

# Enabling End-User Driven Business Process Composition through Programming by Example in a Collaborative Task Management System

Todor Stoitsev<sup>1</sup>, Stefan Scheidl<sup>1</sup>, Felix Flentge<sup>2</sup> and Max Mühlhäuser<sup>2</sup>,  
<sup>1</sup>SAP Research, SAP AG, <sup>2</sup>Telecooperation Group, Darmstadt University of Technology  
 {todor.stoitsev, stefan.scheidl}@sap.com, felix@tk.informatik.tu-darmstadt.de,  
 max@informatik.tu-darmstadt.de

## Abstract

*Letting end users tailor business processes can result in business process management support, which is better turned to users' needs and organizational changes. However, such tailoring requires not only the users' domain expertise but also advanced skills in computer use, which business users mostly lack. The paper presents the design of the Collaborative Task Manager (CTM) prototype which overcomes this limitation and enables end users to become informed participants in business process composition. CTM uses enterprise-wide "programming by example" by exposing common functionalities for personal task management to the end users and tracking their activities to generate end-to-end process execution examples on a central instance. These can be adapted and reused for ad-hoc process support or exported to formal process models, which enables tailoring as collaboration between business users, end-user tailors and developers. The paper finally reports on trial usage of the tool at a partner company.*

## 1. Introduction

Enterprises are constantly struggling to optimize their business processes in order to succeed in the fast evolving global market. Business users are often the only experts understanding the matter and complexity of enterprise processes. Therefore, the need to involve them in process modeling is largely perceived in the context of Business Process Management (BPM) solutions [8]. This calls for bridging the business and technology perspectives into common understanding of processes. As a result, standardized graphical notations such as e.g. the Business Process Modeling Notation [18] have emerged. Visual process modeling is offered in enhanced solutions by leading software vendors like e.g. IBM, TIBCO, Appian and others. However, achieving process support which is better turned to

users' needs and organizational changes by "letting end-users do the tailoring" demands "both domain expertise and advanced skills in computer use" [17]. Upfront process modeling hence remains inaccessible for business users, who have good domain knowledge but limited technical skills. Such modeling can furthermore result in overhead for end users as it can be hardly considered as part of their daily activities. Studies on ad-hoc process support consider this limitation and suggest "the existence of a separate organizational unit for process modeling" [11], yet confirming the disruption between business users and business technology staff, i.e. process designers and developers. Process mining approaches involve end users implicitly in process modeling by generating workflows from logged data on collaboration or events in formal systems [1]. However, this does not allow end users to proactively tailor the emerging processes at use time. The need for user-centric approaches arises, which can enable "informed participation" of end users in business process composition by fostering "social creativity" [7] and allowing domain experts to proactively drive process optimization in enterprises.

This paper presents the Collaborative Task Manager (CTM) prototype and reports on trials of its use. CTM enables end-user driven process composition through "programming by example" [14]. Through this End-User Development (EUD) [15] technique unobtrusive process support is achieved by embedding the process definition in the existing end users' working environment and inferring process models from the captured, executed activities. The major motivation behind the tool is to "render" appropriation of process models to end users and to "exploit the potential of opportunity-based and emergent changes" from the introduction of groupware in enterprises [22].

In section 2 we present the major design goals behind our approach for end-user driven process composition. The approach is described in section 3. Section 4 describes the basic components of the CTM

prototype. In section 5 we report on CTM trial usage at a partner company. Section 6 gives a conclusion and future research directions.

## 2. Design Goals

The presented study builds up on the state of the art research in the areas of task management, ad-hoc workflows, Computer Supported Cooperative Work (CSCW) and EUD. It is based on intra-organizational knowledge sources accumulating customer requirements as well as on dedicated site visits and interviews at three different companies from various industries: textile (120 employees), software (ca. 500 employees), automotive (ca. 150 employees). For enabling end-user driven business process composition we have defined the following major design goals:

**Gentle slope of complexity [16]:** Process tailoring by end users should be ensured through a “gentle slope of complexity”. A solution should be able to create an environment where different stakeholders with different business and IT background can benefit from each-others’ knowledge and can collaboratively evolve a tailoring culture for enterprise processes.

**Seeding, evolutionary growth and reseeded (SER) [7]:** As end users have different level of technical expertise and different attitude towards maintaining process data, emergent process models may have different level of specificity. Therefore SER of such models should be enabled for their iterative exchange, reuse and complementation towards comprehensive business process definitions.

**Support tailoring as collaboration [17]:** Rigidly recurring processes are suitable for formalization and automation through workflow engines. Process formalization should be enabled through “a shared context between developers and end-user tailors” [17] which is able to bridge the business and technology perspectives on processes and to increase “business collaboration in process modeling” [8].

