Creation and maintenance of implementable clinical guideline specifications

Vipul Kashyap[1], Ph.D., Alfredo Morales[2], Eng, MS and Tonya Hongsermeier[1], MD, MBA
[1]Clinical Informatics R&D, Partners HealthCare System, Wellesley, MA
[2]Network Inference, Carlsbad, CA
vkashyap1@partners.org

ABSTRACT

Clinical guidelines form an important component of improving the quality of care and patient outcomes. We are exploring approaches to encode high-quality evidence-based text guidelines as computer implementable specifications. An architecture that includes components such as a ***rules engine*** *that executes declarative if-then rules, an OWL-based* ***classification engine*** *that performs classification and subsumption inferences on an object oriented information model/ontology and a* ***data repository*** *storing patient data is presented. Some design choices and trade-offs we explore in this paper are: (a) Decomposition of a clinical guideline into decisions, actions, patient state transitions and definitions; (b) Use of an ontology engine for* ***definition management****; (c) Use of a rules engine for* ***guideline management****; and (d) Representation of definitions as OWL-axioms or if-then rules based on the expressiveness of these formalisms; and (e) Issues of ongoing maintenance and consistency checking as definitions and clinical guidelines change over time. We discuss how our architecture makes the creation and maintenance of implementable specifications a more tractable process.*

INTRODUCTION

Clinical care guidelines form an important component of improving the quality of care and patient outcomes. As different payer agencies such as the Federal Government (through Medicare/Medicaid) and insurance agencies such as Blue Cross and Blue Shield move towards a pay for performance model, healthcare quality and patient outcome metrics, such as the JCAHO[Error! Reference source not found.] and HEDIS[2] measures have come into focus. At Partners Healthcare, we seek to incorporate these measures within our clinical information systems. Clinical guidelines are tools for encouraging best practices in clinical care and are intended to improve safety, quality and cost effectiveness[3]. We seek to implement clinical guidelines within our clinical systems to enable implementation of JCAHO and HEDIS measures within Partners HealthCare System.

Approaches for modeling of clinical guidelines are: the Arden Syntax[4], EON[5], PRODIGY-3[6], PROforma[7], Asbru[8], GUIDE[9], Prestige[10] and GLIF3[11]. The various stages of the lifecycle in a clinical guideline that have been identified as[12]: (1) conceptual modeling, (2) encoding, (3) validation, (4) dissemination, (5) local adaptation, (6) integration with an implementation system and (7) application and revision. From an architectural viewpoint, the various steps in the GLIF3 guideline model have been delineated as[3]: *action and decision steps* to represent clinical actions and decisions; *patient state steps* to serve as entry points into a guideline; and *branch and synchronization steps* for modeling concurrency.

We focus on the application and revision of clinical guidelines. We are investigating ways of implementing simple guidelines that are likely to have a high impact by using an industrial strength rules engine and an OWL-based ontology engine. Changes in guidelines could result from either a change in the logic of the guideline or a change in the definition of the object and data types referenced in the guideline. These changes could lead to inconsistency in the logic and/or definitions of object types referenced. To enable the maintenance of these guidelines in a tractable manner, we propose further delineation of decision steps into ***definitions*** and ***decisions***. We present various options for representing definitions as OWL-axioms or if-then rules based on the expressiveness of these formalisms. Furthermore, an architecture that introduces modules for ***definition*** and ***guideline/rule management*** and enables tractable maintenance and consistency checking is presented.

We discuss a simple guideline for lipid management in diabetes and discuss representation of its components. An architecture for guideline maintenance is presented. Issues related to definition maintenance and guideline maintenance are discussed in detail followed by a presentation of conclusions and future and ongoing work.

USE CASE

Consider the following guideline for lipid management suggested by the American Diabetes Association (ADA)[13]:

[13]
Lowering triglycerides and increasing HDL cholesterol with a fibrate are associated with a reduction in cardiovascular events in patients with clinical CVD, low HDL and near-normal levels of LDL (A). Lower triglycerides to <150 mg/dL (1.7 mmol/L) and raise HDL cholesterol to

