

Summary of the Guideline Workbenches Evaluation

Purpose of the Evaluation

We (the SAGE Guideline Workbench group) have adopted Protégé-2000 as the basis for developing the SAGE Guideline Workbench. We evaluated a set of existing guideline workbenches to assess the features of Protégé-2000, and to identify the features in the other workbenches that we would like to have in the SAGE Guideline workbench.

Method

We selected the following workbenches to evaluate:

- Protégé-2000
- Arezzo / PROforma
- Design-a-Trial
- GILIF Guideline Authoring Tool
- GUIDE
- AsbruView
- CG-AM
- GEM Cutter
- URUZ

We listed a set of evaluation topics:

- Components,
- Modeling and encoding process,
- Verification, simulation and verification

We provided some general information on each workbench, evaluated the workbench on the topics mentioned above, and summarized its strengths and weaknesses. To perform the evaluation, we tried the workbenches directly and/or examined the papers published on these workbenches.

Results

We used a spreadsheet a column for each workbench to record our evaluations. This document highlights the information in that spreadsheet relevant to the purpose of the study. On each of the evaluation topics, we state what Protégé-2000 has to offer. Along with that, we describe any relevant and desirable features that we identified in other workbenches. We also include a summary of Protégé-2000 evaluation highlighting its strengths and weaknesses.

Components

A. Guideline model – What is the underlying guideline model? Is the guideline model geared towards any specific types of guidelines?

Protégé-2000 is not tied to any guideline model. It can support relatively simple guideline models such as Prodigy to complex guideline models such as EON.

GEM Cutter is a tool for rendering text based guidelines in the Guideline Element Model (GEM). GEM is an XML DTD that contains all the attributes needed in a published guideline.

B. What are the capabilities supporting, or supporting development of, the following guideline features:

a. Enterprise workflow context and modeling

Samson has developed workflow models (in collaboration with University of Pavia in Protégé-2000. However, these models have not been implemented.

PatMan is billed as a Patient Careflow System. GUIDE is the graphical front end to create PetriNet based clinical workflow models. To that end an Enterprise Ontology is incorporated into the system. The Enterprise Ontology is maintained at the Stanford Knowledge Systems Lab.

b. Information processing context and modeling

No experience in modeling system resources in Protégé-2000.

The CG-AM "representation formalism" was designed with the aim of representing "contextual limitations", such as availability of clinical and other resources.

c. Graphical (flowchart logic) depiction

A special-purpose widget in Protégé-2000 called the Diagram Widget allows users to model flowcharts. This widget has been used to model clinical algorithms.

The AsbruView user interface presents two views concurrently: (1) A "Topological View" that displays relationships among plans (i.e., sub-plans within guidelines), and (2) A "Temporal View" that displays the temporal characteristics of plans in more details. In the Topological

View, plans are depicted as segments on a visual "running track" metaphor. It is important to note that the topological view metaphor is from the point of view of the clinician (not the patient) moving along a running track populated with guideline sub-plans. Selected process characteristics (e.g. entry, exit conditions) are also displayed using "traffic" metaphors such as stop-lights and entry gates. The temporal view employs less intuitive symbols (reminiscent of a Gantt chart), to show detailed temporal relations within and among plans.

Arezzo and the CG-AM tool provide a graphical view of guidelines using multiple panels: relations between guideline actions in a format similar to a Windows directory "tree" in the left panel, and a flowchart view (fairly similar to the graphical display in Protege) in the right panel.

d. Data layer instantiation of logical elements into standard data elements

In the EON project, patient data variables were defined in the guideline model using Protégé-2000, and were subsequently mapped to data elements in a relational database.

The CG-KRM can receive data from four databases: (1) the Clinical DB, which provides standard terminology for actions and conditions; (2) the Pharmacological DB, which provides a "structured list" of drugs and their costs (sounds like a formulary-GM); (3) the Resources DB, which lists resources (e.g. CT, NMR) available in a given hospital; and (4) the ICD DB, which contains the international coding of diseases. The CG-AM interacts with these databases to enforce use of standard vocabularies during the authoring/encoding process.

e. Execution engine for run-time support?

