Open-Source Publishing of Medical Knowledge for Creation of Computer-Interpretable Guidelines

Mor Peleg[1], Rory Steele[2], Richard Thomson[2,3], Vivek Patkar[2], Tony Rose[2], John Fox[2,3]

1Department of Management Information Systems, University of Haifa, 31905, Israel
morpeleg@mis.hevra.haifa.ac.il
[2]Advanced Computation Laboratory, Cancer Research UK, Lincoln Inn fields, London WC2A 3PX, UK {rs, vp, tr, jf}@acl.icnet.uk
[3]OpenClinical (www.openclinical.org) rt@openclinical.org

Abstract. Guidelines, care pathways, and other representations of high quality clinical practice can now be formalized and distributed in executable form. It is widely recognized that the ability to apply knowledge at the point of care creates an opportunity to influence clinicians' behavior, encouraging compliance with evidence-based standards and improving care quality. The ability to share formal knowledge may also enable the medical community to build on work done by others and reduce content development costs. We propose a Medical Knowledge Repository and content model that supports assembly of components into new applications. Some types of resources that may be included in such a repository are defined, and a frame-based representation for indexing and structuring the components is described. The domain of breast cancer is used as a case study for demonstrating the feasibility of the approach.

1 Introduction

A number of studies of computer applications that can deliver patient-specific clinical knowledge to the point of care during patient encounters have shown positive impacts on clinicians behavior [1]. Since developing such resources requires much effort, we would like to be able to share them, enabling the community of clinical guideline developers, publishers, and users to work collaboratively, leveraging prior work.

There are several potential ways for sharing medical knowledge that is in a computer-interpretable format. One way is to permit multiple formalisms to exist and to translate between different representation formalisms, as required. This appears to be infeasible based on the experience of the InterMed project [2].

A second option is to adopt a single standard. The HL7 Clinical Guidelines Special Interest Group (CGSIG) is engaged in a process of developing and standardizing components of a standard computer-interpretable guideline (CIG) model. Standardizing CIG components would enable sharing of significant parts of encoded guidelines across different CIG modeling methods. The selection of CGSIG components was influenced by the results of a study that compared CIG formalisms in terms of components that capture the structure of CIGs [3].

A third way to facilitate sharing of computer-interpretable (CI) components of medical knowledge is to create a library of resources that could be used as components for developing executable guidelines. Guideline developers can use this Medical Knowledge Repository (MedKR) to assemble guidelines from CI knowledge components. The idea is similar to Common Object Request Broker Architecture (CORBA) [4], an open, vendor-independent architecture and infrastructure that computer applications use to work together over networks. Similar architectures, which address interoperability of components defined by different standards, security and administration challenges, include Web Services Architecture (WSA) [5] and Semantic Web Services Framework (SWSF) [6]. The knowledge components would be units of operational software that deliver specific functionality (and may include data relevant to the function being delivered). The components will share a common interface for specifying the operation that is performed and its arguments. In this way, a client object could request an operation to be performed on a server, with appropriate parameters, which would perform the operation and return the results to the client.

2 Executable Medical Resources that Could Be Shared

Many types of medical resources could be published and later used to assemble executable CIGs. A few examples include:

1. Medical calculators (e.g., http://www.medalreg.com/; http://www-users.med.cornell.edu/~spon/picu/calc/)
2. Risk-assessment tools (e.g., http://www.yourdiseaserisk.harvard.edu/),
3. Drug databases (http://www.medscape.com/druginfo),
4. Controlled terminologies (e.g., SNOMED-CT http://www.snomed.org/),
5. Authoring, validation, and execution tools for CIGs [7]

How might we create a CIG, care pathway, or other application by bringing subsets of such a wide variety of resources together in a single application?

Components may exhibit different granularities. For example, medical calculators may be shared by linking them as nodes within a CIG's task network. In this way, any CIG formalism that conforms to the task network model [3], such as Asbru, EON, GLIF3, GUIDE, PRODIGY, or PRO*forma* may call upon predefined components. On the other hand, components may be shared at finer granularity. For example, different CIG formalisms might use the same expression language to express decision and eligibility criteria, and may refer to the same patient data model, which may be different than the expression language and patient model that their CIG formalism uses.

