

Social Software for Business Process Modeling

Agnes Koschmider^{a,*}, Minseok Song^b, Hajo A. Reijers^b

^a*Institute of Applied Informatics and Formal Description Methods
Universität Karlsruhe (TH), Germany*

^b*School of Industrial Engineering, Eindhoven University of Technology,
The Netherlands*

Abstract

Formal models of business processes are used for a variety of purposes. But where the elicitation of the characteristics of a business process usually takes place in a collaborative fashion, the building of the final, formal process model is done mostly by a single person. This paper presents the design and implementation of a Recommendation-Based Process Modeling Support System with “social features.” A process builder using this system will receive recommendations to complete or edit a formal business process model on the basis of previous usage of modeling fragments by her peers. Such features potentially improve the modeling process and, as such, the modeling outcome, i.e., the quality of the process model. This paper also contains an evaluation of the system’s usage and effectiveness, which builds on an experimental design. It is shown that process builders are inclined to follow up on the provided recommendations and that this will improve the semantical quality of the created model. However, information on peer usage of modeling fragments does not play a big role in selecting the recommendations being followed up. This paper fits within a stream of research that puts emphasis on the modeling process, rather than on the model artifact.

Key words: business process models recommender systems social networks process modeling support

1. Introduction

In past years, the mapping of business processes in the form of process models has become the primary form of conceptual modeling [11]. A business process model captures elements, typically in some graphical form, such as the *activities* that constitute the business process; the *performers* of these activities; the time, location, and modus of their *execution*; and the *information* that is processed [19]. Process models are being widely used in the development of organizational

*Corresponding author.

Email addresses: koschmider@aifb.uni-karlsruhe.de (Agnes Koschmider),
m.s.song@tue.nl (Minseok Song), h.a.reijers@tue.nl (Hajo A. Reijers)

structures [63], information systems [13], service-oriented architectures [15], and web services [16].

Even with substantial numbers of practitioners modeling processes, with vendors such as Lombardi, IDS Scheer and Pallas Athena providing the commercial tools to do so, and with a very active and expanding research community studying process models and languages, there is still reason to worry about the fate of this type of conceptual modeling. In a research commentary, Wand and Weber argue that any form of conceptual modeling often falls into disuse by practitioners after an initial period of excitement [61]. This view is confirmed by a wide range of empirical studies and anecdotic evidence, e.g., [6, 11]. Indeed, it has been noted that the success of process modeling efforts has already become a critical concern [2] and that there are indications that practitioners struggle with various process modeling aspects and find limited support from academic literature in guiding their efforts [29].

To avert the potential downside of process modeling, we believe that it takes an approach that is distinctly complementary to the focus on the *process model artifact* itself, which is so characteristic for much of the current research into process modeling. For example, one dominant stream of research [48, 59] is concerned with the development of analysis techniques that can be used to detect different types of syntactical mistakes in a process model itself, but does not guide the modeler towards avoiding such mistakes. In line with [2, 46, 56], we subscribe to the view that a focus on the *modeling process* is called for, paying attention to the actual decisions that modelers have to make to arrive at a process model that supports the efficient and effective development of a business or IT system. The sensibility of this view is supported by a survey on conceptual modeling [6] which concludes that it is precisely the lack of interest from the academic community for the *act* of data modeling that is missed by practitioners and has led to their decline of interest in it. The open question that we address in this paper is how an individual modeler can be supported in creating a high quality business process model.

Against this background, we present a system that supports the modeling process by providing *recommendations* to a process modeler. The recommendations consist of process model fragments that are deemed suitable for that modeler to complete a business process model that she is working on. To do so, the recommendation system takes into account a process builder's modeling intention and uses a repository of process parts for composing recommendations. The recommendations that are provided by such a system are extended with a *trust mechanism*, based on the social context of the model builder. Heider's famous "balance theory" [23] suggests that individuals are more prone to interact with friends of friends rather than with unknown peers. Following this line of argument, process builders may also be willing to select recommendations that are based on decisions by known (skilled) persons, in contrast to unknown ones. Both features – the recommendations and the trust mechanism – are aimed at supporting the process builder in such a way that the quality of her modeling effort is positively affected and, as such, its outcome: the formal process model.

Our most tangible contribution is that we show how the existing version of

a recommendation-based modeling support system [28] can be extended with capabilities to take social information into account. We describe in detail how the social networks can be gathered from the recorded history of the system’s usage, which consists of the name of the selected recommendation and the performer (users that selected a specific recommendation). When a modeler creates a new process, the modeling history of others who are close to the user can then be exploited. Additionally, we provide empirical indications for the willingness of process builders to follow up on such recommendations in creating a process model and, when they do, whether social information is relevant in choosing between the suggested alternatives.

At a more abstract level, the contribution of this paper is that it strengthens a research stream into process modeling that is more concerned with the *modeling process* than the *model artifact*. While both views have their merits, we believe that the “process-centered” stream is underdeveloped in a field, which – as we argue – can have serious ramifications for the field as a whole. Just as with the stream of “artifact-centered” research, our approach fits into the design science research tradition of the IS field [25], as it creates and evaluates an IT artifact intended to solve identified organizational problems. However, because we aim to affect and improve the actual decisions that modelers make, we are also duly concerned about the effectiveness of that artifact in that respect. This is addressed with the experimentation of our system, as described in this paper.

The structure of this paper is as follows. We first provide a section with background information on the problem context, aimed at the reader who is less familiar with the domain of conceptual modeling and process modeling in particular, and also provide a summary of works related to our approach. In Section 3, we sketch the functionality of the recommendation-based process modeling support system, after which the generation of social network structures is explained in detail in Section 4. The design and results of our experiment to evaluate the use of the recommendation system is presented in Section 5. The paper ends with conclusions and a reflection.

2. Background

2.1. Problem context

As mentioned in the introduction, *business process modeling* – also referred to as *process mapping* – is a special form of conceptual modeling. The paper by Curtis, Kellner, and Over in 1992 [10] is often seen as the rough birth date of the discipline. In this paper, the authors describe how the research on process modeling evolved from an interest of software organizations to improve their development processes. The comprehensible insight that graphical process models provided were considered very helpful for that purpose. Since then, process modeling has been applied in diverse domains, such as manufacturing [14], the service industry [45], and healthcare [35], for the development of a range of business and IT systems. The overview in [31] covers a large variety of process modeling techniques halfway through the 1990s. Today, that supply

has widened rather than diminished, with some of the most popular techniques being Event-driven Process Chains (EPCs) [50], the Business Process Modeling Notation (BPMN) [42], the Activity Diagrams in the UML family [12], and workflow nets [57]. Dominant research streams are concerned with the verification of process models [48, 59] and the use of process models for process execution support [18, 44], while industrial efforts seem most concerned with standardization issues.

Various authors have accentuated the cooperative and communicative aspects of creating a conceptual model and a process model in particular. According to [26, 27, 56], this act involves two related dialogues. The *elicitation* dialogue takes place between an informer (presumably a domain expert) and a model mediator (typically an information analyst). The *formalization* dialogue, on the other hand, takes place between the model mediator and the model builder (usually some tool is used to capture and verify the actual model). This view is congruent with the phases described in [17]: An analyst interacts in an iterative fashion with domain experts to arrive at an informal model, which is then concretized into a complete formal specification.

Crucial for the motivation of this work is the following observation. While the elicitation phase is commonly recognized as a collaborative effort, the formalization dialogue is mostly a solitary activity, requiring specific skills. In other words, the model mediator typically also *builds* the model. According to [46], the few available descriptive studies on process modeling support the scenario of a single expert modeler who creates a formal model of some part of a business. Indeed, the available business process modeling tools on the market today are mostly “one-person tools”, i.e., (1) they assume that only one modeler is changing the model at any point in time [47], and (2) the created models are not reused and further assembled in IT implementation projects [54]. The former factor potentially results in dissatisfaction of business users with current IT implementations because of missing, sketchy or otherwise inadequate content. The latter factor directly affects the economic risk of projects involving process modeling, with the wheel being re-invented time and time again.

