

WoPeD - An Educational Tool for Workflow Nets

Thomas Freytag, Cooperative State University (DHBW) Karlsruhe, Germany
freytag@dhbw-karlsruhe.de

Martin Sänger, 1&1 Internet AG, Karlsruhe, Germany
m.saenger09@web.de

Abstract

WoPeD stands for **W**orkflow **P**etrinet **D**esigner and is an open-source software distributed over the Sourceforge¹ platform. WoPeD's focus lies on verification, visualization and explanation. Over the years, WoPeD has become a widely-used tool in the academic sector. This demo will provide information about recently added and improved features and functions. The demo will show how to use WoPeD as an e-learning tool to get a better understanding of fundamental BPM concepts².

Keywords: Workflow nets, simulation, explanation, visualization, e-learning, open-source.

1 About WoPeD

The development of WoPeD started in 2003. Since then, the software has been extended continuously, leading to the current version 3.2.0 released in June 2014. WoPeD has been downloaded more than 45,000 times since the initial release and has been successfully utilized in lectures, projects, case studies and scientific publications in many universities worldwide. WoPeD covers a variety of BPM aspects. Firstly, it provides a graphical editor for standard Petri nets and workflow nets including an interactive simulator. Secondly, it supports model verification and property visualization, e. g. has built-in algorithms for soundness, process metrics, and quantitative simulation. Thirdly, it allows the transformation from and to other common model formats like BPMN. Overall, WoPeD has proved to be a handy and easy-to-use e-learning tool mainly for the academic sector allowing students and practitioners easy access, hands-on experience and fast acquisition of the major concepts of business process modeling and verification.

2 Functional Features of WoPeD

Process and Resource Editor

In the WoPeD editor, the user can to draw, annotate, manage and export regular place-transition Petri nets as well as workflow (WF) nets according to the *van der Aalst* notation as introduced in [1]. The editor supports hierarchical sub-processes as transition refinements, enabling WoPeD to handle also large

¹www.sourceforge.org

²Copyright © 2014 for this paper by its authors. Copying permitted for private and academic purposes.

process models. A layout beautifier is provided to re-arrange the process graph in a human-readable representation. Moreover, WoPeD contains a graphical resource model editor covering the organization view of BPM, by defining resource objects, roles, and groups.

PNML Compliance and Import/Export Formats

The underlying file format complies to standard PNML, allowing models created by WoPeD to be exchanged with other PNML-aware tools. WoPeD saves its “proprietary” data into a separate part of the PNML stream such that third-party tools can access the basic control structure and skip all WoPeD-internal extensions. Additionally, WoPeD can export to graphical formats (PNG, JPEG) and (experimentally) to BPEL[4].

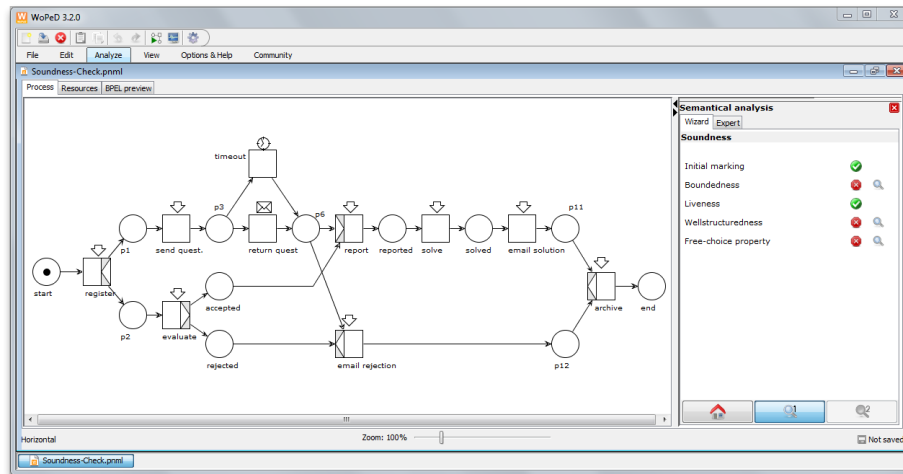


Figure 1: Soundness wizard

Soundness Checker

WoPeD contains an implementation of the soundness algorithm, including contributing structural and behavioural properties like free-choice, S-coverability, and PT/TP handles. The user can choose to run these checks in *wizard mode*, where a visual explanation engine guides him through all verification steps intuitively, or alternatively in *expert mode* showing a compact, tree-like property structure. Figure 1 shows a WF net process model and an analysis session in wizard mode. In both modes, a high focus is laid to visualizing a soundness failure by interactively referring to the associated editor elements.

