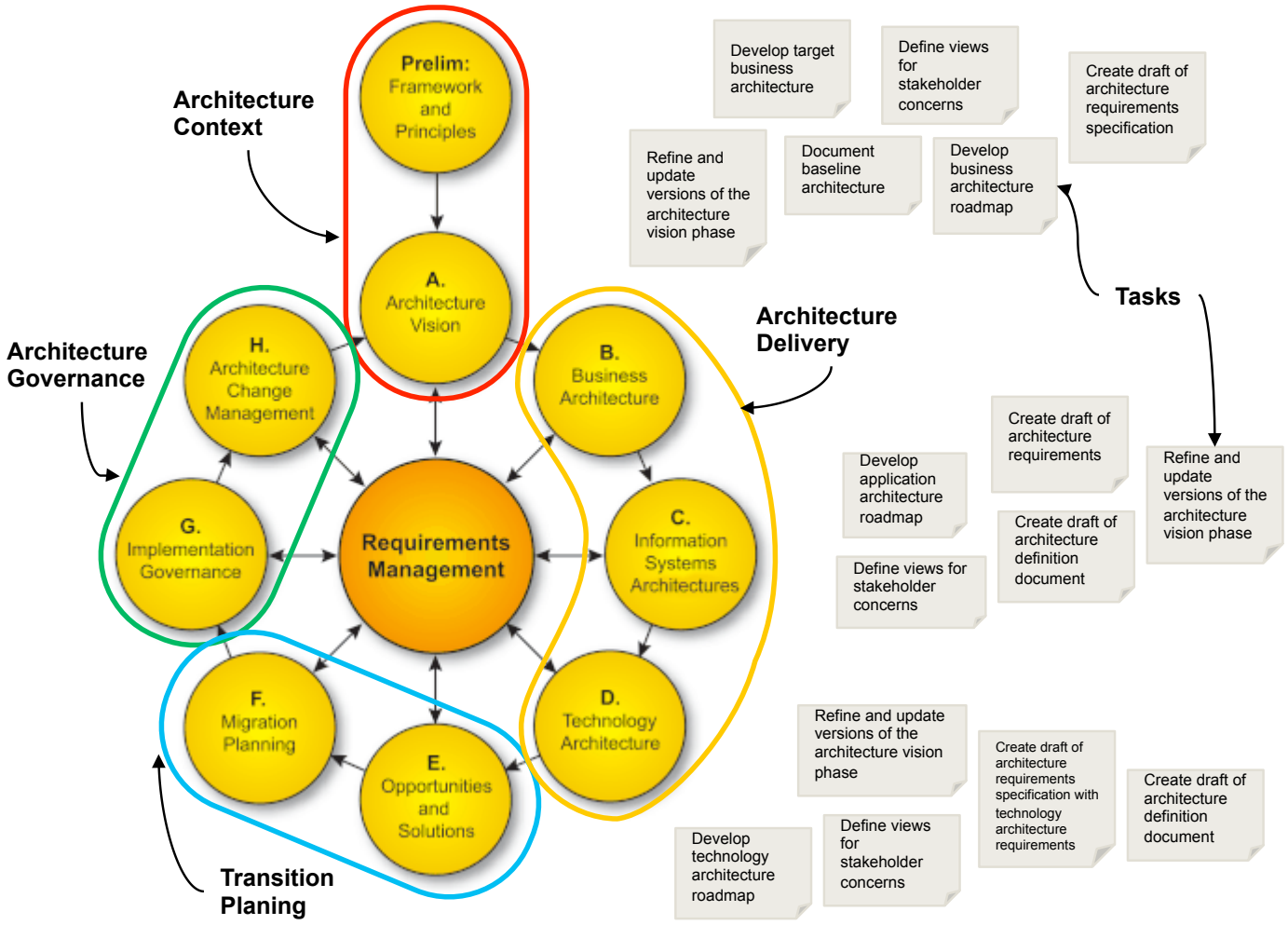

Fig. 2. Iterations in the TOGAF ADM (left) and defined tasks in the prototype (right) for the three phases of the architecture delivery phase [10]

processes that focus only on control flow. Business processes supported through workflow management solutions typically separate between data and process which is not desirable in ACM. As a result the processes have to be tightly integrated with data located in the case files. This requirement is also related to the issue known as context tunneling which is described by Van der Aalst et al. [14]. Context tunneling moves data related to the entire case to the background for the user and only provides information to complete single work-items.

