An Experiment on the Capture of Business Processes from Knowledge Workers

David Martinho[1](✉) and António Rito Silva[2]

[1] Instituto Superior Técnico – UL, Lisboa, Portugal
[2] ESW- INESC-ID, Lisboa, Portugal
{davidmartinho,rito.silva}@ist.utl.pt

Abstract. This paper describes an experiment on the capture of business processes' tacit knowledge owned by knowledge workers. During the experiment, knowledge workers used a functional prototype to achieve a set of goals described to them in natural language. To achieve these goals, they needed to create a set of informational objects. By inspecting these objects, it is possible to decide on the success of a business process instance. The experiment hypothesis states that by enhancing the tool with a recommender system, the nomenclature used by the knowledge workers converges. The experiment took 13 subjects and a confederate. Results show that the proposed approach, empowered with a contextual recommender system, allows work practices to converge, and thus, tacit knowledge to be captured using a practice-oriented strategy.

Keywords: Recommender systems · Ad-hoc workflows · Empirical study

1 Introduction

Capturing knowledge workers' tacit knowledge concerning the organization's business processes is a current problem of the social BPM community [1]. Most organizations execute informal business processes using email instead of formal workflow systems. This happens because the business rules driving the execution of such business processes only exist in the knowledge workers' *heads*, and email provides the necessary flexibility to handle such executions informally.

Traditional requirement engineering techniques focus on the capture of such knowledge, but are known to be time consuming, and to produce biased results due to communication, social or political factors, pondering the cost-benefit of automating such processes. Our goal is to mitigate these problems by providing information about the actual execution of business processes to process experts before they meet with knowledge workers for the first time.

We propose an approach [3,4] to capture tacit knowledge during the execution of business processes. The approach follows a practice-oriented strategy [2], focusing on what knowledge workers actually do, rather than on what they say they do, or on what they ought to be doing. To do so, we developed a modeless

N. Lohmann et al. (Eds.): BPM 2013 Workshops, LNBIP 171, pp. 113–124, 2014.
DOI: 10.1007/978-3-319-06257-0_10, © Springer International Publishing Switzerland 2014

workflow system, with foundations on conversational speech acts [6]. With such foundations, we can structurely capture the knowledge workers' interactions, while still providing the necessary flexibility to continue executing informal business processes with no model constraining their actions. The application allows knowledge workers to address requests to one another and exchange data objects within them. However, since there are no models, knowledge workers must label such requests and data objects in order to tell them apart, i.e. associate some semantic value to them.

With no model constraining a particular nomenclature, an unguided execution is likely to cause divergence on the captured knowledge representation as different knowledge workers can attain the same business goal following different paths of execution, and using different, yet semantically equivalent, labels when refering to requests and data objects. To deal with this problem, we proposed a recommendation algorithm [5] that fosters the reuse of labels used to identify both requests and data objects. Consequently, the algorithm converges knowledge workers' behavior by recommending previously used labels appropriate in similar contexts of execution.

This paper describes an experiment of the approach, demonstrating that the recommender system actually promotes nomenclature convergence among knowledge workers. Although we identified some threats to the experiment's validity, it provides important feedback to enhance both the approach and the tool, improving the outcomes of future experiments.

In Sect. 2 we describe our approach for the non-intrusive capture of tacit knowledge and the tool that implements it. Then, in Sect. 3, we describe the experiment, highlighting its design and analysis. The paper concludes in Sect. 4 with a synthesis of the results and a description of the next steps in our research work.

2 Approach

Setting conversational speech acts as a theoretical foundation, we implemented a modeless ad-hoc workflow system that empowers knowledge workers to send requests to one another, fostering the exchange of business data artifacts in a structured way. The structure consists on keeping track of causality between requests, as well as abstracting and saving the contexts of creation, provision and response of data objects within such requests.

