

Research Paper ■

# Comparing Computer-interpretable Guideline Models: A Case-study Approach

MOR PELEG, PHD, SAMSON TU, MS, JONATHAN BURY, MBChB,  
PAOLO CICCARESE, MSc, JOHN FOX, PHD, ROBERT A. GREENES, MD, PHD,  
RICHARD HALL, MSc, PETER D. JOHNSON, MBBS, NEILL JONES, MBBS,  
ANAND KUMAR, MBBS, SILVIA MIKSCH, PHD, SILVANA QUAGLINI, PHD,  
ANDREAS SEYFANG, MSc, EDWARD H. SHORTLIFFE, MD, PHD,  
MARIO STEFANELLI, PHD

**Abstract Objectives:** Many groups are developing computer-interpretable clinical guidelines (CIGs) for use during clinical encounters. CIGs use “Task-Network Models” for representation but differ in their approaches to addressing particular modeling challenges. We have studied similarities and differences between CIGs in order to identify issues that must be resolved before a consensus on a set of common components can be developed.

**Design:** We compared six models: Asbru, EON, GLIF, GUIDE, PRODIGY, and PROforma. Collaborators from groups that created these models represented, in their own formalisms, portions of two guidelines: the American College of Physicians–American Society of Internal Medicine’s guideline for managing chronic cough and the Sixth Report of the Joint National Committee on Prevention, Detection, Evaluation, and Treatment of High Blood Pressure.

**Measurements:** We compared the models according to eight components that capture the structure of CIGs. The components enable modelers to encode guidelines as plans that organize decision and action tasks in networks. They also enable the encoded guidelines to be linked with patient data—a key requirement for enabling patient-specific decision support.

**Results:** We found consensus on many components, including plan organization, expression language, conceptual medical record model, medical concept model, and data abstractions. Differences were most apparent in underlying decision models, goal representation, use of scenarios, and structured medical actions.

**Conclusion:** We identified guideline components that the CIG community could adopt as standards. Some of the participants are pursuing standardization of these components under the auspices of HL7.

■ *J Am Med Inform Assoc.* 2003;10:52–68. DOI 10.1197/jamia.M1135.

Affiliations of the authors: Stanford Medical Informatics, Stanford University School of Medicine, Stanford, California (MP, ST); Advanced Computation Laboratory Cancer Research UK, London (JB, JF); Medical Informatics Laboratory, Department of Computers and Systems Science, University of Pavia, Pavia, Italy (PC, AK, SQ, MS); Decision Systems Group, Harvard Medical School, Brigham and Women’s Hospital, Boston, Massachusetts (RAG); Sowerby Centre for Health Informatics at Newcastle, University of Newcastle, Newcastle upon Tyne, United Kingdom (RH, PDJ, NJ); Institute of Software Technology and Interactive Systems, Vienna University of Technology, Vienna, Austria (SM, AS); Department of Medical Informatics, Columbia University, New York (EHS).

The authors from Stanford, Columbia, and Harvard Universities are supported in part by Grant LM06594 with support from the Department of the Army, Agency for Healthcare Research and

Quality, and the National Library of Medicine. The EON project has been supported by grant LM05708 from the National Library of Medicine. The Asgaard Project is supported by Fonds zur Foerderung der wissenschaftlichen Forschung (Austrian Science Fund), grants P12797-INF and P15467-INF. The GUIDE team has been partially funded by the European Commission through the project PatMan (Health Care Telematics Applications Programme). The PRODIGY project is funded by NHS Executive Primary Care. The PROforma project is funded by Cancer Research Funding.

Correspondence and reprints: Mor Peleg, PhD, Stanford Medical Informatics, Medical School Office Building X-208, 251 Campus Drive, Stanford, CA 94305-5479; e-mail: <peleg@smi.stanford.edu>.

Received for publication: 03/28/01; accepted for publication: 08/16/02.

Clinical guidelines are increasingly being used to support clinician decision-making. They are intended to improve the quality of patient care and reduce costs. Conventional narrative guidelines present population-based recommendations. However, clinician behavior is most effectively influenced through patient-specific advice, particularly if delivered during patient encounters.<sup>1,2</sup> Unfortunately, accessing information contained in conventional guidelines may be difficult, as is applying it to a specific patient during a consultation. Guideline-based point-of-care decision support systems have the potential to address this problem. A prerequisite for developing these systems is creating computer interpretable representations of the clinical knowledge contained in clinical guidelines. A number of groups are actively developing computer interpretable guideline (CIG) representation languages for this purpose.<sup>3-8</sup> Each has adopted different approaches, reflecting its members' interests and expertise. Nevertheless, many of the approaches share a hierarchical decomposition of guidelines into networks of component tasks that unfold over time. This approach has been described as the "task-based paradigm."<sup>9</sup> We use the term Task-Network Models (TNMs) to describe guideline-modeling formats based on this approach.

There is understandable interest in developing a standardized, consensus guideline representation language or at least in identifying common components that different formats may share. Previous reviews have relied on published descriptions of individual methodologies.<sup>10-12</sup> Although these reviews have identified some similarities and differences, more systematic and objective comparisons are required. This study is a direct comparison of six different guideline representation languages. We studied, in detail, how developers of each language modeled two sample guidelines. The complete case study statement, modeled guidelines, and their documentation, as well as detailed analyses not included in this paper, are available online at <<http://www.openclinical.org/gmmcomparison.html>>.

## Background

The Arden Syntax<sup>13</sup> is perhaps the best-known language for representing clinical knowledge in patient-specific decision-support systems. It is a rule-based formalism developed for encoding individual clinical rules as Medical Logic Modules (MLMs). Augmented Decision Tables (ADTs)<sup>14</sup> have been used to model guidelines. ADTs go beyond Arden's rule-based for-

malism by augmenting decision-table rules with additional information, such as probability and utility information. Although the Arden Syntax has been adapted for representing guidelines by employing interacting MLMs,<sup>13</sup> neither MLMs nor ADTs provide full support for conceptualizing a multistep guideline that unfolds over time. The TNM approach has arisen in response to this problem. TNM languages typically provide modeling primitives specifically designed for representing complex, multistep clinical guidelines and for describing temporal and other relationships between component tasks. Unlike rule-based systems, TNMs can explicitly model alternative pathways or sequences of tasks (i.e., control flow), and they provide tools for visual representation of plans and the organization of tasks within them.

## Guideline-modeling Methodologies Compared in This Study

**Asbru**<sup>7</sup> is being collaboratively developed at Ben Gurion University and the Vienna University of Technology. It is a time-oriented, intention-based, skeletal-plan specification language that is used to represent clinical protocols.<sup>15</sup> Skeletal plans capture the essence of a procedure but leave room for execution-time flexibility in the achievement of particular intentions. Asbru's developers have enriched skeletal plans by (1) characterizing plan attributes such as intentions, conditions, and effects; (2) adding a rich set of ordering of plans; and (3) defining temporal dimensions of states and plans. Uncertainty in temporal scope and parameters can be expressed by bounding intervals.

**EON**<sup>5</sup> was developed at Stanford University and provides a suite of models and software components for creating guideline-based applications. It views the guideline model as the core of an extensible set of models, such as a model for performing temporal abstractions. EON uses a task-based approach to define decision-support services that can be implemented using alternative techniques.<sup>16</sup> Its guideline execution server uses formalized clinical guidelines and patient data to generate situation-specific recommendations. A temporal data mediator supports queries involving temporal abstractions and temporal relationships. A third component provides explanation services for other components.<sup>17</sup>

**The Guideline Interchange Format version 3 (GLIF)**<sup>3</sup> has been collaboratively developed by groups at Columbia, Stanford and Harvard Universities (the InterMed Collaboratory). GLIF stresses the importance of sharing guidelines among

different institutions and software systems. GLIF tries to build on the most useful features of other guideline models and to incorporate standards that are used in health care. Its expression language was originally based on the Arden Syntax,<sup>18</sup> and its default medical data model is based on the HL7 Reference Information Model (RIM).<sup>19</sup> A subsequent object-oriented language, GELLO,<sup>20</sup> is being refined for consideration as an HL7 standard.