ACM requires support for temporal-logical dependencies between tasks in the case [29], [16]. The resulting flexibility in the ACM system might overwhelm case workers without sufficient guidance in case no dependencies between tasks are defined. Users might have to select between too many possible tasks at one time in the system. Temporal-logical dependencies described in case templates reduce the number of open tasks if certain preconditions are not met or other tasks have to be completed before continuing with the process.

Finally, business users without any programming or modeling knowledge have to be able to understand and adapt cases on their own [29], [16]. This is necessary to empower business users to make decision by themselves and plan upcoming tasks in the system at run-time.

V. Development of a Planned State with Adaptive Case Management

In this section we will present the answer for the second research question and illustrate our prototype which implements the requirements from Table I to support EAM processes through ACM. The prototype is evaluated using an EAM process for the development of a planned state that is based on a real world example from a German insurance organization. Although this process is customized and adapted to the organizational context, it can be aligned with the description in the architecture delivery iteration in the Architecture Development Method (ADM) from TOGAF [10]. Required phases in the ADM to develop a planned state and tasks that have to be performed within these three phases are summarized in Figure 2 for our case study. While this process contains almost a complete set of tasks required, an enterprise architect might reduce the number of tasks depending on the requirement in the organization to avoid waste.

Our prototype is based on a wiki system that is extended to fulfill the ACM requirements from Table I. Every wiki page can be associated with a case type, whereas we created our own template called architecture delivery to implement the EAM process. In general, wiki pages can be assigned with any

