Mining the Usability of Process-Oriented Business Software: The Case of the ARIS Designer of Software AG

Tom Thaler, Sabine Norek, Vittorio De Angelis, Dirk Maurer, Peter Fettke, and Peter Loos

Abstract

(a) **Situation faced**: The quality of the technical support of business processes plays an important role in the selection of corresponding software products. Against that background, software producers invest considerable capital and manpower in improving their business software's usability with regard to customers' needs and process-related requirements. However, existing approaches from the field of usability engineering generally require laboratory environments, which do not cover the real user behavior without limitations. Therefore, the case described here seeks to improve a user-centric UX approach based on the idea of automatic identification of real customer needs.

(b) **Action taken**: For that purpose, the German Research Center for Artificial Intelligence (DFKI) and Software AG analyzed the issues in the currently available UX process at Software AG. Research and practice were searched for additional approaches to the critical point of *understanding the user*. Finally, a four-step approach based on process mining, consisting of *user monitoring*, *trace clustering*, *usage model derivation*, and *usage model analysis* was conceptualized and evaluated in a user study.

(c) **Results achieved**: The application of the developed approach showed high flexibility and scalability in terms of the level of detail. Despite the small

T. Thaler (✉) • P. Fettke • P. Loos
Institute for Information Systems (IWi) at the German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany

Saarland University, Saarbrücken, Germany
e-mail: tom.thaler@iwi.dfki.de; peter.fettke@iwi.dfki.de; peter.loos@iwi.dfki.de

S. Norek • V. De Angelis • D. Maurer
Software AG, Darmstadt, Germany
e-mail: sabine.norek@softwareag.com; vittorio.deangelis@softwareag.com; dirk.maurer@softwareag.com

J. vom Brocke, J. Mendling (eds.), *Business Process Management Cases*, Management for Professionals, DOI 10.1007/978-3-319-58307-5_16

number of participants, it was possible to identify several process-related software issues and to reduce significantly needed resources (e.g., cost and time). Hence, a promising alternative to the existing techniques of *understanding* was found, leading to important improvements regarding a comprehensive and continuous lifecycle.

(d) **Lessons learned**: The adapted UX approach increases flexibility and a widens the spectrum to proceed to the development of a user-centric business software. Although the improved procedure had a promising performance for further application in production environments, there are some open questions, such as handling the possibly high amount of upcoming data or privacy aspects that must be addressed in the future. Independently and regarding the transferability to other application scenarios, promising potential were identified, such as a mechanism for controlling the software's evolution, the inductive development of usage reference models, and an approach to measuring the ease of learning a new business software.

1 Introduction

Software AG empowers customers to innovate, differentiate, and win in the digital world. Its products help companies combine their existing systems on their premises and in the cloud into a single platform to optimize and digitize their businesses. The combination of process management, data integration, and real-time analytics on one Digital Business Platform enables customers to drive operational efficiency, modernize their systems, and optimize processes for smarter decision-making. Building on over 45 years of customer-centric innovation, Software AG is ranked as a leader in many innovative IT categories. It has more than 4300 employees in 70 countries and in 2015 had total revenues of 873 million euros.

Business process models play an important role in the Digital Business Platform, from the analysis and design of business processes to their implementation, execution, monitoring, and controlling. Therefore, business process models are key artifacts in business process management. Traditionally, process models are generated by human modeling experts using modeling tools like the ARIS Designer, the market-leading BPM tool and one of Software AG's main products. Although ARIS has stood for professional business process design in more than 20 years in practice and research, many facets of the process of process modeling still need research.

Against that background, the Software AG has applied expert interviews, pre-release usability tests, and other established usability methods in order to improve the ARIS software. Therefore, a comprehensive user experience (UX) approach was established that understands the user as a key to identifying and defining existing usability issues and to designing and testing possible solutions. The user experience research for ARIS has covered 10 years of usability testing and interviews with more than 300 users all over the world, which led to over 400 h of recorded video sessions, thousands of person hours in usability sessions, and the evaluation of the recorded and collected data. Still, there are many aspects of the supported business processes to discover. The challenges in the users' daily work are often unknown or cannot be explicated easily, in part because of the mostly strong focus on particular functionalities or cuttings of the software.

Since we also understand business software as tools that support particular business processes, the aim of improving its usability requires not only analyzing the technical aspects of software that affect their operability, but also analyzing the supported processes for areas of improvement. In addition, since customers have the deepest insights into their processes, we see them as a key to that task. Hence, in the context of a research project, we were looking for new approaches that take user behavior in their real environments and in their daily work into account in order to improve the modeling tool based on actual, not yet identified customer needs. Therefore, the case at hand seeks to develop a method for analyzing the dimensions of usability whereby both the system design (in terms of a technical support of the process of process modeling) and the business process it supports are explicitly addressed. For that purpose, existing methods and techniques from the field of process mining are adapted and enhanced with basic ideas of usability engineering in order to improve the current UX approach.

The resulting approach was applied to a use case that focuses on the process of modeling business processes with the ARIS 9 Designer. Therefore, participants were asked to perform some generic modeling tasks with the software, and the recorded user behavior was then analyzed with regard to the issues mentioned above using the developed lifecycle approach. Finally, the approach covers several phases of the BPM Lifecycle (Dumas et al. 2013) and generally starts with recording the user behavior in order to reveal and continuously improve, implement, monitor, and control the supported processes. We were able to produce promising results and revealed several new improvements that support the process of process modeling.

Section 2 describes the current UX approach at Software AG and the initial situation regarding its issues. The basic ideas of improvement and the corresponding actions that focus on the parts of the UX approach that are lacking are presented in Sect. 3. Section 3 also presents the user study that supports the whole improvement process. The improved UX process is then demonstrated and explained in Sect. 4. Finally, Sect. 5 summarizes the lessons learned and outlines further potential for the proposed approach.