Our work can be seen as relevant within the wider discussion of how IS methods and tools contribute to successful IS development. While vendors of tools often do not adequately understand the needs of IS development organizations, there is a lack of research that considers tool support in the light of the recognition of the concept of “method in action” [37]. The focus of the dominant streams of research on process modeling that we referenced is on the *process artifact*, while we feel that tools that support the decision-making process during the actual *modeling process* should also be considered.

2.2. Related Work

Existing work that is relevant to this paper’s subject can be differentiated into five categories: (1) Social network analysis, (2) recommender systems, (3) web service composition, (4) process reuse, and (5) process modeling support.

Social network analysis: The study of social network analysis has attracted the interest of scientists from different fields such as: (1) sociology [64], (2)

psychology [30], (3) economy [55] and computer science. In the computer science field the studies related to our approach are using social networks as a method for communicating trust or distrust [62], [65], [20], [21], propagating changes [60] or analyzing interpersonal relationships in an organization [3], [58].

Recommender system: Various types of recommender systems can be distinguished. A *content-based* recommender system [4] suggests an item to a user based upon a description of the item and the user's interests in the past. This kind of recommender system has its roots in the information retrieval (IR) community [1] and suggests items containing text documents, web sites or movies. *Collaborative* recommender systems [24] predict what a user wants based on what she and people with similar preferences preferred in the past. The focus of this kind of systems is the similarity calculation of users rather than of items (like in content-based systems). The combination of content-based and collaborative-based recommender systems is described with hybrid recommender systems [9]. The recommendation-based modeling support system, described in the paper, can be regarded as a specific type of a hybrid recommender system which incorporates some features of content-based and collaborative-based systems [34].

Web service composition: Many related approaches focus on a composition of Web services (e.g., [40]) or Web services that are semantically enriched (e.g., [49, 8]). Especially the second type of results uses well-defined knowledge representation languages in order to support automated search capabilities or information analytics [41]. However, the limitations of these approaches are that preliminary work is required to build up a (background) ontology in order to use semantic technologies. Insufficient quality metrics are provided to describe the usefulness of recommendations that ease the service selection. In the next section we sketch our techniques for the usefulness of recommendations.

We believe that a semi-automatic composition of web services is much easier than the composition of business processes (the recommendation of appropriate process model parts with a process model part being edited can be regarded as a process composition). Usually, web services are considered to be black-boxes where no knowledge about the user's modeling purpose, perspective, role or history is required.

Process reuse: Several authors have proposed methods for process model reuse [38, 32, 36, 39], albeit with little impact on the modeling context and user's modeling intention and requirements. None of these solutions proposes techniques that will enable the user to finish her design most efficiently.

Process modeling support: The initial idea of a recommendation-based process modeling support has been described in [28]. In practice, we realized that the recommendation engines often suggested process parts that dealt with the desired process but which were actually modeled for a different purpose. Therefore, we extended the recommendation system by taking the modeling perspective into account [33].

Based upon these extensions of the modeling support system, this paper presents a technology for Web 2.0 which plays an important role by enabling new forms of information sharing and exploitation of social relationships. We note that the generation of a social network from recommendation history require

consideration of more relationships than in the enumerated related works in the field of social network analysis.

3. Recommendation Based Process Modeling Support

A recommendation-based process modeling support system [28] suggests process builders fitting processes to achieve a modeling intention regarding the builder's modeling intention and modeling history of a community of users. This recommendation system implementation embeds two types of modeling support in order to achieve the user's modeling intention:

1. A *query interface* allows users to request process models or process model parts that are of interest to them. We define a process model part as a logically coherent group of process elements belonging together (e.g., approval, billing or shipping).
2. A *recommender component* proposes appropriate process model parts which fit to a business process model that is currently being edited. The user can invoke the recommender component by highlighting the corresponding element group to be completed by process reuse. This component of the modeling support should be used if the user is not sure how to complete the process. In this case the results from the query can be unsatisfying due to the process builder's vague intention of the process model.

However, when the recommendation system has been invoked the process builder can configure the process model (part) suggestions in her workspace by inserting or deleting process elements. Subsequently, the process builder can store the modified process version in a process repository for further process reuse.

Before making models and model parts searchable, we need to index them. Process model parts are handled in the same way that the complete models are handled, but additionally, we store a pointer to the business process model with which they are associated. For example, for a business process which consists of three distinct process model parts, we would include four virtual documents in our index: the whole process and each of the three parts.

After indexing the processes users can use the query interface which uses Lucene's query parser syntax¹, and users can enter seven query arguments:

- *Title*: referring to names of process elements (e.g. *approved request*)
- *First Element*: searching for a specific first element(s) in the process model
- *Last Element*: searching for a specific last element(s) in the process model
- *Property*: referring to specific properties of a process model assigned by users before storing the process in the repository (e.g., *standard* signifies a standard process)

¹<http://lucene.apache.org/java/docs/queryparsersyntax.html>

- *Purpose*: referring to models fulfilling one of the four modeling purposes such as analysis, documentation, execution or reengineering
- *Objective Description*: searching for processes fulfilling an objective (e.g., processes modeling handling of order request). This field is only searchable if process builders have annotated the process with the corresponding data before storing the model in the repository
- *Previous User Selection*: searching for a specific user who selected a recommendation.

To overcome a limitation through a controlled vocabulary, we use WordNet²(a free English taxonomy). With standard Boolean operators, such as AND, OR, and NOT, users can express more complex queries.

Processes that meet process builders’ criteria are displayed first in a table-based result list as depicted in Table 1. The ranking criterion *Score* reflects the name match between the user’s query and each process model in the repository. The *frequency* criterion describes how often a process model has been selected/reused by other users, and the average number of *insertions* and *deletions* describe the number of operations made when selecting a recommendation. To encourage process builders’ trust and participation by those process builders who are unskilled in process modeling, we extended the table-based result by one more criterion, which indicates the names of persons who selected a recommendation (*Previous User*).

#	Process Name	Score	Frequency	Insertion	Deletion	Previous Users
1	CheckOrder	95.02	5	15	3	M. Song
...
10	Handle Orders	48.85	3	20	4	A. Koschmider

Table 1: Extended Table-based Result List with Previous User Selection

We consider these ranking criteria as our “metrics” for usefulness of recommendations. In several empirical designs we discovered that a high match between the user’s query input and the recommendation (in our context reflected by the criterion *Score*) was the greatest influence factor for selecting a recommendation. Therefore, we assigned the greatest weight for the Score criterion in our overall ranking of recommendation results.

When double clicking on recommendation(s) in the table-based view a graphical-based visualization of the process will be opened, as shown in Figure 1. In the example the user query returns 10 suggestions for processes or process model parts, while Figure 1 shows two of them.

In Figure 1 the process builder can preview related process model parts for each recommendation (see *Show related process model parts*). The idea is that

²<http://wordnet.princeton.edu/>

Results 1 of 2 out of 10



<p>CheckOrder</p> 	<p><u>inspection of client orders</u></p> <p>This process gives an overview of the inspection of client orders. This process include...</p>	<p>Show related process parts</p>
	<p>Show related process models</p>	
<p>verify customer order</p> 	<p><u>verify orders delivered by customer</u></p> <p>This process describes provisioning error request through the automated comparison of the...</p>	<p>Show related process parts</p>
	<p>Show related process models</p>	

Figure 1: Graph-based visualization of fitting recommendations

process model parts that succeed or precede the part in question and which are used in the same modeling domain that the process builder is in at the moment (e.g., Manufacturing) can help to estimate the degree of fitness of a recommended part.

By pushing the button *Show related process models*, the user can preview all phases of the BPM life-cycle, from the early documentation of a process through subsequent phases of analysis and execution. To realize the functionality of previewing related process models, we construct a user profile based on the respective search history [33].

With a right mouse click (in the previous user column), the user can open network structures which were generated from process models or recommendation history. In the next section, we discuss social network based recommendation support in detail.

