phase as richer data evaluation techniques are available compared to paper based experiments (e.g., replaying the modeling process).

The remainder of this tool paper is structured as follows. Section 2 introduces a running example, which will be used in Section 3 for describing CEP. Finally, Section 4 concludes the paper with a summary and outlook on future work.

2 Example

To illustrate the functionalities of CEP, we introduce a typical experimental design as a running example (cf. Fig. 1). Let us assume that the goal of the experiment is to investigate whether secondary notations (cf. [9]), for example, layout of a process model has an influence on the quality of a change conducted on that process model. To investigate this question, the subjects (participants of the experiment) are divided into two groups. The first group is asked to conduct a change on a process model with good layout, whereas the second group has to perform the *same* change task on the *same* process model, this time with poor layout. As the subjects' modeling capabilities might differ and therefore influence their modeling performance, the research team wants to collect demographical data of each subject (e.g., experience in business process modeling). In addition, it should be ensured that the lacking knowledge about *how* to use the modeling tool does not influence the results, i.e., the impact of learning how to use the tool should be minimized. Consequently, the research team decides to include a process modeling tutorial in the experiment. Besides, the mental effort necessary for conducting the process change should be documented. For this, a survey on cognitive load should be presented to subjects.

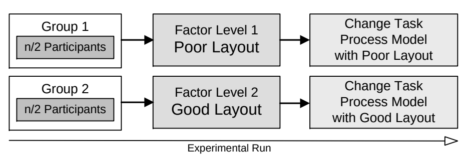

Fig. 1. Exemplary Experimental Design

3 Cheetah Experimental Platform

This section describes CEP. In particular, Section 3.1 illustrates how the platform can be used to support the design of experiments. Then, Section 3.2 deals with the actual operation of the experiment. Finally, Section 3.3 discusses how CEP fosters data analysis.

3.1 Experimental Design

Even tough the creation of experimental designs is a task highly relying on researcher's experience and domain knowledge, tool support can be beneficial in this phase. The majority of controlled experiments consists of a series of tasks that have to be executed by the experiment's subjects, referred to as *Experimental Workflow*. CEP enables experimenters to quickly assemble experimental workflows from components that have proven to work well in several experiments. In particular, CEP offers a set of frequently used components, including surveys, tutorials and *Cheetah Modeler* for creating business processes (cf. Section 3.2).

The exemplary experimental workflow described in Section 2 is supported by CEP as illustrated in Fig. 2. Depending on the number of different groups several branches are available in the experimental workflow configuration. At the beginning of the experiment, subjects are provided with assignment sheets containing an introductory text, instructions for performing the modeling tasks and a *group code*. Irrespective of the code the subjects entered, each participant has to fill out a demographic survey before working through an interactive tutorial. Based on the group code the respective branch of the experimental workflow is entered, presenting subjects with a change task for a process model with good/bad layout. Finally, participants are asked to fill out a survey about the cognitive load of the performed change task. All activities of the experimental workflow are handled using components provided by CEP.

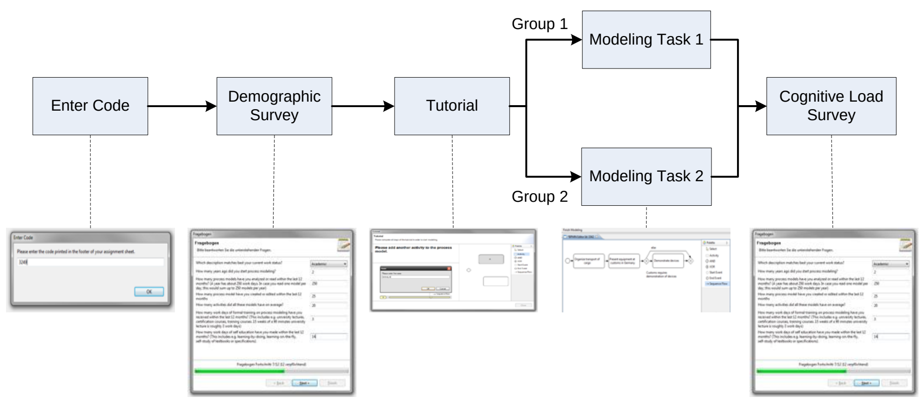

Fig. 2. Cheetah Experimental Workflow

3.2 Experimental Execution

Experimental Workflow When executing the experimental workflow configuration CEP guides the user through the experiment ensuring that the setup is followed. Furthermore, data collected when executing the experimental workflow is stored on a central database server, giving researchers the possibility to check whether all activities were completed and to restore the experiment to a specific state (e.g., in case of a crashed system). If the database server cannot be accessed a local copy is created and the user is asked to send it to the experiment's supervisor via email.