**GUIDE**<sup>8</sup> is part of a guideline modeling and execution framework being developed at the University of Pavia. It supports (1) integrating modeled guidelines into organizational workflows, (2) using decision-analytical models such as decision trees and influence diagrams, and (3) simulating guideline implementation in terms of Petri nets.<sup>21</sup> GUIDE considers issues such as patient data, the implementing facility's organizational structure, and resource allocation. This paper considers the guideline model as presented in the GUIDE tool, which is a graphical authoring tool that a modeler uses to create a guideline flowchart.

**PRODIGY**<sup>4</sup> was developed at the University of Newcastle upon Tyne. It provides support for chronic disease management in primary care. The PRODIGY project's aim is to produce the simplest, most readily comprehensible model necessary to represent this class of guidelines. Teams of clinicians have used Protégé's knowledge engineering environment<sup>22</sup> to encode three complex chronic disease-management guidelines. Over 150 guidelines encoded in PRODIGY's simpler Release One model have been translated into the current model. Two vendors have integrated identical PRODIGY components into their clinical information systems for general practitioners.<sup>23</sup>

**PROforma**<sup>24</sup> was developed at the Advanced Computation Laboratory of Cancer Research, UK. It combines logic programming and object-oriented modeling<sup>24</sup> and is formally grounded in the R<sup>2</sup>L Language.<sup>9</sup> One aim of the PROforma project is to explore the expressiveness of a deliberately minimal set of modeling constructs. PROforma supports four tasks: actions, compound plans, decisions, and enquiries. All tasks share attributes describing goals, control flow, preconditions, and postconditions. The simple task ontology should make it easier to demonstrate soundness and to teach the language to encoders.

## Research Question

We identified similarities and differences among the various guideline-modeling methodologies as well as

factors that account for variability among them. A secondary question was to determine the extent to which areas of similarity could facilitate developing a shared consensus model.

## Methods

The study was coordinated by MP and ST. Members of groups responsible for each of the modeling methodologies submitted models of two clinical guidelines represented in their particular modeling languages. The models were systematically analyzed according to predefined axes of comparison.

### Selection of Guidelines for Modeling

The study coordinators selected sections from larger guidelines relating to cough<sup>25</sup> and hypertension.<sup>26</sup> These sections presented modeling challenges in each dimension of comparison described below, while being concise enough to allow efficient modeling and analysis. We assumed that the guideline statements were valid, because the guidelines were created by panels of experts, based on experimental evidence, and were endorsed by accredited health organizations.

### Dimensions of Comparison

ST identified eight dimensions of comparison and circulated them to all collaborators. These dimensions fell into two broad categories: structuring guidelines as plans of decisions and actions and linking a guideline to patient data and medical concepts. The dimensions were (1) organization of guideline plans, (2) goals, (3) model of guideline actions, (4) decision model, (5) expression language, (6) data interpretation/abstractions, (7) medical concept model, and (8) patient information model.

### Analysis

MP and ST analyzed the resulting guideline models. As far as possible, models were studied using the authoring groups' own modeling environments. Thus, we used Protégé to study models in GLIF, EON, and PRODIGY and Arezzo to study the two models in PROforma. For the Asbru language, we studied AsbruView<sup>27</sup> screen shots showing the overall organization of the models and the XML code representing details of attributes. Similarly for GUIDE, we viewed screen shots for the GUIDE authoring tools and the associated relational database schema and SQL queries showing model details. The modelers of each methodology used computerized tools to

check the syntactic correctness of the encoding, including parsing of logical expressions, data type matching, and integrity constraints (see <http://www.openclinical.org/gmmcomparison.html>). We worked in close collaboration with representatives of each of the formats throughout the analysis, ensuring that their models represented the guideline sections as fully as each format would allow. The results of the analysis were iteratively collated by MP and circulated to all contributors until consensus was reached.

## Results

With one exception, all of the models accurately represented the original guideline knowledge. The exception was that the Asbru encoding of the hypertension guideline did not include a reasoning module or domain ontology for selecting an optimal drug. This encoding assumed that the model would call an external reasoning-module. Analysis of the different models revealed a number of differences in how the different formats approach aspects of guideline modeling. Results here are presented by dimension; more detailed information is online.<sup>28,29</sup>

### Dimension 1. Organization of Guideline Plan Components

#### Plans and Nesting of Components

Two unifying features of TNM languages are the decomposition of guidelines into networks of component tasks and the ability to express various arrangements of these components and interrelationships between them. Although all the languages use the concept of task network to describe collections of tasks, there are differences in the way that the concept is used. This article uses the term *plan* as defined in the Merriam-Webster dictionary: “an orderly arrangement of parts of an overall design or objective” to denote these task networks.

Asbru, PROforma, and GUIDE each use only a single, generic class of plan object, whereas PRODIGY and EON distinguish between *Management Guidelines* (major management decisions and flow of control) and *Consultation Templates* (context-dependent recommendations on actions that are not part of a management decision, on data to collect or display, and on patient education). GLIF also has two kinds of plans: Guidelines and Macros. GLIF *Guidelines* are similar to EON’s Management Guidelines. *Macros* provide a means to declaratively specify a procedural pattern that appears in guidelines as a single con-

struct that is realized by a set of GLIF steps.<sup>3</sup> All of the methodologies provide nesting mechanisms to simplify top-level plans and support the reuse of sub-guidelines (Figure 1).

#### Plan Components

The modeling formats use different terminology to refer to the various types of plan components. However, with minor variations in implementation, all formats provide constructs of some form for representing **decisions** (Dimension 4), **actions** (Dimension 3), and nested **subplans**. In addition to these core constructs of action, decision, and plan/subplan, the different modeling languages provide various other primitives (Table 1).