4. Social Network based Recommendation Support

This section discusses how to exploit social networks in the context of business process modeling. Three networks can be generated from a process repository and a recommendation system. Table 2 shows the three social networks and their properties.

source	nodes	arcs
Process model	org. units	strength between org. units
User history	users	similarity between users
Insertion history	users	insertion history in terms of users

Table 2: Overview of the social networks

From a process repository, process model parts are used to derive a social network in which a node represents an organizational unit, (e.g., performer or department) in which it is described. Arcs among nodes show the relationship between organizational units, and weight values on arcs indicate the strength of the relationship. This network enables users to consider the fitness between two process model parts in terms of organizational units. The other two networks can be derived from a recommendation system. When users model a business process, two typical behaviors occur. First, process builders select some of the process model parts from the process repository and combine them to make a new process model. However, if they cannot find a proper process model part, they can make a new part by editing an existing one or by combining some parts. A recommendation system can register these kinds of behaviors (i.e., user history, insertion history) and two kinds of networks can be generated from the history. From the user history, a network which shows the similarity between users can be derived. It supports reusing the modeling history of “neighborhoods” in order to faster complete an edited process model. A network from

the insertion history shows the insertion structure in terms of users and enables propagating changes of existing process parts across “clique” members.

The remaining section explains how to generate and use the social networks in detail.

4.1. Social Network from Process Models

Process models contain information about performers who will execute processes. From this information, social networks among performers can be derived. Before explaining the methods of deriving social networks, a typical process model can be defined as follows.

[Process model] A process model, PM , is a 5-tuple (P, T, F, R, π) where

- (i) (P, T, F) is a WF-net [57],
- (ii) R is a set of performers,
- (iii) $\pi : T \rightarrow R$, i.e. performer (R) assignment for a task (T).

The process model extends Workflow nets (P, T, F) with performers (R). In a Workflow net, P , T , and F refer to places, transitions, and flow relations, respectively. The models in the process repository are represented in terms of a Petri net. Activities are modeled by transitions, and casual dependencies are modeled by places and arcs. Performers related to activities are specified in the above transitions.

To generate social networks from the process models, three types of metrics defined in [53] can be considered. They are *transfer of work*, *subcontracting*, and *cooperation*. The transfer of work metrics and the subcontracting metrics take into account causal dependencies (i.e., based on ordering of activities) among performers, while the cooperation metrics consider the occurrence of performers. The causal dependency is defined as follows.

[causal dependency, \Rightarrow] Let $PM = (P, T, F, R, \pi)$ be a process model. For $t_1, t_2 \in T$, $t_1 \Rightarrow t_2$ if and only if $path(t_1 \rightarrow t_2)$ is elementary and $(t_1, t_2) \in F^2$.³

If t_1 and t_2 have a causal dependency, t_1 is followed by t_2 in the process model. Within a process model there is a transfer of work from performer i to performer j if there are two subsequent activities where the first activity is assigned to i and the second activity to j . Considering the frequency of transfers, the weight values on arcs between performers can be calculated using the following formula.

$$\bullet r_1 \triangleright_M r_2 = \sum_{p \in M} (| r_1 \Rightarrow_p r_2 |) / \sum_{p \in M} \sum_{i, j \in p} | i \Rightarrow_p j |$$

$r_1 \triangleright_M r_2$ denotes the total number of transfers from performer r_1 to r_2 in process models divided by the total number of transfers in the models. Such formula enables deriving a social network among performers, which shows the transfer of works between them.

³ $path(x \rightarrow y)$ if and only if there is a path of nodes in the graph corresponding to $(P \cup T, F)$. A path is elementary if each node appears only once. If R is a relation, then $R^n = \{(a_1, a_3) \in A \times A | \exists a_2 \in A (a_1, a_2) \in R^{n-1} \wedge (a_2, a_3) \in R\}$ and R^* is the transitive closure.

Subcontracting considers the number of times performer j executes an activity in-between two activities executed by performer i . This may indicate that a work was subcontracted from i to j . The cooperation ignores causal dependencies and simply counts how frequently two performers participate in activities of the same models. The more often two organizational units work together, the stronger their relation is.

The social network from process models provides an organizational view of business processes. When a user queries possible candidate process models, the result could show the related social network and analysis results. As an example, the process repository contains five process models as shown in Figure 2. The user is not sure how to continue her business process under construction. She invoked the recommender component in order to be supported by the recommendation system. In the extended table-based result list (see Table 1) she right-clicked on the first network structure alternative and thus the social network opened, as depicted in Figure 2.

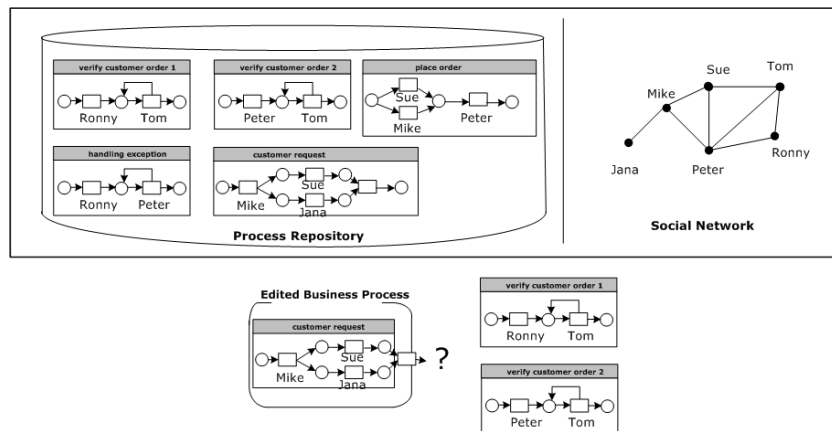


Figure 2: Social Network from process models

In this figure each process model in the repository contains information on performers. The network depicted in the figure is derived by considering the transfer of work between performers. In the network, the nodes refer to the performers in the process models and the arcs show the transfer of work between the performers. For example, since the work is transferred from “Ronny” to “Tom” at the “verify customer order 1” process, an arc between “Ronny” to “Tom” is added in the social network.

If the process builder is not sure which recommendation to select after a customer request (“verify customer order 1” vs. “verify customer order 2” which have the same process structure, but are performed by different performers), she can consult the social network. If the user takes into account the connections between the performers (i.e., “Mike,” “Sue,” “Jana”) in the existing process (i.e., “customer request”) and the two performers (i.e., “Ronny,” “Peter”), there

is no direct connection to “Ronny,” but two connections to ”Peter” (i.e., “Sue” to “Peter” and “Mike” to “Peter”). Thus, “verify customer order 2” process could be more attractive to the user, since the performers are more tightly connected. The process builder can combine the edited business process with “verify customer order 2” and save it as a new business process. Note that the names of performers are used in the process models and the social network in the example for ease of understanding. Of course, in a real process model, a group of people, such as roles, department, etc., is assigned to a task. And nodes in a social network also represent groups of people.

4.2. Social Network from User History

This section discusses the use of social networks from a recommendation history. The user history in a recommendation-based modeling support system consists of the name of the selected recommendation and the performer (user that edited or selected a specific recommendation)⁴. In Figure 3 the user selected the process model part *customer request* (edited and selected by Jane and Peter) and the directly succeeding process model part *verify customer order* (edited and selected by Anna, John and Jim).

