The DCR graphs process portal

Debois, Søren; Hildebrandt, Thomas; Marquard, Morten; Slaats, Tijs

Published in:
Proceedings - BPM 2016 Demonstration Track

Publication date:
2017

Document version
Publisher's PDF, also known as Version of record

The DCR Graphs Process Portal

Søren Debois[1], Thomas Hildebrandt[1], Morten Marquard[2], and Tijs Slaats[3] *

[1] IT University of Copenhagen, Denmark
{debois, hilde}@itu.dk, http://www.itu.dk
[2] Exformatics A/S, Denmark
{mmq}@exformatics.com, http://www.exformatics.com
[3] Department of Computer Science, University of Copenhagen, Denmark
{slaats}@di.ku.dk, http://www.diku.dk/

Abstract. We demonstrate the dcrgraphs.net Process Portal: a cloud-based solution for collaborative continuous development and analysis of knowledge intensive processes. This tool is the result of a long-term collaboration between Exformatics A/S, a Danish provider of Adaptive Case Management (ACM) solutions, and researchers from IT University of Copenhagen (ITU) and the Department of Computer Science at University of Copenhagen (DIKU). The tool draws heavily on current research into declarative process modelling notations: it is built upon the declarative Dynamic Condition Response (DCR) Graphs notation. In our demonstration we will introduce the primary features of the tool: (1) a portal for creating, storing and sharing processes, (2) a declarative process designer, (3) a collaborative simulator, and (4) process analysis modules. The intended audience are researchers and practitioners interested in process modelling notations and techniques for knowledge workflow management and similarly flexible process environments.

Introduction In this demonstration we present dcrgraphs.net, a cloud-based commercial solution for modelling, simulating and analysing declarative process models, built and supported commercially by Exformatics A/S. The solution is free for academic use.

The core philosophy of the tool is that the declarative and imperative modelling paradigms are solving fundamentally distinct problems, and therefore tooling around them must be similarly distinct. Declarative modelling notations, such as Declare [6] and Dynamic Condition Response (DCR) Graphs [2, 4, 8] achieve flexibility by encoding a multitude of potential executions of a workflow into a set of rules. This means on the one hand that a single model embodies potentially orders of magnitudes more potential workflow executions—more flexibility!—than imperative notations such as BPMN [5], but on the other that the modeller has to account for the increased complexity that comes with these many different possible executions. Thus, declarative models

* Authors listed alphabetically. This work is supported in part by the Hybrid Business Process Management Technologies project (DFF 6111-00337, 2016-2019) and the Computational Artifacts project (VELUX 33295, 2014-2017)

buy flexibility at the cost of complexity. A declarative model is not as readily understood from "looking at the model" [7] as is a model in an imperative notation such as BPMN [5], but this is to be expected: imperative models express less complexity. They solve a different problem than declarative models.

The tool is grounded in the belief that flexibility is a necessity for Adaptive Case Management, and that this complexity is a necessary evil of that flexibility. The only way forward is to aid the modeller in creating and understanding his model. *Providing this support is the core aim of the* `dcrgraphs.net` *solution.*

The portal, its main features, and the underlying DCR Graph notation were reported on in [3]. The formal semantics of the notation and some of its primary extensions are described in [2, 4, 8].

In the present paper we describe the main features of the portal and how they help curb complexity:

1. The main portal, which acts as a social repository for declarative process models in which models can be shared and discussed with other users.
2. The process designer, which enables the user to model their processes.
3. The simulator, which supports both individual and collaborative simulation
4. Analysis tools, which allow the user to understand model behaviour.

At the live demonstration we will familiarise the audience with each of these features by creating, modelling, simulating and analysing an example model. Audience members will be invited to participate in collaborative aspects from their own devices.

