Reviewer's report

Title: Evaluation of PROforma as a language for implementing medical guidelines in a practical context.

Version: 1 Date: 28 November 2005

Reviewer: Samson Tu

Reviewer's report:

General

The paper describes the experience of using PROforma as a language for implementing clinical guidelines in a deployed system that provides advice to pharmacists on the management of hypertension. The criteria used to evaluate PROforma are logical adequacy, heuristic power, notational convenience, and explanation support. The evaluation is appropriately qualitative. However, without quantitative data, to be convincing, judgmental evaluations should be backed up with detailed examples and observations. The paper can be strengthened considerably if the authors can provide such qualitative information to back up their observations. There should be no unsurmountable problem in fixing the paper for publication.

The paper is very well-written. The background section is particularly lucid. The paper made very important observations about PROforma.

-----------------------------------------------

Major Compulsory Revisions (that the author must respond to before a decision on publication can be reached)

p. 13 What kinds of logical distinctions should be encoded in PROforma?

The authors state that “One of the aims of the PROforma language is to describe the logic of processes in a way that allows them to be examined and validated by domain experts. Thus one possible answer…is to say that knowledge needs to be represented in PROforma if it needs to be validated by domain experts.”

It’s not clear what’s the argument being made here. Is the authoring saying that, because PROforma allows encoded knowledge to be validated by domain experts while other software components do not, it’s a good idea to encode the knowledge that requires validation? It would be highly desirable for the authors to give specific examples. It’s not clear what formalized knowledge doesn’t require some kind of validation.

The authors found the essential logic of the medical process involved to be relatively simple. This reviewer, having created a guideline-based decision-support system for the management of hypertension, finds that hard to believe. If the system gives advice about how a patient’s medication should be modified, the necessary knowledge is quite complex. It is very frustrating that the authors didn’t give more details about the kinds of advice and knowledge available in their system.

p. 14: Weaknesses in PROforma’s logical adequacy

- Support for abstraction and information hiding - please give examples.

- “…the Tallis implementation of PROforma provides no support for record structures or for
collections other than ordered lists" Is it a problem with Tallis implementation or is it a more general problem in PROforma? In (Peleg et al., 2003), it’s said that “PROforma, GUIDE, and Asbru do not constrain the possible classes of complex objects.”

p. 15. “The PROforma language proved adequate…may be because the reasoning requirements of the study were relatively simple” Again, it’s very frustrating not being told what the reasoning requirements were.

p. 17. “UML has more complete graphical representation. This was one of the main reasons why UML activity diagrams were used during the knowledge acquisition phase of the project.” This statement is surprising. UML activity diagrams models processes similar to those of PROforma. The authors really need to describe in detail why UML activity diagrams are more convenient. For example, it is not clear at all that UML activity diagrams can support graphical representation of arguments and candidates that the authors say PROforma lacks. My understanding is that arguments are Boolean expressions for or against a particular candidate (which is a pointer to the next task). Please argue for your assertions and give concrete examples!

p. 20. “In particular we considered that strengths and weaknesses of the PROforma language are revealed by: ….” The authors didn’t really argue for the assertions made in this paragraph.

1. Very little is said about the knowledge needed to provide the advice for the users.

2. No example is given on why activity diagrams are better than PROforma processes in terms of usability by domain experts. Are the difficulties intrinsic difficulty with PROforma or is it a problem of tool support? Suppose you model the PROforma task ontology in Protégé, would PROforma be more usable? Much of the complexity in encoding computer-interpretable guidelines is in writing the decision criteria. That’s why in the EON system, for relatively simple criteria, we use a set of object templates that clinicians can just fill in the fields.

3. Little is said about how the needs for distributed system and user interface impact the use of PROforma.

-------------------------------------------------------------------------------------------
Minor Essential Revisions (such as missing labels on figures, or the wrong use of a term, which the author can be trusted to correct)

p.17-18: Guard conditions: GLIF3.x doesn’t have guard conditions on transitions.

-------------------------------------------------------------------------------------------
Discretionary Revisions (which the author can choose to ignore)

It seems that the way to simulate guards in PROforma is to “reify” the condition by creating a decision task where the guard condition is a “conclusive” argument for the subsequent task.

What next?: Unable to decide on acceptance or rejection until the authors have responded to the major compulsory revisions

Level of interest: An article of importance in its field

Quality of written English: Acceptable
Statistical review: No

Declaration of competing interests:
I declare that I have no competing interests