Whenever a new request is created, different contextual information is gathered, and saved in the form of a request template. The request templates are used as the past memory to recommend future requests and data object labels. Finally, when a business process instance is marked as completed, the overall set of data object labels used in that process instance represents the business process goal attained by that process instance. We say that two process instances achieve the same goal when they produce data objects that have the same labels.

With these entities of process goal and request template, we developed a recommender system that is goal-driven as it computes its recommendations based on previously attained goals that best fit the current execution context. Based on that comparison, the algorithm uses the computed set of missing data object labels, and recommends the best fitting request templates that will produce such missing labels.

Since the approach is goal-driven, it only focus on what data objects are produced, instead of how they are produced. This is because different combinations of request templates' executions may produce the same data objects. Therefore, our approach falls better in data-driven approaches, e.g. [7], rather than to activity-centric approaches[1].

2.1 Recommender System

The recommender system recommends both request and data object labels, whenever the knowledge worker initiates a new request or creates a new data object, respectively.

To provide request label recommendations, the recommender system considers three different metrics: goal match, request template fitness and support. The goal match metric ranges between 0 and 1 considering how conform the data labels of the current process instance are to a previously attained process goal. By computing this value for all previously attained process goals, the recommender system then values request templates that drive the process instance towards the achievement of such goals. To do this, the recommendation algorithm analyzes previously captured configurations of requests that created data object labels missing in the process instance. Such analysis results in the request template fitness metric that considers 5 different contexts, which are explained with the required detail in [5]. The request templates that produce data object labels leading the process instance to attain the most conforming process goals are then recommended based on their fitness with the evolving context: who is initiating the request, what data labels existed at that time, what data labels will be created if the request is executed.

Concerning the recommendation of data object labels, the recommender system attempts to find the most matching request template with the current executing request (the one with the higher request template fitness), and recommends the creation of missing data objects based on the creation context captured in the request template.

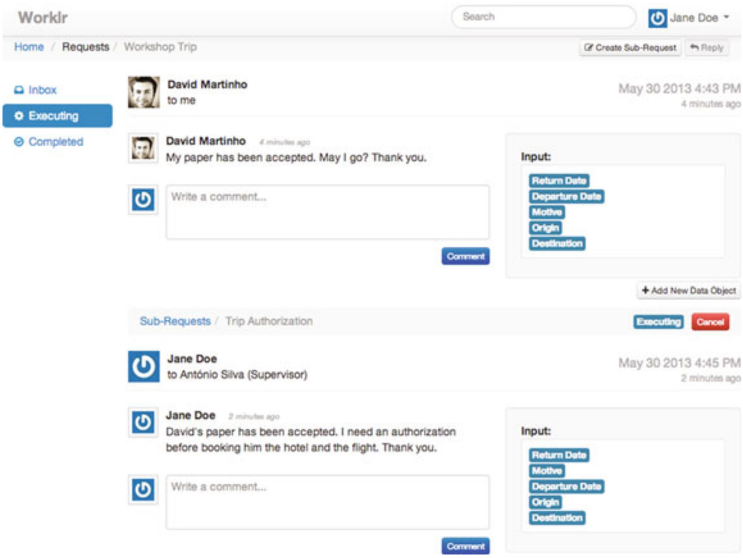
The use of recommendations in workflow systems is not new [8–11]. However, our approach differs essentially in not relying on a pre-existing process model, even if a probabilistic one. Rather than looking into activity traces, our recommender system considers mainly the data objects created and exchanged within those activities. Additionally, such research work focus on process execution assistance considering a particular performance strategy, whereas our approach addresses the elicitation of business processes.

[1] BPMN-like approaches: http://www.bpmn.org/.

2.2 Workflow System

To promote the non-intrusive quality of the approach, the tool follows an email-like experience where requests are assigned to boxes: Inbox, Executing, and Completed. A user accepts requests located in her Inbox, which are move to the Executing box until they are responded. To respond to a request, the executor replies to the request initiator with a set of new data objects, known as the output data objects. These data objects may have been created in the context of that request, or received from a sub-request the executor sent to another co-worker. A business process results in a causal tree of requests, along with information about the data objects provided, produced, and responded to request solicitations, as well as the organizational roles played by the request initiators and executors.