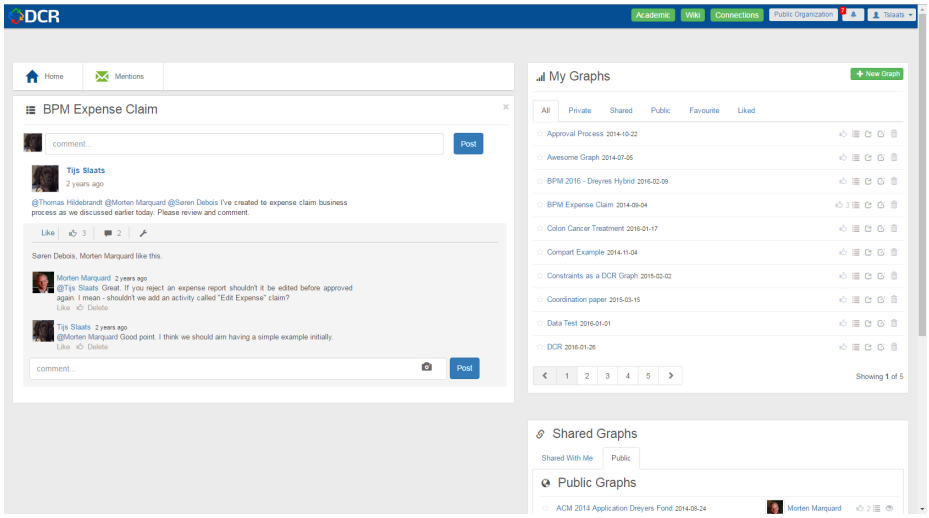

Fig. 1. The Process Portal

Process Portal Figure 1 shows the main portal. The right half of the screen is used to display lists of personal, shared and public process models. The left half of the screen

displays *activity streams*, which can be used to discuss with other users. Users can add colleagues and friends to their personal network, which allows them to share models and join collaborative simulations.

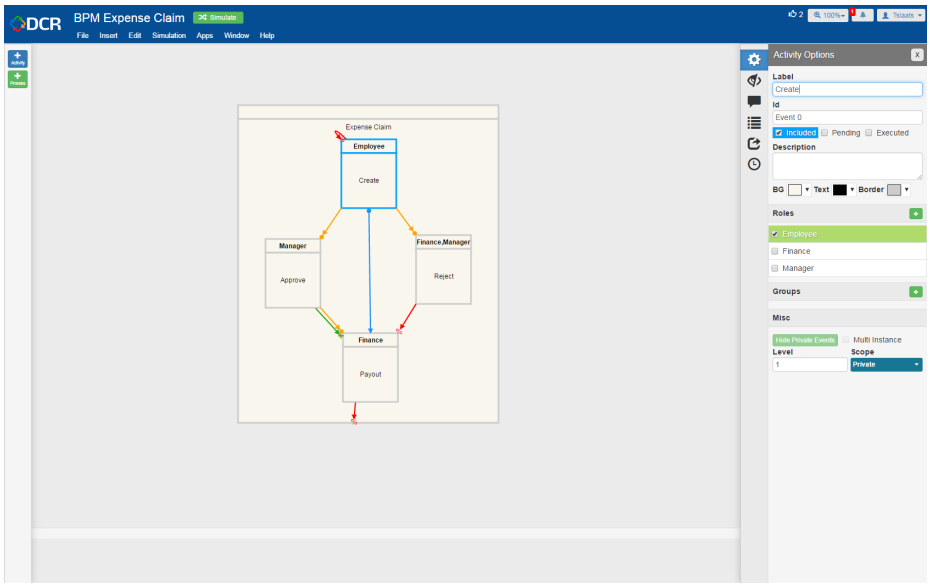

Fig. 2. The Process Designer

Process Designer Figure 2 shows the process designer. At the centre is the process model, consisting of boxes representing activities and relations between the boxes that represent the rules governing the process. Clicking on activities and rules brings up an options pane which allows one to customise that element. To facilitate the understandability of process models a number of visual filters are offered by the designer: activities can be assigned a *level of abstraction* at which they are shown, meaning that the users can start to view the process at the highest level of abstraction, showing only a few activities and then gradually introduce more details by lowering the level of abstraction. In addition activities can be assigned to any number of *user-defined groups*, which e.g. allows the user to view only those activities that are relevant to a particular organisational unit. The designer also supports fine-grained revision tracking, which allows any change to the model to be inspected and reverted. The designer can import and export DCR Graphs in a standardised XML format which is used by a number of external tools.