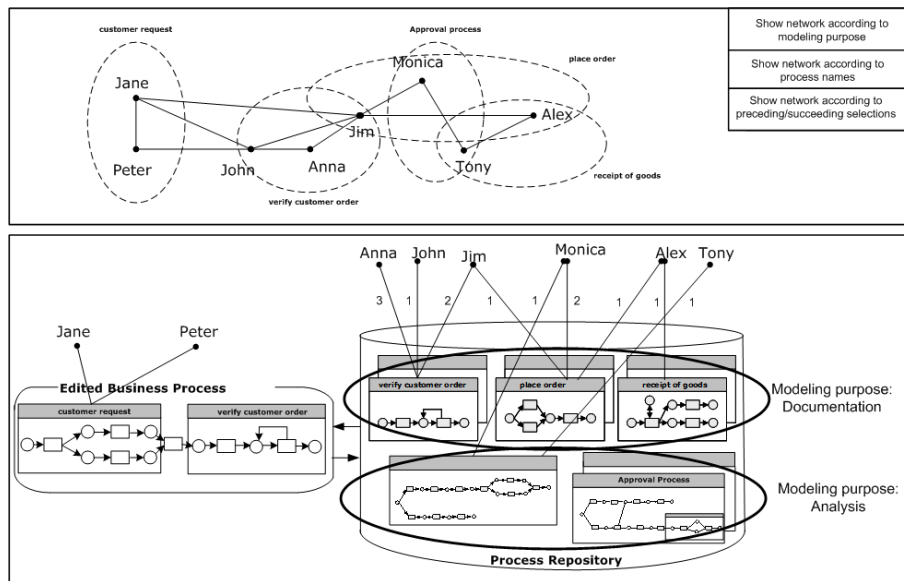


Figure 3: Social Network according to recommendation names

The recommendation system also stores (for each modeling purpose e.g., analysis, documentation, execution or reengineering) the order of selected pro-

⁴Note the different consideration of users. In the previous section the transfer of work between performers is taken into account. In this section users who selected and edited processes lay the foundation for the social network generation.

cess model parts into the workspace of the user. Table 3 shows an example. In the table each row refers to an operation of a user. For example, $user_1$ uses process P_1 and process model parts P_2 and P_1 is located before P_2 . $User_2$ uses the same process model parts (i.e., P_1 and P_2), but the order is different from that of $user_1$ (while the modeling purpose is the same).

	Preceding process	Succeeding process
$user_1$	P_1	P_2
$user_2$	P_2	P_1
$user_1$	P_2	P_3
$user_3$	P_1	P_2
$user_4$	P_4	P_5
$user_2$	P_1	P_5
:	:	:

Table 3: Preceding/Succeeding History

From the information on the selected processes and preceding/succeeding history, social networks can be derived. Table 4 shows an example of the user history generated from the modeling history of the community of users. In the table, each row refers to a user (u) and a column corresponds to a process model (p) in the repository. Each cell (c_{ij}) shows the number of uses of the process model (p_j) by the user (u_i). The right part of the table shows how frequently users use a certain order of process models. Each cell represents the number of uses of an order of processes ($P_j \rightarrow P_k$) by a user (u_i).

	P_1	P_2	P_3	P_4	...	P_N	$P_1 \rightarrow P_2$	$P_1 \rightarrow P_3$...	$P_{N-1} \rightarrow P_N$
$user_1$	2	1	0	0	..	2	2	0	..	2
$user_2$	2	1	0	0	..	2	0	1	..	2
$user_3$	0	0	1	0	..	0	1	0	..	0
:	:	:	:	:	:	:	:	:	:	:
$user_M$	0	0	1	0	..	0	0	0	..	0

Table 4: User History

From the matrix, the distance between two users can be measured by comparing the corresponding row vectors. Several distance measures, such as *Minkowski distance*, *Hamming distance*, *Pearson's correlation coefficient*, etc., can be applied. They are defined as follows.

- Minkowski distance(u_i, u_j) = $(\sum_{k=1}^m |c_{ik} - c_{jk}|^n)^{1/n}$
- Hamming distance(u_i, u_j) = $(\sum_{k=1}^m \delta(c_{ik}, c_{jk})/m)$,

$$\text{where } \delta(x, y) = \begin{cases} 0 & \text{if } (x > 0 \wedge y > 0) \vee (x = y = 0) \\ 1 & \text{otherwise} \end{cases}$$

- Pearson's correlation coefficient $(u_i, u_j) = (\sum_{k=1}^m c_{ik}c_{jk}) / (\sum_{k=1}^m c_{ik}^2 + \sum_{k=1}^m c_{jk}^2 - \sum_{k=1}^m c_{ik}c_{jk})$,

The *Minkowski distance* is a generalization of the Euclidean distance. It has a parameter n : $n = 1$ is the Rectilinear distance also referred to as Manhattan distance, $n = 2$ is the Euclidean distance, and for large values of n the metric approximates the Chebyshev distance. The *Hamming distance* does not consider the absolute frequency but only whether it is 0 or not. Another metric is *Pearson's correlation coefficient* which is frequently used to find the relationship among users. Its value ranges from -1 to +1. If two users have similar preferences, the value between them is close to 1. -1 means a maximal negative line relationship between users. Using the distance measure, the distance matrix among users can be calculated and then transformed into a network form. From the network, by applying a certain threshold value, unimportant arcs can be removed. If two users are connected in the network, it means that the two users used a similar set of process models during their previous process designs.

For example, there is a recommendation history shown in Table 5. Note that the preceding/succeeding information is not considered in this example. The ta-

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}
$user_1$	0	0	0	0	1	0	1	2	0	1	0	0	0
$user_2$	0	0	0	1	0	0	1	2	1	0	0	0	0
$user_3$	1	0	0	0	0	0	0	0	0	0	0	0	2
$user_4$	0	0	1	1	0	0	0	0	1	0	1	1	0
$user_5$	0	0	1	0	0	0	0	0	0	1	0	1	1
$user_6$	1	0	0	1	0	0	0	1	0	0	0	1	0
$user_7$	0	0	1	0	0	1	0	0	0	0	1	1	0
$user_8$	0	0	1	0	0	0	0	0	0	1	0	1	1
$user_9$	0	1	1	2	0	0	0	0	0	0	0	1	0
$user_{10}$	1	0	0	0	0	0	0	0	0	0	0	1	0

Table 5: Realistic User History (P_1 =check Order2, P_2 =check Order3, P_3 =order approval, P_4 =Order Checking, P_5 =CheckOrderAvailability, P_6 =order_approval, P_7 =handle rejected order, P_8 =manage order approvals, P_9 =customer order, P_{10} =check credibility, P_{11} =checked order availability, P_{12} =validate Order, P_{13} =approve order)

ble shows a recommendation history where 10 users were involved who selected a total of 13 process models. The purpose of the models was documentation. From this history, a social network shown in Figure 4 is derived. Note that the threshold value of 0.3 is applied, and arcs (of whichever weight value is less than the threshold) are removed. There are five cliques in the network. They are $\{user_1, user_2\}$, $\{user_3\}$, $\{user_4, user_7, user_9\}$, $\{user_5, user_8\}$, $\{user_6, user_{10}\}$. The user that is directly connected to a certain user is the neighbor-

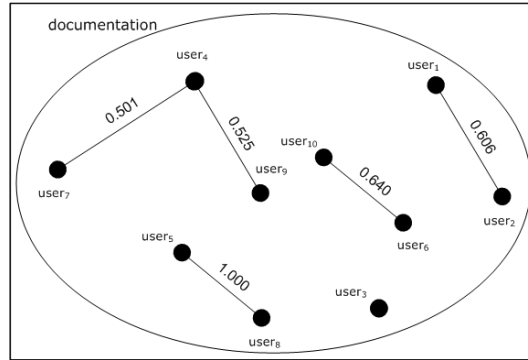


Figure 4: Social Network from Table 5

hood of the user, who has similar process usage patterns. Thus, when the user designs a new process model, she can use the history of her neighborhoods. For example, in the figure, the history of *user₄* is used in the recommendation system for *user₇* but can be used to find a neighborhood of a user. For example, Table 5 can be extended to include the history of the order of process models, and a social network can be calculated from the combined table.