Figure 1 shows a request (Workshop Trip) being executed by Jane Doe (Secretary), which was originally sent by David Martinho (Employee). On the top-right, Jane has two buttons that allow her either to create new sub-requests or reply to the current executing request. Bellow these buttons, there is the request she is executing (Workshop Trip), and one of the sub-requests (Trip Authorization) she sent to António Silva (Supervisor). In the left side of both the request and sub-request detailed views there is a thread of conversation. The top thread allows Jane to chat with the initiator of the Workshop Trip request, i.e. with David Martinho, while the bottom thread allows conversations with the executor of the Trip Authorization sub-request, i.e. with António Silva. In the right side of both detailed views, Jane identifies which data objects were provided as input to her, and those that she provided to the sub-request. That box shows data



David Martinho
to me
May 30 2013 4:43 PM
4 minutes ago

David Martinho 4 minutes ago
My paper has been accepted. May I go? Thank you.

Jane Doe
to António Silva (Supervisor)
May 30 2013 4:45 PM
2 minutes ago

Jane Doe 2 minutes ago
David's paper has been accepted. I need an authorization before booking him the hotel and the flight. Thank you.

Fig. 1. The tool interface for a request with an opened sub-request

objects that were received with the request, created in its context, and received from its sub-requests.

3 Experiment

3.1 Design

This experiment involved 13 participants, all members of a software development team. Before the experiment, the subjects did not know the business process, neither had interacted with the tool. Only in the day before the experiment, the subjects received instruction about the tool in a common demonstration session, which took about 20 min. During such demonstration, the subjects asked questions about the tool functionalities, but did not discussed their experiment script because they only received it after the demonstration session ended.

Different roles were randomly assigned to the subjects, and each subject only received the description of his competences. Therefore, none of the subjects knew the complete process, neither what roles were played by the other subjects. The confederate was responsible to start the business process instances, address the first request, and receive the last response. Consequently, he could analyze the complete process and decide wether or not the business process instances achieved one of the expected goals.

The experimental case was a simple travel request process, where an employee, played by the confederate, sends a request to his secretary including, as input data objects, the travel motivation, the destination, and both the departure and return dates. The secretary is then responsible for getting two approvals, an operational and a financial, and confirm the hotel and flight bookings. To do so, she must contact three travel agencies to get three different tenders for the trip, and choose the cheapest one. She also interacts with the operational and financial managers to get the respective operational and financial authorizations.

In what concerns to role distribution, six subjects played the secretary role, two the operational manager, another two the financial manager, and three played the travel agencies, one for each agency.

Overall, a process instance may attain one of three expected goals: one where the travel is approved and the employee receives both authorizations, and consequently the hotel and flight information; another two where the travel is denied, either for financial or operational reasons. Although there are only three expected goals, these goals can be achieved through the production of more than three groups of data object labels. This is because the subjects may use different labels for the data objects or may interpret their script differently. These differences will be identified and analyzed in the next subsection.

During the experiment, 16 business process instances were executed and completed. The recommendation system was only activated after the 5th execution, and the operational and financial managers were asked to only authorize 4 of each 5 instances in order to generate all three expected goals.

3.2 Analysis

Given our approach hypothesis, we analyze the experiment results from different perspectives. In the first perspective, we are interest in investigating whether there is convergence in the usage of data object labels. Hence, we want to study the number of data object labels, their frequency (the number of data objects associated to each label), and their usage evolution (how the frequency changes with time). Does a label become popular? Does its frequency increase, or does it become unused instead? To do so, we analyzed each one of the emerged data object labels and grouped them according to semantic similarity. For instance, Vôo and Voo (the Portuguese word for flight written with a selling error in the second case) will be under the same semantic group. This will allow us to identify the more popular labels.