2 Situation Faced

In order to develop useful, efficient, and appealing products, Software AG defined a user-centric development process that is influenced by the ideas of design thinking, collaborative design, and lean UX. Embedded in an agile development

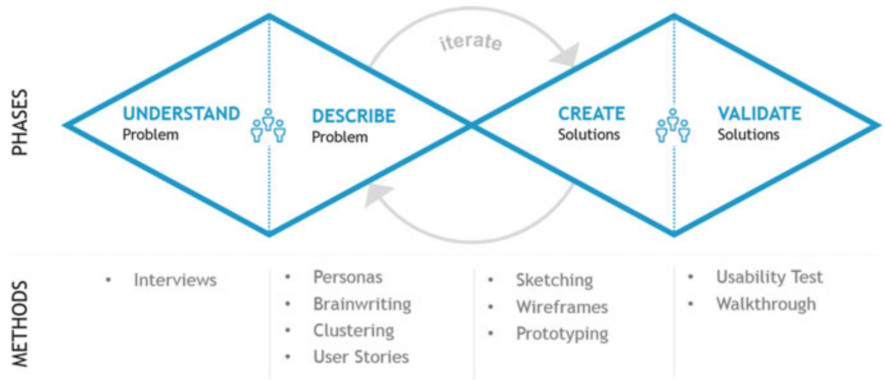

Fig. 1 The UX process at Software AG

environment, the process leans heavily on learning early and often. The basic phases and methods of this user experience process are illustrated in Fig. 1.

The process starts with understanding the users and their problems (*understand problem*). Methods like contextual interviews with members of the target group, surveys, and non-participatory observations are used to gather as much information as possible about the users' goals, tasks, and pain points. These methods also help the team to build empathy with the user and to keep the user in mind during the whole product-development process.

The first step generates many useful insights and considerable amounts of data. Therefore, the second step, *describe problem*, creates a common understanding about the most important pain points in the process. Methods like brainwriting, clustering, and voting help to facilitate this process, and instruments like personas and user stories are used to document the results.

Only when the problem is well-defined does the team start to ideate on possible solutions in the *create solutions* step. In addition to typical creativity techniques, sketches, wireframes and prototypes are used to visualize, communicate, and select ideas.

The last step, *validate solutions*, tests selected ideas with real users and real tasks by running local or remote usability tests or by doing internal walkthroughs of typical tasks.

If needed, this process runs in iterations. For example, a usability test conducted in the validation phase may reveal that the test participants are not able to pursue their goals. In this case, additional interviews are needed, which requires re-entering the understanding phase. In other cases, a series of usability tests may show that none of the tested ideas leads to satisfactory task completion rates or execution times. In these cases, the ideating phase, *create solutions*, must be re-entered.

While large projects like projects for creating new products or developing new components require the full set of methods, smaller projects like projects for implementing new features can get along with a subset or slight variations of the methods mentioned above.

For an existing application, the problem may be easiest to understand by conducting a combination of interviews and usability tests. Interviews focus on identifying typical tasks, which may be known to a certain extent, as they reveal users' opinions. Usability tests, which focus on identifying problems that occur when performing these tasks within the current solution, reveal users' behavior.

In a smaller project, defining the problem might mean agreeing about the most important problems found in a usability test. Creating solutions might be done by discussing a few hand-drawn ideas by hand, and then implementing the solution in the code itself. To validate this solution, the usability test may be repeated.

The basic process approach has been applied successfully in many development and customer co-innovation projects, and considerable effort is expended to improve the software. However, the investigations so far have been focused on particular functionalities and have often relied on (pre-release) tests, the statistical validity of which is limited. In addition, the costs of performing these tests and analyzing the results are high. For example, since it is necessary to talk to the test users during the test run(s), it is also necessary to have employees on site, which may lead to travelling or workshop expenses in addition to the costs of the user experience research's core activities.

Moreover, the functional focus and the essential laboratory environment hinder a holistic view of the software since the test users have the considered function in their minds. However, distorted results are generally expected in non-participatory observations of real-life scenarios. At the same time, a holistic view of the software is necessary to identify the customers' everyday challenges. There are also typical situations in which users cannot adequately explicate the problems they have with a particular functionality or the software as a whole. Hence, the main objective is to develop a comprehensive, methodical approach that facilitates the target-oriented further development of process-oriented business software, especially in order to improve the user experience. Therefore, it is necessary to observe the users in their daily work without spying on them or measuring their performance. In addition, the costs incurred should be as low as possible.

The general idea is to use established process-mining techniques to derive usage models automatically, which can then be enriched by data like GUI information (e.g., element positions) and random user, system, or context data (e.g., user experience). Therefore, a second objective is to extend and improve the current UX approach in a cost-efficient way by using the possibilities of automating that result from the application of process-mining techniques. This approach makes it possible to analyze real user behavior in order to improve the IT support of a business process, especially in terms of its usability. The term "business process usability" should be understood as referring to the extent to which a business software can be used for the effective, efficient, and satisfactory execution of business processes.

3 Action Taken

As a first step toward improving the existing UX approach in the intended way, alternative techniques for *understanding* and covering the state of the art are identified and analyzed in both business information systems engineering (especially process mining) and software engineering (especially human computer interaction and usability engineering). Adequate knowledge in all fields involved is required, as only then can the two disciplines be combined meaningfully. A broad analysis of relevant aspects in these fields was carried out in Thaler et al. (2015a). Approaches from the field of usability engineering and the resulting effort in terms of data analysis are expensive, and they usually require a laboratory environment. Although some work has already been done (Hilbert and Redmiles 2000; Siochi and Ehrich 1991) in deriving process models (petri-nets) and addressing some possibilities for usability analyses, the mining techniques that were used lean toward the rudimentary, as they do not take into consideration such aspects of process mining as dealing with noise and harmonizing the log data of multiple systems. Consideration of current methods and techniques from information systems research is also missing.

