

# Linking Clinical Guidelines with Formal Representations

Peter Votruba<sup>1</sup>, Silvia Miksch<sup>1</sup>, and Robert Kosara<sup>2</sup>

<sup>1</sup> Vienna University of Technology, Inst. of Software Technology & Interactive Systems,  
Favoritenstraße 9-11/188, A-1040 Vienna, Austria  
{peter,silvia}@asgaard.tuwien.ac.at  
www.asgaard.tuwien.ac.at/

<sup>2</sup> VRVis Research Center for Virtual Reality and Visualization,  
TechGate Vienna, Donau-City-Strasse 1, A-1220 Vienna, Austria  
Kosara@VRVis.at www.VRVis.at/vis/

**Abstract.** Clinical guidelines have been used in the medical domain for some time now, primarily to reduce proneness to errors during the treatment of specific diseases. Recently, physicians have special software at their disposal, which supports them at decision-making based on computerized protocols and guidelines. Using such tools, physicians sometimes want to know the reason why the computer recommends a particular treatment method. To comprehend the suggestions, a connection between the original guideline and its computerized representation is needed. This paper introduces a tool that was designed to provide a solution for that, the so-called *Guideline Markup Tool* (GMT). This tool enables the protocol designer to create links between the original guideline and its formal representation.

## 1 Introduction

Clinical guidelines have been introduced to standardize treatment methods of physicians. Clinical *guidelines* are systematically developed directions to assist the medical practitioner in making decisions about appropriate healthcare for specific conditions. Guidelines are intended to define each step of a treatment for specific diseases to reduce proneness to errors. Conventional guidelines are written as plain text documents, sometimes including tables or flow charts for better illustration of important facts. Often these documents contain ambiguities or even contradictions, which reduce their usefulness.

Several approaches have been carried out with the aim of improving the usefulness of guidelines, by trying to model them in a machine-readable form using a guideline modelling language. In many cases, the main goal is to systematically validate guidelines. In addition, software systems have been developed, which support the computerization and application of guidelines.

The *Asgaard* project<sup>1</sup> [6] outlines some useful task-specific problem-solving methods to support both designers and users of clinical guidelines and protocols. The key part of this project is the guideline representation language *Asbru* [3,5].

The *Guideline Markup Tool* (GMT) has been developed within the *Asgaard* pro-

---

<sup>1</sup> *Asgaard* Project website: <http://www.asgaard.tuwien.ac.at/>

ject to support the translation of a free-text guideline into its *Asbru* representation.

### 1.1 Related Work

Several tools for acquiring guidelines have been proposed. *AsbruView* [1], a knowledge acquisition and visualization tool, was developed within the *Asgaard* project to facilitate the creation, editing and visualization of *Asbru* files. To be suitable for physicians, *AsbruView* uses graphical metaphors, such as a running track with a finishing flag, to represent *Asbru* plans.

*GEM Cutter* [4] is a tool that was developed to support the transformation of a guideline into the GEM format. It shows the original guideline document together with the corresponding GEM document, similar to our *Guideline Markup Tool*, and makes it possible to copy text from the guideline to the GEM document.

There are two tools for translating guidelines into *PROforma* [2] - both make heavy use of the same graphical symbols representing the four task types in *PROforma*. *AREZZO* is designed to be used on client-side only, whereas *TALLIS* [7] supports publishing of *PROforma* guidelines over the World Wide Web.

## 2 The Guideline Markup Tool

None of the existing tools supports (i) the linking of informal and formal representation of guidelines to increase the structuring and understanding of guidelines in both representation and to trace back flaws and errors and (ii) the facility of design patterns to ease the authoring of guidelines in a formal representation. This leads to the two main features of the *Guideline Markup Tool* (GMT) [8].

Firstly, GMT allows the definition of links between the original guideline and the *Asbru* representation, which gives the user the possibility to find out where a certain value in the *Asbru* notation comes from. If someone wants to know the origin of a specific value in the *Asbru* XML file, the GMT can be used to jump to the correlating point in the HTML file where the value is defined and the other way round.

The second main feature of the GMT is the usage of macros. A macro combines several XML elements (in other words, *Asbru* elements), which are usually used together. Thus, using macros allows creating and extending *Asbru* XML files more easily through the usage of common design patterns. Such design patterns are often used behaviours, which can be found in guidelines.

Through these two features, GMT is able to support the following tasks:

- **Authoring and Augmenting Guidelines.** We want to be able to take a new guideline in plain text and create an *Asbru* version of it, and to add links to the corresponding parts of a guideline to an already existing *Asbru* file.
- **Understanding *Asbru* Guidelines.** For an *Asbru* guideline, we want to be able to see where values in the different parts of the *Asbru* code come from, and how parts of the original text were translated into *Asbru*. This is important not just for knowledge engineers, but also for physicians wanting to get an understanding of the language *Asbru*.