Figure 2 presents the eleven data object labels semantic groups created during the experiment. For each group it is depicted, using different patterns, the variations of labels inside the group. For instance, for the Hotel Name group there are five different data object labels with their frequency, Nome Hotel:20, Número Hotel:15, Hotel:3, Hotel Estadia:2 and N. Reserva Hotel:1. The analysis of this diagram shows that for most of the semantic groups, one data object label was predominantly used. Note that the data object labels created by the confederate are not included in the figure, and that the Total Price semantic group has a single instance and corresponds to a situation where a secretary decided to create a data object with the sum of the hotel and flight prices to send to the financial manager.

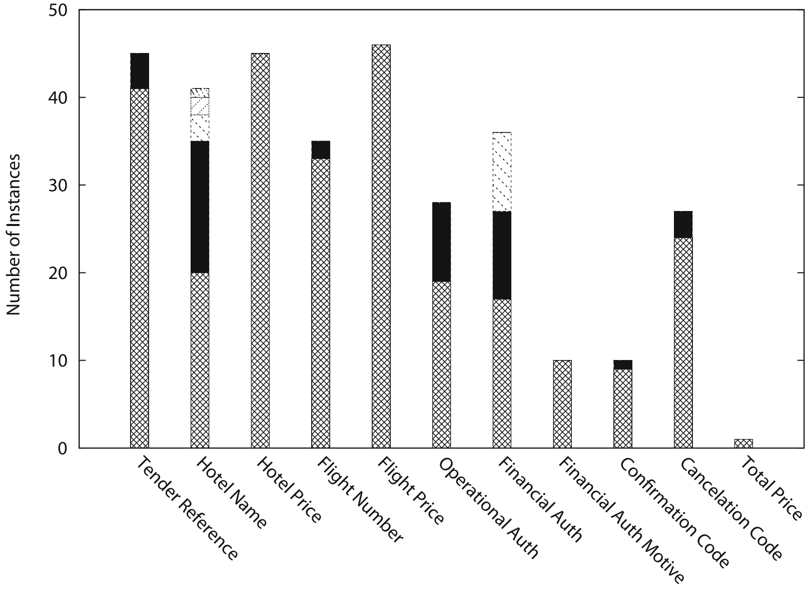
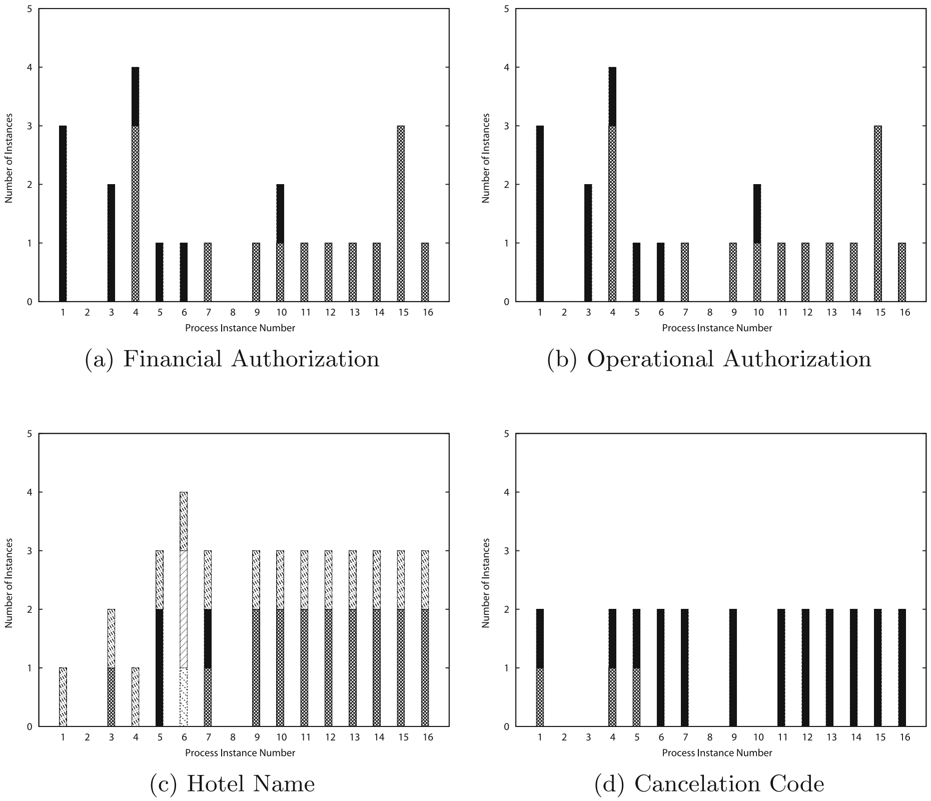