Figure 1 shows an example of our proposed component definition for sharability and interoperability. It includes input and output types, which can be conventional data types or semantic types that are defined in an external ontology. The definitions of CORBA components and Web services descriptions have been extended by developing a frame-based representation that includes, in addition to the above-mentioned attributes, attributes by which the resources could be indexed and organized in a repository. One of these is the component's *goal* — a semantic type based on an ontology originally developed for the breast cancer domain [8], which includes about 20

general classes of goals, broadly categorized as knowledge goals and action goals. Currently, most of these semantic types contain only a "goal sentence", which is free text, but some semantic types also have additional slots. Work is under way to structure the goal sentence [9], and to suggest a restricted vocabulary for coding.

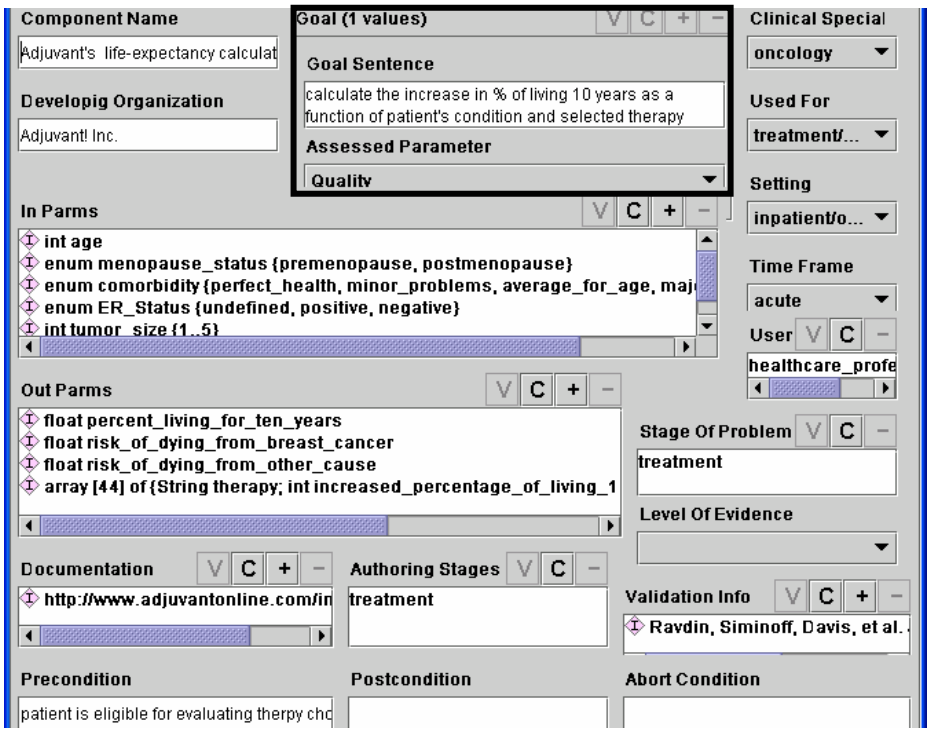

Fig. 1. The interface of Adjuvant's life-expectancy calculator specified as a frame using the Protégé-2000 knowledge modeling tool. Input and output parameters are specified as instances of the well-defined Parameter_Definition class hierarchy, which consists of classes for simple and complex parameters and for enumerated types. In the figure, only the short names for the parameters are shown

The goal shown in Fig. 1 is of type Assessment and has a slot called "assessed_parameter" which can hold the following symbols: urgency, need, risk, quality. Additional attributes describe the clinical sub-domain that the component addresses. They are taken from a schema for indexing CIGs that was developed by Bernstam and colleagues [10] and is an extension of the classification scheme used by the National Guideline Clearinghouse. Additional attributes include pre- and post- and abort-conditions for using the component, the authoring stage at which the component could be used, information about the component's developers, version, pointers to documentation, the level of evidence supporting the component, and information on how the component has been validated, including pointers to test cases.

Components can be indexed and searched based on their attributes. Especially useful are attributes that specify the clinical sub-domain, the relevant authoring stages, and goals.

Components could be assembled into CIGs by specifying the guideline's skeleton into which components can be integrated. When a CIG engine that enacts the skeletal CIG formalism calls a component, it passes control to a broker that acts as an interface between the engine and the MedKR, passing parameter values and returning results. The broker will have services for converting vocabulary terms, measurement units, and data-type casting.