At the same time, current approaches, such as genetic algorithms (Alves de Medeiros 2006) and cluster techniques that handle noisy data or avoiding spaghetti-like models (van der Aalst 2011) could be helpful in making it possible to derive meaningful models or meta-information that enables practitioners and researchers to draw concrete conclusions about the usability of business software. Moreover, metrics from the various research disciplines may make it possible to quantify automatically some aspects of the quality, understandability, and usability of business processes and their application. However, a practical, integrated application of the concepts discussed in the context of usability has not yet been observed.

Hence, the identified methods and techniques from the fields of process mining and usability engineering were analyzed with respect to their applicability in the given case. The relevant methods and techniques were consolidated in a four-step approach to *understanding the user*. That approach is described in detail in what follows and is evaluated by means of a user study in the context of a Software AG product.

3.1 Design of the User Study

Several users were asked to work on tasks in the context of business process modeling using Software AG's ARIS Designer, Version 9. The objective was to evaluate whether the identified techniques could serve as an alternative to existing approaches to *understanding the user*. Hence, the focus is on an automatic identification of the problems the users have in performing the tasks. Thirteen students were asked to use the rich modeling client of Software AG's ARIS Designer, which is based on natural language text descriptions, to (1) model an organigram, (2) model an EPC, and (3) modify an EPC. The exercises cover basic actions of business process modeling and do not focus on particular functionalities. The study's objective was to learn how users act in reaching a solution, not to evaluate the correctness of the solution itself. Ten of the students had modeling experience, while the other three had not, so the exercises also revealed information about how users with no prior knowledge interacted with the software. The average time needed to execute the tasks was 47 min. In order to identify concrete usability issues from the analyses, the user screen was recorded by a screen-capturing software.

3.2 A Four-Step Approach to Understanding the User

For the design of an adequate approach with corresponding tool support, the idea of process mining was adapted with regard to the specific requirements of usability engineering. Referring to Thaler (2014) and Thaler et al. (2015b), the steps (Fig. 1) are *user monitoring*, *trace clustering*, *usage model derivation*, and *usage model analysis*. In contrast to previous approaches, the steps can be applied during live operation, so they take real user behavior into account and expensive laboratory experiments are not usually necessary (Fig. 2).

3.2.1 User Monitoring

Process execution data (instance data) is the basis for business process usability mining. Depending on the analysis's objectives, the requirements for log data differ. It is usually necessary to fulfill the traditional log data requirements of process mining (case, task, originator, timestamp) (van der Aalst 2008), and these data should then be extended with additional information depending on the context. When weak points in usability are identified, it might be helpful to log information like GUI information related to element positions and case-specific data. Collecting additional information may require the use of other data sources, such as an enterprise database, external services, or sensors. Since Software as a Service plays an increasing role in the business context, (web) server logs or error logs—which are not traditionally considered in the context of process mining—are possible sources as well. Against that background, one must design a logging strategy based on the

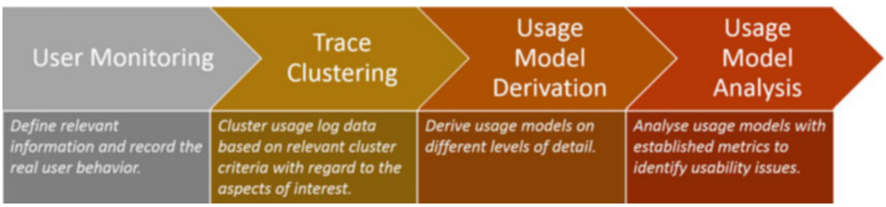

Fig. 2 Understand the user: a four-step approach

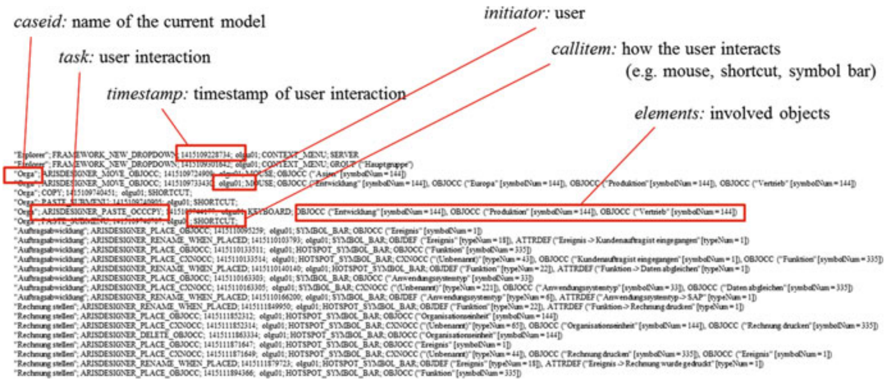

Fig. 3 Sample log file generated in the user monitoring phase

objectives of the analysis or the application scenario and implement it in the software being addressed. Consolidating the various data sources is also important.

Evaluation in User Study In addition to the traditional log information of process mining, he callitem, which describes how the user triggers particular actions (e.g. with a mouse, using a shortcut, using an item on the symbol bar), and the objects involved (e.g., the elements on the grid) are recorded as well. A sample log file generated by the ARIS 9 Designer in the user monitoring phase is presented in Fig. 3.

3.2.2 Trace Clustering

Trace clustering, the task of clustering traces within log data using a specific cluster criterion, is a traditional challenge in the context of process mining. As business software and BPM tools usually cover a multitude of business processes, a corresponding log file covers all of these processes too. Discovering a process model based on a non-clustered log file usually leads to highly complex models that cannot be read by humans (so-called spaghetti-like models), so processes or instance classes must be identified in order to generate less complex process models (e.g., Ekanayake et al. 2013; Jagadeesh and van der Aalst 2010). Therefore, the choice of a particular trace-clustering technique depends on the analysis's objectives (Thaler et al. 2015c). Trace clustering may have any of many criteria, including:

- Processes: variants, patterns, occurrence of loops or tasks, etc.
- Resources/performance: time, budget, hardware, load values, etc.
- Cases: value of a shopping cart, etc.
- Users: experience, age, groups, etc.
- Software: version, device, etc.