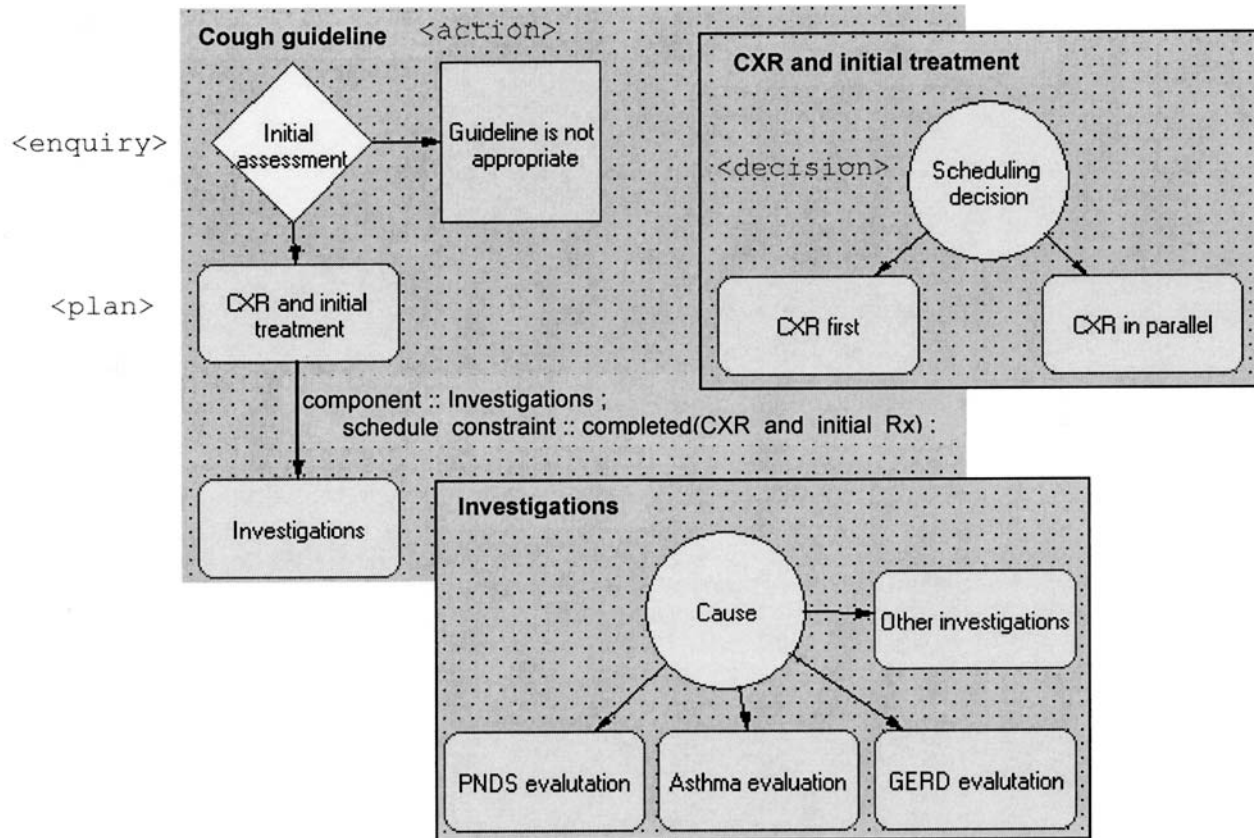
A **Scenario** (EON and PRODIGY) or **Patient State** (GLIF) is a plan component that defines a particular patient management context. It is characterized by patient conditions (e.g., hypertension) and/or their treatments (e.g., taking low-dose thiazide diuretics)—and possibly by clinical settings (e.g., outpatient clinics).<sup>12</sup> Scenarios serve as entry points into guidelines and are useful in multi-encounter guidelines, in which patients may enter the guideline at different places in the plan (see below). In EON and PRODIGY, consultation templates are associated with scenarios to describe scenario-specific actions.

**Branch Steps** (EON, GLIF, PRODIGY, and GUIDE) and **Synchronization Steps** (EON, GLIF, and GUIDE) model parallel paths in a guideline plan. PROforma and Asbru can achieve parallel as well as sequential execution without having branch and synchronization steps (see below).

The **Wait Step** in GUIDE is used to introduce delays. GUIDE’s **Monitor** task checks for conditions at specified durations, and can be used to monitor goal conditions. GUIDE measures the monitored variable with a certain frequency, and the modeler can specify actions that should occur, depending on the value.

#### Sequential, Parallel, Cyclical, and Iterative Plans

Care processes may involve sequenced, iterative, and possibly concurrent activities occurring over time. All the methodologies support these modes of plan organization. All except PROforma use different constructs to specify sequential versus parallel plan execution. PROforma uses scheduling constraints to govern task execution, and both sequential- and parallel-execution are implicitly supported (see Figure 1).



**Figure 1.** Arezzo's graphical view of the cough guideline encoding in PROforma. The top-level cough guideline is shown on the top left. The two inserts show nesting of the two plans of the top-level guideline. The scheduling constraint in the top-level guideline states that the component "Investigations" should not be executed until the component "CXR and initial treatment" has completed.

All of the modeling methods can combine guideline steps in directed cyclical graphs. All except PRODIGY have explicit constructs to support iteration or cycling of plans and/or plan-components. Generally, iteration is specified by providing the time or trigger of the first repeat, the duration of each cycle, the repeat frequency, the maximum number of repeats, the completion condition, and the abort condition. Asbru, EON, and GLIF can define fuzzy iteration frequency (e.g., take drug every 3–4 hours).

Note that sequential-, parallel-, and cyclical-execution define part of the control flow of guideline plans. Other elements of control flow are typically handled by decision models (Dimension 3), which conditionally direct control flow into selected branches of the guideline model.

#### Entry Points into Guideline Plans

Different entry points into a guideline are necessary to represent multistep clinical processes, which may take

place over several encounters. Ideally, CIG-based decision-support systems should keep track of a patient's state at one encounter and use this information to automatically provide an appropriate entry-point at subsequent encounters. Although this process is complicated by changes in a patient's health, all of the methodologies aim to support multiple entry points into a guideline. We identified two distinct approaches. PRODIGY, EON, and GLIF use specific modeling components to represent entry points. They are called Scenario or Patient State components and facilitate a patient's automatic entry into an appropriate plan or subplan. Other models use expressions referring to patient states in decision criteria or preconditions that affect guideline control flow.

#### Dimension 2. Specification of Goals/Intentions

Guideline modeling methods fall into two groups according to how they model guideline intentions or goals. PRODIGY and GLIF specify goals as text

Table 1 ■

## Terms Used by Guideline Modeling Methodologies to Refer to Plans and Actions\*

Model	Plan	Plan component					
		Branching	Action	Decision	Scenario	Special	Subplan
Asbru	Plan	Plan type	Plan	Plan precondition			Recursive plan
EON	Management Guideline	Branch Synchronization	Action	Decision	Scenario		Subguideline step
	Consultation Guideline	Consultation—branch	Consultation—action				Consultation guideline part of scenario
GLIF	Guideline, Macro	Branch Synchronization	Action	Decision	Patient-state		Guideline or Macro called in Action or Decision steps
GUIDE	Guideline	Synch-&, Synch-Or	Task	Deterministic decision, non-deterministic decision		Wait Monitor	Any task can be decomposed
PRODIGY	Decision/Management map		Action		Scenario		Subguideline Step or called in Action step
	Consultation Template	Consultation—branch	Consultation—action				Consultation template part of scenario
PROforma	Plan	Action, Enquiry, Decision	Action, Enquiry	Decision			Plan task

\*See Dimension 1.

strings. Text is presented to users or used for indexing libraries of guidelines and searching them. The other methods represent intentions and goals as formal expressions used to control the execution-state of plans. Asbru represents plan intentions as context-dependent temporal patterns (e.g., “give three courses of chemotherapy within three months”). Encoders can specify that an intention *achieve*, *maintain*, or *avoid a state* or *action*, either *during* a plan’s execution or *after* it has terminated. Thus, Asbru can specify 12 different intentions (Figure 2). Although this particular syntax is perhaps the most developed of those studied here, EON, PROforma, and GUIDE also represent goals formally and use the resulting expressions to influence control flow and recommendation generation or interpretation.

### Dimension 3. Model of Guideline Actions

Guideline actions are the modeling primitives used to represent the tasks described by a clinical guideline (e.g., prescription, clinical investigation). Table 2 summarizes the characteristics of these constructs.

#### Structuring Medical Actions

Although all of the models can specify medical actions, not all of them do so with specialized struc-

ured classes. Medical actions in GLIF, EON, PRODIGY, and GUIDE contain slots for mapping their instances into controlled vocabulary terms. In GLIF, guideline encoders specify medical actions by defining the attributes of a patient data item according to the data model of the HL7 Reference Information Model (RIM).<sup>19</sup> The HL7 RIM is general enough to represent the data structure for a wide range of medical data and concepts in a uniform manner while using a small number of classes. Patient data can simply be modeled as *observations*, *medications*, and *procedures*. These classes contain a *mood* code that distinguishes each as an event that occurred, a definition, intent, order, etc. Figure 3 shows a Patient Data Item representing a medication order for angiotensin-converting enzyme inhibitor (ACEI). EON and PRODIGY can represent many specialized medical actions. They include referrals, acute prescriptions, scheduling, asserting conclusions, and modifying drug treatments.