Collaborative Simulation Many processes involve a collaboration between multiple actors, often in different organisations. For the successful implementation of such processes it is imperative that the end-users understand their own role. To facilitate this the DCR Portal supports both individual and collaborative simulation, which can be started

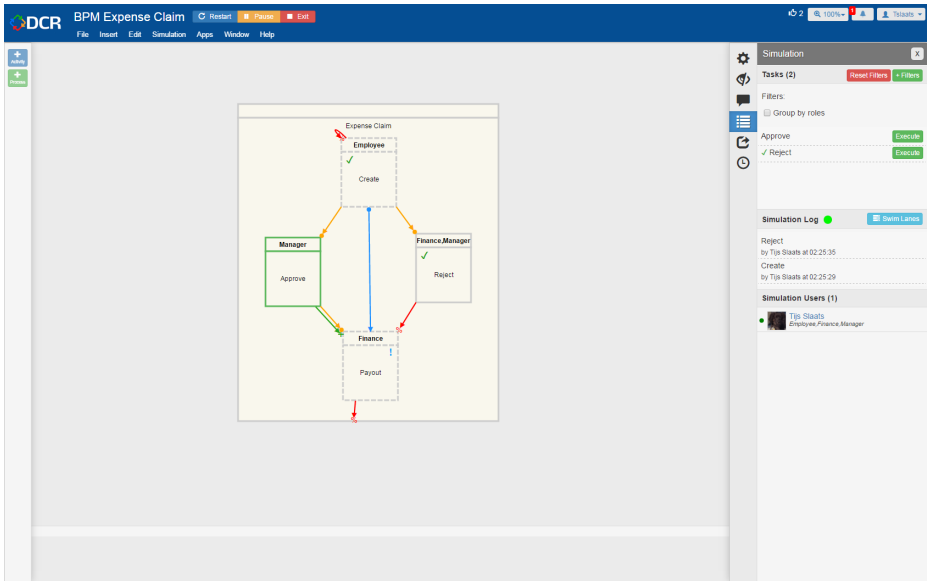

Fig. 3. The Simulation Window

from the designer. Here the user can invite connections to their simulation and assign roles to them. Figure 3 shows the simulation window; on the left one can see the current DCR Graph, with its marking updated according to the actions taken by the users. On the right the user is presented with a task list, the log of previously executed tasks and a list of participants.

Analysis Tools The DCR Portal offers a flexible plug-in framework which can be used by third parties to create additional functionality for the portal. As a proof-of-concept several analysis tools have been developed as plug-ins. The first is the *path analyser*, essentially a GPS for processes: the user can enter a start and goal activity and the analyser will determine the shortest possible execution path to get to the goal. Typical GPS-like functions, such as intermediate activities or activities to avoid are also supported. The other plug-in is the *dead-end analyser*, which can be used to detect dead- and live-lock in processes. Figure 4 shows an example of the path analyser plug-in. Here the user has asked for a path from *Create* to *Payout*, using the activity *Reject* at least once. The proposed path is drawn as a swimlane diagram in the bottom window.

Maturity, Availability, Documentation and Video Tutorials Altogether, the development of DCR Graph technologies and tools started in 2010 [9]. The development of the portal in its present form [3] started in the spring of 2014 and the first official release was in the autumn of the same year. It has since been continuously updated.