Therefore, the recorded log data can be interpreted as a multidimensional data cube whose dimensions are partially not known a priori. In fact, some dimensions are given by the log specification (the recorded attributes), while others, such as the actual recorded process, are unknown. Therefore, it is possible, at least in part, to apply slicing-and-dicing approaches from the area of data warehousing. However, especially in the context of business process usability analysis, the identification of new information and patterns related to the processes and variants, possibly depending on user profiles, is important. A good overview of existing trace-clustering techniques with their corresponding implementations and an evaluation of their applicability in various contexts is presented in Thaler et al. (2015c).

Evaluation in User Study Since the user study contains three exercises, in the trace-clustering phase we initially focused on separating the log file based on the processes that are equivalent to that exercise. Prior to the clustering, we knew that each of the exercises contained actions that are usually not performed in the other ones. Referring to Thaler et al. (2015c), we used the known causal dependencies (case-specific tasks) as a basis for separating the log file. Then we separated the log file based on the information concerning whether a user had experience working with ARIS.

3.2.3 Usage Model Derivation

Process mining distinguishes three fields of inquiry: process discovery, conformance checking, and enhancement (van der Aalst 2011). Process discovery seeks to derive a new process model based solely on log data, while conformance checking compares the as-is process to the to-be process. Enhancement focuses on the derivation of new information from log data in order to extend or improve an existing process model.

Against that background, all three of these fields of inquiry play important roles in business process usability mining in general and in the phase of usage model derivation in particular. In that phase, one derives a process model based on the clustered log data and enriches it with information like performance data, execution probabilities, correlation matrices, and additional (scenario-specific) data and metrics. Many process-mining techniques that produce output models with differing characteristics already exist; they differ in terms of the fitness and appropriateness of the resulting models to the underlying log files, such as in their simplicity, their abstraction level, and the resulting model type (e.g., petri-nets, EPC, FSM) (Thaler 2013), or in their approach to calculations. Therefore, a concrete algorithm should be selected based on the concrete objectives of the analysis (e.g. Alves de Medeiros 2006; van der Aalst et al. 2004; Weijters and Ribeiro 2011; Weijters et al. 2006). In contrast to discovery and enhancement, conformance checking should be seen as belonging to the phase of usage model analysis (phase 4), as it might be important to know whether the users use the software in the intended way.

Evaluation in User Study We applied the Heuristics Miner (Weijters and Ribeiro 2011) with default parameters to derive the usage models based on the clustered log

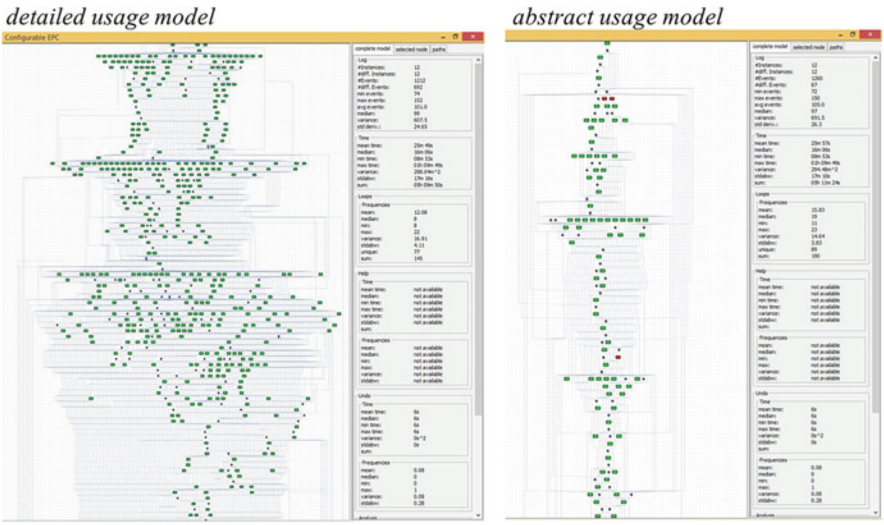

Fig. 4 Detailed vs. abstract usage model visualized with the RefMod-Miner

files. In so doing, we prepared the log files in two settings: (1) task, callitem, and elements are consolidated and use a task description like “place event ‘Event’ by HOTSPOT_SYMBOL_BAR,” and (2) the task is used as it is recorded, such as “place event.” Extracts of the resulting EPC models from the two settings are presented in Fig. 4, which shows that the models’ complexity differs to a high degree, grounded in the fact that setting 1 produces much more detailed node labels and, thus, a significantly higher absolute number of nodes than setting 2. Hence, the degree of detail in the task description again depends on the analysis’s objectives.

3.2.4 Usage Model Analysis

There are several possibilities for analyzing the usage model. Many metrics from a variety of research fields exist to characterize the model(s) and give first indications of weak points:

- Model metrics: complexity, extent, cross-connectivity, etc. (Melcher 2012; Mendling et al. 2012).
- Process metrics: execution count, execution time, error rates, cancellation rates, etc.
- Usability metrics: irrelevant actions, undo actions, using help function, etc.

These categories can be broken into subcategories, such as size and complexity in terms of model metrics or placement, and time aspects in terms of usability metrics. Other performance metrics may include:

- Achievement of objectives/conformance checking: In the context of business processes and their management, there are often objectives that should be achieved at process execution. These objectives could include the process's overall execution time and consumption of resources. Business rules that are obligatory at the process execution, such as legalities, might be important in determining conformance.
- Causal dependencies: Process models may contain causal dependencies between activities or process fragments that are not evident in the process model. A correlation matrix may reveal those dependencies.
- Core and exception fragments: Process models often contain activities or fragments that are executed in a high percentage of cases (core actions) or those that are seldom executed (exception actions). Knowledge about frequency helps in keeping the focus on the most important system points during development.
- Non-supported processes: Sometimes users use a system to execute processes that were not intended by the system's producer. Identifying these processes helps in the effort to improve a system in light of customer needs or to identify other business areas that can use the system.
- System avoidance: Users can avoid a software's particular functionalities, even though they are available, which may indicate a non-working or badly implemented functionality that needs analysis.