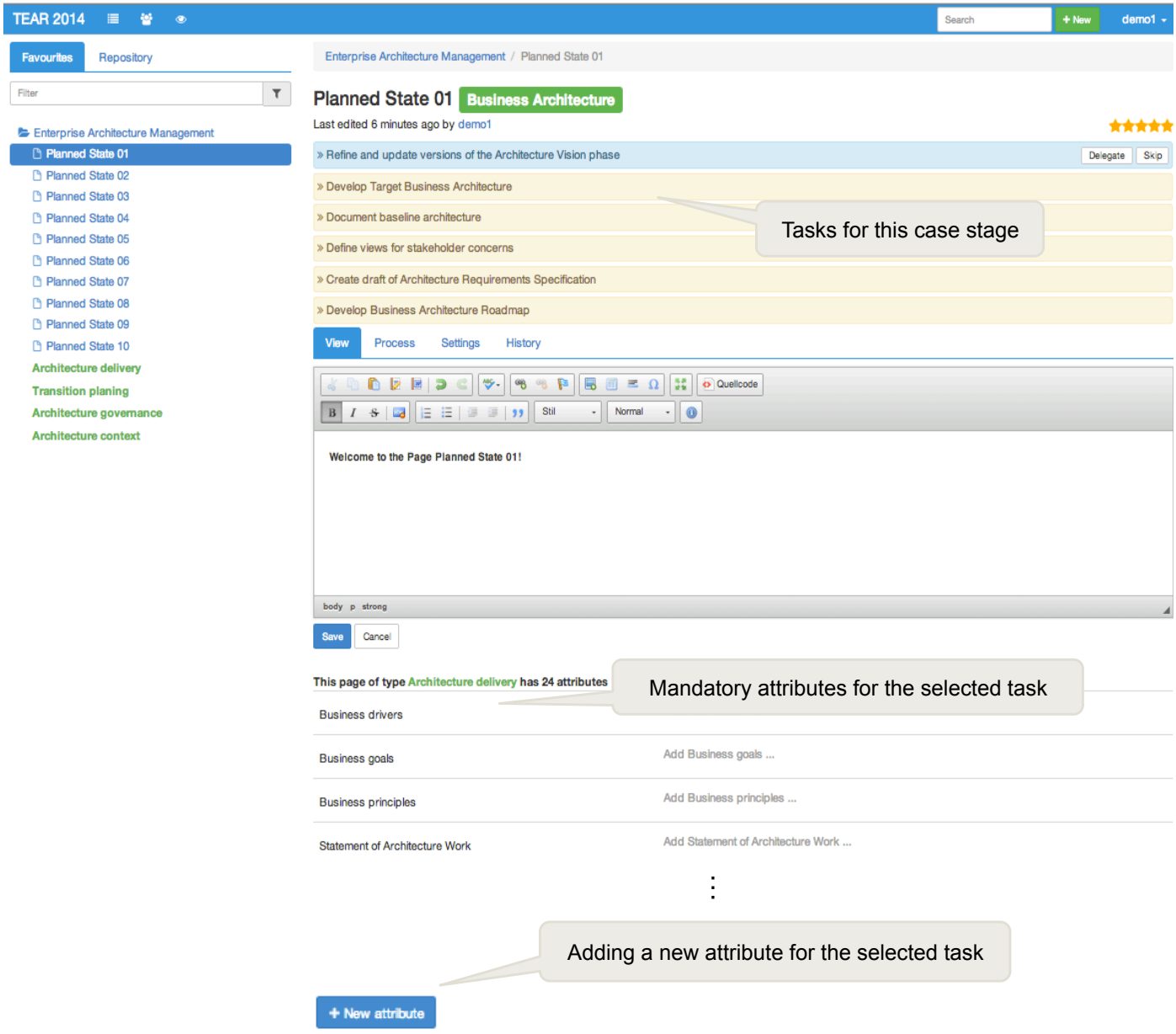

Fig. 3. Screenshot of a case in the implemented prototype for the development of a planned state with the required tasks in the current stage business architecture

type and the prototype is not limited to architecture delivery. Figure 3 shows a screenshot of a planned state called *Planned State 01* that is divided into a view tab and a process tab. The view tab contains an editor for unstructured information and structured information with attributes to capture the whole case data, while the process tab shows an interactive diagram with consecutive case stages to define the overall process.

Case data is associated with tasks so that values for all mandatory attributes need to be provided to complete a certain task. As soon as all required tasks for one case stage are completed the case stage is automatically changed to the subsequent stage. These stages represent high-level tasks that can be refined with tasks at the lower level in the hierarchy. At any time authorized users can create new mandatory attributes for a task or create completely new tasks or stages. Next to an execution role these tasks also have a redo and skip role that can be assigned to a person or group during the definition of a new task. Users can search for cases at any time through the explorer on the left-hand side of the window or the search function in the navigation bar.

Figure 3 shows a screenshot of the implemented prototype with a case that is currently in the case stage business architecture. During this stage several tasks need to be accomplished to reach the subsequent stage application architecture (refer for details of phase B to TOGAF [10]). During the *refine and update versions of the architecture vision phase* task, some attributes need to be provided which are highlighted in the attribute list after the task has been selected by the user at the top of the page. Among them are for instance business drivers, business goals, business principles as well as statement