## 3. Approach

The increase of human-centric business processes, which are executed in distributed teams in a rather informal manner led to extensive research on ad-hoc workflows and agile BPM [11, 13, 19]. Despite the variety of tools for flexible process support, email and to-do lists, provided in standard office applications, such as e.g. an email client, dominate end users’ practices for managing ad-hoc work [2, 3]. Deficiencies in software support for knowledge work are addressed in the Chandler project [6], which introduces a comprehensive solution for personal

information management featuring among others collaborative content sharing. Although we find this approach adequate for addressing ad-hoc work, it focuses on supporting individual actions in a group context rather than on involving end users in composition and adaptation of enterprise process models as discussed in the presented paper.

It becomes apparent that end users have different strategies for organizing their work and gain efficiency through the possibility to manage their individual tasks. Therefore, we suggest that end users can be involved in business process composition by providing added value on personal task management and leveraging their experience with standard tools for task management and collaboration towards definition of process models. In this respect a “gentle slope of complexity” [16] for process tailoring can be provided by closely integrating the process definition in the actual user working environment and unfolding emergent processes behind the scenes in an unobtrusive, implicit manner. For achieving this we propose enabling of enterprise-wide, collaborative “programming by example” [14] by implicitly reconciling data on personal task management of multiple process participants to end-to-end process execution examples. Concretely, the presented approach enables end users to create hierarchical to-do lists by breaking down tasks into sub tasks. Tasks can be delegated over email, whereby the recipients can further break down the received tasks and delegate resulting (sub)tasks to other end users. Changes of individual tasks in the personal end users’ to-do lists are tracked over web services on a central server instance and task data is replicated in a tracking repository in a database. Tracking of email exchange for task delegation integrates the personal to-do lists of different process participants to overall Task Delegation Graphs (TDG) [20] on the server.

TDGs represent weakly-structured process models, which are captured as actual process execution examples and contain all task data including artifacts (attachments) and stakeholders’ information. A generic view of a task delegation graph is shown in Figure 1.

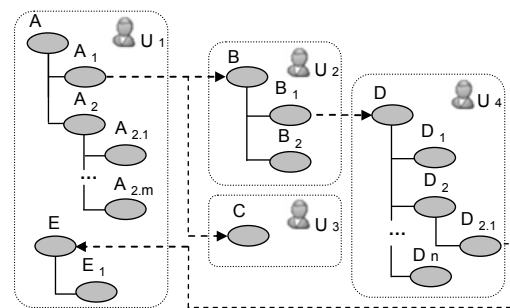


Figure 1. Task Delegation Graph (TDG)

The dotted-line areas represent the personal workspaces with the individual to-do lists of users  $U_1 - U_4$ . The ovals represent user tasks, where e.g. task A has sub tasks  $A_1$  and  $A_2$ , task  $A_2$  has sub tasks  $A_{2,1}$  to  $A_{2,m}$  etc. The dotted line arrows represent task delegations, e.g. user  $U_1$  has delegated task  $A_1$  to users  $U_2$  and  $U_3$ . Tasks B and C are thereby the tasks resulting from the delegation, which are respectively contained in the personal workspaces of users  $U_2$  and  $U_3$ . TDGs deliver added value to the users by providing a workflow-like overview of evolving, collaborative tasks, resulting in transparency beyond the capabilities of common email and to-do lists. Unlike approaches for modeling of collaborative processes [10, 21], TDGs do not require any initial process model or preliminary knowledge of a process. They are themselves emerging process models which unfold during end users' task management activities.

SER of weakly-structured process models is enabled through extraction, adaptation and reuse of Task Patterns (TP) [19, 20]. In the following a TP is considered as a reusable task structure, comprising one task with its sub task hierarchy and the complete context information of the contained tasks like e.g. description, used resources, involved persons etc. TPs can be enacted to create a new process instance and execute it along the provided example flow. This flow can be altered by changing suggested task delegations or reusing referenced TP hierarchies. TP adaptation and reuse can result in evolution and complementation of captured processes. This evolution is traced through task instance-based ancestor/descendant relationships [20]. TPs deliver added value to the end users by allowing them to reuse previous knowledge of tasks without the need to manually assemble all task-relevant information. TPs further enable end users to establish best-practices and the ancestor/descendant relationships enable tracing of best-practice deviations in different application cases.

For supporting tailoring as collaboration, the presented approach enables transformation of user-defined TDGs to formal workflows, based on the task change and evolution history. The resulting workflows are hence implicitly modeled by all process participants and can be extended by process designers or developers in a shared context, containing ad-hoc and formal process model representations.

In the next section we describe how this approach is supported in the CTM prototype. An evaluation of the approach based on a CTM trial is presented later on.

## 4. Collaborative Task Manager (CTM)

The CTM is a task management tool with extensive support for end-user driven composition of business process models. CTM addresses two main issues: (i) light-weight composition of weakly-structured process models for ad-hoc process support; (ii) formalization of weakly-structured process models for automation of rigidly recurring processes through workflow engines.