In short, simple statistical indicators might provide hints concerning process or software usability issues, although these indicators cannot be used to analyze these issues in detail. In most cases, further information on the process and its execution logs is needed based on input from human experts.

To calculate the metrics, we implemented a tool support in the research prototype RefMod-Miner.[1] An extract of the available metrics and statistical analysis techniques is illustrated in the screenshots presented in Fig. 5. It is also necessary for the tool to have functionalities that allow graphical navigation through the model, like the visualization of a node's predecessor and successor nodes or the highlighting of particular nodes (like help or undo calls). The application of these analysis techniques may provide concrete hints about weak points in the technical support of modeling and how to prevent them.

Evaluation in User Study Analyzing the models with the RefMod-Miner led to the identification of several aspects from which to view the models, ranging from a purely technical perspective to a professional one. Three of these perspectives are presented in Fig. 6.

The first example shows that users could not understand the toggled-edge mode. When the mode was activated, they expected an automatic connection of edges based on the element positions, which led to the effect that some modelers

[1]RefMod-Miner: http://refmod-miner.dfki.de

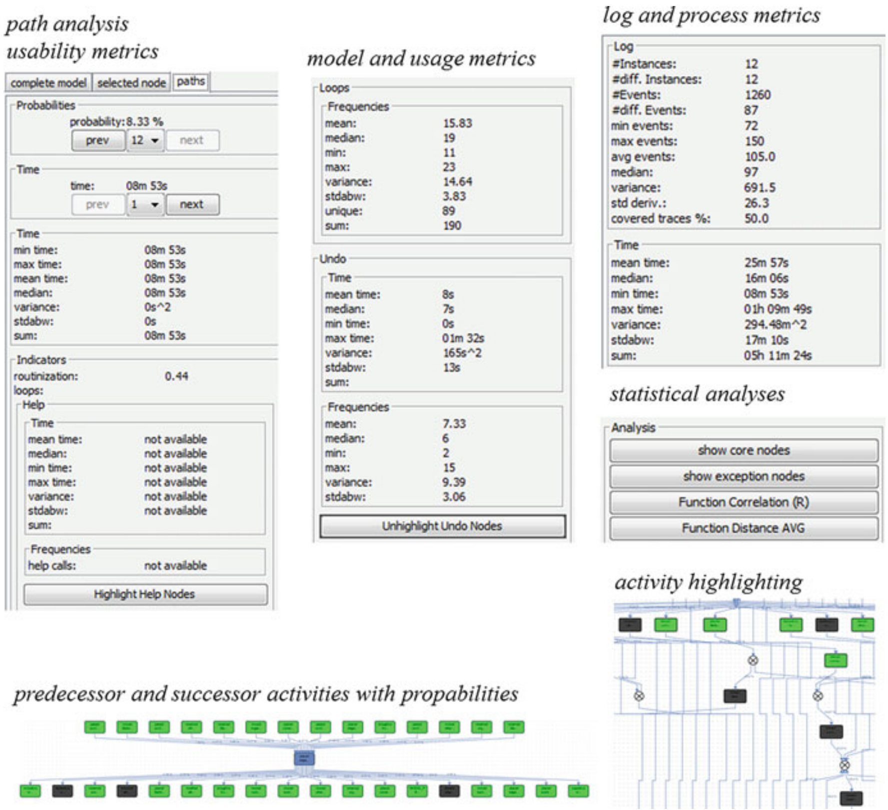

Fig. 5 Extract of metrics and analysis tools implemented in the RefMod-Miner

placed connectors over the edges that connected several nodes. In contrast to their expectations, the connector was not automatically connected.

The second case takes a more professional perspective. Some modelers placed organizational units on the grid and connected them to an activity. They expected the connecting edge to be undirected, but the system automatically produced a directed edge from the organizational unit to the activity. The only solution to the problem was to delete the edge direction manually for all corresponding edges. A similar case showed that it was not possible to modify the edge direction, and this limitation might be meaningful in many contexts.

The third case reveals other strategies in modeling. While some modelers placed the nodes and labeled them immediately, others added a set of nodes and labeled all of them afterwards, a strategy that consumes much more time.

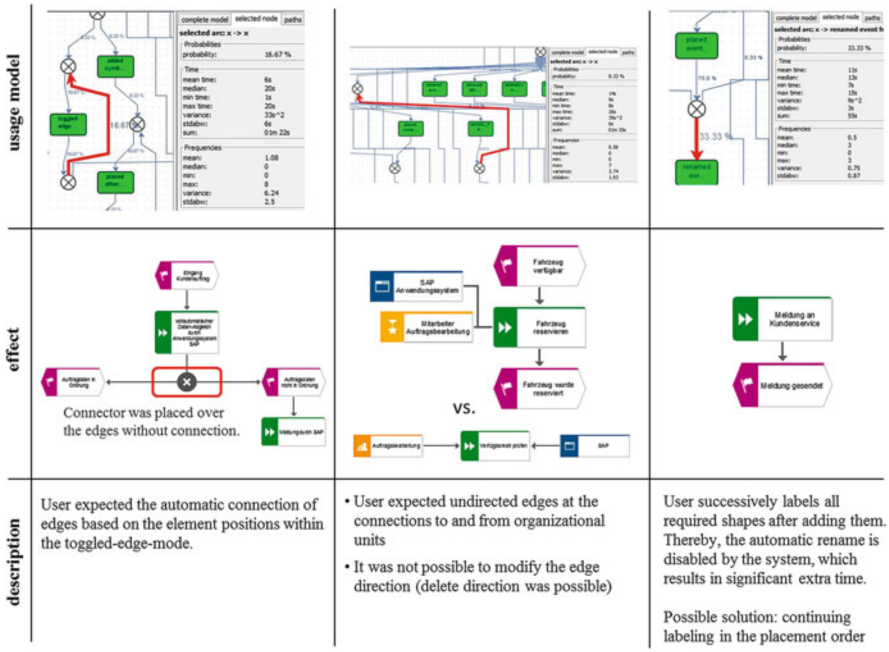

Fig. 6 Presentation of three identified issues

4 Results Achieved

In order to clarify whether the four-step usability-mining approach meets its objective, one must determine whether the subsequent phase of the overall UX approach, *describe problem*, can be processed based on the generated results. In other words, it should be possible to explicate the customers' needs in order to design improved IT support for their business processes.

