The Process Highlighter
From Texts to Declarative Processes and Back

Hugo A. López
Søren Debois
Thomas T. Hildebrandt
Morten Marquard
The Process Highlighter
From Texts to Declarative Processes and Back

Hugo A. L´opez
Søren Debois
Thomas T. Hildebrandt
Morten Marquard

Abstract

The adoption of formal models by process specialists has faced two challenges: First, it requires process specialists to get training in formal modeling. Second, the resulting specifications bear little resemblance wrt. the original descriptions. We introduce a tool that supports translations between natural language descriptions and declarative process models. The resulting models are given in a graphical formalism, DCR Graphs. Traceability is at the core of the tool: Later changes in the process model due to, e.g., ambiguity resolution are traced back into the text. This allows users to either correct and complete their descriptions, or to derive models more refined than the text. In this paper, we describe the mechanics of the tool and provide examples of its use. Finally, we report on experiences using the tool in a Danish Municipal government.

1 Introduction

This paper describes an implementation of tool support for a method for extracting declarative process models from textual process descriptions. The tool aims at reducing the gap between textual descriptions of processes and their interpretations in a declarative process model. The tool is particularly aimed at supporting users who are not trained in formal methods.

Both the tool and the method exploit the regularity of contemporary constraint-based process notations with trace semantics such as DECLARE [2, 10] or DCR graphs [9, 5]: That constraints are expressed as relations between a fixed number of activities. This regularity is actually present and respected already in textual descriptions of processes. For instance, suppose our text says:

"Payout requires pre-approval by the line manager" (1)

Then activities “Pre-approval” and “Payout” are related by a DCR or DECLARE condition relation. Thus, there is a correspondence between the constraint in the declarative process model on the one hand and this particular sentence on the other. The highlighter tool reported on here helps to establish and maintain that correspondence. It allows users to “mark up” a textual process description, as if with a yellow marker; the tool then generates a formal declarative DCR process model from that markup. The correspondence between model and text is bi-directional: later changes to the model may change the markup and vice versa.

The purpose of the tool is to allow domain experts with some familiarity with the modelling notation, to easily construct and maintain declarative models. The tool is currently being used at both industrial and academic environments, and user feedback indicates that both the development method and the tool are effective in establishing and maintaining a correspondences between a textual process descriptions and a formal constraint-based process models.
2 Example Usage

We illustrate the tool via an example process description from the BPM Academic Initiative, presented in Fig. 1:

The process includes two major roles, agents (supporting customers outdoor) and clerks (work indoors). When the insurance company receives a new claim, the clerk calls the agent to actually check the claim, and creates a new case. As both tasks are executed by different roles (that are mapped to different people), the activities are scheduled in parallel. After the agent has confirmed the claim to the clerk, he supports the customer with additional assistance (e.g. getting a new id-card from the public authority). After the clerk has received the confirmation from the agent, she issues a money order for the claim. If the agent has completed his additional support and the clerk has issued the money order, the claim is closed.

Figure 1: Insurance process description

DCR Graphs are the formal foundations of the process engine developed by DCR Solutions. For sake of space, we do not provide formal definitions, referring the reader to [9, 6] instead. As the name suggests, a DCR graph is a (directed multi-)graph: nodes of the graph are activities in a process, and edges denote constraints between activities.

Activities are there to be executed, and relations regulate the state and executability of activities:

- A condition relation from $A \rightarrow B$ means that $B$ can only execute if $A$ already has been executed, or has been excluded (see below).
- A response relation $A \rightarrow B$ indicates that whenever $A$ executes, $B$ changes state to pending, that is, it needs further execution (or exclusion) of $B$ for the workflow to be complete.
- An exclusion (resp. inclusion) relation $A \rightarrow \% B$ (resp. $A \rightarrow + B$) indicates that whenever $A$ is executed, $B$ is excluded from (resp. included in) the workflow: An excluded activity cannot execute and is ignored as condition and not required to execute if pending, unless it is re-included by an include relation.

From texts to processes. We start by creating process models from texts. Fig. 1 contains three explicit roles (agent, clerk, customer) and one implicit role (insurance company). Using the tool, we mark the text of each role and mark it up as such.

We proceed by identifying activities. For instance, fragments “confirm the claim” and “support customer” are marked as activities in the tool (see top of Fig 2). Changes in the title of each activity to improve readability are supported, e.g., “confirm the claim” vs. “confirmed the claim”.

We proceed to find relations. The word “After” in line 5 of Fig. 1 suggests causality between agents’ activities “confirm claim” and “support customer”. When marking up “After” as a relation, the tool allows setting the type of relation, as well as which activities are related. Such mark-up creates a relation in the graph (a condition arrow). Fig. 2 shows the highlighter in action in an excerpt of Fig. 1; we see the text fragment at the top, and (part of) the generated graph at the bottom. The full mapping is shown in Fig. 3.