### 4.1. Programming by example of weakly-structured process models

In order to ensure integrated support in a common user working environment, the CTM front-end is delivered as a Microsoft Outlook (OL) add-in. CTM extends OL mail and task items and enables "programming by example" by capturing OL events and using web services to replicate task data in a tracking repository on the CTM server. The CTM to-do list is shown in Figure 2. Extensions to the standard OL tasks enable end users to create hierarchical to-do lists. When the end user is creating or editing a CTM task they work with the familiar OL task fields. Files can be added to CTM tasks as common OL attachments. A CTM task is delegated through a "Request" email message, which recipients can "Accept", "Decline" (similarly to meeting requests in OL) or "Negotiate". The latter action allows iterative clarifications on tasks. When a request is accepted, and later on completed by a recipient, they issue a "Declare Complete" message, to which the requester can respond with "Approve Completion" or "Decline Completion". The actual discourse takes place in the email text, independently from the given message type. This allows open-ended collaboration and prevents

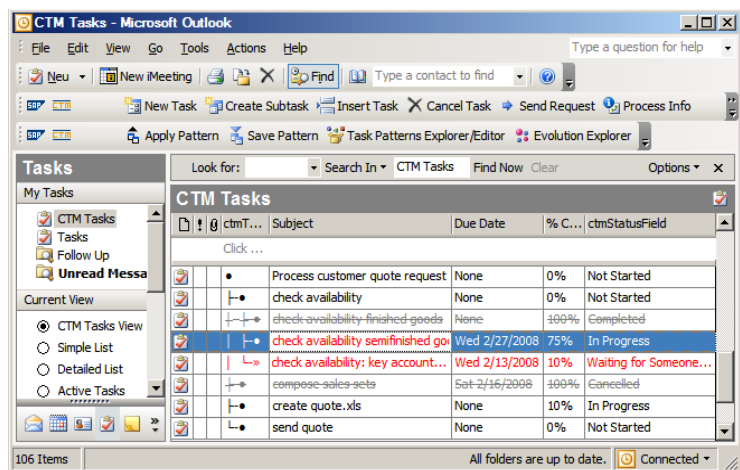


Figure 2. CTM to-do list

from submitting user behavior to strict speech-act rules, which is a known limitation in speech-acts adoption [5]. All task-related email exchange is associated to a dialog and stored on the server. Dialogs can be inspected through a process tree web overview, where the nodes provide links, opening task and email information including text and attachments.

CTM tracks the task-related email exchange and integrates the to-do lists of different process participants to a TDG [20] as shown in Figure 3 (user data is blacked-out for privacy reasons in all figures in the paper). TDGs provide a workflow-like overview of collaborative activities, aiming to facilitate “the creation of a shared understanding leading to new insights, new ideas, and new artifacts as a result of collaboration” [7]. While known process mining approaches [1] generally aim at the generation of process models from underlying data *after* a process has finished, TDGs enable end users to evaluate their current work situation *while* a process is executed. In that sense TDGs enable end users to become informed participants in the composition of emerging processes and to influence these processes according to their problem solving strategies. For example, in a TDG users can view status of related tasks, evaluate work distribution and identify potential bottlenecks. Currently, due date, task processing status and percent complete indications are provided. Description links in task nodes open dialogs with full task description. Attachments, added in OL tasks, are replicated in a central artifacts repository in a database on the CTM server and are accessible in the task nodes.