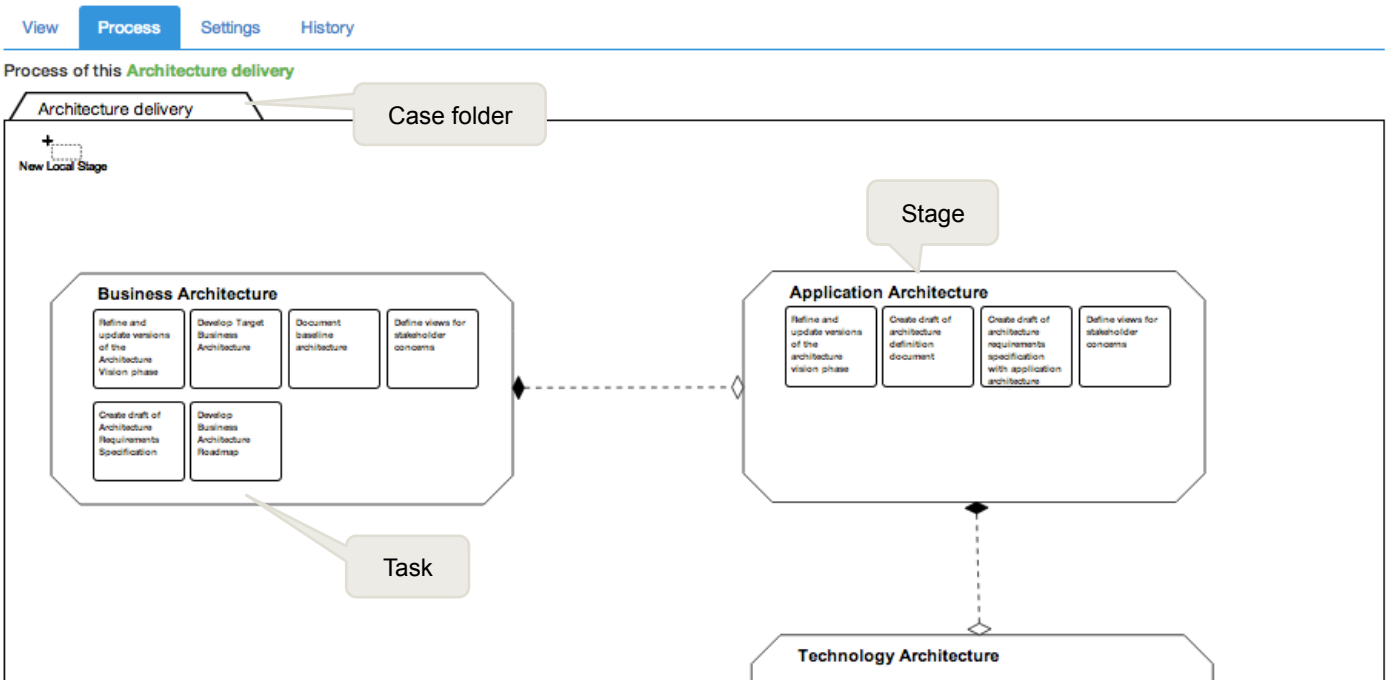

Fig. 4. Screenshot illustrating the visualization of stages and tasks for the development of a planned state in the implemented prototype

of architecture work.

We defined this task so that it can be skipped and it could be assigned to another enterprise architect in case someone else is better qualified for this task. Mandatory attributes can be directly added in the text field or a file containing this data can be uploaded, whereas we decided to use a simple string value for the sake of brevity to evaluate our implemented prototype in the case study.

Furthermore, *the target business architecture needs to be developed* in an own task with data regarding business functions, business goals, business roles, and organization structure. This data describe the future state of the business architecture based on the results of the architecture vision phase [10]. The *document baseline architecture* task models the current state of the business architecture. During *the definition of views for stakeholder concerns* the enterprise architect defines necessary views on the business architecture. The final tasks before this stage is completed are the *draft of the architecture requirements specification* and the *development of the business architecture roadmap*. In the first request formal requirements are described and the second task defines the necessary steps to achieve the target architecture. Despite this guidance it is still possible to provide data for tasks that are due in a later stage of the process if no task is selected, so that these tasks are completed automatically in forthcoming stages.

An enterprise architect might also adapt the template and add new mandatory attributes while a task is selected using the new attribute button. This might be necessary in case further information is required to complete a task or this task was not predictable during template definition and has to be refined after new insights are available. As a result the process execution planning can take place during execution of the case and evolutionary changes can be incorporated into template. Next to mandatory attributes that are the primary driver for the progress of the case, further optional attributes can be added to provide users with all required information so that context tunneling is avoided.

Case stages are defined in an interactive editor on the process tab of the wiki page. The notation of this editor is based on the Case Management Model and Notation (CMMN) published by the OMG[1]. Figure 4 shows the process for the development of planned states within the architecture delivery iteration. Every stage in this diagram is associated with several tasks that have to be completed before proceeding to a subsequent stage is possible. Next to the definition of required attributes, options for execute, delegate and skip roles for the task can be defined (cf. Figure 5). Enterprise architects can use this tab to extend the case with new stages at run-time to structure their EAM processes. Adapted case templates are persisted in a library and can be reused by other enterprise architects using the new button at the top of the page. Similar to previously described tasks in the business architecture phase subsequent case stages have associated tasks that guide enterprise architects during development of a planned state for the EA. Despite this guidance an enterprise architect retains his flexibility since he/she is empowered to adapt the case and skip/redo tasks in the template if necessary.

Regarding the requirements in Table I the prototype supports them as follows. Flexibility at run-time is possible through new attributes and tasks that can be added on the wiki pages by authorized users (R1). A dedicated role in the system is able to monitor these changes and propagate them to the template to allow continuous improvement (R2). Responsibilities in the case are visible for other users with the defined roles for every task (R3). Progress of the case