The first identified issue results from users' inability to interpret the meaning of the "toggled-edge mode," a problem derived from the users' interaction with the software, where the corresponding button was clicked several times in sequence without the intended effect. Renaming the functionality might improve its understandability. The second issue is a bug that can be fixed by allowing the user to modify the edge directions. Since the third issue uncovers a user demand (a modeling strategy that had not yet been considered), a new functionality that supports that strategy is necessary. Continuous labeling in the placement order might be a meaningful feature in meeting that demand.

Hence, despite the early stage of studying and applying the approach in a real context and although the number of participants in the user study was small, we identified ten issues, ranging from minor bugs and general weak points to specific user demands. The derived information was also sufficiently detailed to be described in a professional way and to be addressed with concrete improvements, which are currently being implemented. Therefore, there is potential for application of the approach in a broader user study with pilot users and beyond.

The instantiation of the phases showed that the approach is highly flexible and scalable. Starting at the user monitoring phase, the log information could be enhanced with information like the heart rate of a user or parallel interaction with other cases or software tools. An analysis's level of detail can also be individually scaled, as shown in the usage model derivation phase. Moreover, the graphic representation of usage models enables human observers to easily conceive the human interactions in a broad context and, thus, to replay critical cases in order to detect issues and develop possible solutions.

Especially in a real business environment, the application requires additional thought and action on privacy. Since the procedure might enable employee monitoring, concepts for reliable anonymization are essential to ensure that the recorded information is not traceable to a particular employee or used to measuring the performance of departments. The end users must be able to decide whether they agree to the process of recording and transferring the resulting log data to the software-providing company.

In summary, the four-step usability-mining approach meets expectations. Since it can be applied in customers' production environment, a laboratory environment is not necessary (although it may be necessary in the case of user studies). Problem situations and their locations in software can be detected automatically based on metrics from the areas of usability engineering and business process models. Thus, much of the work that is traditionally performed by usability analysts is done by the analysis tool. Even so, it is not possible for these metrics to detect all issues, so analysts are still necessary; additional application of qualitative interviews in particular might be meaningful. At the same time, visualization of the usage models enables analysts to understand real user behavior and the issues that result from them more easily, more effectively and more quickly. Thus, the necessary resources in terms of costs and time can probably be reduced significantly, although concrete data is missing in the current state.

Finally, promising results led to an enhanced UX process that includes an additional *understand* approach and several effects on the overall UX process (see Fig. 7). In fact, the proposed usability-mining approach reveals a new way to understand the software users based on their real behavior in their daily work. Hence, an alternative was found to interviews in terms of both the procedure and the results. Interviews can provide both high-level deep-level insights on particular aspects of a process and are appropriate for discussing solutions that have not yet been designed or implemented. In contrast, usability mining focuses on the details with which users are confronted, facilitating a broad or overall evaluation of all user interactions with a software, which is not possible with interviews. However, interviews can also be an appropriate technique with which to analyze that quantitative results of usability mining in a qualitative way.

Since the necessary data for usability mining are acquired automatically, it is possible to redo the usability-mining steps on that data, as well as on new data.

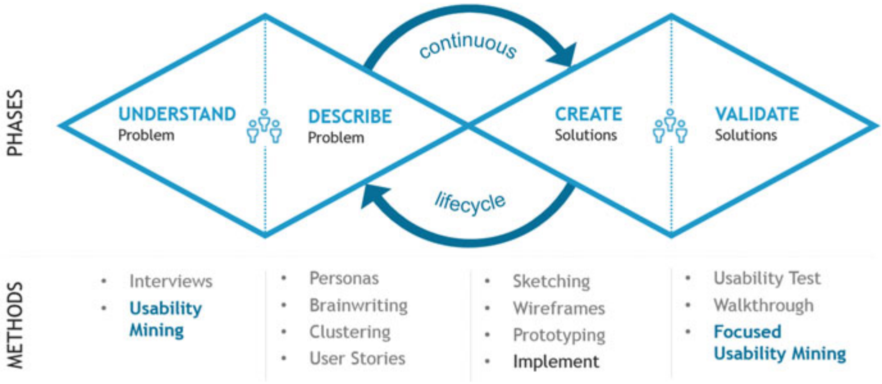

Fig. 7 The improved UX process

Hence, it is easy to determine whether a particular issue is fixed by recalling the corresponding usage data in the next software version. This process, *Focused Usability Mining*, has two important effects. First, comprehensive and perhaps expensive prototyping with a corresponding usability test walkthrough is not necessary before the general roll-out, as the effect of a fix can be measured with the upcoming version. As a consequence, the release cycle can be shortened so there are more resources for the further development of the software. Second, the UX process should not be considered a straightforward process with a fixed start and end and several iterations but as a lifecycle, which allows the continuous and target-oriented further development of a software. Hence, the borders between *validate* and *understand* are often blurred since *understand* in particular covers the users' understanding not only to identify unknown issues but also to verify the solution to known issues in a continuous way.

In summary, the developed method significantly enhances the capabilities of the UX work since it enables continuous screening of usability issues in a semi-automated way. At the same time, focused work is still important, especially in cases like those of developing new products or functionalities and redesigning existing ones.

5 Lessons Learned

Although the results are promising, there are still some aspects of the user study that remain to be discussed. First, the statistical relevance of the found issues and user demands to general user is currently unknown. Another study with more participants would have to use statistical tests, such as the p-Value to make this determination. Second, from a technical point of view, the amount of upcoming log data will require the use of methods that can handle it. Depending on the degree of detail of the log files (e.g., every click, every mouse movement), the number of monitored users, and their intensity of use, its processing will require considerable memory. Therefore, the content of these log files will become more complex and challenges like user tracking, clustering of the log files, and the potentially high complexity of resulting usage models, which are difficult for humans to interpret, will have to be met. First methods and techniques that address these challenges already exist (e.g., Ekanayake et al. 2013; Evermann and Assadipour 2014) and should be evaluated with regard to their applicability in this context.