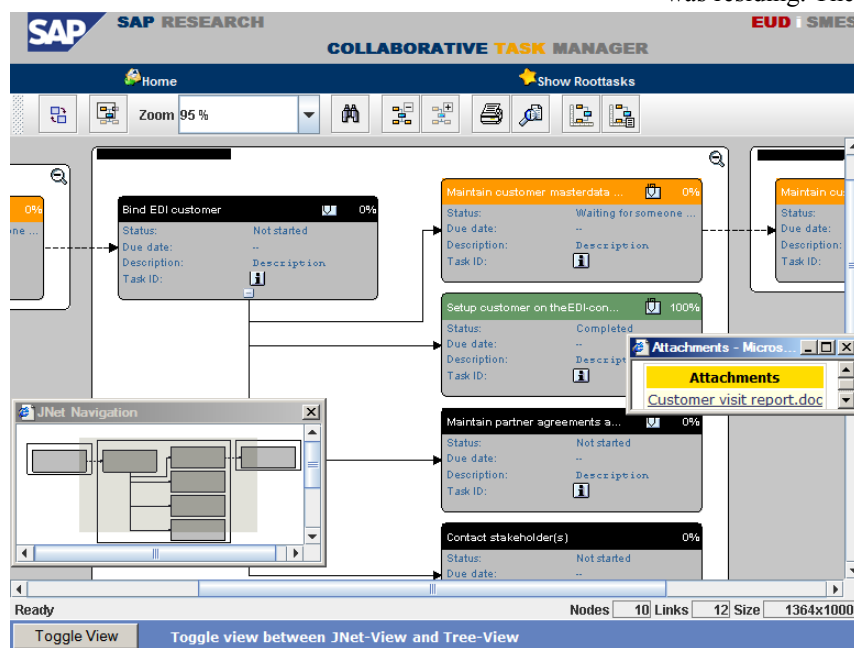


Figure 3. CTM Task Delegation Graph (TDG) overview

## 4.2. SER of weakly-structured process models

CTM enables export of a local task from the personal to-do list to a single TP, and export of a complete TDG from the server to multiple TPs which represent the personal task hierarchies of different users and are interlinked through suggestions according to the delegation flow. TP extraction is currently done manually, i.e. whenever a user decides that they could reuse a certain TDG or task (sub)hierarchy. No automated detection of TPs in the tracking repository e.g. through machine learning approaches is currently provided. TPs can be saved in local or remote TP repositories. A local TP repository is a XML document [20] whereas remote TP repositories reside in a database on the CTM server. TPs are managed in the Task Pattern Explorer shown in Figure 4, which provides rich editing and search functionality on task trees and on data in context fields on the right hand side, and enables also task search and extraction of TPs from the tracking repository. Editing of process execution examples (interlinked TPs) in this component is realized through direct manipulation of the task fields, whereby “the user is not required to interact in the interface domain of computational abstraction, but works directly with the data that interests him or her” [15]. The “Name”, “Description” and “Suggested Execution Time” fields hold simple task information in text format and are self-explanatory. The “Owner” field recommends expertise, i.e. when a task is extracted from an executed process the owner is the person, in whose to-do list the task was residing. The field “Suggested Delegates” contains

information about the persons, who have the expertise to execute a given task, i.e. upon task extraction from a collaborative process the task recipients are set in this field. The “Suggested Pattern” field holds a reference to a TP which should be used for the further processing of a task. In case of TDG extraction, such references in requester tasks point at recipient tasks, used for the further task processing. The recipient tasks are themselves extracted as separate TPs. Task attachments are represented as “Artifacts”. Adding of custom artifacts in the TP Explorer replicates these to the artifacts repository.

TPs can be reused through an “Apply Pattern” operation in the

to-do list. It opens the TP Explorer, where the user can search for TPs in TP repositories and for reusable tasks in the tracking repository. Applying a TP reactivates the process example by generating the task hierarchy and filling the pre-modeled content information in the to-do list. Available delegates are automatically suggested when delegation is initiated. The anticipated example flow can be changed by entering different recipients. Suggested TP references are also included in the resulting tasks and can be used by the person, activating the TP, to accomplish the task themselves without further delegations. If a delegation is issued, the recipient task receives a reference to the suggested TP so that the recipient(s) can adapt and reuse it.

SER of TP through their iterative adaptation and reuse can result in refinement of captured process examples. CTM enables tracing of evolving TPs through task instance-based ancestor/descendant relationships [20]. Such are set iteratively between the tasks in the originating hierarchy and the corresponding tasks in the resulting hierarchy always, when a task hierarchy is reused, e.g. on copy/paste in the TP Explorer or save/apply pattern operations. Through navigating in evolution hierarchies, the user can view the TDG and dialog flow of tracked

ancestors/descendants. Task evolution can be viewed in an Evolution Explorer in the CTM OL add-in.

### 4.3. From email and to-do to formal workflows

In CTM, rigidly recurring process fragments can be detected based on the captured TP evolution resulting from SER. For process formalization CTM uses the JBoss Business Process Management (jBPM) solution [12]. jBPM workflows are modeled in a graph-oriented, visual language – the jBPM Process Definition Language (JPDL). The workflows can be deployed and executed on a JBoss server, where these are accessed over a web front-end. jBPM process modeling is originally performed in an Integrated Development Environment (IDE) – the JPDL designer, provided as an Eclipse plug-in. However, CTM enables transformation of user-defined TDGs to formal JPDL workflows in the CTM OL add-in, by bridging ad-hoc and formal process representations.

**4.3.1. Ad-hoc to formal workflow conversion.** The conversion from TDG to jBPM workflows is based on the task change and evolution history. Task changes altering task status, percent complete or task artifacts, are considered as Task Processing Changes (TPC), denoting that the user is acting on a given task. Parallel flows in a formal workflow are created for tasks, which have received TPCs in parallel. For example if task  $T_1$  has received a first TPC in given time  $t_1$  and a further TPC at given time  $t_n$ , each task and each delegated task on the same tree level under the parent task of  $T_1$  is considered parallel to  $T_1$  if it has received a TPC at a given time  $t_i$  such that  $t_1 \leq t_i \leq t_n$ . The period  $t_1$  to  $t_n$  is referred to as the *range* of task  $T_1$ .

Task ranges are a simplified way to suggest sequencing. This is due to the fact that ad-hoc tasks can be executed without meeting any pre- or post-conditions. The resulting sequencing is hence based on suggestions and during model conversion, the user can view the task change and evolution history and estimate whether the suggested flow is correct. SER can improve the accuracy of the generated workflows, i.e. if a given TP is reused multiple times and given task ranges overlap in multiple executions, the tasks can be considered parallel with greater certainty.