Fig. 2. Data object label semantic groups



(a) Financial Authorization

(b) Operational Authorization

(c) Hotel Name

(d) Cancelation Code

Fig. 3. Evolution of data object label

Although Fig. 2 suggests that there may exist convergence of data object labels, we want to effectively know whether the predominant data object labels become popular as the recommender system is warmed-up by past executions of business process instances. Figure 3 shows, for the four semantic groups that have more variations, that there is convergence on the use of data object labels.

It shows the usage frequency of data labels for each of the business process instances. Business process instances with a higher number occurred latter in time. Note that, in the Hotel Name semantic group, only one of the travel agencies does not follow the recommendation that has a higher rate.

It is also important to analyze whether subjects changed their behavior during the experiment due to the recommendations received by the tool. Note that, it is not surprisingly that a subject commits to the same behavior, and repeats the names of the labels she already used. Such behavior can also be explained by the autocomplete facility of a browser. Nevertheless, it is is interesting to know whether the user accepts recommendations of labels used by co-workers, changing her behavior.

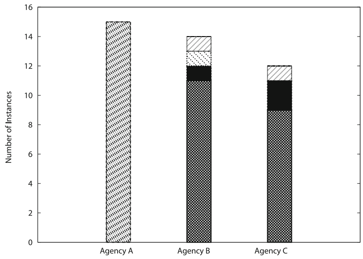
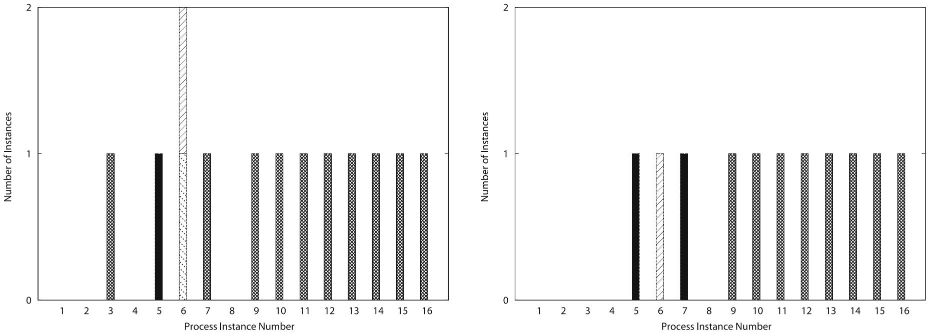