#### Action Refinement

PRODIGY refines drug choice through a model of drug regimen (e.g., ACEI), regimen component (low-dose ACEI), and prescribable item (captopril, 12.5 mg; 0.5 tablets bd). A regimen component is refined to prescription details, accounting for drug interactions, contraindications, and patient sensitivities.

```

<plan name="hypertension-treatment">
  <intentions>
    <intention type="overall-state" verb="achieve">
      <parameter-proposition parameter-name="systolic-blood-pressure">
        <value-description type="equal">
          <qualitative-constant value="normal"/>
        </value-description>
        <time-annotation>
          <time-range>
            <starting-shift include-limit-value="yes">
              <latest>
                <numerical-constant unit="month" value="1">
                  </numerical-constant>
                </latest>
              </starting-shift>
            </time-range>
          </time-annotation>
        </parameter-proposition>
      </intention>
    </intentions>
  </plan>

```

**Figure 2.** Expressing intentions in Asbru. The “hypertension-treatment” plan’s intention is achieving an overall state of normal systolic blood pressure within 1 month of the plan’s execution.

PRODIGY suggests the best product, based on previous use of the drug, formulary status, and guideline-specific criteria. EON refines drug choice through a drug-ontology: a drug category, such as ACEI, is refined to preferred drugs (e.g., lisinopril). The other guideline models can provide similar functionality of action refinement by defining a subplan that expands the details of a drug-order action.

#### Temporal Constraints

All of the models can specify constraints on the start times of guideline plan components. The models differ in their abilities to specify constraints on end time and duration. Figure 2 shows a “latest start time” constraint in Asbru.

#### System Actions

All methodologies model patient data queries and message sending in roughly the same way. The actions of the methodologies differ in their abilities to accept parameters, return results, assign variables with values, and inherit knowledge roles, such as complete and abort conditions.

#### Representing Effects of Actions and Reasoning with Them

Asbru and PROforma are the only modeling languages that support expressing effects of actions, thus allowing reasoning about actions based on these effects. In Asbru, plan effects can be used to select among alternative plans and to express causal relationships (e.g., chronic cough is caused by PNDS with a likelihood of 0.33). PROforma’s postconditions are semantically different in that they represent assertions that become true after an action is completed (e.g., immunocompromization is a postcondition of chemotherapy). Task selection on the basis of postconditions has not yet been implemented in PROforma, although they can be used to affect downstream control flow.

#### Dimension 4. Decision Models

Decision-making is central to most clinical guidelines. The modeling methodologies that we studied use a variety of decision models, including switch constructs, argumentation schema, decision trees, and influence diagrams. Some support multiple

Table 2 ■

Characteristics of Actions are Modeled by Different Guideline Modeling Methodologies\*

	Asbru	EON	GLIF	GUIDE	PRODIGY	PROforma
Medical actions	+	+	+	+	+	+
		(Structured)	(Structured)	(Structured)	(Structured)	
Action refinement	+	+	+	+	+	+
		(Using concept model)		(Using concept model)	(Using drug model)	
Nesting	+	+	+	+	+	+
	See (1)	See (1)	See (1)	See (1)	See (1)	See (1)
Temporal constraints on actions/ activities						
Start time	+	+	+	+	+	+
End time	+			+	+	+
Duration	+	+	+	+	+	
System actions						
Actions accept arguments (e.g., action of determination of best drug for a patient accepts patient comorbidities and current medications)	+		+			+
Actions return values (e.g., drug determination returns drug)	+		+	From sub-guideline to higher-level decision		+
						(Decision)
Inheriting action knowledge roles (e.g., complete conditions inherited by subplans)	+					
Actions assign variables (e.g., test_done := true)	+		+	+	+	+
Iterative and cyclical actions	+	+	+	+	+	+
	See (1)	See (1)	See (1)	See (1)	See (1)	See (1)
Representing and reasoning with effects of actions	+					+

\*The comment "See (1)" indicates that Dimension 1 has further explanations.

decision models (Table 3). Some of the methods also support decision-making through calls to external functions.

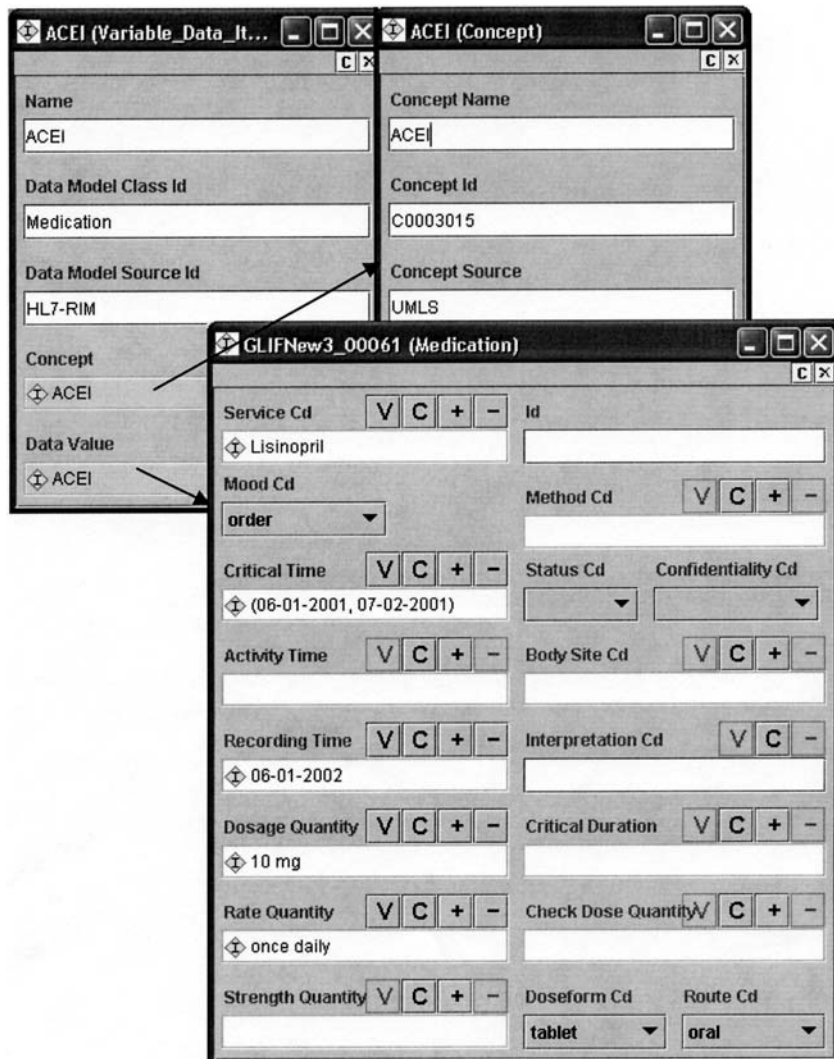
Switch Constructs

Switching describes mutually exclusive branching of guideline control flow. The decision on which branch is taken is deterministic. GLIF, EON, and GUIDE use specific primitives to model this process. Each tests whether an expression matches one of a number of constant values and forces execution to branch accordingly. The other formats do not use explicit constructs to represent switching, but achieve similar results through mutually exclusive expressions in precondition or argumentation rules, for example.

Argumentation Rules For/Against Choice Alternatives

Several methods use argumentation schema to express preferences for alternative candidates of non-deterministic decisions (i.e., decisions where multiple alternatives are justifiable). Different options are associated with different sets of *arguments*—conditions, which if true, provide some degree of support for that option. Note that this support may be negative. Support may be expressed numerically, as in cost/utility schemas. Alternatively, symbolic weights, such as *for*, *against*, *confirming*, and *excluding*, may be used. Both symbolic and *numeric* approaches may the use of methods for the aggregation of weights. Figure 4 shows PRODIGY rules for and against adding a diuretic to an ACE inhibitor.





**Figure 3.** A Cough data item in GLIF is defined by a concept whose code is taken from UMLS (right) and by an HL7 RIM Medication class (lower part of figure).