- **Structuring *Asbru*.** The GMT provides a structured list of *Asbru* elements – the macros – that needs to be done in a way that best supports the authoring of plans. This list will also provide a good starting point for teaching material and possible subsets of the language for special purposes.

## 2.1 Features

According to the requirements presented above, the user interface is designed to show the contents of the HTML file (original guideline), the XML file (*Asbru* representation) and the macros file together in one window. Therefore, the GMT window is divided into three main parts – Fig. 1 shows a screenshot of the GMT with loaded HTML-, XML- and macros files.

The upper left part of the window (component #1 in Fig. 1) shows the contents of the HTML file. The XML part consists of a hierarchical view of the XML file (component #2a) and a detail view of the current XML node (component #2b). The macros part contains a view of the macros structure (component #3a) and a pre-view of the currently selected macro (component #3b).

- **Inserting a macro/link.** To insert a macro (or a link, which is a special kind of macro), the target XML element in the XML view and a proper macro in the structure view have to be selected. After clicking on the insert-macro button, an input dialog appears where the attribute values can be entered.
- **Activate a link.** If links have been defined during the translation of a guideline, they can be used to comprehend the connections between the original guideline and its *Asbru* representation (see Fig. 2).
- **Link visualization.** A useful add-on is the possibility to visualize the spread of links in an *Asbru* file, where each element in XML view gets coloured differently – all link elements get a green background, the elements that belong to a link, are coloured blue, and the other elements are grey. In case links are inserted into an existing *Asbru* file, this feature provides a good overview of all unlinked parts.

## 3 Evaluation and Conclusions

In this paper, a new guideline tool, called *Guideline Markup Tool* (GMT), is introduced. It supports knowledge engineers in translating clinical guidelines into their *Asbru* representation. It does this by providing macros to facilitate assembling *Asbru* guidelines. However, the main feature of the GMT is the ability to create and maintain links between a guideline HTML file and its representing *Asbru* XML file. The knowledge engineer should always define links during the translation task. If the resulted *Asbru* XML file is used as an input of another *Asgaard* tool, it may happen that someone wants to know the reason for the choice of a particular *Asbru* element or the origin of a specific attribute value. The GMT can be used to answer such questions or to retrace errors.

To be consequently, links also work in the other direction, which allows easier comprehension of the translation process and thereby facilitates learning of the quite

complex language *Asbru*. Therewith, the GMT can be used to find out how a particular passage in the text of the original guideline has been modelled in *Asbru*.

We performed a small, qualitative study on the usability of the GMT [8]. We chose eight knowledge engineers, who were familiar with the *Asbru* language.

The evaluation procedure consisted of three phases: (i) a questionnaire assessing the computer skills of the participants; (ii) an exploration session with the GMT, where the participants examined the functionality of the GMT; (iii) a questionnaire assessing the overall impression and the three views in particular.

The second and third phase confirmed that the three views (HTML, XML, Structure View) are very appropriate to author clinical guidelines and to translate such clinical guidelines into a formal notation, like *Asbru*. The linking features in both directions facilitated structuring guidelines' text, the retrieval of knowledge parts, and retracing of possible flaws and errors. One drawback of the GMT was that everybody was expecting a fully functional editor for XML code in the XML View, which was out of scope. In summary, the participants rated the GMT as a very powerful and useful tool, which supports the implementation of clinical guidelines.

**Acknowledgments.** We wish to thank Mar Marcos and Marije Geldof for their valuable suggestions. Furthermore, we would like to thank Katharina Kaiser, Georg Duftschmied, Andreas Seyfang, Christian Popow, Monika Lanzenberger, Wolfgang Aigner, Peter Messner and Klaus Hammermüller for participating in the evaluation.

This tool is part of the *Asgaard* Project, which is supported by "Fonds zur Förderung der wissenschaftlichen Forschung" (Austrian Science Fund), grant P12797-INF.