Interactive Simulator and Coverability Graph Visualization

WoPeD provides a highly intuitive token game simulator allowing to navigate through Petri net executions stepwise or automatically. For sub-processes, there exists a *step-into/step-over* logic similar to an interactive source code debugger. WoPeD contains algorithms to construct the coverability graph of a Petri net, allowing various behaviour-related property checks. Additionally, the coverability graph can be displayed, showing the behaviour of the associated Petri net and allowing manual inspection and navigation, e. g. to study and understand the concepts of occurrence rule, marking and reachability visually.

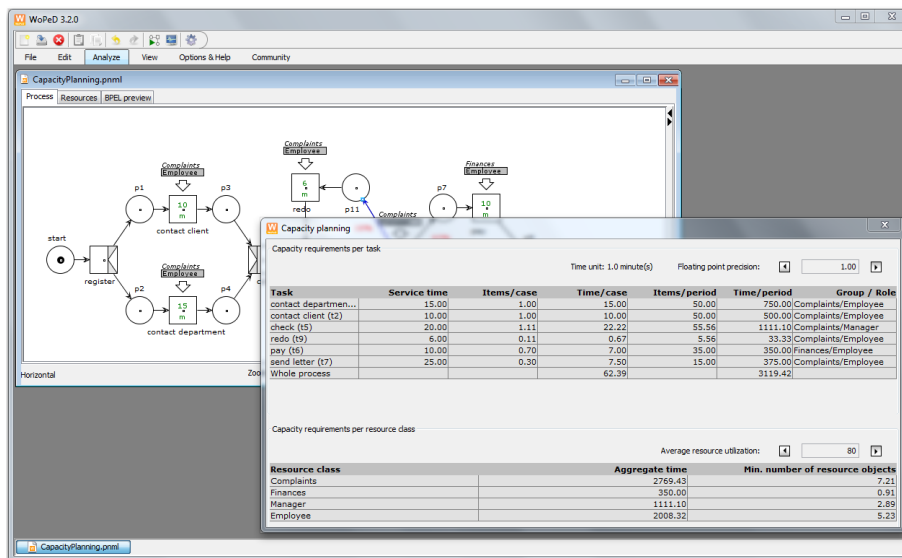


Figure 2: Capacity planning results

Quantitative Simulation and Capacity Planning

WoPeD supports also quantitative simulations. A built-in resource editor allows for each process the definition of a resource model, i. e. roles, groups, and objects. Additionally, transitions can then have resource class assignments. Two different algorithms are implemented, following the suggestions of [1], both based on stochastic distributions of arrival rate, task service times and XOR branching probabilities: Firstly, a quantitative simulator engine for the random construction of execution traces, allowing to derive quantitative properties like average waiting or completion time. Secondly, a capacity planner in order to calculate the optimal number of resource objects for each resource class. Figure 2 shows a capacity planning result based on quantitative parametrization of a process model.