Since Protégé-2000 can support different guideline models, there is no generic execution engine. It provides a rich set of API to access the elements in the knowledge base. An execution engine needs to be built for each guideline model.

C. EMR – What is the model of patient information?

Protégé-2000 is a generic knowledge acquisition tool. It does not come with a built-in access to a virtual EMR. However, the functionality can be added as part of the guideline models.

The GLIF3 model uses a set of the HL7 RIM classes as the model of patient information. However, the workbench does not support that at this time.

In CG-AM, there is a patient data model (Patient DB). According to the authors, "the schema of the Patient DB parallels that of the Clinical DB [which] makes it possible to automatically retrieve from the Patient DB at execution time. Access to patient-specific data is obtained through "Query Actions" -- for data from: (a) patient history, (b) physical examination, or (c) laboratory results.

D. Controlled Terminology Services – Does it provide access to controlled terminology services? How smooth is it to use standard terminologies when entering guideline knowledge? Are there utilities for loading and maintaining versions of external terminologies?

The mode of operation to use external terminologies is to import the whole ontology into the Protégé-2000 environment. Protégé-2000's component-based open architecture facilitates integrating utility functions and custom-built applications into the system. For example, the developers can add new functional tabs to the standard set. At knowledge acquisition time, users can access the utility functions via the new tabs. One functional tab that is relevant to encoding guidelines allows online access to UMLS. The UMLS tab allows users to browse online sources, to verify existence of a medical concept within UMLS, and to import sub-trees of the UMLS ontology directly into the knowledge acquisition environment. Apelon recently built a Protégé tab that provides access to the Distributed Terminology Service (DTS). Protégé-2000 does not provide any utilities for maintaining terminology versions. No guideline group has yet used terminology services within Protégé-2000 when entering guideline knowledge.

The PatMan Careflow system is built using SNOMED terminology. When the user issues an exception to the workflow they are supposed to indicate the exception utilizing a SNOMED browser. When utilizing GUIDE the user does not interact with a controlled vocabulary

Modeling & Encoding Process

E. Mode of Operation – What is the general process to encode guidelines? Does it support multi-layered modeling that allows clinical experts to interact easily with knowledge experts?

In Protégé-2000, the developer with the help of domain specialist creates an ontology of domain concepts, and builds a patient model and a guideline model using these concepts. The domain specialist with the help of the developer enters guideline knowledge using user-interface forms and special-purpose widgets. Currently there is no explicit support for multi-layered

modeling but it can be achieved to an extent through special-purpose widgets and conventions on division of labor between clinical experts and knowledge experts.

Design-a-Trial interviews a physician, prompts and guides them through suitable design options, comments on the statistical rigor and feasibility of their proposed design, and generates a 6-page structured draft protocol document.

The GLIF3 methodology is to have clinicians encode a top-level conceptual view of a guideline and have knowledge engineer encode the computable parts. A third layer involves mapping and customization of encoded guidelines to deployment institutions. The GLIF workbench does not yet support these layers.

The purpose of URUZ is to allow a domain expert to convert a free text clinical practice guideline into a marked-up guideline through cut and past like gem cutter. The next function is to allow a knowledge engineer to further mark-up the text using ASBRU such that it could be implemented into a CIS for decision support or retrospective review for quality assessment.

F. Multi-user support – What kind of multi-user support does it provide? Does a client software allow multiple remote users to work collaboratively?

When using a database backend, multiple Protégé-2000 users can work on the same knowledge base at the same time. Currently users are limited to working on different parts of the knowledge base. There is ongoing effort to build a client-server architecture to improve on the functionality. There will be an indication when a new user starts to edit the knowledge base, and a user's changes to the knowledge base will be propagated to other users.

G. Extensibility – How extensible is the system? Is it customizable for different situations? Does it have a library of components that can be assembled in different ways? Does the database or programming environment create any known restraints of scale?