The hierarchical order of tasks in TDGs is considered during model transformation by allowing end users to select different export modes for a task with subtasks: (i) as sub process, containing the sub tasks – this mode is pre-selected if a parent task contains data like e.g. attachments, detailed description etc., which is transferred to one or more of the sub tasks; (ii) as atomic task before the sub tasks' sequence – this mode is pre-selected if the parent task data is not

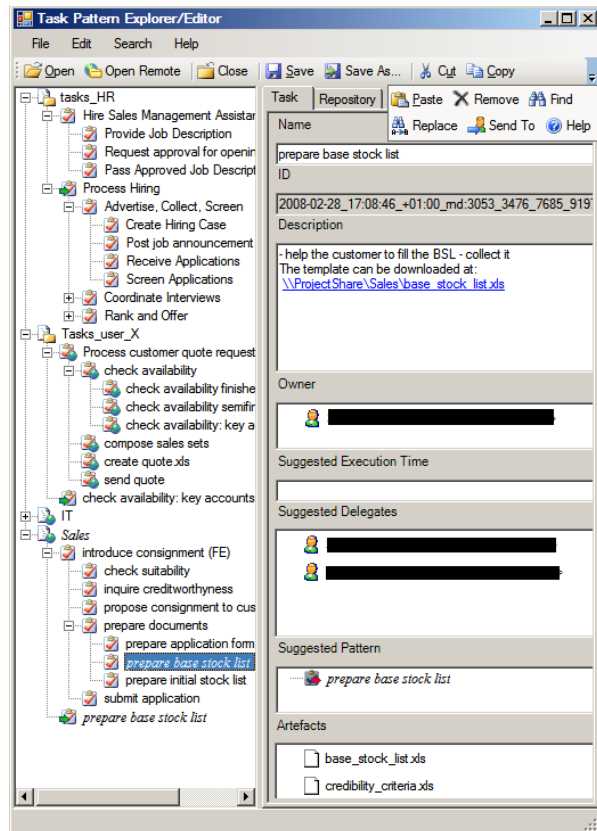


Figure 4. Task Pattern Explorer/Editor

transferred to any of the child tasks; (iii) as group element (jBPM super state), embracing the sub tasks as logical association – this mode is pre-selected if the parent task contains only a subject.

Delegations in a TDG are considered as follows: (i) if a delegated task has no sub tasks on requester side it can be omitted, or preserved along with the recipient task in the resulting model. Omission is pre-selected as it results in model simplification when the task was fully processed by the recipient. (ii) if a task was delegated, but the requester has added subtasks to it in their to-do list, requester and recipient tasks can be preserved as independent process nodes, or they can be merged by selecting one of them as the preferred, resulting jBPM task. In the latter case requester and recipient sub tasks are handled as children of the same parent and checked for overlapping ranges.

We should stress here that the transformation of weakly-structured TDGs to formal process models needs to be done by end-user tailors with higher IT skills, capable of dealing with workflow diagrams. The benefit from the introduced approach is that the process modeler is able to work with data, which was implicitly defined by the business users during their daily activities. The tracked task data for TDG generation is available during formal process modeling, even if the participants in the initial ad-hoc processes did not interact on TDG level and did not extract or reuse TPs. The resulting formal models hence closely relate to the real-life context and embody process knowledge which is not explicitly documented or of which business users may not be explicitly aware.

**4.3.2. CTM process definition environment.** The CTM process definition environment is shown on Figure 5. The upper left corner contains a view,