4.3. Social Network from Insertion History

The third way of generating social network is taking into account the insertion history of process models. The recommendation system stores the order of inserted process model parts into the workspace of a user. A user of the recommendation process system can generate a new process model and insert it into the repository. A user can search for suitable process models and generate a new model by combining them. Table 6 shows an example of insertion history. In the table, *user₁* generates “customer order” process by merging “check order availability” and “check credibility.” A user can also take one process model and modify it to make a new process model. In the table, *user₂* modifies “customer order” and generated “check order.”

	used processes	inserted process
<i>user₁</i>	check order availability + check credibility	customer order
<i>user₂</i>	customer order	check order
<i>user₃</i>	customer order + handle reject order	order approval
<i>user₄</i>	order approval + notify customer	approval process
:	:	:

Table 6: Process Insertion History

From this information a social network reflecting the history can be generated. Figure 5 shows the social network derived from Table 6. Figure 5(a) shows

a graphical view of the insertion history from the table. In the figure, nodes represent process models and arcs show relationships between process models. For example, since the “customer order” is designed with “check credibility” and “check order availability,” there are arcs from “check credibility” to “customer order” and from “check order availability” to “customer order.” From this figure, by replacing the process model with the user who generated it, the social network depicted in Figure 5(b) is generated.

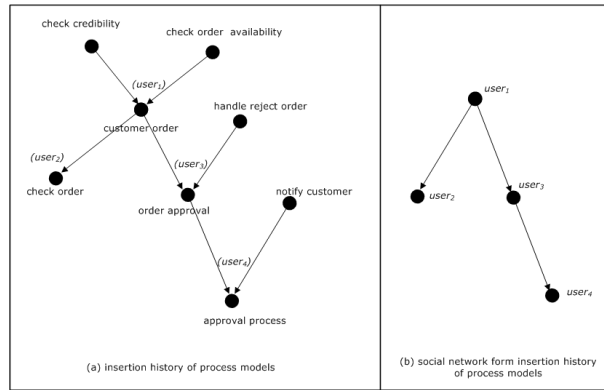


Figure 5: Generating social network from insertion history

Social networks from insertion history cannot be used to find neighborhoods, but for change propagation. Whenever one of the members of the clique changes a process the other neighbors who are affected by this modification will be informed about this change. This notification is optional and will be performed only if users have activated this functionality.

5. Research Method

5.1. Design

To evaluate our modeling support system and investigate our main conjectures regarding its effectiveness, an experiment has been conducted. Basically, the experiment should gain insight into the following questions:

1. Will process builders be inclined to follow up on the suggestions of our recommendation-based process modeling support system? It is more or less a basic assumption underlying all our endeavors in this field that process builders will be interested in, and follow up on modeling recommendations. However, this has not been tested in an empirical fashion so far.
2. Will the support of a recommendation-based process modeling support system improve the modeling process? The second basic assumption behind our work is that the use of a recommendation system – beyond perhaps being appreciated by model builders – will increase the quality of the process model and would not be less efficient than modeling from scratch. After

all, some balance is expected between the additional time that is required to evaluate recommendations and the saved modeling time when following up a recommendation (both compared to a traditional, unsupported situation).

3. If process builders follow up recommendations of the support system, will social information play any role in their decision-making? The specific expectation we have is that recommendations that are based on modeling decisions of people who are socially close to a model builder will receive favorable attention. On the basis of the extension of our recommendation system, as outlined in the previous sections, we are now able to approach this question with an experimental design.

The specific hypotheses with respect to these questions are as follows (numerals corresponding with the previous questions):

- **H1:** Process builders that receive modeling recommendations to create part of a process model will more often use a recommendation than build that model part from scratch.
- **H2a:** Process builders using a recommendation system will create process models in the same time as those without the support of a recommendation system.
- **H2b:** Process builders using a recommendation system will create process models of a higher semantic quality than do those without the support of a recommendation system.
- **H2c:** Process builders using a recommendation system will create process models of a higher syntactic quality than do those without the support of a recommendation system.
- **H3a:** Process builders that follow up recommendations will favor those recommendations that refer to previously used model parts by well-known people over recommendations without any information on previous use.
- **H3b:** Process builders that follow up recommendations will favor those recommendations that refer to previously used model parts by well-known people over those that refer to people they do not know well.

The experiment that has been conducted took place in the semester of 2008 and involved the participation of 28 graduate and post-graduate students. All students followed a course in Workflow Management at the University of Karlsruhe. We refer to this group as the *experiment group*. Subsequently their members were asked to model a business process handling business trips on the basis of an informal description of the procedure in use. Note how this phase coincides with the *formalization phase* that has been distinguished in the introduction of this paper as being of central interest to us.

The informal process description was divided into three subsequent parts: (1) the checking of the travel application, (2) the booking of the hotel room

and the car, and (3) the activities which needed to be handled after returning from the trip. The participants were given access to a repository of process model parts, and the system generated recommendations on the basis of this repository. Overall, each participant received a set of recommendations on each of the three parts of the informal description of the process. For each part, they were given three alternative modeling parts (thus, participants received 9 recommendations in total). The participants were not obliged to follow up on these recommendations, that is, they were free to ignore them. The obvious alternative was to build a modeling part themselves.

To prepare the repository and the recommendations, the following procedure was undertaken. Previously, the same modeling exercise was given to 24 different students of the same university who followed the same course. We refer to these students as the *preparation group*. These students had to create the model on the basis of the same informal description, but were given no modeling support whatsoever. The time that they spent on building the model was recorded using a self-assessment form. From the models created in this way, 19 solutions were considered to be acceptable (and 5 were not). The main criteria that were employed were the semantic quality (or validity) of the model and the syntactic quality. With respect to the former, issues were addressed such as: “Does the model represent the exercise truthfully?” and “Are no parts missing and are no parts spurious?” The syntactical quality was checked with questions such as “Does the model contain wrongly matched pairs of logical connectors?” and “Are all parts of the mode reachable?”

To prepare our system for the evaluation the repository has been populated with process model parts that were derived from three business process models randomly selected from the 19 acceptable solutions. Then, the three process models have been fragmented into nine process model parts. Each of the process models have been fragmented at the same position so that each model part related to exactly one part of the modeling exercise. Consequently, the repository contained three process model parts of roughly similar quality for the “beginning” of the process, three process model parts of equal quality for the “middle” and finally three equal parts for the process “end.”

Subsequently, each of the 28 persons were asked regarding their relationships within this particular group (who knew whom) with the question, “To what extent do you know individual X?” for each of the other 27 persons. The interviewees could choose between three answers: 1. not all, 2. somewhat, and 3. very well. Based on their answers, a social network of the interviewees has been generated. This information was used to annotate the nine process model parts *individualized* for each of the participants. To be precise, for each participant we randomly attached 1) to three of the nine model parts (one for each of the three parts of the modeling exercise) the name of a person whom that participant knew well, 2) to three model parts the name of a person whom the participant did not know well, and 3) for the three remaining model parts no information on previous usage at all was provided. Note how the information on the social relationships was used to mimic information that could have been gathered in a real setting on historic, actual usage of the modeling fragments

by the support system. During the experiment the participants were not aware that this information was artificial.

In the actual experiment, each participant was individually observed carrying out the modeling exercise, e.g., what recommendations were followed up, which adjustments were made, which parts were modeled from scratch, etc. Also, an automatic log was generated for the actual use of both the modeling tool and the recommendation tool. The observations were then matched with these automatic logs to improve and validate their quality. The automatically generated logs were also used to record the modeling time. After the experiment, each individual participant was interviewed about the reasons for following up a recommendation (or not) and the information that they used in this decision making process.

Students volunteered to participate in the experiment and received a small financial reward. The reward was dependent only on their participation - not on the quality of the created model or the modeling decisions.

5.2. Results

To investigate hypothesis **H1**, the occasions that participants selected a recommendation at each of the three parts in the modeling exercise was considered. We advance the null hypothesis that the average number of times that a recommendation is used to cover a part is equal to the average number of times the model part is created from scratch. In other words, the average difference between the two equals zero. The alternative hypothesis would be that the number of parts modeled on the basis of recommendations is bigger than the parts modeled from scratch. To test this, 28 paired frequencies of modeling decisions are at our disposal. (For example, participant Thorsten decides to select a recommendation for parts 2 and 3 of the model, but models part 1 himself. In his case, the difference equals 1.) For the 28 participants, the mean value is 2.5 and the median 3, hinting at a strong preference for using a recommendation. A t-test was used to determine the significance of this result. It provides a P-value of $0.000 \ll 0.05$, which shows that the preference is highly significant at a 95% confidence level. A sign test on the difference in medians gives a similar small P-value, which confirms the strong tendency to base all parts of the formal model on recommendations. Therefore, the alternative hypothesis must be accepted, leading to the acceptance of **H1**: Modelers are inclined to follow up recommendations, assuming that they satisfy a certain quality level.