Protégé-2000 has an open-source Java-based extensible architecture that allows developers to build special-purpose GUI widgets and utility functions that can be easily integrated with the core system. Protégé-2000 has been used to build decision-support systems based on guideline models that embody very different assumptions, such as EON and Prodigy. EON is very expressive and uses complex constructs such as PAL constraints and temporal abstractions to represent complex decision-criteria and patient

states. Prodigy is a simpler model that stresses being intuitive to domain-specialists, and relies more on clinicians to recognize complex clinical patterns at the time of guideline execution. Protégé-2000's plug-and-play framework allowed both the modeling groups to customize the knowledge acquisition environment to suit their models.

I. User-friendliness – How does it make it easy for domain experts to enter guideline knowledge? How well does it hide the complexities of the underlying guideline model? What visual metaphors does it use to aid the knowledge entry process? Are the component modes of operation understandable, scalable and useful for:

- a. the clinical domain expert
- b. the knowledge engineer
- c. the software maintenance vendor?

Protégé-2000 generates user-interface forms automatically based on class definitions. Users build knowledge bases by filling out the form. Besides these generic forms, special-purpose user-interfaces can be integrated to facilitate knowledge acquisition. For example, a diagram widget that presents information graphically as a network of nodes and arcs has been successfully used to encode clinical guideline algorithms. Such widgets can be effectively used to also hide the complexities of the underlying guideline model.

Arezzo has a diagramming tool allows assembling of tasks as a network of nodes. It also has a knowledge editor to specify the details of a task, an decision editor to enter decision rules, and a condition editor to define a wide range of logical conditions that may be relevant during task execution. The user-interfaces are special-purpose and greatly simplify the modeling process.

The Design-a-Trial (DaT 2.0) interface, implemented in Prolog, employs a simple graphical representation of the components of a trial emphasizing the typical order in which the main design subtasks should be undertaken. The user is presented several data entry forms. The Prolog sophisticated graphical user interface was felt to be impractical to implement. Therefore, with DaT 2.0, uses a visual basic interface and an Amzi Prolog logic server module. The environment was designed for a naive clinical user, and would have less utility for domain experts or knowledge engineers.

For it's purpose GEM Cutter is easy to use with an adequate help file and a graphical navigation display. GEM Cutter uses an outline metaphor to display the GEM attributes with their attached text.

It seems very easy for domain experts to enter knowledge using Uruz. The tool hides the complexity of the model very well. Uruz takes a free text guideline and allows a domain expert to either create a guideline or cut and paste a guideline into a semi-structured format.

H. Evidence – When entering guideline rules, is there a way to specify the references to medical literature and/or enterprise standards of care that justify the rules?

In EON, using Protégé-2000, the guideline model had place holders associated with the knowledge rules to specify references that justify the rules.

One of the pillars of the Arezzo/PROforma approach is providing argumentation for a specific recommendation. Therefore, when entering recommendations, designer can specify evidence for and against such as recommendation.

One of the main purposes of GEM is to support all the information about the guideline in an organized structure. The actual logic of the guideline is a small part of the GEM Ontology. Every guideline step has associated data about the source, strength of evidence and other explanatory information.

I. Does the software support maintenance of multiple versions with rollback and compare functionality?

Protégé-2000 does not provide any explicit support for maintaining multiple versions.

In GEM Cutter, the XML files can be saved with different names and compared against each other using XML utilities.

Verification, Simulation & Localization

J. Verification – What are the mechanisms to verify the guideline knowledge base? Internal scenario data integrity and consistency? Compliance with external vocabulary standards? Compliance with syntax standards for logic expression?

Protégé-2000 supports a constraint language called PAL which can be used to write complex integrity constraints on the knowledge base. PAL allows developers to make general assertions about relationships among objects in Protégé-2000 (e.g., “all criteria instances are referenced, “nodes” in a

diagram should be connected to other nodes”), and to check if these relationships hold directly in the knowledge base. There are also inherent type checking.