Since the spring of 2015 the portal has been used in classes at IT university of Copenhagen, most prominently a process modelling class for 75 bachelor and M.Sc. students. The portal has also been used for teaching at the Federal University of the

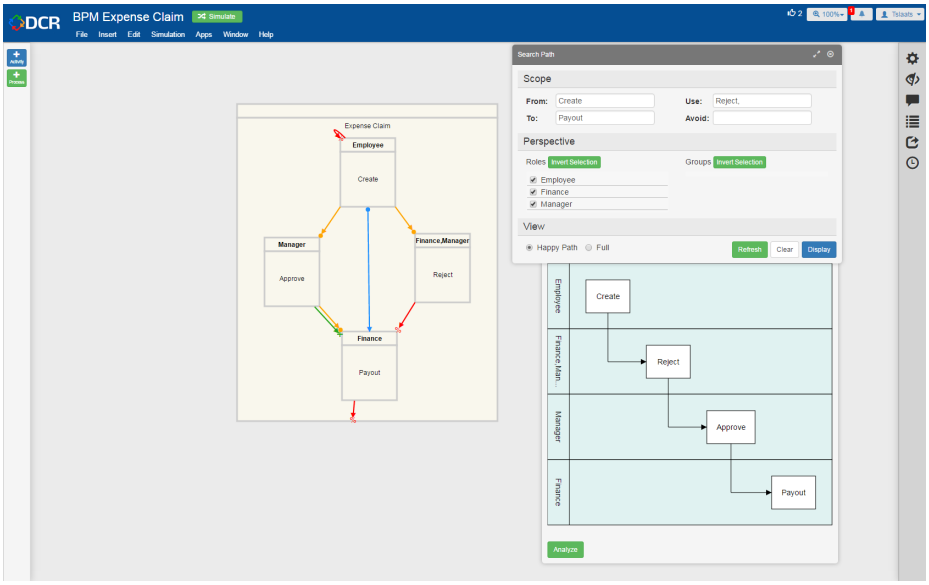

Fig. 4. Process Analysis

State of Rio de Janeiro. Exformatics uses the portal to model and maintain the processes of their customers, in particular a Danish foundation [1], whose Adaptive Case Management solution uses a declarative process engine for executing its processes.

The portal is accessible for free online at `http://www.dcrgraphs.net/`. Documentation is available through a wiki at `http://wiki.dcrgraphs.net/`, which contains video tutorials explaining how to use the different features of the portal.

References

1. S. Debois, T. Hildebrandt, M. Marquard, and T. Slaats. A case for declarative process modelling: Agile development of a grant application system. In *AdaptiveCM '14*.
2. S. Debois, T. Hildebrandt, and T. Slaats. Hierarchical declarative modelling with refinement and sub-processes. In *BPM '14*, volume 8659 of *LNCS*, pages 18–33.
3. M. Marquard, M. Shahzad, and T. Slaats. Web-based modelling and collaborative simulation of declarative processes. In *BPM '15*.
4. R. R. Mukkamala. *A Formal Model For Declarative Workflows - Dynamic Condition Response Graphs*. PhD thesis, IT University of Copenhagen, March 2012.
5. OMG BPMN Technical Committee. Business Process Model and Notation, version 2.0.
6. M. Pesic, H. Schonenberg, and W.M.P. van der Aalst. DECLARE: Full Support for Loosely-Structured Processes. In *EDOC '07*, pages 287–. IEEE.
7. P. Pichler, B. Weber, S. Zugal, J. Pinggera, J. Mendling, and H. A. Reijers. Imperative versus declarative process modeling languages: An empirical investigation. In *Business Process Management Workshops*, volume 99 of *LNBIP*, pages 383–394. Springer, 2012.
8. T. Slaats. *Flexible Process Notations for Cross-organizational Case Management Systems*. PhD thesis, IT University of Copenhagen, January 2015.
9. T. Slaats, R. R. Mukkamala, T. Hildebrandt, and M. Marquard. Exformatics declarative case management workflows as dcr graphs. In *BPM '13*, volume 8094 of *LNCS*, pages 339–354.