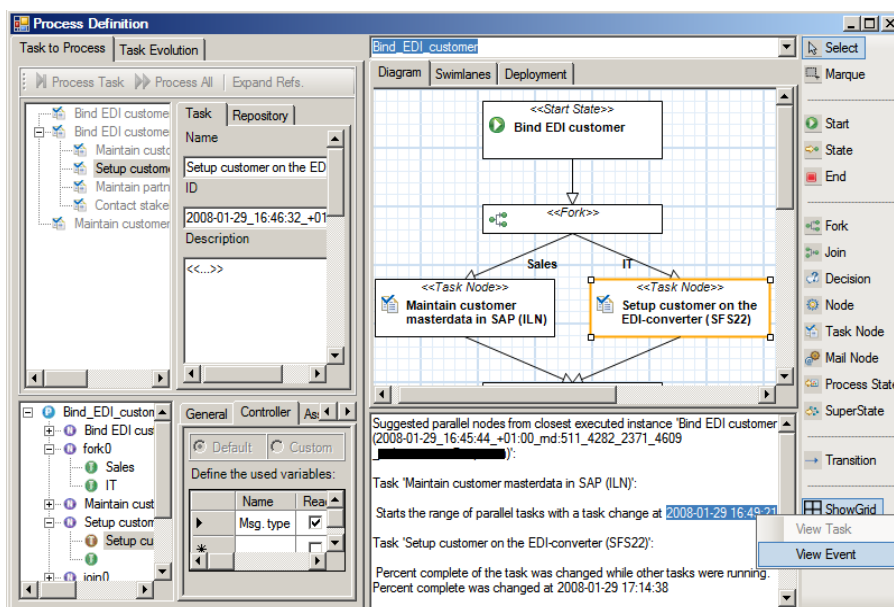


Figure 5. CTM process definition environment

displaying the task hierarchy in the same manner as the TP Explorer. Processed tasks receive the jBPM task icon and a gray foreground. Tasks can be processed along the hierarchy through the “Process Task” (stepwise) and “Process All” (iteration) buttons. The upper view in the center contains the generated JPDL graph. A toolbox on the right hand side allows advanced users to select appropriate tools and edit the models. The tree in the lower left part of Figure 5 contains the generated jBPM process entities (nodes and transitions). A tab control for setting their properties is provided on the right. In the “Controller” tab, users can set parameters for task nodes, used during workflow execution. An “Assignment” tab allows setting of jBPM task assignments such as e.g. swimlanes. The latter are automatically generated based on task owner information, where each swimlane is defined through an expression “user(email\_address)” (swimlanes can be edited in a dedicated “Swimlanes” tab - see upper central part of Figure 5). The task properties’ tab control further contains a “Form” tab where the xhtml code of a jBPM task’s web form is provided. CTM automatically generates this code by embedding also links to the original TDG and used artifacts (available in the artifacts repository). Advanced users can edit the code to enhance the runtime task views.

A textual explanation of the relevant transformations for each task is given in the lower central part of Figure 5. It describes the overlapping ranges and refers to the appropriate change events. Task change and evolution history is provided in the “Task Evolution” tab, shown on Figure 6. The task evolution tree in the upper left part contains on root level the task ancestors and their references resulting from delegations, followed by the currently processed

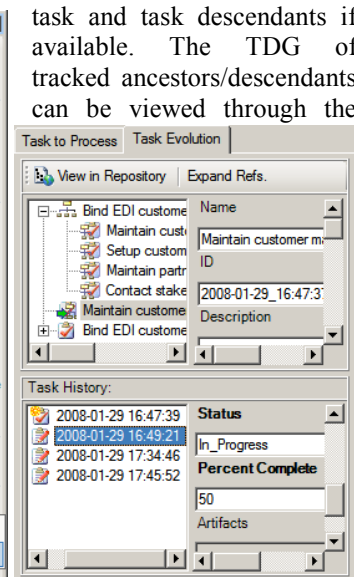


Figure 6. Task change and evolution history

“View in Repository” button. Task change history is displayed in the lower tree. Changes are given with their time of occurrence. The changed fields e.g. status, percent etc. are shown on the right.

Generated jBPM workflows can be saved as process files or deployed as fully functional workflows on the jBPM server. Both functionalities are provided in the “Deployment” tab in the upper central part of Figure 5. Process files can be copied in the jBPM IDE, where the workflows can be extended by developers.

## 5. Case study

### 5.1. Setting and extent of use

The CTM case study was conducted at the textile production company (cf. 2) and involved 6 users, selected for having related tasks: COA - Chief Officer Assistant; CSO - Chief Sales Officer; SL1 & SL2 - Sales Employees; ITL - IT Department Lead; ITE - IT Employee. ITL and ITE were dealing with computers at an advanced level but did not have any process modeling or programming skills and hence matched the type of end-user tailors. The other participants were typical business users. All users used OL as email client. CSO, SL1 and ITL also used OL tasks before the CTM installation. The trial was initiated with a workshop in which we gave a 1 hour presentation on the tool, followed by 30 minutes individual training of each user on the basic functionalities. Detailed user guides were provided to all participants. The jBPM export functionality was not included in the installations and manuals to preserve the focus on informal process support, addressing equally IT and business users. The trial lasted 8 weeks. Daily backups of the CTM database were scheduled and collected for evaluation each week. The evaluation concluded with a short video recording and transcription of the tool use, followed by a structured debriefing interview, in which we asked each participant to assess the basic features and rate to what extent CTM improved their ability to manage work using Likert scales and freeform explanations.

In a second iteration with ITL and ITE we additionally performed formal modeling exercises for a recurring process, which we detected in the database backups. We first gave ITL and ITE a 40 minutes tutorial on the jBPM process modeling (in Eclipse), and a 30 minutes tutorial to the CTM workflow transformation environment. Then we asked ITL and ITE to model the process in each of the two environments, using think-aloud and contextual inquiry [4] methods to track their strategies and intents. The exercises were videotaped for analysis. As the focus

was on process modeling as result from systematic interactions in CTM rather than on modeling with the JPDL visual notation, cognitive dimensions [9] of JPDL modeling were not considered.