Advanced use. Constraints may be explicit (e.g. “after”, “and”, “if-then”), or implicit (i.e.: the first highlighted comma). Other aspects may hinder the mapping between texts and DCR graphs:

i. A sentence might denote the refinement of an activity. i.e.: “The evaluation process might contain evaluations regarding chronic and lifelong conditions of the patient”. Here, activity (evaluation) needs to be refined into several (separate assessments for lifelong and for chronic diseases).

ii. A sentence might denote internal activities (performed by a single role) as well as communication activities (i.e.: lines 2–3 in Fig. 1).

iii. Elements identified in textual paragraphs might be referenced later.

iv. Relations might include one-to-one, many-to-one and one-to-many cardinality between activities.

The tool supports these scenarios. First, multiple markings of the same text will denote groups (abstractions) and refinements (nested activities). Second, multiple highlights denote communication activities, one role per marked text. Third, the tool allows the creation of references to existing elements. Finally, the creation of relations within the text allows users to include more than one relation per marked text, with possibly different actors and types involved.

2.1 . . . “and back”: Support for Model Changes

In practice, both models and texts may evolve over time. These changes do not break correspondences with the model. The filter view (c.f.: Fig. 3) shows the mapping between components and highlighted texts. Elements in the model that do not have a mapping to the text are shown in red. In the example, those elements are the new activities that refine additional support. This distinction helps users identify deviations from the model, and to correct them either by linking the model components to the text, or by updating the text description.

3 Concluding Remarks

Despite a raising interest in the alignment of process models and textual descriptions [8, 7, 11], authors are not aware of other works exploring such alignment with declarative process models. At its current stage, the highlighter depends
on the knowledge of the process modeler to resolve inherent ambiguities coming from the interpretation of natural
language descriptions. In future work, we would like to explore (semi)-automated declarative process discovery from
texts, via the integration of the highlighter tool with NLP techniques [8].

The highlighter tool has been validated in industrial and academic environments. First, the tool has been introduced
to a small number of end-users from the Syddjurs municipality in Denmark, who have used it to build models
corresponding to the municipal governments internal guidelines on processes for implementing Section 42 of the Danish
Consolidation Act on Social Services, which describe rights to compensation for loss of earnings awarded to parents
of children with long-term illness or disabilities. Users had a background in law and social work, and had previously
received training in process modelling using DCR graphs. Users’ interaction with the tool resulted in their independent
process models [2]. End-users reported that the highlighter was effective and easy to use for the intended purpose [3]. Finally,
the highlighter tool is currently being used in B.Sc. and M.Sc. courses on business process modelling at the ITU.

3.1 Availability, Documentation and Screencast

The tool has been implemented by DCR Solutions A/S as part of their set of tools for declarative process modelling and
execution, and is part of their commercial on-line offering [4]. The highlighter tool is free for non-commercial and
academic use. Documentation, installation and further examples of use are available in our wiki site [4]. A screencast
documenting its usage is available at https://youtu.be/JB9ueRu_asE.

Acknowledgments. Work supported by the Innovation Fund Denmark project EcoKnow.org (7050-00034A). This
project has received funding from the European Union’s Horizon 2020 research and innovation programme under the
Marie Sklodowska-Curie grant agreement BehAPI No.778233. We thank Nicklas Healy for providing industrial tests of
the tool, and to Muhammad Shazhad for providing programming support of the tool.

References

[1] Bekendtgørelse af lov om social service (Aug 2017), Børne- og Socialministeriet
(2017)
482–496. Springer (2011)

[3]However, they did report that some aspects of the guidelines, e.g., the criteria an assessment of a child’s special needs must meet to be lawful,
were not straightforward to map to the BPM concepts of “Activity”. This concern is orthogonal to the tool itself: such difficulty would manifest also
in a pen-and-paper model.
A Example catalog

A list of case studies has been compiled in addition to the working example:

<table>
<thead>
<tr>
<th>Description</th>
<th>Source</th>
<th>DCR Graph</th>
</tr>
</thead>
<tbody>
<tr>
<td>Danish Consolidation Act on Social Services (excerpts)</td>
<td><a href="https://danskelove.dk/serviceloven/42">https://danskelove.dk/serviceloven/42</a></td>
<td><a href="http://www.dcrgraphs.net/Tool?id=7079">http://www.dcrgraphs.net/Tool?id=7079</a></td>
</tr>
</tbody>
</table>

B Industrial test

The Syddjurs Municipality tested the tool via the construction of models corresponding to the municipal governments internal guidelines on processes for implementing Section 42 of the Danish Consolidation Act on Social Services [1], which describe rights to compensation for lost of earnings awarded to parents of children with long-term illness or disabilities. The model is publicly available at: [http://www.dcrgraphs.net/Tool?id=7119](http://www.dcrgraphs.net/Tool?id=7119).

C Obtaining a license

No license needs to be obtained to test the highlighter tool. In order to test the tool a user in the DCR graphs portal[5] needs to be created. The portal details a (rather standard) procedure required to create a user. The creation of new accounts requires a valid email address. Reviewers of this paper are welcome to create anonymous emails to preserve the anonymity of their reviews.

The highlighter app is provided as a plugin to the DCR portal. Further documentation on how to install the tool and its usage can be retrieved at [3].