[1]http://www.omg.org/spec/CMMN/, last accessed on: 2014-06-05

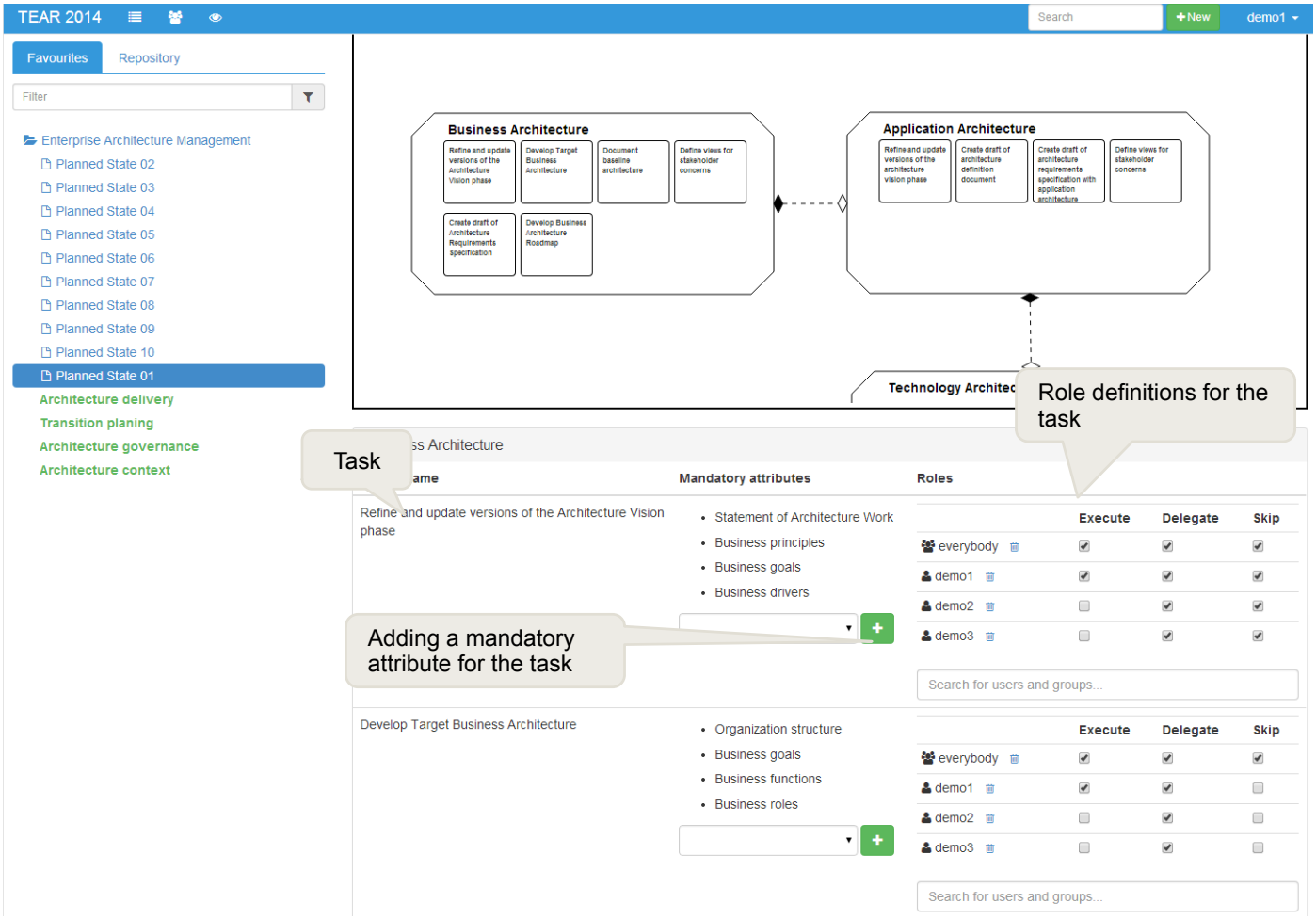

Fig. 5. Screenshot showing the adaption of the template at run-time through a new mandatory attribute that is added to a task in the implemented prototype

is visible to the users since only active tasks are shown to the user and current stages are highlighted to the user (R4). At any time tasks can be assigned to new roles and persons (R5). The definition of case objectives is not directly visible in the editor due to missing constructs in the current version of CMMN. As a workaround the case objectives can be defined with separate attributes on the page (R6). The solution support a hierarchical structure of tasks through stages (R7). Data integration is supported through attributes that are linked to tasks. Attribute values of these tasks drive the process and tasks can also be completed by directly entering data in attribute values on the wiki page if no task is selected (R8). Although logical dependencies cannot be expressed directly within the tasks, stages can be used to define dependencies by structuring tasks (R9). The prototype only uses a limited set of constructs from CMMN, so that the process is adaptable for end users (R10).

VI. Evaluation with Expert Interviews

The implemented prototype for the case study is evaluated through expert interviews in the entire EA department of a leading German insurance organization. This department consists of three enterprise architects and two EA consultants (cf. Table II). Both EA consultants are very familiar with this organization and answered questions on behalf of this insurance organization. All interviews were guided through an eight step questionnaire, whereas the participants have a very diverse experience in EA ranging from less than one year up to ten years. One of the enterprise architects recently joined the team from another department in the organization and has no fundamental experience in EA. The organization established the EA department in 2008 and already has several processes for EAM documented in textual descriptions although the maturity is still on an early state.