### Decision Trees and Influence Diagrams

GUIDE uses decision trees or influence diagrams to represent nondeterministic choices<sup>21</sup> by providing a link to Java applets that build trees or diagrams that are specific to a situation.

### The Relationship between Decision-making and Control Flow

In all of the methodologies except *PROforma*, decision-making is explicitly coupled to commitment to a choice (switch constructs are one example of this process). Action sequencing in *PROforma* is governed by satisfying scheduling constraints and preconditions of individual tasks rather than through explicit *go to* relationships between tasks. If a decision's result will govern subsequent tasks, the result can be referenced in the precondition expressions of relevant subsequent tasks.

### Authorizing Decisions

All of the groups recognize that user intervention or confirmation may be required. All except *PRODIGY* allow the system to make some decisions automatically. Decisions in *PRODIGY* always require confirmation. This fact reflects the *PRODIGY* team's emphasis on maintaining clinician autonomy and their goal of producing interactive systems for use during clinical encounters.

### Dimension 5. Expression/Criterion Languages Used to Specify Decision Criteria

The methodologies use expression languages to represent decision criteria, including pre- and postconditions of guideline plan components, criteria that control plan execution states (filter, setup, suspend, reactivate, complete, and/or abort conditions in *Asbru*),

Table 3 ■

## Decision Models Used by the Different Methodologies\*

	Asbru	EON	GLIF	GUIDE	PRODIGY	PROforma
Switch (see dimension 4)	+	+	+	+	+	+
	(Task pre-conditions in XOR)	(Case)	(Case)	(Deterministic one-of)	(Using argumentation rules)	(Plan preconditions in XOR)
Argumentation rules		+	+		+	+
Decision trees/influence diagrams				+		
				(Non-deterministic one-of)		
External functions	+		+	+	+	+
Extensibility		+	+	+	+	+
Decision-making commits choice of alternative	+	+	+	+	+	Not necessarily
Authorization required	±	±	±	±	±	±
Preferences	Symbolic	+	+		+	+
					Preferred option	
	Weighted numeric					+
	Cost function	+			+	+
	Formal utility theory			+		

\*The symbol +/- means that the guideline encoder may specify that user authorization is required.

rules for and against decision alternatives, goal criteria, and definitions of patient states. All models support various standard logical, arithmetic and comparison operators. The different methodologies use quite different expression languages, although EON and PRODIGY share some criteria templates defined as objects with certain attributes. Note that, when this study was conducted, GLIF used the Guideline Expression Language (GEL) to specify criteria and expressions.<sup>30</sup> GLIF now uses GELLO,<sup>20</sup> an extensible object-oriented expression language that supports a superset of the functions supported by GEL.

#### Presence Criteria

Presence criteria check for patient data items (e.g., presence of diabetes). PRODIGY, EON, GLIF, and GUIDE model presence criteria by giving explicit definitions of terms to be checked, together with a method for looking for the terms in the local EMR. Figure 5a shows a presence criterion in EON. PROforma and Asbru model presence criteria by defining a Boolean data item called "concept present." This item is treated like all other data items.

#### Template-based Criteria

Clinicians can use templates in EON and PRODIGY to encode relatively simple decision criteria. The tem-

plate criteria look for qualitative and quantitative observations, medications, and other types of EMR entities (see Dimension 8). They can declaratively express simple temporal constraints on these entities of the form within a certain interval of a time point (see Figure 5a).

#### First-order Logic Criteria

EON uses Protégé-2000's constraint language (PAL) to define decision criteria in a subset of first-order predicate logic written in the Knowledge Interchange Format syntax.<sup>17</sup> Figure 5b shows a PAL criterion for ACE Inhibitor contraindications. PROforma is a first-order language, and Arezzo—a PROforma toolkit—supports a number of first-order logic features through the use of open formulas in its criterion language (i.e., variables in conditions). These features are particularly useful in decision-making. GUIDE uses SQL to write decision criteria. It thus supports some first-order logic criteria.

#### Temporal Criteria

All of the methodologies support temporal criteria. Temporal criteria may refer to time stamps of data and task enactment events. Figure 5c shows an example of a temporal criterion in EON. The methodologies differ in the complexity of temporal expressions

**Figure 4.** The PRODIGY choice model. Note the rules for and against giving diuretics to a patient who is already on an ACE inhibitor. The “rule in condition” expresses strong preference for an alternative. The “greyed in condition” is a possible argument for the alternative. The “greyed out condition” is an argument against the alternative. The “rule out condition” is a rule excluding the alternative. The rule in and rule out conditions are objects that include formal criteria that can be evaluated against patient data and natural language descriptions of the criteria. The figure shows only the natural language form. If none of the conditions apply, one alternative can be marked “preferred” by default.

they represent. Asbru and EON support temporal abstractions (see Dimension 6). GLIF supports temporal operators of the Arden Syntax logic grammar.<sup>18</sup>

#### If...Then...Else and Switch Statements

GLIF, Asbru, and PROforma's Arezzo can use *if...then...else* and *switch* statements in their expression languages. For example, the following PROforma expression assigns a value of 140 to the variable *TargetSBP* if the value of the variable *Target\_BP\_decision* is equal to Standard. Otherwise, *TargetSBP* is assigned the value 130:

```
TargetSBP = if( result_of( Target_BP_decision) =
Standard, 140, 130);
```

#### Context-dependent Expressions

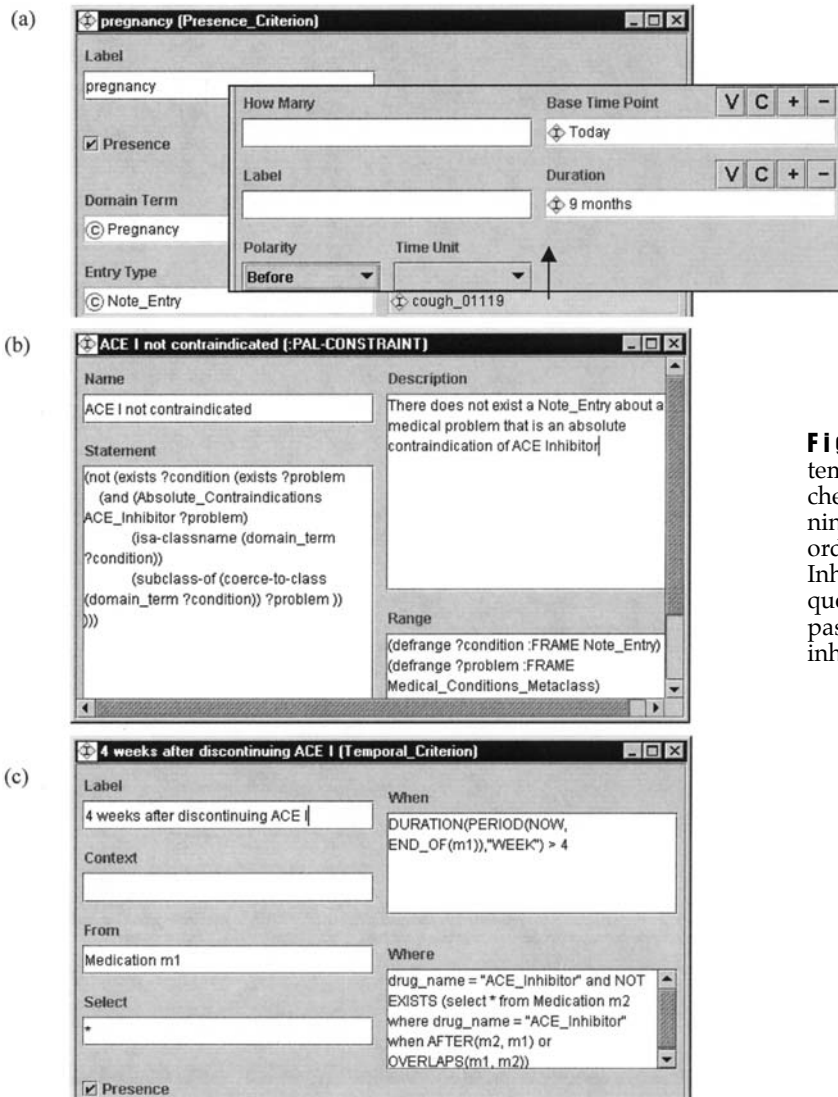
Asbru supports expressions that depend on special context variables such as “presence of diabetes mellitus.” Values of context variables can be Boolean or

qualitative symbols. They can be set through user input, obtained through data abstraction from other parameters, or set by plans during execution. Context expressions are used in setting limits of parameter definitions and in specifying temporal patterns, argument dependencies on measurable parameters, and plan effects.