Fig. 4. Hotel name semantic group for each travel agency



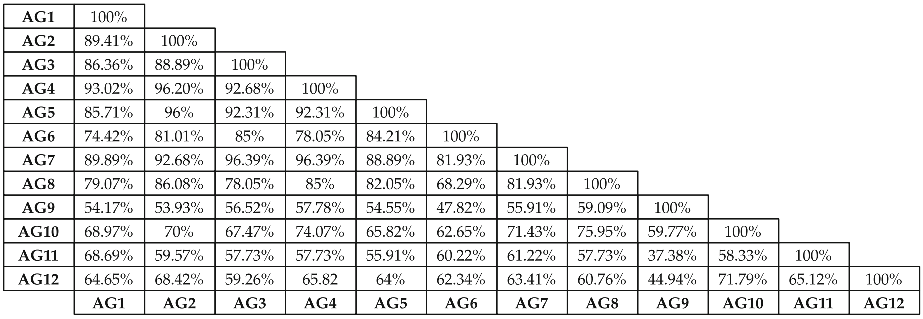
(a) Agency B (b) Agency C

Fig. 5. Hotel name data labels evolution for agencies B and C

To answer this last question, we analyze the labels used by the travel agencies for the Hotel Name semantic group because each agency was played by a single subject. Figure 4 shows that Travel Agency A always use the same data object label while Travel Agencies B and C used different object labels.

If we look more detailedly to these agencies (see Fig. 5), we observe a convergence of data object labels after the 6th process instance as the agencies started to use the same data object label to refer to the hotel name. We can conclude that recommendations fostered the subjects to change their behavior.

After analyzing data object labels convergence, we want to measure the similarity between the achieved goals of the same expected goal. An achieved goal is represented by the labels of data object produced by the business process instance, and it also includes the number of times a particular label was used in the process. For instance, it may be the case that the responses received by the secretary from different travel agencies contain different data objects with the same label, e.g. (Hotel Price). We also want to consider the number of variations among the achieved goals of an expect goal. These variations fit into two aspects: semantic variation and interpretation variation. In the former, two achieved goals contain different labels but they belong to the same semantic group, whereas in the latter the two achieved goals denote different interpretations of the scripts by the



AG1	100%											
AG2	89.41%	100%										
AG3	86.36%	88.89%	100%									
AG4	93.02%	96.20%	92.68%	100%								
AG5	85.71%	96%	92.31%	92.31%	100%							
AG6	74.42%	81.01%	85%	78.05%	84.21%	100%						
AG7	89.89%	92.68%	96.39%	96.39%	88.89%	81.93%	100%					
AG8	79.07%	86.08%	78.05%	85%	82.05%	68.29%	81.93%	100%				
AG9	54.17%	53.93%	56.52%	57.78%	54.55%	47.82%	55.91%	59.09%	100%			
AG10	68.97%	70%	67.47%	74.07%	65.82%	62.65%	71.43%	75.95%	59.77%	100%		
AG11	68.69%	59.57%	57.73%	57.73%	55.91%	60.22%	61.22%	57.73%	37.38%	58.33%	100%	
AG12	64.65%	68.42%	59.26%	65.82	64%	62.34%	63.41%	60.76%	44.94%	71.79%	65.12%	100%
	AG1	**AG2**	**AG3**	**AG4**	**AG5**	**AG6**	**AG7**	**AG8**	**AG9**	**AG10**	**AG11**	**AG12**

Fig. 6. Achieved goal matrix for approved travel

subjects. For instance, a secretary created a data object, Total Price, that sums the hotel and flight prices, to send to the financial manager, while other forwarded the two data objects received from the travel agency. These measures will allow us to analyze goal convergence according to these two aspects.

Figure 6 shows the comparison of the achieved goals for the expected goal: Approved Travel. It compares all the achieved goals, from AG1 to AG12, where achieved goals with lower number were achieved by the latter business process instances, after the recommender system warmed-up.

The similarity measure for two achieved goals, AG_i and AG_j, is computed with the formula:

$$\frac{nEqual(AG_i, AG_j)}{nEqual(AG_i, AG_j) + nSem(AG_i, AG_j) + 1.5 \times nInter(AG_i, AG_j)}$$

where $nEqual$ represents the number of shared data object labels between the two achieved goals, $nSem$ represents the number of data object labels belonging to the same semantic group, and $nInter$ the number of data object labels that are not semantically equivalent.

The analysis of this matrix shows that there is convergence between the goals that were achieved latter, top-left part of the matrix, and divergence between the goals that were achieved earlier, bottom-right part of the matrix. This data reinforces our analysis on the convergence of data object labels as the recommender system warms-up and subjects start following recommendations.