To test hypothesis **H2a**, the differences in modeling between the experiment and the preparation group were considered. Note that these are not paired observations since the members of the groups differ. The standardized skewness and kurtosis values for the modeling times of both groups are within the range expected when assuming normal distributions of the modeling times. The null-hypothesis is that the mean modeling times for both groups are the same; the alternative hypothesis is that these times differ. The 95 % confidence interval for the mean modeling time of the experiment group equals [69.418, 80.916], while that of the preparation group is [71.455, 85.212]. Since the intervals overlap, there is no significant difference at this confidence level. This is also the case

for the standard deviations for these groups. Therefore, we accept **H2a**: The same time is needed to build a process model with or without recommendation support.

With hypothesis **H2b**, we turn our attention to the semantic quality of the process models created by both groups. Of the 24 process models in the preparation group, only 4 displayed a completely valid representation of the exercise, in the sense that the represented business logic was completely correct and that no obvious parts were missing or spurious. For the 28 process models built by the experiment group using the recommendation system, 19 were completely valid. Since the data is not normally distributed for the preparation group, the non-parametric Kruskal-Wallis test is used to test the null hypothesis that the medians of both groups are the same (the alternative hypothesis being that the median for the experiment group is higher). This test gives a P-value of 0.000, which is much smaller than the 0.05 threshold. Therefore, the alternative hypothesis must be accepted, leading to the acceptance of **H2b**: The semantic quality of the process models created under support of the recommendation system is higher.

The other aspect in comparing the quality of the process models concerns the syntactical quality. Of the 24 process models created by the preparation group, 14 displayed no syntactical mistakes; of the 28 process models created by the experiment group, 11 showed no syntactical mistakes whatsoever. Once more, the data are not normally distributed, thus the Kruskal-Wallis test is used to test the null hypothesis that the medians of both groups are the same (the alternative hypothesis being that the median for the experiment group is higher). The resulting P-value equals 0.911, which means that the null hypothesis cannot be rejected. In other words, hypothesis **H2c** must be rejected: The syntactical quality of the process models created under support of the recommendation system is not higher.

The final set of hypotheses deals with the influence of the social information that accompanies some of the recommendations. To investigate hypothesis **H3a**, we compare per participant from all the recommendations that he/she followed up the proportion that relate to previous well-known users to the ones containing no information on previous use. Thus, for example, from the three recommendations that Thorsten followed up, one of them was connected to (a supposed) previous use of that model part by someone he knows well, while he followed up one recommendation that did not contain any information on previous use (the remaining recommendation relating to someone he does not know well). A t-test to test the null-hypothesis that there is no difference between the mean number of decisions for either type of recommendation leads to a P-value of 0.190. Therefore, the null hypothesis cannot be rejected and we must reject hypothesis **H3a**: Process builders that follow up recommendations will not favor those recommendations that relate to previously used model parts by people in their social network.

Similarly, to test hypothesis H3b we compare per participant the number of recommendations he/she followed up that refer to well-known previous users to the number of recommendations that relate to people not known by the

modeler. The t-test to check the null-hypothesis illustrates that there is no differences between the mean number of decisions for either type of recommendation, resulting in a P-value of 0.155. The null hypothesis must therefore be rejected when assuming a 95% confidence interval. Therefore, **H3b** must be rejected as well. Information on previous usage of recommendations by peers of the process builder seems to play no big role in their selection.

5.3. Discussion

Interpretation. The experiment provides strong support for process builders' preference for and active use of recommendations as provided by the proposed support system. Clearly, for each part of the modeling exercise each participant in the experiment received three recommendations, meaning that someone would need to feel very strongly about the lack of quality of *all* recommendations in order to model a process part himself/herself. Recall that the recommendations are derived from previous acceptable solutions. It is open to question whether it would be possible, in general, to provide such strong recommendation support, but our reference situation is that of organizations that have hundreds or thousands of process models in use from highly related domains. This is increasingly becoming a realistic situation, considering the reports on actual process modeling practice in industry [7, 22, 52]. In other words, the attractiveness of a recommendation system receives strong support from this experiment.

The obvious next issue is whether such support is of any value. What follows from the experiment is that, in accordance with our expectation, the modeling time itself does not change with respect to a traditional situation. So, the support system does not make the building effort less efficient. What is surprising is that the *semantic* quality of the process models is positively affected, while no relation can be found between recommendation usage and improved *syntactical* quality. In other words, a relation can be inferred from our data that a recommendation system prevents process builders from forgetting relevant parts of the informal model or including spurious parts. Clearly, this is a highly relevant outcome and supports further development of this support. However, syntactical mistakes are not prevented as effectively. That syntactical mistakes can be created *at all* when using recommendations can be explained from the practice that recommended process parts are often edited by the process builders, as noted in our observations.

Finally, the experiment gives no reason to believe that the kind of social information that was included in this setting is of much importance in the model building process. On the one hand, this is disappointing and contrasts with our initial assumptions about the impact that social information would have. On the other hand, it gives support for the existing practice of solitary process building.

To get a better insight into the role of social information, in our follow-up to the experiment our participants were asked whether they (a) noticed and (b) considered the social information tags to the recommendations at all. All but one was aware of this information, but only two stated that this information affected their decision making in some way. Much more important in deciding

to follow up a recommendation was the semantical match between the model part and modeling part content (mentioned 21 times), the tag score of the recommendation (mentioned 8 times), and the name of the recommended model part (mentioned 5 times). (Note that multiple answers were possible.) From this feedback, it can be inferred that social information plays only a minor role in the decision to follow up recommendations.

Implications. The most important implication from our work for practice is that the usage of recommendation systems should be actively considered by vendors of process modeling suites, like ARIS, Protos, Lomardi, and TIBCO. In this way, large repositories that are at the disposal of their clients can be more actively used to support their process builders. Our results show an increased interest of users and enhanced semantic quality. This is an extremely important benefit in the setting of individual modelers building process models, at a time when little opportunity exists to receive feedback on the created work by peers.

Because the syntactical quality is not visibly affected and syntactical errors still occur under the guidance of a recommendation system, the combined usage of tools to support process modelers in checking the syntactical quality of their model should be actively considered, e.g., as provided by the open source tool Woflan⁵. As mentioned, the recommendation system does provide support for avoiding syntactical errors, but this feature was deactivated during the experiment in order to provide transparency. The open source tool WoPeD⁶ is a fine example where model building and verification features are closely integrated in the same user interface.

For research, a very preliminary insight from our work is that it seems more fruitful for work on providing collaborative process modeling support (e.g., in [46]) to focus on the elicitation dialogue. After all, process building can be supported by recommendations as evaluated in this paper, while social information and, in turn, interaction does not seem to play too big a role in the formalization dialogue. Nonetheless, in the presented work social networks derived from data were used that had been gathered within the use of that recommendation system. To build social networks, other information could have been used that was available in the wider context of process modeling which would have perhaps been more attractive to interest process builders. For example, the reputation of people, their skill level in process building, their history in process model creation, etc., could all be features to consider. Because the infrastructure is there in the form of the recommender and social network tools that was described, further investigations could take place with relatively small start-up costs.

Limitations. Because students participated in the conducted experiment, the usual external validity problems arise. Despite this, we agree with [5, 43] that in the area of conceptual modeling in general and process modeling in particular the selection of students over practitioners can, in fact, be advisable. Results from both domain understanding and problem solving tasks could have

⁵<http://is.tm.tue.nl/research/woflan.htm/>, last consulted on November 9, 2008

⁶<http://www.woped.org/>

been confounded by participants that were able to bring to bear prior business knowledge in one of the areas [51].