The experiment described in Section 2 is supported by CEP as follows. After entering the code identifying the group, the upcoming survey is collecting the user's demographic data. The survey ensures that all questions marked as mandatory are answered before the user continues with the next step in the experimental workflow. Before starting the actual modeling task the experimental workflow contains an interactive tutorial explaining the functionalities of Cheetah Modeler to make sure the used notation is well understood and participants know how to utilize the tool to change the process model. Therefore, each important functionality is presented by a screencast and users have to perform the corresponding modeling step. Depending on the entered code users are presented with process models with good/bad layout serving as a basis for the change task. Afterwards, a final survey assessing the mental effort for performing the change task is displayed.

Cheetah Modeler In order to enable the investigation of how process models are created, CEP offers Cheetah Modeler, which is a rather simple modeling component providing only basic modeling functionalities for simulating a pen and paper modeling session using a subset of BPMN (cf. Fig. 3). The focus was put on developing a tool facilitating the investigation of how process models are created, rather than providing a full fledged modeling suite. Currently, BPMN is the only process modeling language supported by CEP. Nevertheless, support for other notations was kept in mind when designing CEP and can easily be integrated.

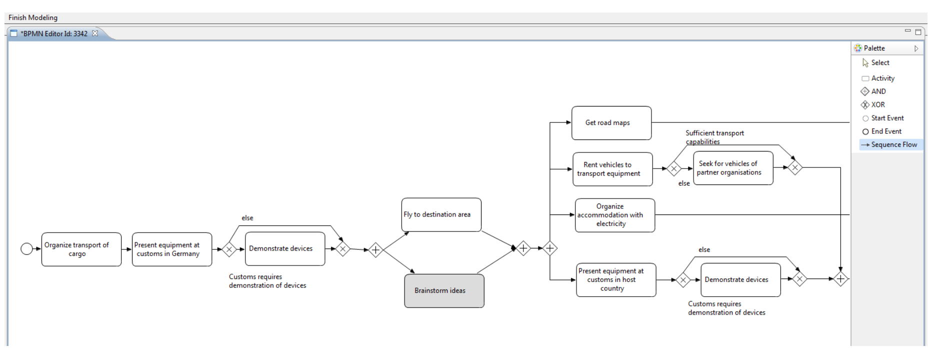

Fig. 3. Cheetah Modeler

Logging: Besides monitoring the experiment's correct execution and gathering the results of surveys, the collection of data on how users create process models was one of the main objectives when implementing Cheetah Modeler.

Consequently, every change to the process model (e.g., add/delete/move activity, add/delete/move edge) and the corresponding timestamp is automatically recorded and stored in a separate process log, offering the possibility for detailed investigations concerning the process of modeling (cf. Section 3.3).

3.3 Experimental Analysis

In addition to efficiently executing and monitoring experiments, data analysis was one of the main objectives when developing CEP. This section sketches the provided functionalities of *Cheetah Analyzer*, offering various data export features and means for replaying process models.

Experimental Workflow To be able to analyze data collected when executing the experimental workflow an export system is in place. By providing the option to export data as Comma-Separated Values (CSV) files, several tools for performing statistical analysis can be addressed (e.g., SPSS, Excel).

Process of Process Modeling One of the main advantages of using CEP is the possibility of replaying process models created with Cheetah Modeler. Recording all modeling steps enables researches to investigate *how* business process models are really created. For this purpose Cheetah Analyzer was implemented allowing for a step by step execution of modeling processes (cf. Fig. 4). Additionally, researches can export modeling processes using the Mining XML (MXML) format, allowing them to apply process mining techniques using ProM [10].