For the second perspective on convergence, we analyzed the use of request labels: Did subjects reuse the request labels? To analyze this hypothesis we group request labels in semantic groups, as we have done with data object labels, and measure the variations within each group.

Figure 7 shows divergence of request labels. For instance, the Tender Request semantic group has 18 variations. This result is not surprising given the recommendation algorithm emphasis on goal achievement, and the definition of a goal being based on data objects production. It may be the case that if the experiment had took longer, request labels may eventually converge. This will be one the research issues to be addressed in future experiments.

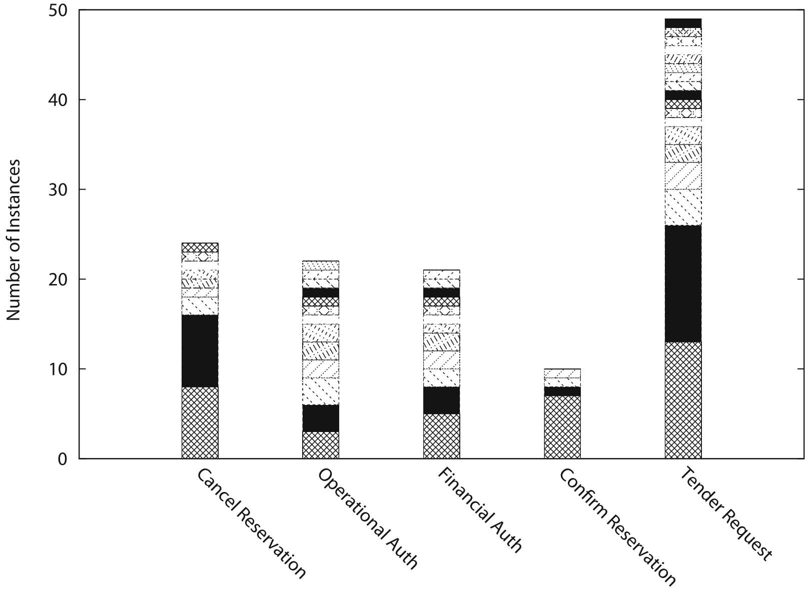


Fig. 7. Request label semantic groups

3.3 Threats to Validity

The subjects received a guided demo of the tool before the experiment, and a script with its responsibilities written in natural language, where there is not an explicit reference to request labels or data objects labels. The business process case fits into the kind of knowledge processes the approach wants to address.

However, we identified some threats to validity of the experiments. The subjects, although being knowledge workers, belong all to the same group, they are software developers, whereas in a business process usually participate different kind of knowledge workers, from technical professionals to administrative staff. On the other hand, the experiment took place in a shared common room, and, although we restricted the communication among subjects during the experiment, it was impossible to completely avoid it during the two hours of experimentation.

These threats were essentially due to logistic limitations, and we were aware of them before the experiment. However, the goal of this first experiment was to receive feedback about the approach and the tool, which will be useful for their improvement and tuning, and to design a new experiment where the identified threats will be dealt with.

As a final note on the threat to validity and the results achieved: from the analysis we can conclude that there is convergence on the goals whereas the same is not true for request labels. In some sense, it confirms our hypothesis that business process goals achievement can be driven by the production of data objects, and that the subjects behavior was not biased because there is divergence in the request labels. They were not aware whether we were studying data object labels convergence or request labels convergence, and if any communication occurred among subjects it makes no sense that it had only impact on the data object labels.

4 Conclusions

In this paper, we present the first experiment of our approach to non-intrusively capture the tacit knowledge of knowledge workers. The experiment addresses the main open issue: can the approach foster the convergence of knowledge workers' behavior during the capture of their knowledge? This issue is particularly relevant because one of the challenges of requirements engineering activities is mutual agreement between stakeholders. This agreement ranges from the terminology of the domain to the goals of the business.