Other models do not have special constructs that hold the contexts of parameters. However, in EON and PRODIGY, scenarios define contexts for actions and expressions encoded in the consultation guidelines associated with them. The other models can use their general expression constructs to set context, as shown in *if . . . then . . . else* and *switch* statements.

#### Dimension 6. Data Interpretations/Abstractions

All of the models define abstractions, which aid in conceptualizing guideline logic and interpreting data. We found four types of abstractions. Three are



**Figure 5.** Criteria languages in EON. (a) A template query for a presence criterion that checks for pregnancy in note entries dating up to nine months before the current date. (b) A first-order logic criterion that checks if an ACE Inhibitor is not contraindicated. (c) A temporal query that checks whether four weeks have passed since the administration of an ACE inhibitor.

discussed below. The fourth, scenarios and patient-state steps, was discussed in Dimension 1.

#### Temporal Abstractions/Temporal Patterns (Trends)

Asbru and EON can use systems that perform temporal abstractions to abstract conditions that persist over time, based on raw, time-stamped values.<sup>29</sup>

#### Definitions of Abstract Terms

Using given data, all of the methodologies can use formal expressions to define abstract terms. For example, *isolated systolic hypertension* can be defined as a situation in which patients not taking anti-hypertensive agents have systolic blood pressures of at least 140 mmHg and diastolic blood pressures less than 90 mmHg. Note that this definition requires a definition of antihypertensive agent (concept).

#### Terminology Abstractions via Classification Hierarchies

PRODIGY, EON, and GLIF can create hierarchies of medical concepts and reason about them by writing expressions that utilize the hierarchies. In a taxonomic hierarchy, a concept that is a parent of other concepts creates an abstraction for the lower-level concepts. For example, *ACE inhibitor* is an abstraction of individual drugs, such as *lisinopril*. PRODIGY has a terminology mediator that uses hierarchies in READ codes to map specific terms to abstract terms. EON creates taxonomic hierarchies using Protégé's frame-based knowledge-engineering environment. GLIF can define hierarchies of concepts using the *Concept\_Relationship* class, with a relationship-type of "is-a." In GUIDE, each guideline task can be associated with a single SNOMED code that represents a

Guideline_Outputs				
Code	Description	Table name	Data Type	Guidelines
111	Patient_id	Patient_anagraphic	Number	GERD
112	Pregnancy	Patient_table2	Boolean	GERD

**Figure 6.** Mapping guideline data items to EMRs in the GUIDE model. The description column gives the name of the data item that guidelines use (Guidelines column). The three other columns are related to the EMR; each code is unique for that attribute. If possible, it is the SNOMED code. Data Type is the data type of the attribute, and Table name is the name of the EMR table where the attribute value is stored.

clinical task (e.g., cardiovascular stress testing). The hierarchy of SNOMED codes is also utilized for reasoning: when a user disagrees with a task proposed by GUIDE, he or she must select a new code that corresponds more closely to the desired task. The terminology server displays a set of tasks that are at the same hierarchical level of the SNOMED taxonomy. The rationale is that a physician may wish to use a different method to achieve a goal.

#### Dimension 7. Representation of a Medical Concept Model and Its Use

In *PROforma* and *Asbru*, the concept to which a variable refers is represented by the variable's name. In contrast, medical concept models in other methodologies include classification hierarchies (terminology abstractions via classification hierarchies in Dimension 6), concept definitions, and relationships between concepts that convey medical knowledge.

In all systems that use some kind of concept model, concepts are defined, at least in part, through mapping to terminology systems. In *EON*, concepts in hierarchies can also be defined through explicit definitions of abstract terms (Dimension 6).

*PROforma* models medical knowledge representing contraindications, indications, and drug interactions as part of arguments for decision alternatives. Medical knowledge is not represented as part of *Asbru*'s guideline model, but *Asbru* can access medical knowledge by using function calls. *GUIDE* can represent knowledge in relational database tables. In *GLIF* and *EON*, modelers can represent medical knowledge as instances of *Concept\_Relationships*. In addition, *EON* and *PRODIGY* model medical knowledge as classes in the medical ontology (action refinement in Dimension 3). *PRODIGY* also models this kind of knowledge as part of the rules for selecting an action for a relevant scenario (see Figure 4), or it can use queries to refer to outside information sources.

#### Dimension 8. Patient Information Model

The patient information model is concerned with representing patient data and mapping it to institutional EMR data models. This model defines terminologies (e.g., codes for routes of administration) and the structure of patient data (e.g., the properties of a medication order).

All of the methodologies can represent and manipulate complex data items that group related data values in a single structure. *PROforma*, *GUIDE*, and *Asbru* do not constrain the possible classes of complex objects. *PRODIGY*, *EON*, and *GLIF* define a constrained set of patient data and concept classes. *PRODIGY* and *EON* view patient data as instances of *virtual medical record (VMR)* classes.<sup>31</sup> These VMRs use medical records to abstract a small set of patient data classes that are needed in decision support applications. These classes include Qualitative Observation (e.g., indicating presence of a disease), Quantitative Observation (e.g., a test result), Medication Authorization, Procedure (e.g., a pancreatectomy), and Allergy State. *GLIF* defines data items by associating them with controlled vocabulary codes and by structuring patient data according to medical data models chosen by guideline developers (see Figure 3). The default data model used in *GLIF* is based on the HL7 Version 3 Reference Information Model (RIM)<sup>19</sup> (action refinement in Dimension 3).

By fixing the structure of patient data and the necessary terminology, the patient information models of *EON* and *PRODIGY* facilitate mapping concepts and data to institutional EMRs. The developers of *EON*, *PRODIGY*, and *GLIF* plan to make their patient information models consistent with HL7 Version 3 RIM. They will rely on HL7's messaging standards for their instantiation. Fixing the structure of patient data could facilitate mapping guideline terms to EMRs, but is not required for mapping. For example, in *GUIDE*, tables such as the one

shown in Figure 6 define mapping between the EMR and data needed by the guideline. *PROforma* and *Asbru* use a similar approach.

## Discussion

Authoring CIG models can be time-consuming and may require clinical knowledge as well as technical skill. Moving guidelines encoded in one format into systems using other formats would reduce duplication of effort. The GLIF project originally intended to devise an interchange format to facilitate this process. However, GLIF's developers have acknowledged that this goal is impractical at present. As a result, they are now trying to create a versatile modeling language that will allow guideline models to be shared between different institutions and software platforms. GLIF is being developed with an evolutionary life-cycle approach, in which functional requirements for the sharable CIG language are continuously refined and incorporate important features of other modeling environments.

The aim of this study has been to facilitate standardization by identifying common components that the CIG community can adopt as standards, while allowing different groups to continue exploring their own approaches to aspects of guideline modeling lacking consensus. If mapping large parts of the methodologies to a common representation is possible, then sharing significant components of encoded guidelines across methodologies might be feasible. Guideline formalization could also support authoring, designing, and maintenance of guidelines. We have identified areas of considerable similarity between models and areas where groups have adopted diverse approaches to specific challenges.