## References

1. Kosara, R.; Miksch, S.: Metaphors of Movement - A Visualization and User Interface for Time-Oriented, Skeletal Plans. In: *Artificial Intelligence in Medicine*, Special Issue: *Information Visualization in Medicine*, pp. 111-131, 22(2) (2001)
2. Bury, J.; Fox, J. and Sutton, D.: The PROforma Guideline Specification Language: Progress and Prospects. In: *Proceedings of the First European Workshop on Computer-Based Support for Clinical Guidelines and Protocols (EGWLP) 2000*, Volume 83 of *Studies in Health Technology and Informatics*, pp. 12-29. IOS Press. (2000)
3. Miksch, S.; Shahar, Y.; Johnson, P.: *Asbru*: A Task-Specific, Intention-Based, and Time-Oriented Language for Representing Skeletal Plans. In: Motta, E.; Harmelen, F. v.; Pierret-Golbreich, C.; Filby, I.; Wijngaards, N. (eds.), *7<sup>th</sup> Workshop on Knowledge Engineering: Methods & Languages (KEML-97)*, Milton Keynes, UK (1997)
4. Polvani, K.-A.; Agrawal, A.; Karras, B.; Deshpande, A.; Shiffman, R.: *GEM Cutter Manual*. Yale Center for Medical Informatics (2000)
5. Seyfang, A.; Kosara, R.; Miksch, S.: *Asbru's reference manual, Asbru version 7.3*. Technical Report Asgaard-TR-2002-3, Vienna University of Technology (2002)
6. Shahar, Y.; Miksch, S.; Johnson, P.: The Asgaard Project: A Task-Specific Framework for the Application and Critiquing of Time-Oriented Clinical Guidelines. In: *Artificial Intelligence in Medicine*, 14, pp. 29-51 (1998)
7. Steele, R. and Fox, J.: *Tallis PROforma Primer*. Advanced Computation Laboratory, Cancer Research, UK. (2002)
8. Votruba, P.: *Structured Knowledge Acquisition for Asbru*. Master's Thesis, Vienna University of Technology (2003)

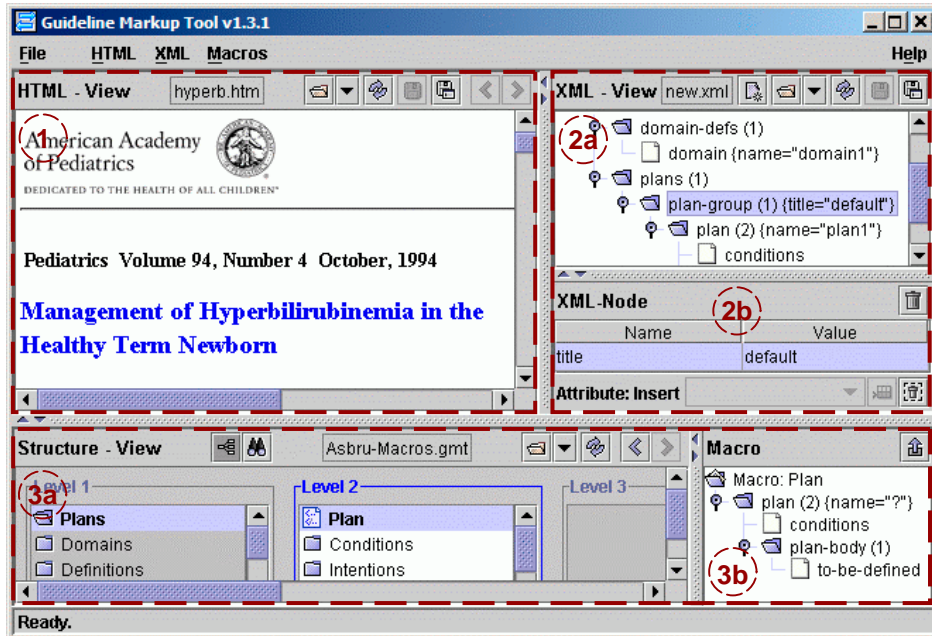


Fig. 1. Screenshot of the GMT window

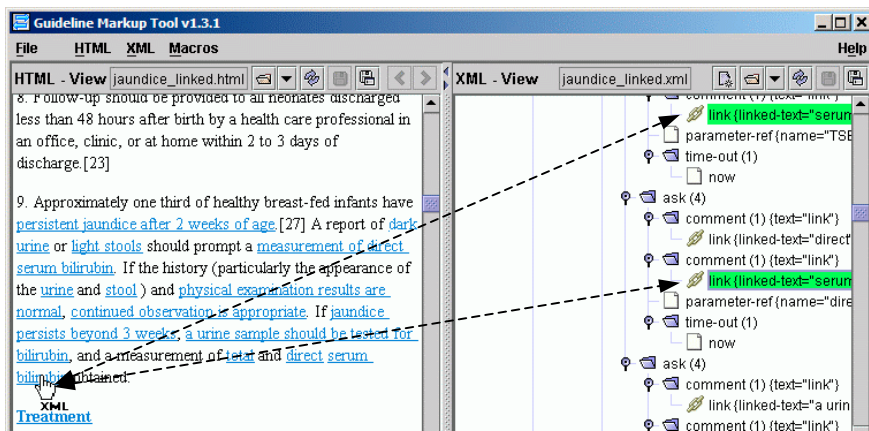


Fig. 2. Link activation. When clicked on a link endpoint in the HTML view, the counterparts are highlighted in the XML view