Another challenge refers to the case of hidden tasks. Hidden tasks are user actions that cannot be recorded by the logging mechanism since they occur apart from the analyzed software, such as manual tasks or tasks done using another software. A typical example is asking a colleague about a functionality by telephone or in person.

One might also ask whether it is more expedient to improve the usability of software tools that support a particular business process than it is to train the end users. While adequate training might help the users to work more efficiently, effectively, and satisfactorily with a modeling tool, individual approaches to modeling business process models based on the end users' habits and preferences are equally important. Therefore, it is necessary to do both: train the end users and improve the software's usability based on the end users' needs. Whether user training or software improvement leads to more promising results in terms of business process usability—efficiency, effectiveness, satisfaction, and costs—must also be determined, as these are important factors, especially for small and medium-sized enterprises. The cost factor is a major strength of the method developed here.

The UX approach (Fig. 7) used here uses an interview technique for *understanding the user*, as usability mining does not replace it. Interviews may also be important in evaluating qualitatively the quantitative results from usability mining in terms of indications regarding particular usability issues. When one is developing new functionalities or software, it is also necessary to include the users in the design phase. Usability mining is not applicable here since the software is already rolled out. Nevertheless, it can meaningfully applied to validate generated solutions.

In addition the important improvements to the UX approach, as presented in Sect. 4, usability mining can deliver insights into the software usage that have not yet been addressed. Among others, several design-related questions can be answered as well:

- What are the core application scenarios on the user side? Which implemented processes or functionalities are not used?
- Are there observable user profiles other than user role and user experience? Are there significant differences in how users use a system?
- Are there observable case profiles for a process that influence its execution?
- Are there other functional requirements on the user side?
- Are there ways to improve the supported process (e.g., user- or case-sensitive processing, adding new functionalities, data preloading, reorganization of forms)?

With regard to the transferability of the developed method to other domains and applications, the analysis of the real user behavior can also be seen a basis for working in several application scenarios (Thaler 2014; Thaler et al. 2015b):

- *Controlling the Software Evolution*. Further development of a software is in the nature of a product's lifecycle, although it can be challenging to determine whether such a development leads to the desired effect and whether it is used as it is intended. This issue affects both new supported processes and existing ones. Since the developed method analyzes real user behavior, it is possible to follow the software evolution from the user side. This benefit can also be seen in Thaler et al. (2013).
- *Inductive Usage Reference Model Development*. Information about a process's performance, resource consumption, and other collected data facilitate the inductive development of reference models with a best-practice character. Based on the process instances, a process model could be derived that pursues objectives like minimizing costs or resources or optimizing output quality.
- *Ease of Learning*. One quality criterion of a business-process-supporting software is the effort required to learn to execute it. An analysis of users' usage models over time would visualize their learning effects and allow the derivation of individual learning curves.

In summary, the paper at hand presents a method for assessing the usability of process-supporting software tools based on process mining. It addresses important areas that must be investigated in order to gather insights about the further development of the software based on users' needs. While the phases of user monitoring, trace clustering, and usage model derivation have an established theoretical and technical foundation that can be adapted to apply to usability, a detailed analysis of the resulting data is challenging. Several ideas have been proffered to quantify the usability of a software system and process models, but these ideas are yet to be developed, conceptualized, implemented, and evaluated.

We showed that the developed method links the software-engineering view and the process-oriented view on business-process-supporting software, which suggests the potential for their design and further development. We also identified several promising scenarios for meaningful application of the method in other domains that will be addressed in future work.

Finally, the developed method has several advantages over extant approaches. It can be applied to processes already in production and in real environments, so it involves real user behavior. At the same time, it clarifies measurement results, which are usually problematic in direct observations. Moreover, the measurement and analysis of aspects of usability can, in many cases, be arranged automatically or with only a little input, so costs are lowered, making the method attractive for use by small and medium-sized enterprises. The approach also significantly improves the overall UX approach by considered it as a broad and continuous lifecycle instead of a focused process with fixed start and end points.

References

Alves de Medeiros, A. K. (2006). *Genetic process mining*. Dissertation. Eindhoven: CIP-Data Library Technische Universität Eindhoven.

Dumas, M., Rosa, M. L., Mendling, J., & Reijers, H. A. (2013). *Fundamentals of business process management*. Berlin: Springer.

Ekanayake, C. C., Dumas, M., García-Bañuelos, L., & La Rosa, M. (2013). Slice, mine and dice: Complexity-aware automated discovery of business process models. In F. Daniel, J. Wang, & B. Weber (Eds.), *Business process management, LNCS 8094* (pp. 49–64). Berlin: Springer.

Evermann, J., & Assadipour, G. (2014). Big data meets process mining: Implementing the alpha algorithm with map-reduce. In *Paper Presented at the Proceedings of the 29th Annual ACM Symposium on Applied Computing*, Gyeongju, Republic of Korea.

Hilbert, D. M., & Redmiles, D. F. (2000). Extracting usability information from user interface events. *ACM Computing Surveys, 32*(4), 384–421.

Jagadeesh, R. P., & van der Aalst, W. M. P. (2010). Trace alignment in process mining: Opportunities for process diagnostics. In *Paper Presented at the Proceedings of the 8th International Conference on Business Process Management*, Hoboken, NJ, USA.

Melcher, J. (2012). *Process measurement in business process management – Theoretical framework and analysis of several aspects*. Karlsruhe: KIT Scientific Publishing.

Mendling, J., Sánchez-González, L., García, F., & La Rosa, M. (2012). Thresholds for error probability measures of business process models. *The Journal of Systems and Software, 85*(5), 1188–1197.