### 5.2. Findings – supporting ad-hoc work

An excerpt from the case study metrics is given in Table 1. All participants reported that creating CTM tasks did not impede their work. We observed that users generally manage percent complete and status information, however not as precise estimation of work completion, but moreover “to indicate that I’m working on it [a task] and avoid getting calls and emails from the others [sales], asking about status” (ITE). We further encountered that users maintained attachments in CTM tasks, which was considered “faster than email, as I only needed to attach the updated document and the others can pull the latest version [from the TDG]” (SL1). Users further considered that having “a kind of checklist [TP] with all things I need to do and the documents I need is very useful ... especially if she [CSO] is not in the office [vacation]” (SL2). The overall attitude was that global TP should be delivered by a (senior) domain expert, who can handle also the responsibility for providing them. Due to the restricted CTM usage, it was not possible to distribute TPs throughout the company, which prevented from developing a global strategy for TP management e.g. as alternative to text-based documents. Eventually, 2 remote TP were finally available (from ITL & CSO) whereas SL2 and ITE had developed local TPs.

**Table 1. Excerpt of case study metrics**

Created root tasks (ad-hoc processes)	8
Created tasks (overall)	46
Delegations	14
Unique attachments added	25
Attachment changes (diff. checksum, same name)	12
Percent complete changes	45
Task changes overall (only edit, no create/delete)	68
Created remote TP	2
Created local TP (files on user PCs)	4
Reused remote TP	1
Reused local TP	2

### 5.3. Findings – process formalization

The binding of new customers for Electronic Data Interchange (EDI) occurred 3 times in the collected database backups. A screenshot of a captured process is shown in Figure 3 (task names are freely translated by the authors from German, customer name is

removed for privacy reasons). A formalization of the process based on the real-life execution is visible in Figure 5. The process is initiated by ITL, who receives a customer visit report describing what EDI messages will be exchanged. ITL sends a “Bind EDI customer” task with the attached report to ITE, who asks SL2 to “Maintain the customer master data” in the SAP R/3 system and starts himself to “Setup customer on the EDI-converter” by creating the EDI message structure as requested. When SL2 is ready, ITE “Maintains the partner agreements” by mapping internal SAP R/3 message types to the EDI message types for external communication. ITE finally “Contacts the customer” to initiate the EDI transfer.

When first modeling the process in the jBPM IDE, ITL ordered all tasks sequentially (task names given by both – ITL and later on by ITE slightly differed from the captured real-life process but had the same meaning). Although he found drawing the task nodes and connecting them with transitions straightforward, he considered the environment very technical: *“If you show this to him [SL2] he’ll probably give up the CTM trial [laughing]”* (ITL). While modeling, ITL omitted the “Maintain partner agreements” task.

In the next exercise, ITL was able to perform the process formalization in CTM, by evaluating the generated flow through the explanation and the corresponding TDG in the tracking repository: *“Ah, I didn’t think that they do it in parallel [customer master data & EDI converter setup] ... but yes, both things are independent”*. Regarding the omitted task, ITL commented: *“Yes, I know that but it didn’t come to my mind ... he [ITE] is our expert on the topic... but here [CTM] they [ITE & SL2] have done the fine work for me, right ... I need at most to cross-check with them”*.

When ITE modeled the process in the jBPM IDE, he was able to create a complete diagram, by adding also parallel flow. Later on ITE performed the model transformations in CTM successfully: *“I always liked the other overview [TDG], but this [jBPM graph] I like even better ... they are complementary as the old [TDG] gives the logical work breakdown and this [jBPM] shows you how things actually happened”*. ITE also appreciated the fact that common business users like SL2 can be involved in the modeling of the “flow diagrams” without doing more than managing their CTM tasks: *“Yes, it can happen that someone misses to maintain their percent or status ... but errors are OK, they will focus our attention and help us understand how work is managed or why not”*.

ITL developed the model in the jBPM IDE for 23 minutes, whereas the formalization in CTM took him 9 minutes (including evaluation of correctness). ITE needed 18 minutes for modeling in the IDE and 7 minutes in CTM. We observed that modeling in the

IDE demanded a lot of time alone for thinking of how the process is executed and for writing the task names. Time consuming were also the setting of assignments (swimlanes) and the generation of the task forms, which are automated in CTM. The workflow developed by ITL in the IDE was furthermore inconsistent due to the omitted task and the strictly sequential flow. CTM delivered a real-life compliant process by only requiring comparison with implicitly generated TDG and selection of export mode options.

#### 5.4. Summary of findings

The case study showed that the presented approach for involving end users in business process composition through enhanced personal task management is adequate and efficiently reduces the cognitive distance between work tasks and EUD (modeling) tasks. The primary perceived benefits for task management were the transparency in collaborative activities and the reuse of previous experience. During the case study users were able to develop several weakly-structured process models, as well as personal and global TPs. End-user tailors could successfully transform weakly-structured processes to formal workflows, by using complementary representations of formal processes and user-defined, ad-hoc tasks.