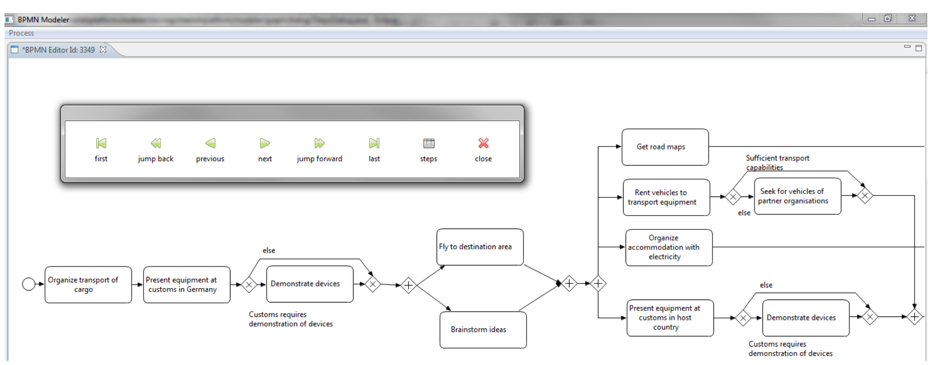

Fig. 4. Cheetah Analyzer

In context of the experiment presented in Section 2 researchers can have a detailed look on *how* the given process models were changed and if the layout had an influence on the change process. For example, it might be possible that users presented with a bad process layout rearranged activities before performing the actual change.

4 Summary and Outlook

Cheetah Experimental Platform, described in this tool paper, supports researches in conducting controlled experiments on business process modeling. In particular, CEP provides a repository of typical components (e.g., surveys, tutorials, process modeling tools) which can be used for assembling experimental workflows. Furthermore, the risk of producing invalid data is mitigated as the user is guided throughout the experiment's execution, reducing the number of accidental errors. In addition, richer analysis of data is possible compared to paper based experiments.

Future developments include a graphical experimental workflow and survey builder to further facilitate the creation of experimental designs as well as a dashboard simplifying the supervision of experiments. Furthermore, we would like to investigate the influence of collaborative modeling on *how* process models are created. For this purpose, CEP is currently extended toward collaborative modeling support.

Acknowledgements: We thank Dirk Fahland, Jan Mendling, Hajo A. Reijers, Matthias Weidlich and Werner Wild for their much appreciated feedback when developing Cheetah Experimental Platform.

References

1. Antonucci, Y.L.: Using workflow technologies to improve organizational competitiveness. Int'l. J. of Management **14** (1997) 117–126
2. B. Weber, B.M., Reichert, M.: Investigating the effort of using business process management technology: Results from a controlled experiment. Science of Computer Programming **75** (2010) 292–310
3. Myers, G.J.: A controlled experiment in program testing and code walkthroughs/inspections. Commun. ACM **21** (1978) 760–768
4. Lott, C.M., Rombach, H.D.: Repeatable Software Engineering Experiments for Comparing Defect-Detection Techniques. Empirical Software Engineering **1** (1996) 241–277
5. Weber, B., Reijers, H.A., Zugal, S., Wild, W.: The declarative approach to business process execution: An empirical test. In: CAiSE. (2009) 470–485
6. Pinggera, J., Zugal, S., Weber, B.: Alaska simulator supporting empirical evaluation of process flexibility. In: WETICE. (2009) 231–233
7. Zugal, S.: Agile versus Plan-Driven Approaches to Planning - A Controlled Experiment. Master's thesis, University of Innsbruck (2008)
8. Pinggera, J.: Handling Uncertainty in Software Projects A Controlled Experiment. Master's thesis, University of Innsbruck (2009)
9. Green, T.R.: Cognitive dimensions of notations. In: Proc. BCSHCI '89. (1989) 443–460
10. van der Aalst, W.M.P., Reijers, H.A., Weijters, A.J.M.M., van Dongen, B.F., de Medeiros, A.K.A., Song, M., Verbeek, H.M.W.E.: Business process mining: An industrial application. Inf. Syst. **32** (2007) 713–732