3 Case Study Example

We have chosen guidelines in the domain of breast cancer to test the applicability of these proposals. This is a good area because knowledge in the domain is abundant, evidence-based, and structured. Figure 2 shows an example of a PRO*forma* CIG that call external components: Adjuvant's life expectancy calculator, which calculates the density of breast masses from radiographs (www.adjuvantonline. com), and Gail's breast cancer risk assessment module (http://www2.swmed.edu/breastcarisk/).

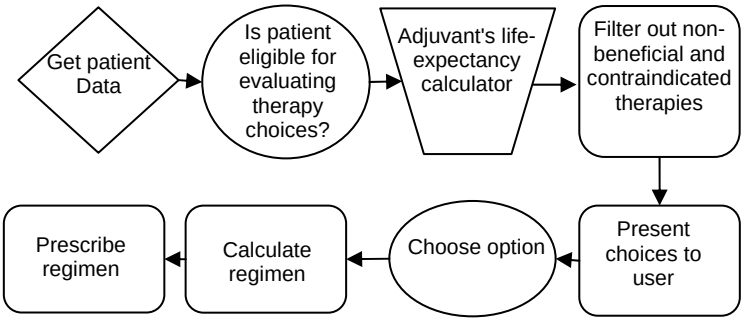

Fig. 2. A PRO*forma* CIG that provides advice on regimens for treating breast cancer. The CIG includes an external *component* (trapezoid), whose interface definition is given in Fig. 1. Different types of native PRO*forma* tasks are shown: *enquiry* (diamond), *decision* (circle), and *plan* (round-cornered square)

4 Discussion

A repository of executable medical knowledge components that would be published on the Web would enable guideline developers to piece together CIGs from components that have been independently developed and tested. In this paper, we suggested a framework for specifying the interface of executable components so that they could be searched for and integrated within a CIG specification. Unlike publets, which are encoded solely in the PRO*forma* formalism, our approach permits a CIG to be en-

coded in any CIG formalism, integrating components written in different formalisms. We hope that the notion of creating a repository of tested knowledge components and CIGs would be appealing to the community of content developers, and developers of CIG formalisms. We hope developers and researchers will contribute components and tested CIGs to the MedKR. We are currently working on implementing a simple guideline using our approach

References

1. Shea S, DuMouchel W, Bahamonde L. A Meta-analysis of 16 Randomized Controlled Trials to Evaluate Computer-based Clinical Reminder Systems for Preventative Care in the Ambulatory Setting. J Am Med Inform Assoc 1996;3(6):399-409.
2. Peleg M, Boxwala AA, Tu S, Zeng Q, Ogunyemi O, Wang D, et al. The InterMed Approach to Sharable Computer-interpretable Guidelines:A Review. J Am Med Inform Assoc 2004;11(1):1-10.
3. Peleg M, Tu SW, Bury J, Ciccarese P, Fox J, Greenes RA, et al. Comparing Computer-Interpretable Guideline Models: A Case-Study Approach. J Am Med Inform Assoc 2003;10(1):52-68.
4. Siegel J. CORBA 3 Fundamentals and Programming, 2nd Edition: John Wiley & Sons; 2000.
5. W3C. Web Services Architecture. Champion M, Ferris C, Newcomer E, Orchard D, editors. 14 November 2002 ed: W3C; 2002.http://www.w3.org/TR/2002/WD-ws-arch-20021114/
6. Domingue J, Cabral L, Hakimpour F, Sell D, Motta E. -III: A Platform and Infrastructure for Creating WSMO-based Semantic Web Services. Proc Workshop on WSMO Implementations 2004.
7. Clercq PAd, Blom JA, Korsten HH, Hasman A. Approaches for creating computer-interpretable guidelines that facilitate decision support. Artif Intell Med 2004;31(1):1-27.
8. Fox J, Alabassi A, Black E, Hurt C, Rose T. Modelling Clinical Goals: a Corpus of Example and a Tentative Ontology. Proc Symp Comput Guidelines and Protocols 2004: 31-45.
9. Fox J, Alabassi A, Patkar V, Rose T, Black E. An ontological approach to modelling tasks and goals. Comput in Biol & Med, in press 2004.
10. Bernstam E, Ash N, Peleg M, Tu S, Boxwala AA, Mork P, et al. Guideline classification to assist modeling, authoring, implementation and retrieval. Proc AMIA Symp. 2000: 66-70.