#### 6. Conclusions and future work

This paper presents an integrated approach for enabling informed participation of end users in business process composition by introducing several gentle slopes of complexity and providing added value on personal task management as motivation to overcome each one of them. The approach is implemented and validated through the CTM prototype. Usage of CTM tasks is motivated through transparency in collaborative processes, exceeding the capabilities of common email and to-do lists. The extraction and adaptation of TPs is motivated through the ability to exchange and reuse previous experience. The transformation of ad-hoc processes to formal workflows benefits from multiple representations, fostering tailoring as collaboration between business users, end-user tailors and developers.

Our future research efforts will aim at extending the SER capabilities by allowing deviations from formal workflows at runtime, resulting in on-demand extensions of formal process models with user-tailored hierarchies of ad-hoc tasks. We will continue to investigate further scenarios of CTM usage.



## 7. Acknowledgements

The reported work was supported financially by the German "Federal Ministry of Education and Research" (BMBF, project EUDISMES, number 01 IS E03 C). We thank to all participants in our user studies for their time and cooperation.

## 8. References

- [1] Aalst, v. d. W., and Weijters, A., "Process mining: a research agenda", *Computers in Industry*, Vol. 53 (2003), Elsevier B.V.
- [2] Bellotti, V., Dalal, B., Good, N., Flynn, P., Bobrow, D. G., Ducheneaut, N., "What a To-Do: Studies of Task Management towards the Design of a Personal Task List Manager", *Proc. CHI 2004*, ACM Press, 2004, pp. 735-742.
- [3] Bellotti, V., Ducheneaut, N., Howard, M., Smith, I., Grinter, R., "Quality Versus Quantity: E-Mail-Centric Task Management and Its Relation With Overload", *Human-Computer Interaction*, Vol. 20 (2005), Lawrence Erlbaum Associates, pp. 89-138.
- [4] Beyer, H., Holtzblatt, K., *Contextual Design: Defining Customer-Centered Systems*, Morgan Kaufmann, 1998.
- [5] Button, G., "What's Wrong With Speech-Act Theory", *Computer Supported Cooperative Work*, Vol. 3, No.1 (Mar. 1994), Springer, pp. 39-42.
- [6] Chandler project, <http://chandlerproject.org/>
- [7] Fischer, G., Giaccardi, E., Ye, Y., Sutcliffe, A., Mehandjiev, N., "Meta-Design: A Manifesto for End-User Development", *Communication of the ACM*, Vol. 47, No. 9 (Sep. 2004).
- [8] Forrester Research. *Increase Business Agility with BPM Suites*. Forrester Research Inc., 2006.
- [9] Green, T. R. G., "Cognitive dimensions of notations", A. Sutcliffe and L. Macaulay (Eds.) *People and Computers V*. Cambridge University Press, Cambridge, 1989, pp 443-460.
- [10] Grundy, J., Hosking, J., "Serendipity: Integrated Environment Support for Process Modelling, Enactment and Work Coordination", *Automated Software Engineering: An International Journal*, Vol. 5, No. 1 (Jan. 1998), Kluwer Academic Publishers, 1998, pp. 27-60.
- [11] Holz, H., Maus, H., Bernardi, A., Rostanin, O., "From Lightweight, Proactive Information Delivery to Business Process-Oriented Knowledge Management", *Journal of Universal Knowledge Management*, Vol. 0, No. 2 (Nov. 2005), pp. 101-127.
- [12] JBoss Business Process Management, jBPM, <http://www.jboss.com/products/jbpm/docs>
- [13] Jorgensen, H. D., *Interactive Process Models*, Ph.D. Thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2004.
- [14] Lieberman, H., *Your Wish is My Command: Programming by Example*, Morgan Kaufmann, 2001.
- [15] Lieberman, H., Paterno, F., and Wulf, V., *End-User Development*, Springer, 2006.
- [16] MacLean, A., Carter, K., Löfstrand, L., and Moran, T., "User-tailorable systems: pressing the issues with buttons", *Proc. CHI 1990*, ACM Press, 1990, pp. 175-182.
- [17] Mørch, A., and Mehandjiev, N., "Tailoring as Collaboration: The Mediating Role of Multiple Representations and Application Units", *Computer Supported Cooperative Work*, Vol. 9, No 1 (2000), Kluwer Academic Publishers, pp. 75-100.
- [18] Object Management Group, BPMN, <http://www.bpmn.org/>
- [19] Riss, U., Rickayzen, A., Maus, H., and van der Aalst W.M.P., "Challenges for Business Process and Task Management", *Journal of Universal Knowledge Management* Vol. 0, No 2 (2005), pp. 77-100.
- [20] Stoitsev, T., Scheidl, S., Spahn, M., "A Framework for Light-Weight Composition and Management of Ad-Hoc Business Processes", *LNCS 4849*, Springer, 2008.
- [21] Swenson, K.D., "A Visual Language to Describe Collaborative Work", *Proc. 1993 IEEE Symposium on Visual Languages*, pp. 298 - 303.
- [22] Wulf, V., Jarke, M., "The Economics of End-User Development", *Communications of the ACM*, Vol. 47, N. 9 (Sep. 2004).