Siochi, A. C., & Ehrich, R. W. (1991). Computer analysis of user interfaces based on repetition in transcripts of user sessions. *ACM Transactions on Information Systems, 9*(4), 309–335. doi:10.1145/119311.119312.

Thaler, T. (2013). *Entwicklung einer Methode zum Process Mining unter besonderer Berücksichtigung von Organisationswissen* (p. 121). Master thesis. Best Diploma Award 2012 – Semiramis Research and Service Unit (SeReS Unit), Institut für Wirtschaftsinformatik im Deutschen Forschungszentrum für Künstliche Intelligenz (DFKI) GmbH.

Thaler, T. (2014). *Towards usability mining*. In Paper Presented at the Lecture Notes in Informatics. Jahrestagung der Gesellschaft für Informatik, Stuttgart.

Thaler, T., Fettke, P., & Loos, P. (2013). Process mining – Fallstudie leginda.De. *HMD Praxis der Wirtschaftsinformatik, 50*(5), 56–65.

Thaler, T., Maurer, D., De Angelis, V., Fettke, P., & Loos, P. (2015a). *Mining the usability of business process modeling tools: Concept and case study*. In Paper Presented at the 13th International Conference on Business Process Management, Industry Track (BPM 2015), Innsbruck, Austria.

Thaler, T., Maurer, D., De Angelis, V., Fettke, P., & Loos, P. (2015b). Mining the usability of business process modeling tools: Concept and case study. In *Paper Presented at the Proceedings of the Industry Track at the 13th International Conference on Business Process Management 2015*. Business Process Management (BPM-15), Innsbruck.

Thaler, T., Ternis, S., Fettke, P., & Loos, P. (2015c). *A comparative analysis of process instance cluster techniques*. In Paper Presented at the 12th International Conference on Wirtschaftsinformatik, Osnabrück, Germany.

van der Aalst, W. M. P. (2008). Decision support based on process mining. In F. Burstein & C. W. Holsapple (Eds.), *Handbook on decision support systems 1. Basic themes* (pp. 637–657). Berlin: Springer.

van der Aalst, W. (2011). *Process mining: Discovery, conformance and enchancement of business processes*. Berlin: Springer.

van der Aalst, W., Weijters, T., & Maruster, L. (2004). Workflow mining: Discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering, 16*(9), 1128–1142. doi:10.1109/TKDE.2004.47.

Weijters, A. J. M. M., & Ribeiro, J. T. S. (2011). *Flexible heuristics miner (FHM).* In Paper Presented at the IEEE Symposium on Computational Intelligence and Data Mining, Paris.
Weijters, A. J. M. M., van der Aalst, W. M. P., & de Medeiros, A. K. A. (2006). *Process mining with the heuristics miner-algorithm* (BETA Working Paper Series, WP 166).

Tom Thaler is Product Manager at Software AG and responsible for Business Process Modeling and Analysis with ARIS. He received his master degree in information systems from Saarland University, where he graduated with honors and received the Best Diploma Award from SERES Unit. He worked at META-LEVEL Software AG as a Software Developer and at SAP as a Business Intelligence Consultant. In 2012, Tom started working at the German Research Center for Artificial Intelligence (DFKI). He managed and worked in various research projects. He received the Outstanding Matcher Award 2013 in Beijing for his work on process model matching and developed an outstanding performing approach in the Process Model Matching Contest 2015. His work has been published in leading scientific conferences such as BPM and WI. In parallel to his Product Management position, he finalized his doctoral thesis in the area of Business Process Management.

Sabine Norek is Director User Experience at Software AG, Germany. She holds a master's degree in information science from Saarland University with minors in business administration and psychology. She works as a UX professional since 20 years in the areas of software development, web design, online media and consulting. Her special interest areas are user research, design thinking and lean UX.

Vittorio De Angelis is a Senior Lead Software Engineer at Software AG, Germany. He holds a Master degree in computer science from HTW Saarland with a focus on Business Information Systems. Next to his Software Engineering position, he works as a UX professional since over 10 years in the area of business software development and user interface design. As a full stack web developer, he has a long time passion in developing cutting edge enterprise web solution.

Dirk Maurer is development manager for the ARIS Design Platform products at Software AG.He studied electrical engineering and started with the company (formerly IDS Scheer) in 1995 as part of the ARIS development team. Along with ARIS Toolset version 4 he took over first project responsibilities for several core components. Starting in year 2000 Dirk has been in charge for the Java migration of the ARIS products that are sold till this day as ARIS Architect/ARIS Designer. In parallel he took over product management responsibility for the ARIS BPA and EA solutions. In the framework of the acquisition of IDS Scheer by Software AG Dirk became Vice President ARIS R&D. Amongst others he got the lead for an innovation program in order to enable ARIS products for the cloud and provide a new set of fully HTML5 based user interfaces, resulting in the current ARIS Connect products.

Peter Fettke is Professor of Business Informatics at Saarland University and is a principal researcher and a Research Fellow at the German Research Center for Artificial Intelligence (DFKI GmbH), Saarbrücken, Germany. He heads the Business Process Management Group at DFKI. His research interests embrace all facets of socio-technical processes, including process management, process modelling and process mining. He uses a broad spectrum of research methods comprising engineering methods/design science and empirical/experimental research approaches. His publications rank among the most-cited works of renowned journals of Business Informatics and he belongs to the top 10 of the most-cited researcher at DFKI. Furthermore, he is a highly-demanded reviewer for renowned conferences, journals and research institutions.

Peter Loos is director of the Institute for Information Systems (IWi) at the German Research Institute for Artificial Intelligence (DFKI) as well as full professor and head of the chair for Information Systems at Saarland University. His research activities include business process management, conceptual modeling and enterprise systems. During his earlier career, Peter had been professor at University of Mainz and at Chemnitz University of Technology. Furthermore, he had worked for 6 years as manager of the software development department at a software and consulting company. Peter has written several books and contributed to more than 50 books. He also published more than 30 journal articles and more than 300 articles in proceedings and papers.