Reviewer's report

Title: Leveraging workflow control patterns in the domain of clinical practice guidelines

Version: 2
Date: 18 September 2014

Reviewer: Claudio Eccher

Reviewer's report:

The authors an approach for the development and authoring of Clinical Practice Guidelines models, based on the analysis and semi-automatic transformation of a set of suitable workflow control patterns in an implementation-independent notation (BPMN 2.0) to two CPG languages (Asbru and PROForma).

The paper is well written, the approach is sound and the methods are presented with sufficient clarity. The need of an implementation-independent language that can be transformed in the most used guideline languages has been a research topic of great interest since several years ago. The related work is well covered.

Minor Essential revisions.

Sections Motivation and Discussion

The choice of BPMN 2.0 as the implementation-independent language is fully justified by the fact that it is a standard and it is suitable for the medical domain. In view of this, however, the authors should explain the advantages of modelling patterns in BPMN and then translating them in a guideline language instead of directly using BPMN models for guideline execution, considering that several (freely available) BPMN engines exist. See for example the jbpm suite.

Section Results, Table 1

A brief explanation of the workflow patterns e.g., adding a column to Table 1, could help the readers less accustomed with these patterns in reading the paper without the need of consulting the Russel's paper.

Page 17, Subsection 3.2, row 359

"In Asbru there are two ways to model this pattern combination..."

Which is the other? The if-then-else plan? Are there differences or did the authors make an arbitrary choice between the two?

Page 18, Subsection 3.2, row 396

"... it can only be done in Asbru through programming workarounds.” Referring to the more general problem of what can be done and what cannot be done in each language (+/- or - in Table 2), howuch complex is the programming workaround
in the case of +/- ? Is there a general rule that can be applied? Can be the work around done only by complicating the model, without a general rule to do so, or some external software is necessary?

Moreover, in the case of the (–) mark, does it mean that the corresponding pattern cannot be implemented at all? Does it mean that I cannot model e.g. in PROforma the piece of guideline corresponding to the interleaved routing in Table 1?

Page 23, Section 3.3.

“…the common representation in BPMN is via an intermediate message event. … we do not consider this intermediate message event (nor the message event In the triggering task”

I’m a bit confused by this sentence. If the first part holds, I do not understand the second one. Moreover, it seems to me that the intermediate message event in the BPMN schema in Figure 9, the intermediate message event, which waits for a message different from the persistent trigger message, as far I as understand, is not necessary. The confusion is increased by the sentence

“New scheduling constraints are generated for the actions task with a preceding message event”

Do you consider intermediate message events or not? Can you explain better in order to render the paragraph clearer?

Level of interest: An article of importance in its field

Quality of written English: Acceptable

Statistical review: No, the manuscript does not need to be seen by a statistician.

Declaration of competing interests: Nothing to declare