After a short introduction to the user interface and core concepts in the solution, all participants were asked to use the prototype to execute the previously described process. The questionnaire contains the following eight steps: 1) Role of the participant, 2) experience in EA, 3) existing documentation of EAM processes in the organization, 4) possible improvements of EAM processes, 5) suitability of the proposed solution for the development of planned states, 6) benefit of process adaptions at run-time, 7) additional application scenarios, and 8) general feedback as well as improvements of the implemented prototype. All interviews were conducted with the same initial configuration of the prototype and individually with every expert for approximately 60 minutes. In the following we will describe the answers to the questionnaire in detail.

Currently EA management processes are documented in textual descriptions in the organization and all five participants are aware of these documents. Regarding a question whether it would be beneficial to facilitate a better support of these processes, all five of them confirmed this assumption for several different reasons. One of them highlighted that current textual descriptions are useless as long as they are not actually supported by a tool. It was mentioned by one expert that EAM processes could be standardized and provide guidelines in the organization. Further advantages that are desirable according to the experts are clear role and responsibility definitions, a history of activities in the tool, a continuous improvement and optimization of EAM processes as well as the possibility to delegate tasks in the processes.

TABLE II. PARTICIPANTS IN THE EXPERT INTERVIEWS

Job Title	**# Years EA Experience**	**Industry Sector**
(Lead) Enterprise Architect	10	Insurance
EA Consultant	10	IT Consulting
Enterprise Architect	<1	Insurance
Enterprise Architect	6	Insurance
EA Consultant	<1	IT Consulting

All five participants agreed that the proposed solution would improve the development of a planned state in the organization. In one of the interviews similarity to checklists is mentioned which could be helpful to ensure that no important steps or information are omitted. This was confirmed by another expert, who mentioned that the solution would ensure completeness of the developed artifacts. Adaptions of the process at run-time are regarded as very important, whereas one expert noticed that a review process would be necessary to constantly asses changes on the template. Although not being part of the questionnaire one expert emphasized the importance of the skip and redo role to retain flexibility during the process.

According to one of the experts process templates have to be designed ahead of time, e.g. through a template designer, and instances of these templates should be adapted by enterprise architects. Other processes that were mentioned by the experts related to EAM in the organization that could be supported with this solution are compliance and regulatory audits, elicitation of architectural requirements in projects, security assessments, and collaborative documentation of the architecture with several participating information providers.

The distinction in view and process tab was considered as positive by all participants since experts expect attribute changes, e.g. adding of new attributes for a task, more frequently than process changes, e.g. new tasks and case stages. Another suggestion by two enterprise architects was to add more meta-information about tasks, so that they could be prioritized through a due date and analyzed ex post. One expert also suggested predefined attribute types on the wiki page, which we omitted for the sake of brevity in our solution. These attributes should also be searchable in the tool solution according to this expert. A future version could also provide more visualization capabilities, e.g., gantt charts, to assess the progress of the case. Furthermore an integration of the prototype with other tools has been suggested to incorporate information on the EA from other information sources.

VII. CONCLUSION

Frameworks for EAM are not rigid enough and provide no formal steps in defining, implementing, and maintaining EA in practice. As a result many organizations are struggling with missing EAM governance structures in their projects. While traditional workflow management is not suitable to support EAM processes, ACM as an emerging paradigm to support flexible and knowledge-intensive processes provides manifold new opportunities for this problem. We derived ten major requirements for ACM from an extensive literature review and implemented an ACM solution to evaluate our approach. We applied this solution in a case study for the development of planned EA states in a leading German insurance organization. The prototype to support EAM processes through ACM has been evaluated through five expert interviews within the EA department of this organization.

Our first research question can be answered with a list of ten requirements that has been derived from current ACM literature. These requirements provide the baseline to extend existing EA tools with ACM capabilities. Due to missing reference implementations that fulfill all of the requirements for ACM, we implemented a prototype to answer our second research question. Although we were able to implement the most important requirements for ACM, productive solutions might have a different user interface since ACM is still under constant development. The Object Management Group (OMG) recently published the Case Management Model and Notation (CMMN) specification that provides a first step towards a standardized notation and model for case management. Separate from the user interface it was possible to implement a TOGAF compliant EAM process from a German insurance organization in the prototype.