A major limitation in our experimental set-up is the use of inter-personal acquaintance as a proxy for historical usage patterns by process building peers. An alternative would have been to add an extra phase to our experimental set-up. In this way, *actual* usage of (tagless) recommendations by one group could have been used as a basis for creating the social usage tags for the recommendations in the final phase with the experiment group. The use of this artificial information may have been a cause for the low usage in the experiment of recommendations that (supposedly) had been used by peers. Alternatively, social information is really not of any importance, but this should be backed up by further research.

6. Conclusion

Business process modeling tools on the market today are mostly “one-person tools” and mostly do not support an efficient reuse of process models, resulting in dissatisfaction of business users with current IT implementations. The focus of this paper was on improving the *formalization dialogue* of creating process models using a particular tool and thus improving the current IT implementations. Users were guided in this respect within the context of a recommendation-based process modeling support system to which “social” features were added. Three kinds of social networks were used: (1) a social network from a process model repository, (2) a social network from a user history and (3) a social network from an insertion history. The social network from *process models* provides an organizational view of business processes. The social network from *user history* shows the relationship among modelers who use the recommendation system. The social network from *insertion history* shows the relationship among modelers who decided for equal recommendations.

To investigate the effectiveness of the recommendation system a two phase experiment has been conducted. In the first phase students were instructed to model a business process regarding handling a business trip, but without giving them any modeling support. In the second phase a group of students had to create the same model, but this time was given the modeling support system. The participants were not obliged to follow up these recommendations, and they were free to ignore them.

The experiment confirms that modelers are inclined to follow up recommendations as provided by the system. If users decide to reuse recommendations, then they do not increase their modeling time and they can improve the semantic quality of the process models. Thus, the syntactical quality of the process models created under support of the recommendation system is not notably affected. Our assumption could not be confirmed within the experiment that users favor recommendations that refer to previously used model parts by well-known people over those that refer to people that they do not know well.

While some of the results of the recommendation system are very promising to deal with the inherent risks of process building as a solitary activity –

in particular with respect to semantic quality – many challenges remain. Central problems include finding more informative ways to catalogue and represent parts of business process models, better ways to identify relationships between the model parts, discovering the user’s intentions and requirements, and from there automatic prediction of the process model parts that will enable the user to finish her design most efficiently. Furthermore, we believe that the recommendation system would get a significant improvement if researchers from the Human Computer Interaction domain were brought into the conversation. A tight collaboration with that discipline should establish a continuous feedback loop between the technical solutions being envisioned and the development of process model by process builders.

We recall from our experiment and our discussion of its limitations that the annotation of model parts was artificial, which, in retrospect, may have been a serious impediment. Therefore, more research work seems required within realistic settings, such as intra- or inner-organizational process modeling collaborations, where capabilities and skills of group members are crucial for the project success. Because the feasibility and attractiveness of a recommendation system have been demonstrated, it seems worthwhile to extend its support. The direction would then be towards generating better process models on the basis of information that is readily available – be it from a social context of the process builder or otherwise.

To conclude this paper, we feel that the line of our research with its focus on the *modeling process* rather than on the *model artifact* shows strong signs of viability. We hope it will inspire other researchers to join this stream, thus further developing the discipline of process modeling towards providing actual support for supporting organizational development initiatives.

References

- [1] R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, Addison Wesley, 1999.
- [2] W. Bandara, G. Gable, M. Rosemann, Factors and measures of business process modelling: model building through a multiple case study, European Journal of Information Systems 14 (4) (2005) 347–360.
- [3] G. Barnes, P. Cerrito, I. Levi, A mathematical model for interpersonal relationships in social networks, Social Networks 20 (1998) 179–196.
- [4] C. Basu, H. Hirsh, W. W. Cohen, Recommendation as classification: Using social and content-based information in recommendation, in: In Proceedings of the Fifteenth National Conference on Artificial Intelligence, 1998, pp. 714–720.
- [5] D. Batra, J. A. Hoffer, R. P. Bostrom, Comparing Representations with Relational and EER Models, Communications of the ACM 33 (2) (1990) 126–139.

- [6] D. Batra, G. Marakas, Conceptual data modelling in theory and practice, *European Journal of Information Systems* 4 (1995) 185–185.
- [7] J. Becker, M. Rosemann, C. Uthmann, Guidelines of business process modeling, in: W. M. P. van der Aalst, J. Desel, A. Oberweis (eds.), *Business Process Management. Models, Techniques, and Empirical Studies*, vol. 1806 of *Lecture Notes in Computer Science*, Springer Verlag, 2000, pp. 30–49.
- [8] M. Born, C. Brelage, I. Markovic, D. Pfeiffer, I. Weber, Auto-completion for executable business process models, in: *3rd International Workshop on Semantics for Web services (semantics4ws 2008)*, Milan, Italy, 2008, pp. 1–6.
- [9] R. Burke, Hybrid Recommender Systems: Survey and Experiments, *User Modeling and User-Adapted Interaction* 12 (4) (2002) 331–370.
- [10] B. Curtis, I. Kellner, J. Over, Process modeling, *Communications of the ACM* 35 (1992) 75–90.
- [11] I. Davies, P. Green, M. Rosemann, M. Indulska, S. Gallo, How do practitioners use conceptual modeling in practice?, *Data & Knowledge Engineering* 58 (3) (2006) 358–380.
- [12] M. Dumas, A. Hofstede, UML activity diagrams as a workflow specification language, in: *the 4th International Conference on The Unified Modeling Language, Modeling Languages, Concepts, and Tools*, vol. 2185 of *Lecture Notes in Computer Science*, Springer Verlag, 2001, pp. 76–90.
- [13] M. Dumas, W. M. P. van der Aalst, A. H. M. ter Hofstede (eds.), *Process Aware Information Systems: Bridging People and Software Through Process Technology*, John Wiley & Sons, Hoboken, New Jersey, 2005.
- [14] D. Elzinga, T. Horak, C. Lee, C. Bruner, Business process management: survey and methodology, *IEEE Transactions on Engineering Management* 42 (2) (1995) 119–128.
- [15] T. Erl, *Service-oriented Architecture: Concepts, Technology, and Design*, Prentice Hall, Upple Saddle River, New Jersey, 2005.
- [16] C. Ferris, What are web services?, *Communications of the ACM* 46 (6) (2003) 31–32.
- [17] P. Frederiks, T. van der Weide, Information modeling: The process and the required competencies of its participants, *Data & Knowledge Engineering* 58 (1) (2006) 4–20.
- [18] D. Georgakopoulos, M. Hornick, A. Sheth, An overview of workflow management: From process modeling to workflow automation infrastructure, *Distributed and parallel Databases* 3 (2) (1995) 119–153.

- [19] G. Giaglis, A Taxonomy of Business Process Modeling and Information Systems Modeling Techniques, *International Journal of Flexible Manufacturing Systems* 13 (2) (2001) 209–228.
- [20] J. A. Golbeck, Computing and applying trust in web-based social networks, Ph.D. thesis, College Park, MD, USA, chair-James Hendler (2005).
- [21] R. Guha, R. Kumar, P. Raghavan, A. Tomkins, Propagation of trust and distrust, in: *Proceedings of the 13th International Conference on World Wide Web*, ACM, 2004, pp. 403–412.
- [22] J. Gulla, T. Brasethvik, On the challenges of business modeling in large-scale reengineering projects, in: P. P. Chen, D. W. Embley, J. Kouloumdjian, S. W. Liddle, J. F. Roddick (eds.), *4th International Conference on Requirements Engineering*, IEEE, 2000, pp. 17–26.
- [23] F. Heider, *The Psychology of Interpersonal Relations*, Wiley, 1958.
- [24] J. L. Herlocker, J. A. Konstan, L. G. Terveen, J. T. Riedl, Evaluating collaborative filtering recommender systems, *ACM Trans. Inf. Syst.* 22 (1) (2004) 5–53.
- [25] A. Hevner, S. March, J. Park, S. Ram, Design science in information systems research, *Management Information Systems Quarterly* 28 (1) (2004) 75–106.
- [26] S. Hoppenbrouwers, L. Lindeman, H. Proper, Capturing Modeling Processes-Towards the MoDial Modeling Laboratory, in: *Lecture Notes in Computer Science*, vol. 4278, Springer Verlag, 2006, pp. 1242–1252.
- [27] S. Hoppenbrouwers, H. Proper, V. van Reijswoud, Navigating the Methodology Jungle-The communicative role of modelling techniques in information system development, *Computing Letters* 1 (3) (2005) 97–106.
- [28] T. Hornung, A. Koschmider, G. Lausen, Recommendation based process modeling support: Method and user experience, in: *27th International Conference on Conceptual Modeling*, vol. 5231 of *Lecture Notes in Computer Science*, Springer Verlag, 2008, pp. 265–278.
- [29] M. Indulska, J. Recker, M. Rosemann, P. Green, Business process modeling: Current issues and future challenges, in: P. van Eck, J. Gordijn, R. Wieringa (eds.), *21st International Conference on Advanced Information Systems Engineering (CAiSE 2009)*, No. 5565 in *Lecture Notes in Computer Science*, Springer Verlag, 2009, pp. 501–514.
- [30] D. Kenny, L. LaVoie, The social relations model, *American Journal of Sociology* 105 (1984) 141–182.
- [31] W. Kettinger, J. Teng, S. Guha, Business process change: a study of methodologies, techniques, and tools, *Management Information Systems Quarterly* 21 (1997) 55–80.