Our hypothesis, i.e. design-by-doing approach performs better on the convergence of behavior, stems from the fact that while performing their daily work, knowledge workers may be more isolated from the social and political factors occuring during requirements elicitation sessions and that are responsible for divergence or requirements misalignment.

To test our hypothesis, we considered that the business process goal is represented by a set of data object labels that are produced during the execution of a process instance, and that it is not so relevant how they are produced, inline with existing workflow data-driven approaches. Therefore, we developed a recommender system and integrated it with an ad-hoc workflow system based on conversational acts. We used these tools in a first experiment, where a group of 13 subjects participated in a travel request process. The results of this first experiment are promising as they show convergence on the data object labels and process goals, despite the divergence in the request labels, which is according to what was postulated by our hypothesis. Although we identified some threats to the validity of the experiment, related to the group composition and environment of the experiment, due to time and logistic restrictions, it gave us good hints on how to pursue our research and design the next experiment.

For the next experiment we will integrate the recommendations presentation with the browser auto-complete to avoid subjects to be confused between our recommendations and the browser auto-completion suggestions. Moreover, we will limit the number of presented recommendations to up to N (we need to identify the threshold value) in order to reduce the noise and allow subjects to carefully choose between a small number of potential recommendations. One of the emphasis of the next experience is to analyze when request labels converge, if they do.

Acknowledgment. We would like to thank the thirteen members of the development team that gently participated in the experiment as subjects. This work was supported by national funds through FCT Fundação para a Ciência e a Tecnologia, under project PEst-OE/EEI/LA0021/2013.

References

1. Bruno, G., Dengler, F., Jennings, B., Khalaf, R., Nurcan, S., Prilla, M., Sarini, M., Schmidt, R., Silva, R.: Key challenges for enabling agile BPM with social software. J. Softw. Maintenance Evol.: Res. Pract. **23**(4), 297–326 (2011)
2. Pickering, A.: From science as knowledge to science as practice. Science as practice and culture 4 (1992)
3. Silva, A., Rosemann, M.: Processpedia: an ecological environment for BPM stakeholders' collaboration. Bus. Process Manage. J. **18**(1), 20–42 (2012)
4. Martinho, D., Silva, A.R.: Non-intrusive capture of business processes using social software. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM Workshops 2011, Part I. LNBIP, vol. 99, pp. 207–218. Springer, Heidelberg (2012)
5. Martinho, D., Silva, A.R.: A recommendation algorithm to capture end-users' tacit knowledge. In: Barros, A., Gal, A., Kindler, E. (eds.) BPM 2012. LNCS, vol. 7481, pp. 216–222. Springer, Heidelberg (2012)
6. Winograd, T.: A language/action perspective on the design of cooperative work. Hum.-Comput. Interact. **3**(1), 3–30 (1987)
7. van der Aalst, W., Reijers, H., Liman, S.: Product-driven workflow design. In: 2001 The Sixth International Conference on Computer Supported Cooperative Work in Design, pp. 397–402. IEEE (2001)
8. Barba, I., Weber, B., Del Valle, C.: Supporting the optimized execution of business processes through recommendations. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM Workshops 2011, Part I. LNBIP, vol. 99, pp. 135–140. Springer, Heidelberg (2012)
9. Burkhart, T., Werth, D., Loos, P., Dorn, C.: A flexible approach towards self-adapting process recommendations. Comput. Inform. **30**(1), 89–111 (2011)
10. Dorn, C., Burkhart, T., Werth, D., Dustdar, S.: Self-adjusting recommendations for people-driven ad-hoc processes. In: Hull, R., Mendling, J., Tai, S. (eds.) BPM 2010. LNCS, vol. 6336, pp. 327–342. Springer, Heidelberg (2010)
11. Schonenberg, H., Weber, B., van Dongen, B.F., van der Aalst, W.M.P.: Supporting flexible processes through recommendations based on history. In: Dumas, M., Reichert, M., Shan, M.-C. (eds.) BPM 2008. LNCS, vol. 5240, pp. 51–66. Springer, Heidelberg (2008)