The third research question can be discussed based on five expert interviews with members of the EA department of the participating insurance organization. All of the experts agreed that the proposed solution would be useful to support EAM processes. In particular the ACM requirements for run-time flexibility and optional roles are crucial for EAM processes in practice. The biggest advantages of our approach could be clearly defined roles and responsibilities as well as guidelines for less experienced enterprise architects in the organization. Since many of the EAM processes are already documented in informal descriptions in the organization, our approach could incorporate this information to develop templates for other EAM processes likewise. Thus, we think that many concepts from ACM are very useful to support EAM processes in future. These concepts could be provided by existing EA tool solutions or through standalone ACM tools. Although these initial research results as well as the expert interviews are very convincing, a larger empirical basis and further case studies in organizations are necessary to validate our approach.

In future work we will integrate the solution with our previously developed method building blocks [4] and apply it to other EAM processes. Our long-term goal is to develop a template library that can be adapted to the organizational context. The interviews conducted already provide valuable

suggestions as a starting point for further application scenarios of our solution. Furthermore, feedback which is mainly related to improvements of the user interface as well as the integration of other information sources that could provide data on the EA during the interviews will be incorporated.

ACKNOWLEDGMENT

This research has been sponsored in part by the German Federal Ministry of Education and Research (BMBF) with grant number TUM: 01IS12057.

REFERENCES

[1] J. A. Espinosa, F. Armour, and W. F. Boh, "The role of group cognition in enterprise architecting," in *System Sciences (HICSS), 2011 44th Hawaii International Conference on*. IEEE, 2011, pp. 1–10.

[2] S. H. Kaisler, F. Armour, and M. Valivullah, "Enterprise architecting: Critical problems," in *System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on*. IEEE, 2005, pp. 224b–224b.

[3] P. Weill and J. W. Ross, *IT savvy: what top executives must know to go from pain to gain*. Harvard Business Press, 2009.

[4] S. Buckl, "Developing organization-specific enterprise architecture management functions using a method base," Ph.D. dissertation, Munchen, Technische Universitat Munchen, Diss., 2011, 2011.

[5] C. Moser, S. Junginger, M. Brückmann, and K.-M. Schöne, "Some process patterns for enterprise architecture management." in *Software Engineering (Workshops)*, 2009, pp. 19–30.

[6] J. A. Espinosa, F. Armour, and W. F. Boh, "Coordination in enterprise architecting: an interview study," in *System Sciences (HICSS), 2010 43rd Hawaii International Conference on*. IEEE, 2010, pp. 1–10.

[7] S. Roth, M. Hauder, M. Farwick, R. Breu, and F. Matthes, "Enterprise architecture documentation: Current practices and future directions." in *Wirtschaftsinformatik*, 2013, p. 58.

[8] C. Lucke, S. Krell, and U. Lechner, "Critical issues in enterprise architecting–a literature review," in *Americas Conference on Information Systems*, 2010.

[9] C. Lucke, M. Bürger, T. Diefenbach, J. Freter, and U. Lechner, "Categories of enterprise architecting issues–an empirical investigation based on expert interviews," in *Multi-Konferenz Wirtschaftsinformatik 2012-Tagungsband der MKWI 2012*. Braunschweig: GITO Verlag, 2012, pp. 999–1010.

[10] The Open Group, "TOGAF 9 - The Open Group Architecture Framework Version 9," p. 744, 2009.

[11] S. Taylor, S. Lacy, and I. MacFarlane, "ITIL: Service transition," *Published by TSO (The Stationery Office) for the Office of Government Commerce under licence from the Controller of Her Majestys Stationery Office*, 2007.

[12] D. M. Strong and S. M. Miller, "Exceptions and exception handling in computerized information processes," *ACM Transactions on Information Systems (TOIS)*, vol. 13, no. 2, pp. 206–233, 1995.

[13] W. M. van der Aalst, "The application of petri nets to workflow management," *Journal of circuits, systems, and computers*, vol. 8, no. 01, pp. 21–66, 1998.

[14] W. Van der Aalst, M. Stoffele, and J. Wamelink, "Case handling in construction," *Automation in Construction*, vol. 12, no. 3, pp. 303–320, 2003.

[15] A. P. Sheth, W. Van Der Aalst, and I. B. Arpinar, "Processes driving the networked economy," *IEEE concurrency*, vol. 7, no. 3, pp. 18–31, 1999.

[16] K. D. Swenson *et al.*, "Mastering the unpredictable," *How Adaptive Case Management Will Revolutionize The Way That Knowledge Workers Get Things Done. Meghan-Kiffer Press, Tampa*, 2010.