### Common Components and the Implication for Standards Development

A degree of cross-format standardization may be possible in areas with common components. For example, although the underlying computational models of the methodologies studied vary significantly,<sup>12,28,32</sup> all except for *PRODIGY* organize guidelines as plans that unfold over time. All link plan components in sequence, in parallel, and in iterative and cyclic structures, thus defining control-flow. In addition, all the models support nesting of plans as well as expression of temporal constraints on plan components. As part of the HL7 Clinical Decision Support Technical Committee (CDSTC), some of the authors have been

evaluating the Workflow Management Coalition's (WfMC) Workflow model as a common control-flow model.<sup>33</sup> The Workflow model has constructs for expressing nesting, loops, activities, branching, and synchronization and can express temporal constraints. By specializing activities into guideline-specific tasks, such as enquiries and decisions, the different guideline models should be able to map to this formalism. Indeed, *GUIDE*'s developers already map their guideline representation to this model. As well as being a tested standard of the WfMC, the Workflow model has well-defined formal foundations derived from Petri Nets (PN).<sup>34</sup> It remains to be seen whether the clinical guideline extensions to the workflow model will allow mapping into PNs. Such mapping should support formal verification of a guideline model's properties.

Standardizing the expression language used by the various guideline models may also be possible. A standardized language could be used for specifying and sharing decision and eligibility criteria, patient state definitions, conditions, and system actions. It would need to support operators that are common to all models (logical, arithmetic, and comparison operators), function calls, presence criteria, and temporal expressions. The CDSTC is evaluating *GELLO* as a possible standard.

A third component that would benefit from standardization is the patient information model. An object-oriented Virtual Medical Record (VMR) would ease the process of mapping guideline patient data items to real EMRs, allowing decision criteria, eligibility criteria, and patient states to be defined in guideline models by reference to the VMR rather than to specific EMRs. In addition, structured medical actions can derive their structures from the VMR classes, as is done in *GLIF*. The CDSTC may also standardize the VMR, based on experiences with the patient information models of *PRODIGY*, *EON*, and the HL7 RIM, which is also the basis of *GLIF*'s default patient information model.

A standard medical concept model would also be beneficial. However, this goal is currently out of reach because existing vocabularies have not been explicitly designed for clinical decision support and have limitations for such applications (e.g., mixed hierarchies and missing abstractions). Standardizing definitions of abstract terms would only be possible after a common expression language, patient information model, and medical concept model have been standardized.

### Significant Differences and Their Causes

A key difference between formats is their intended scopes. Some groups, such as the developers of Asbru and PROforma, have deliberately not attempted to include methods for representing static knowledge such as medical concept models and ontologies of actions. Instead, they emphasize the provision of clean interfaces for accessing this information externally. The developers of PRODIGY, which focuses on chronic disease management, made a different scoping decision. These variations in scope probably reflect pragmatism and differences in the research interests of the various groups rather than strong convictions that representing a particular class of knowledge is fundamentally unnecessary for guideline-based decision support systems.

PRODIGY emphasizes a scenario-based approach, in which a guideline is organized as a collection of clinical contexts. Users select contexts from relevant clinical actions. EON and GLIF, which are influenced by PRODIGY, also support scenarios. The other methodologies can support scenario functionality by using expressions that refer to patient states in decision criteria or preconditions to affect guideline control flow. Some researchers argue that the latter approach enables guideline modeling to remain task-based, rather than state-based.<sup>11,35</sup> However, clinical scenarios are intuitive concepts that may be useful to domain experts when they are authoring guideline models.<sup>4</sup>

There is significant variation in the decision models. GLIF, EON, PRODIGY, and GUIDE have adopted extensible approaches, the developers' philosophy being that extensions to the core language to represent a specific decision model are legitimate. The Arezzo tool does not provide support for adding predicates, functions, and task subclasses, but the PROforma developers will support extensibility of tasks and functions in future tools.

The methodologies vary considerably in the ways that clinical goals are represented and utilized. Only Asbru, EON, PROforma, and GUIDE represent goals formally and allow reasoning about them. Similarly, only Asbru and PROforma represent effects of plans and reason with them. Representing goals and the effects of actions is central to Asbru's approach. Guidelines are viewed as plans that may fulfill goals, and plan selection can be based on satisfying preconditions or on matching the effects of a plan to target states. PRODIGY and GLIF adopt a more limited approach, with goals represented as text strings. This approach allows clinicians to browse goals of a

guideline, without allowing machine reasoning about those goals.

### Limitations of Our Study

This study has identified a number of areas in which various groups have adopted different approaches to particular aspects of guideline modeling. However, where such differences have been identified, we have not attempted to judge the superiority of one approach over another. There is no normative framework to allow judgment, and it may be that only experience will show which approach to specific problems in guideline modeling is most effective in the long term. However, we hope that this article will serve as a starting point for focused discussions on the relative merits of the different approaches.

Our study is obviously limited by the dimensions of comparison we used. In practice, wide-scale deployment of CIG-based clinical decision support systems (CDSS) will require other services, such as documentation, tool support, and methods for integration in real world settings.

CIG models will require documentation. For example, a model's authorship, the evidence the guideline is based on, and its intended context of use must all be captured and made available. Currently, only some of these attributes are supported by the various modeling methods, although the rest of them can be added easily (see the website referenced above for details). For example, since we performed this study, all of the documentation attributes suggested by the GEM formalism have been added to GUIDE and GLIF. A common set of documentation attributes, such as those described by the GEM formalism,<sup>36</sup> may well be of value. The CDSTC is considering requirements for standardizing documentation attributes.

Software tools are required to support model authoring and editing. As discussed, the modeling languages provide various facilities to manage complexity, such as nesting of subplans, separating guideline knowledge from consultation templates in EON and PRODIGY, and action refinement. However, the software environment's features will also be important in determining how easily authors can create, edit, and interact with guideline models. Although all of the modeling methods that we considered use graphical environments for authoring, we did not consider this aspect in our study. The successful deployment of CIG-based CDSS will also require suitable software to allow clinicians to interact with guideline models

to derive decision support. We studied expressiveness and syntax, but not how they are used to encode guidelines that are part of real clinical decision-support systems.

Many of the differences among modeling approaches result from their particular classes of intended applications. Adding a new feature to a model is easy, but it may not be practically implementable, or it may add too much complexity to a model. Furthermore, a feature may have limitations that are not discovered until it is used by clinicians. Thus, a feature-rich model may not be the best choice for many users. Such validation of the models requires implementation and integration of guideline-based system in working EMRs. In this respect, we are limited by the fact that neither GLIF nor Asbru has integrated guidelines in real-world settings. At the time of the study, Asbru and GLIF did not have execution engines. Since then, an execution engine for GLIF has been developed. The hypertension guideline modeled in EON has been deployed in many clinics of the U.S. Department of Veterans Affairs hospitals.<sup>37</sup> Three complex chronic-disease-management guidelines and over 150 simple guidelines encoded in PRODIGY have been implemented using two vendor systems.<sup>38</sup> GUIDE has been used to implement guidelines for managing stroke patients in the acute and subacute phases in four Italian neurological departments. An evaluation study demonstrated a positive impact of the guideline on health outcomes, i.e., survival and disability.<sup>39</sup> Another application concerns a guideline for pressure ulcer prevention, implemented in a general medicine ward in Pavia, Italy. PROforma-based systems that are in clinical use include RetroGram<sup>40</sup> (assistance in the interpretation of genotype data and decisions for the management of HIV+ patients), Early Referrals Application (early referral of suspected cancer patients), and MACRO (Clinical Trial Manager). Several other systems have undergone clinical evaluation and advanced prototypes (<http://www.openclinical.org/gmminuse.html>). A detailed comparison of these implemented systems is beyond the scope of this article.