D User Manual

WP 3 – Highlighter App Documentation

Introduction
Highlighter app in DCR is a plugin through which user can design graphs just by highlighting the text. Using this app user can create Activities and roles and can also create relationships between activities. Initially the text shown in the app would be the one available in graph’s description, but user can also update the text within the highlighter app window.

App can be enabled from App Store, as shown in below image

Once the app is enabled, it will be available in Apps section on tool page

When user clicks on the app, it will be opened in a new window as shown below
Working of the App

The idea of highlighter app is to create activities, roles and relations just by simply highlighting the text. When user opens the highlighter app the text available in graph description will be available in it, however user can update the text within the app using text edit button.

When the user selects the text then a list will be opened with three options, whether to create an Activity or Role or Relation with the selected text.

Whatever user chooses, that item will be created in the graph and will also be listed down in the highlighter app.
On the left side of each item in the highlighter app, flags are added to identify the type of the item (Activity, Role or Relation). User can also use filters to see specific type of items or show all.

Activities

Create Activities:
When user choose to create an activity then an activity with the selected text will be created in the graph, the created activity will be shown on canvas as well as will be listed within the highlighter app, the label and description of the created activity will be the text highlighted. Once an activity is created with the text then the respected text will be marked as light green.

Multiple activities of same text can also be created, in this case text marking will go on darker in color. User can create as many activities as he wants to, even using the very same text.
Select Activities:
If we select some activity from canvas then that activity will also get selected in the highlighter app, the blue bar beside the activity shows the selection. And also if we select some activity from within the app then that activity will get selected on the canvas. In both the cases text on the right section of the app will also get highlighted with a very minor visual effect.

Delete Activities:
Activities can be deleted from the app by using the delete button with the activity name. When user choose to delete an activity then that activity will no longer exists in graph as well as in app and also the marking of the text will be removed.
Mapping Un-Mapped Activities:
The user can work directly on the graph, and those activities that are created without connection to the text will be marked red in the list. In order to link such activities with a text, the user needs to select that activity in red in the controller. A selected activity will have a blue shade (see figure below). After this, the user can simply highlight the text as when creating activities, and click on “Assign text” to confirm the highlight. In this case the highlighted text will just go in the description of the text and its label remains the same.

Roles
Create Roles:
When user choose to create a role then a role will be created with the highlighted text. The created role will be available in resources section of the graph and also will be listed within the app. The title and description of the role will be the text highlighted. Once a role is being created with the text then the respected text will be marked as light green.
Make note that names in roles are unique, and highlighting two roles with same names will generate an error. See “Assign texts”

Delete Roles:
Just like activities, roles can also be deleted using delete button with role. Keep in mind that once a role that is in use by certain activities is deleted, then it will be removed from such activities as well. In addition, the role will no longer appear in graph resources, and the marking of the text will be removed.

Assign Roles to Activities:
In item listing on the left, when we click some role, all the activities of the graph will be shown and user can assign the selected role to different activities.
Relations
Create Relations:
In order to create relations between activities, a user needs to select the text and choose the option ‘Add as Relation’. When user chooses to create a relation then a list of relations and activities available in the graph will be shown to user. After, the direction and the relation type need to be specified, confirming it with “done”.

The description of the relation will be the text highlighted. Once a relation is being created with the text then the respected text will be marked as light orange.

On clicking on the relation within the app, activities will be listed down between them the relation exists.

Create Multiple Relations:
If a user wants to create relation with multiple activities in one go then multiple activities can also be selected in ‘To’ field, then the selected relation will be made between ‘From’ activity and all the activities in ‘To’ field.
When a user click on relation text from the left panel in the app then all the relations that belongs to this text will be listed down on right section. A fine grained treatment of relations per highlighted text is possible, by either adding new relations, or deleting previously created ones.

**Delete Relations:**
Just like activities and roles, relations can also be deleted using delete button, and when a relation is being deleted from item list on the left then all the relations associated with the text will be deleted from the graph and also the marking of the text will be removed.
Remove Mapping from Text:
If a user wants to remove a mapping from text but not the item itself, then a cross button over text is available just to remove the mapping, the original item (activity, role or relation) will not be deleted in this case and will then be treated as un-mapped.

Add Relation button for existing relations:
In order to create more relation(s) with existing relation’s text, Add button is available. When the user press ‘ADD’ button then he will be moved to create relations screen and there user can create relations with already existing relation’s text.
**Assign Text:**
Assign text functionality is there to map the un-mapped items, user just have to select the desired item and then hover the text and select Assign option, to which he needs to map the item with.

For activities, title and description of activity will be updated and for roles and relations, only the description will be updated with the text selected.

**Filter:**
Filter is also available on the left side to filter out items.
Edit Text:
If user wants to edit text within highlighter app then it can be done using the edit button. This edition of text will just be for highlighter app and will not be synced to graph’s description.

Help Button:
There is also a help button within the highlighter app that simply downloads the Highlighter App guide (this document).
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry’s standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.