[17] T. H. Davenport, *Thinking for a living: how to get better performances and results from knowledge workers*. Harvard Business Press, 2005.

[18] M. Hauder, S. Pigat, and F. Matthes, "Research challenges in adaptive case management: A literature review," in *3rd International Workshop on Adaptive Case Management and other non-workflow approaches to BPM (AdaptiveCM)*, 2014.

[19] R. H. von Alan, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS quarterly*, vol. 28, no. 1, pp. 75–105, 2004.

[20] J. Webster and R. T. Watson, "Analyzing the past to prepare," *MIS quarterly*, vol. 26, no. 2, pp. 13–23, 2002.

[21] S. Buckl, A. Ernst, J. Lankes, and F. Matthes, "Enterprise architecture management pattern catalog," *Technical Report*, 2008.

[22] M. Farwick, B. Agreiter, R. Breu, S. Ryll, K. Voges, and I. Hanschke, "Automation processes for enterprise architecture management," in *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2011 15th IEEE International*. IEEE, 2011, pp. 340–349.

[23] S. Roth, M. Hauder, F. Michel, D. Münch, and F. Matthes, "Facilitating conflict resolution of models for automated enterprise architecture documentation," in *Americas Conference on Information Systems (AMCIS)*, 2013.

[24] C. Herrmann and M. Kurz, "Adaptive case management: Supporting knowledge intensive processes with it systems," in *S-BPM ONE-Learning by Doing-Doing by Learning*. Springer, 2011, pp. 80–97.

[25] H. R. Motahari-Nezhad, C. Bartolini, S. Graupner, and S. Spence, "Adaptive case management in the social enterprise," in *Service-Oriented Computing*. Springer, 2012, pp. 550–557.

[26] C. W. Guenther, M. Reichert, and W. M. van der Aalst, "Supporting flexible processes with adaptive workflow and case handling," in *Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2008. WETICE'08. IEEE 17th*. IEEE, 2008, pp. 229–234.

[27] H. R. Motahari-Nezhad and C. Bartolini, "Next best step and expert recommendation for collaborative processes in it service management," in *Business Process Management*. Springer, 2011, pp. 50–61.

[28] H. Schonenberg, B. Weber, B. van Dongen, and W. van der Aalst, "Supporting flexible processes through recommendations based on history," in *Business Process Management*. Springer, 2008, pp. 51–66.

[29] M. Kurz, "Taming diversity: A distributed acm-based approach for cross-enterprise knowledge work," in *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on*, vol. 3. IEEE, 2013, pp. 87–91.

[30] T. T. K. Tran, M. J. Pucher, J. Mendling, and C. Ruhsam, "Setup and maintenance factors of acm systems," in *On the Move to Meaningful Internet Systems: OTM 2013 Workshops*. Springer, 2013, pp. 172–177.

[31] M. Hauder, C. Schulz, S. Roth, and F. Matthes, "Organizational factors influencing enterprise architecture management challenges," in *21st European Conference on Information Systems (ECIS)*, 2013.

[32] J. A. Espinosa, F. Armour, W. F. Boh, and M. A. Clark, "A self-fueling coordination model for enterprise architecting effectiveness," in *System Science (HICSS), 2012 45th Hawaii International Conference on*. IEEE, 2012, pp. 4240–4249.

[33] ——, "Team knowledge in enterprise architecting," in *System Sciences (HICSS), 2013 46th Hawaii International Conference on*. IEEE, 2013, pp. 3910–3919.

[34] J. Koehler, J. Hofstetter, and R. Woodtly, "Capabilities and levels of maturity in it-based case management," in *Business Process Management*. Springer, 2012, pp. 49–64.

[35] W. M. Van der Aalst, M. Weske, and D. Grünbauer, "Case handling: a new paradigm for business process support," *Data & Knowledge Engineering*, vol. 53, no. 2, pp. 129–162, 2005.

[36] K. D. Swenson, "Designing for an innovative learning organization," in *Enterprise Distributed Object Computing Conference (EDOC), 2013 17th IEEE International*. IEEE, 2013, pp. 209–213.

[37] I. Bider, A. Jalali, and J. Ohlsson, "Adaptive case management as a process of construction of and movement in a state space," in *On the Move to Meaningful Internet Systems: OTM 2013 Workshops*. Springer, 2013, pp. 155–165.

[38] H. R. Motahari-Nezhad and K. D. Swenson, "Adaptive case management: Overview and research challenges," in *Business Informatics (CBI), 2013 IEEE 15th Conference on*. IEEE, 2013, pp. 264–269.