## Conclusion

Our study identified guideline components that the CIG community could adopt as standards. We outlined how some groups are pursuing standardization of these components under the auspices of HL7. We recognize, however, that because of the different goals of various research groups, a consensus model will be acceptable to the research groups only if it

concurrently allows them to continue their investigations of unique features.

## References ■

- Overhage JM, Tierney WM, Zhou XH, McDonald CJ. A randomized trial of "Corollary Orders to Prevent Errors of Omission." *J Am Med Inform Assoc* 1997;4(5):364-375.
- 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.
- Peleg M, Boxwala A, Ogunyemi O, et al. GLIF3: The evolution of a guideline representation format. *Proc AMIA Annu Fall Symp* 2000:645-649.
- Johnson PD, Tu SW, Booth N, Sugden B, Purves IN. Using scenarios in chronic disease management guidelines for primary care. *Proc AMIA Annu Fall Symp* 2000:389-393.
- Tu SW, Musen MA. A flexible approach to guideline modeling. *Proc AMIA Symp* 1999:420-424.
- Fox J, Rahmazadeh A. Disseminating medical knowledge: the PROforma approach. *Artif Intell Med* 1998;14:157-181.
- Shahar Y, Miksch S, Johnson P. The Asgaard Project: a task-specific Framework for the application and critiquing of time-oriented clinical guidelines. *Artif Intell Med* 1998;14:29-51.
- Quaglino S, Stefanelli M, Lanzola G, Caporusso V, Panzarasa S. Flexible guideline-based patient careflow systems. *Artif Intell Med* 2001;22:65-80.
- Fox J, Das S. *Safe and Sound: Artificial Intelligence in Hazardous Applications*. AAAI and MIT Press, 2000.
- Elkin P, Peleg M, Lacson R, et al. Toward standardization of electronic guideline representation. *MD Comput* 2000;17(6):39-44.
- Wang D, Peleg M, Tu SW, Shortliffe EH, Greenes RA. Representation of clinical practice guidelines for computer-based implementations. *Proc MedInfo* 2001:285-289.
- Tu S, Musen M. Representation formalisms and computational methods for modeling guideline-based patient care. *Proceedings of the First European Workshop on Computer-based Support for Clinic Guidelines and Protocols*. Leipzig, 2000:125-142.
- Starren J, Hripcsak G, Jordan D, Allen B, Weissman C, Clayton PD. Encoding a post-operative coronary artery bypass surgery care plan in the Arden Syntax. *Comput. Biol Med* 1994; 24(5):411-417.
- Shiffman RN. Representation of clinical practice guidelines in conventional and augmented decision tables. *J Am Med Inform Assoc* 1997;4(5):382-393.
- Seyfang A, Kosara R, Miksch S. *Asbru's Reference Manual, Version 7.3*. Vienna University of Technology, Institute of Software Technology, Vienna, 2002. Report No.: Asgaard-TR-2002-1.
- Tu SW, Musen MA. From guideline Modeling to guideline execution: Defining guideline-based decision-support Services. *Proc AMIA Annu Symp* 2000:863-867.
- Tu SW, Musen MA. Modeling data and knowledge in the EON guideline architecture. *Proc Medinfo* 2001; 280-284.
- Hripcsak G, Ludemann P, Pryor TA, Wigertz OB, Clayton PD. Rationale for the Arden Syntax. *Comput Biomed Res* 1994; 27(4):291-324.
- Schadow G, Russler DC, Mead CN, McDonald CJ. Integrating medical information and knowledge in the HL7 RIM. *Proc AMIA Annu Fall Symp* 2000:764-768.
- Ogunyemi O, Zeng Q, Boxwala A. Object-oriented guideline expression language (GELLO) specification: Brigham and



- Women's Hospital, Harvard Medical School, 2002. Decision Systems Group Technical Report DSG-TR-2002-001.
21. Quaglini S, Dazzi L, Gatti L, Stefanelli M, Fassino C, Tondini C. Supporting tools for guideline development and dissemination. *Artif Intell Med.* 1998;14(1-2):119-137.
  22. Grosso WE, Eriksson H, Fergerson R, Gennari JH, Tu SW, Musen MA. Knowledge modeling at the millennium (the design and evolution of Protégé-2000). *Proc 12th Banff Knowledge Acquisition for Knowledge-Based Systems Workshop.* Canada; 1999:7-4-1 to 7-4-36.
  23. Johnson P, Tu S, Jones N. Achieving reuse of computable guideline systems. *Proc Medinfo 2001:*99-103.
  24. Fox J, Johns N, Rahmzadeh A, Thomson R. PROforma: A method and language for specifying clinical guidelines and protocols. *Proceedings of Medical Informatics Europe, Amsterdam, 1996.*
  25. Irwin RS, Boulet LS, Cloutier MM, et al. Managing cough as a defense mechanism and as a Symptom: a consensus panel report of the American College of Chest Physicians. *Chest* 1998;114(2):1335-1815.
  26. National Institute of Health. The Sixth Report of the Joint National Committee on Prevention, Detection, Evaluation, and Treatment of High Blood Pressure. 1997 November. Report No: 98-4080.
  27. Kosara R, Miksch S. Metaphors of movement: a visualization and user inter-face for time-oriented, skeletal plans. *Artif Intell Med.* 2001;22(2):111-131.
  28. Peleg M, Tu SW, Bury J, et al. Comparing Models of Decision and Action for Guideline-based Decision Support: A Case-study Approach. Stanford University, 2002. Report No: SMI-2002-0922.
  29. Peleg M, Tu SW, Bury J, et al. Comparing Models of Data and Knowledge for Guideline-based Decision Support: A Case-Study Approach. Stanford University, 2002. Report No: SMI-2002-0923.
  30. Peleg M, Ogunyemi O, Tu S, et al: Using features of Arden Syntax with object-oriented medical data models for guideline modeling. *Proc AMIA Annu Symp 2001;* 523-527.
  31. Johnson PD, Tu SW, Musen MA, Purves I. A virtual medical record for guideline-based decision support. *Proc AMIA Annu Symp 2001:*294-298.
  32. Tu SW, Johnson PD, Musen MA. A Typology for Modeling Processes in Clinical Guidelines and Protocols. Stanford University, 2002. Report No: SMI-2002-0911.
  33. Fischer L (ed). *Workflow Handbook.* Future Strategies Inc., Lighthouse Point, FL, 2002.
  34. van der Aalst WMP. The application of Petri Nets to Workflow Management. *J Circuits Syst Comput* 1998;8(1):21-66.
  35. Bury JP, Saha V, Fox J. Supporting 'scenarios' in the PROforma guideline modelling format. *Proc AMIA Annu Symp 2001:*870.
  36. Shiffman RN, Karras BT, Agrawal A, Chen R, Marengo L, Nath S. GEM: a proposal for a more comprehensive guideline document model Using XML. *J Am Med Inform Assoc* 2000; 7(5):488-498.
  37. Goldstein MK, Hoffman BB, Coleman RW, Musen MA, Tu SW, Advani A, et al. Implementing clinical Practice guidelines while taking account of evidence: ATHENA, an easily modifiable decision-support system for management of hypertension in primary care. *Proc AMIA Annu Symp 2000:*300-304.
  38. Johnson P, Tu S, Jones N. Achieving reuse of computable guideline systems. *Proc Medinfo 2001:*99-103.
  39. Micieli G, Cavallini A, Quaglini S. Guideline compliance improves stroke outcome: a preliminary study in 4 districts in the Italian region of Lombardia. *Stroke* 2002;3(5):1341-1347.
  40. Tural C, Ruiz L, Holtzer C, Schapiro J, Viciano P, González J, et al. Clinical utility of HIV-1 genotyping and expert advice: the Havana trial. *AIDS* 2002;16:209-218.