- [32] J. H. Kim, W. Suh, H. Lee, Document-based Workflow Modeling: a Case-based Reasoning Approach, *Expert Syst. Appl.* 23 (2) (2002) 77–93.
- [33] A. Koschmider, F. Habryn, F. Gottschalk, Real support for perspective-compliant business process design, in: *BPM 2008 Workshops*, vol. 17 of LNBIP, Springer, Milan, Italy, 2008, pp. 30–41.
- [34] A. Koschmider, A. Oberweis, Designing Business Processes with a Recommendation-based Editor, in: J. vom Brocke, M. Rosemann (eds.), *International Handbook on Business Process Management*, Springer Verlag, 2009, to appear.
- [35] S. Liaw, E. Deveny, I. Morrison, B. Lewis, Clinical, information and business process modeling to promote development of safe and flexible software, *Health Informatics Journal* 12 (3) (2006) 199.
- [36] R. Lu, S. W. Sadiq, On the Discovery of Preferred Work Practice Through Business Process Variants, in: *26th International Conference on Conceptual Modeling*, Lecture Notes in Computer Science, Springer Verlag, 2007, pp. 165–180.
- [37] B. Lundell, B. Lings, Method in action and method in tool: a stakeholder perspective, *Journal of Information Technology* 19 (3) (2004) 215–223.
- [38] T. Madhusudan, J. L. Zhao, B. Marshall, A Case-based Reasoning Framework for Workflow Model Management, *Data Knowl. Eng.* 50 (1) (2004) 87–115.
- [39] I. Markovic, A. C. Pereira, Towards a formal framework for reuse in business process modeling, in: *Business Process Management Workshops*, vol. 4928 of Lecture Notes in Computer Science, Springer Verlag, 2007, pp. 484–495.
- [40] N. Milanovic, M. Malek, Current solutions for web service composition, *Internet Computing*, IEEE 8 (6) (2004) 51–59.
- [41] M. Nagarajan, Semantic Annotations in Web Services, in: J. Cardoso, A. P. Sheth (eds.), *Semantic Web Services, Processes and Applications*, Springer Verlag, 2006.
- [42] OMG, Business Process Modeling Notation, V1.1, Specification (January 2008).
URL <http://www.omg.org/spec/BPMN/1.1/PDF>
- [43] J. Recker, A. Dreiling, Does it matter which process modelling language we teach or use? an experimental study on understanding process modelling languages without formal education, in: M. Toleman, A. Cater-Steel, D. Roberts (eds.), *18th Australasian Conference on Information Systems*, The University of Southern Queensland, Toowoomba, Australia, 2007, pp. 356–366.

- [44] M. Reichert, P. Dadam, ADEPT flexsupporting dynamic changes of workflows without losing control, *Journal of Intelligent Information Systems* 10 (2) (1998) 93–129.
- [45] H. Reijers, *Design and Control of Workflow Processes: Business Process Management for the Service Industry*, vol. 2617 of *Lecture notes in Computer Science*, Springer Verlag, 2003.
- [46] P. Rittgen, Negotiating models, in: *19th International Conference on Advanced Information Systems Engineering*, vol. 4495 of *Lecture Notes in Computer Science*, Springer Verlag, 2007, pp. 561–573.
- [47] P. Rittgen, *COMA Handbook: Collaborative Modeling Architecture, Version 2.0*, University College of Boras, School of Business and Informatics, internet: http://coma.nu/COMA_Handbook.pdf (2008).
- [48] W. Sadiq, M. Orłowska, Analyzing process models using graph reduction techniques, *Information Systems* 25 (2) (2000) 117–134.
- [49] J. Schaffner, H. Meyer, M. Weske, A formal model for mixed initiative service composition, in: *Proceedings of The IEEE International Conference on Services Computing (SCC 2007)*, 2007, pp. 443–450.
- [50] A.-W. Scheer, *ARIS - Business Process Modeling*, 2nd ed., Springer Verlag, Berlin, 1998.
- [51] K. Siau, P.-P. Loo, Identifying difficulties in learning UML, *Information Systems Management* 23 (3) (2006) 43–51.
- [52] J. Siegeris, O. Grasl, Model driven business transformation - an experience report, in: M. Dumas, M. Reichert, M.-C. Shan (eds.), *6th International Conference on Business Process Management*, vol. 5240 of *Lecture Notes in Computer Science*, Springer Verlag, 2008.
- [53] M. Song, *Organizational mining in Business Process Management*, Ph.D. thesis, Pohang University of Science and Technology, Pohang, South Korea (2006).
- [54] L. Thom, M. Reichert, C. Chiao, C. Iochpe, G. Hess, Inventing Less, Reusing More, and Adding Intelligence to Business Process Modeling, *Lecture Notes in Computer Science* 5181 (2008) 837–850.
- [55] N. Tichy, M. Tushman, C. Fombrun, Social network analysis for organizations, *Academy of Management Review* 4 (4) (1979) 507–519.
- [56] P. van Bommel, S. Hoppenbrouwers, H. Proper, T. van der Weide, Exploring Modelling Strategies in a Meta-modelling Context, *Lecture Notes in Computer Science* 4278 (2006) 1128–1137.

- [57] W. M. P. van der Aalst, The Application of Petri Nets to Workflow Management, *The Journal of Circuits, Systems and Computers* 8 (1) (1998) 21–66.
- [58] W. M. P. van der Aalst, H. A. Reijers, M. Song, Discovering social networks from event logs, *Computer Supported Cooperative Work* 14 (6) (2005) 549–593.
- [59] W. M. P. Van Der Aalst, A. Ter Hofstede, Verification of workflow task structures: A petri-net-based approach, *Information systems* 25 (1) (2000) 43–69.
- [60] A. J. W. A. Mason, R. L. Goldstone, Propagation of innovations in networked groups, *Journal of Experimental Psychology*, in press.
- [61] Y. Wand, R. Weber, Research commentary: Information systems and conceptual modeling—a research agenda, *Information Systems Research* 13 (4) (2002) 363.
- [62] J.-C. Wang, C.-C. Chiu, Recommending trusted online auction sellers using social network analysis, *Expert Syst. Appl.* 34 (3) (2008) 1666–1679.
- [63] D. Wastell, T. McMaster, P. Kawalek, The rise of the phoenix: methodological innovation as a discourse of renewal, *Journal of Information Technology* 22 (1) (2006) 59–68.
- [64] D. J. Watts, Networks, dynamics and the small-world phenomenon, *American Journal of Sociology* 105 (2) (1999) 493–527.
- [65] C.-N. Ziegler, G. Lausen, Propagation models for trust and distrust in social networks, *Information Systems Frontiers* 7 (4-5) (2005) 337–358.