During authoring/encoding, CG-AM provides three types of consistency checking: (1) Name and range checking against values in the Clinical DB; (2) Logical consistency checking (e.g., are decision actions always preceded by query actions); and (3) Temporal consistency checking -- a semantic check of temporal constraints within the guideline (e.g. can overall duration specified contain all necessary actions).

K. Simulation – Does it provide support for guideline simulation so that new guideline knowledge can be rapidly tested?

In Protégé-2000, end-user applications that take the ontologies and the knowledge base as input can be plugged in as tabs just like utility functions. Since changes in the knowledge base are immediately available to the application, they can be tested rapidly using the application tab. This facility was effectively used in the ATHENA project when building a hypertension advisory system. Using the application tab, domain experts could rapidly test the advisory system and the entered hypertension guideline knowledge base. They could modify parts of the knowledge base, and immediately see the effects of their changes in the advisories generated by the application. They could also verify the knowledge base against different patient data.

Arezzo's Protocol Tester is part of the development environment. It has an execution engine that is able to execute tasks by carrying out actions or finding out the current state of the environment by making requests to a human user or software system (such as a database). It displays a set of decision options that it is recommending, arguments for each option. Thus a complete guideline can be executed within the development environment facilitating rapid testing of the system.

L. Localization – Does it provide support for localizing guidelines for specific situations and institutions?

There is no explicit support for localization in Protégé-2000.

The CG-AM "representation formalism" was designed with the aim of representing "contextual limitations", such as availability of clinical and other resources. In addition it integrates a "Resource DB", which lists resources (e.g. CT, NMR) available in a given hospital.

SUMMARY of Protégé-2000 Evaluation

M. What are its strengths?

- Protégé-2000's extensible component-based architecture and configurable GUI greatly facilitates customizing knowledge-acquisition for given domains.
- Automatic generation of domain-specific user-interface forms cuts down on the time and effort needed to go from building knowledge models to acquiring knowledge via the models. It exposes the guideline model to the domain-specialists immediately. This rapid turnaround can be vital to guideline model evolution and experimentation.
- Custom user-interface widgets such as the Diagram widgets can be integrated to ease knowledge acquisition of complex information.
- Utility functions such as terminology services, and end-user applications can be plugged-in easily to expand the support for knowledge acquisition.
- PAL constraint language is expressive and can be used to write complex integrity constraints on the knowledge base. Another use of PAL is in writing decision-criteria which define patient-specific constraints that must be evaluated during guideline execution.
- The organization of knowledge bases as projects, and the notion that a project can include other projects allows building larger custom-tailored knowledge bases using reusable smaller knowledge bases. For example, separate knowledge bases can be built to store guideline model concepts and different domain concepts, such as hypertension or cancer domains. Another key feature advanced by the inclusion facility is the notion of integrating multiple guidelines. For example, a guideline on treatment of diabetic patient may include guideline on management of hypertension or foot care.

N. What are its weaknesses?

- Special-purpose knowledge acquisition tools such as Arezzo and AsbruView are tightly coupled with the underlying guideline model. Such tools generally provide elegant and sophisticated user-interfaces that are highly directed. Protégé-2000 provides generic user-interface forms that may not be intuitive to use for a domain-specialist. For example, Protégé-2000 associates one form with each class and does not facilitate logical grouping of classes into a single form. Therefore, it provides a general forms-based view of guideline knowledge in a knowledge base, but not a concise and domain-specific view. Domain

specialists can find it daunting to review the entered knowledge form by form.

- There are no 'wizards' to guide the domain-specialist through the knowledge acquisition process. Thus knowledge-entry can be unstructured, and fragmented. Domain specialists may lack the sense of how to go about entering knowledge, what they have entered so far and what needs to be entered.
- There has been no demonstration of how a standard terminology service would be integrated with Protégé-2000, and used in the guideline modeling and knowledge acquisition process.
- During encoding a guideline, domain specialists need to be able to share the information easily for review, and more often all of them will not have access to, or want to use the knowledge acquisition tool. Protégé-2000 does not support an export function to translate the contents of the knowledge base into a human-readable format. This puts an additional burden on the domain specialists to maintain a textual document of the guideline knowledge rules.