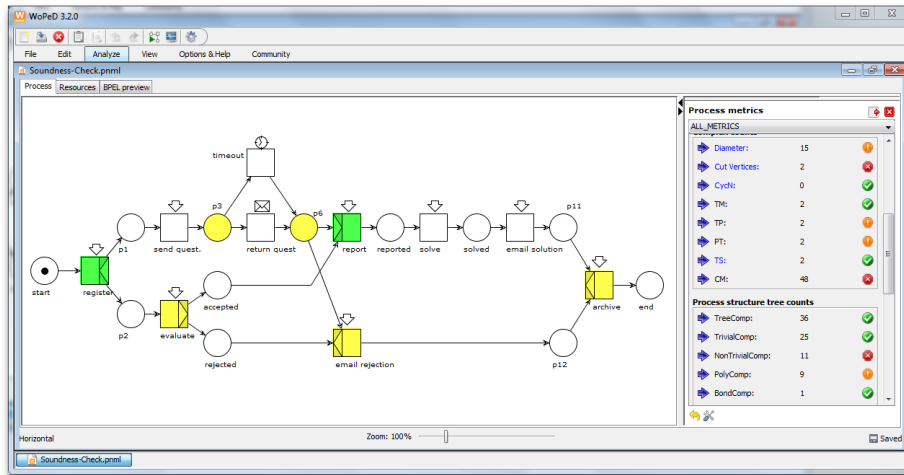


Figure 3: Operator coloring and process metrics

Research Support Tool

WoPeD has been used as a *proof-of-concept* tool for several research ideas in the BPM community. Firstly, WoPeD can detect so called *matching operator pairs*, using a coloring scheme to visualize corresponding split and join transitions in the style of syntax highlighting in programming editors. A summary of these research results can be found in [6], the implementation aspects were discussed in [4]. Secondly, a component was developed within WoPeD to edit and visualize process metrics and their relation to the process graph [5]. The algorithms make use of a built-in variant of BPStruct [3, 7]. Figure 3 shows WoPeD with enabled operator coloring and a list of process metric values and their visual classification on the sidebar.

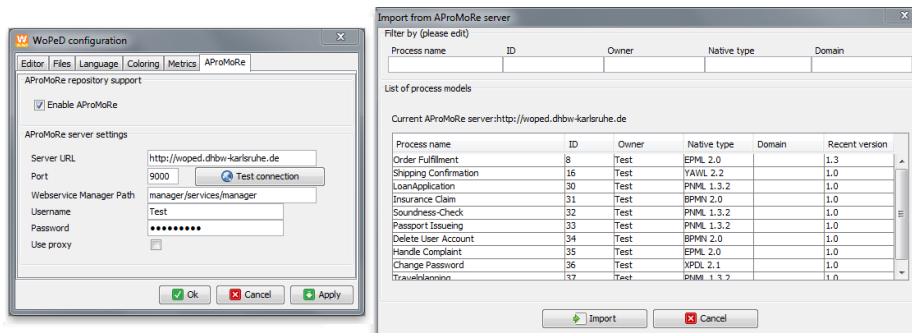


Figure 4: AProMoRe configuration panel and import dialog

AProMoRe Repository Front-End

AProMoRe is another open source software developed at QUT Brisbane [2]. It provides roughly a set of conversion algorithms between most common process model formats, e. g. BPMN, EPML or YAWL. The most recent feature of WoPeD is an interface to access documents stored in an AProMoRe repository. It is possible to import any type of process model from AProMoRe (no matter what its native format is) into WoPeD and also to export a PNML process from WoPeD to AProMoRe. This allows to make use of WoPeDs analysis or visualization functionality also to process models not created with WoPeD and even in different process model formats. Figure 4 shows the configuration panel and the import dialog to load a process model into WoPeD.

3 Availability of WoPeD

WoPeD can be downloaded on its homepage [8] or directly from Sourceforge. The code is open source and is published under the LGPL license. WoPeD is written in Java. For convenience, there exist three different installer packages for the major platforms Windows, Linux and Mac OS.

References

- [1] W. M. P. van der Aalst, K M. van Hee: *Workflow Management: Models, Methods, and Systems* MIT Press 2002.
- [2] AProMoRe website: www.apomore.org, accessed in May 2014.
- [3] BPStruct. www.sep.cs.ut.ee/Main/bpstruct, accessed in May 2014.
- [4] A. Eckleder, T. Freytag, J. Mendling, H.A. Reijers: Realtime Detection and Coloring of Matching Operator Nodes in Workflow Nets. In T. Freytag and A. Eckleder (ed.), *16th German Workshop on Algorithms and Tools for Petri Nets*, volume 501 of CEUR Workshop Proceedings, pages 56-61, 2009.
- [4] A. Eckleder and T. Freytag: *WoPeD 2.0 goes BPEL 2.0*. AWPN 2008 - Research Report, University of Rostock, 2008.
- [5] J. Mendling: *Metrics for Process Models*, Springer Verlag, Berlin, 2008.
- [6] H. A. Reijers, T. Freytag, J. Mendling, A. Eckleder: Syntax highlighting in business process models. *Decision Support Systems* 51(3): 339-349 (2011).
- [7] M. Weidlich, A. Polyvyanyy, J. Mendling, and M. Weske. *Petri Nets*, volume 6128 of *Lecture Notes in Computer Science*, pages 63– 83. Springer, 2010. In: LNCS 6128, *Petri nets*, pages 63– 83. Springer, 2010.
- [8] WoPeD website: www.woped.org, accessed in May 2014.