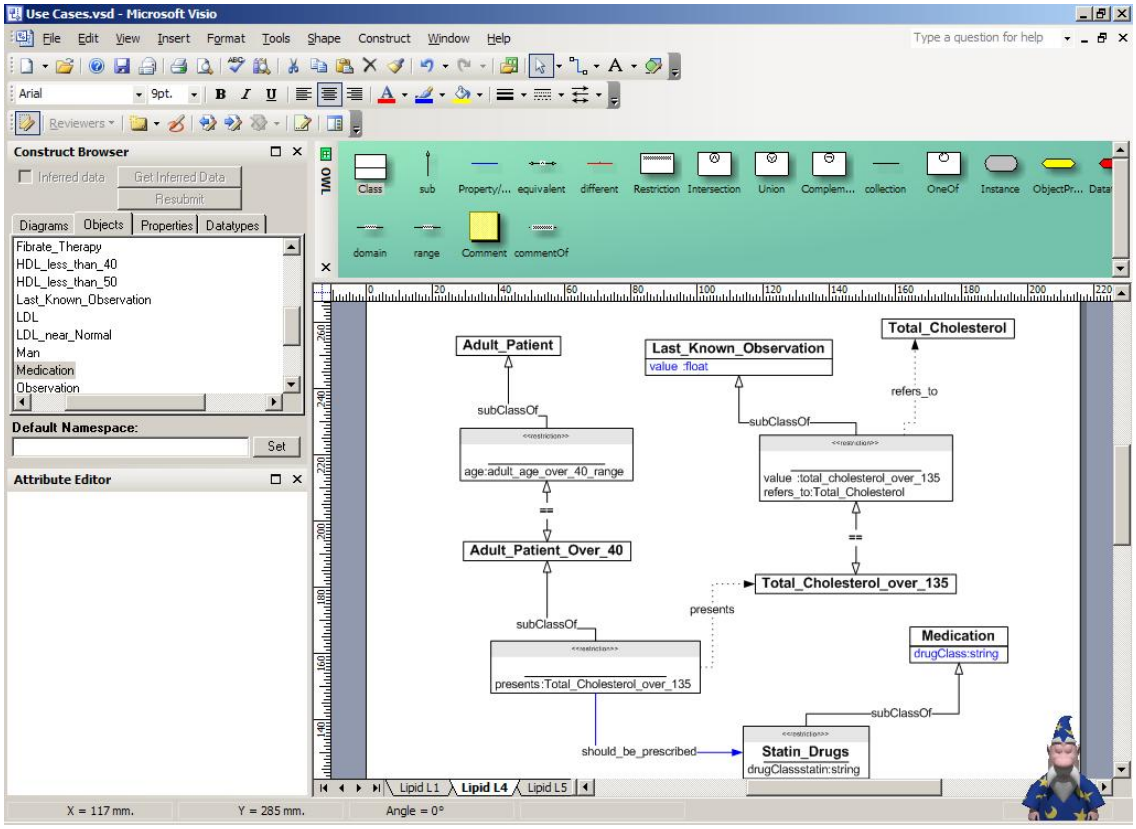

Figure 1: The ontology underlying the clinical guideline

>40 mg/dL (1.15 mmol/L). In women, an HDL goal 10 mg/dL higher may be appropriate (C). Patient has CVD, triglycerides >150 and/or HDL<40 (for women HDL<50) and LDL levels are near normal.

The first step in decomposing the above guideline is to construct the underlying ontology (or object model). A sample ontology created using the Network Inference's Construct tool is illustrated in **Figure 1**.

The following elements may be observed in the above recommendation. Decisions are processes by which a course of action is undertaken when one or more conditions are satisfied. Conditions may typically consist of definitions or patients state transitions, and their occurrence is necessary to require an action to be undertaken. In general decisions are better represented as if-then production rules and definitions are represented as OWL-axioms. In the recommendation above, a decision to prescribe fibrate therapy with the goal of triggering a patient state transition for all patients defined to be suffering from clinical CVD. Patient State Transitions describe changes in patient state during the progression of a disease or therapy. Each state could be conceived as the equivalent to a node in an activity graph, representing one well defined, understood classification of a patient based on the combination of known qualitative or quantitative measurements. Transitions from one state to another will occur based on the fact that certain conditions arise, triggering the reclassification of the patient.

In the example above, a female patient suffering from cardiovascular disease could be consider to be in a state of "higher risk for cardiovascular events" if her clinical presentation describe her as having HDL cholesterol levels below normal and LDL levels near normal. This statement can be represented as the following OWL axiom:

$$\texttt{DiabeticWomanWithHigherRiskofCVD} \equiv \texttt{WomanWithCVD} \cap \forall \texttt{presents.(LDLNearNormal} \cap \texttt{HDLLessThan50} \cap \texttt{TriglyceridesOver150)}$$

In similar fashion, the state of not being at "high risk for cardiovascular events" could be represented as the complement of the axiom above:

$$\texttt{DiabeticWomanWithNormalorLowerRiskofCVD} \equiv \neg\texttt{DiabeticWomanWithHigherRiskofCVD}$$

The patient state transition will take place once the conditions in the first axiom are not present in the patient's clinical presentation, which will reclassify the patient into the complementary state. The state transition may be represented as:

```
IF DiabeticWomanWithHigherRiskforCVD AND
goal(∀presents.TriglyceridesUnder150)
THEN "Prescribe fibrate therapy"
```

The predicate goal(X) may be implemented by the rules engine and executes state transitions. However the goal state definitions are stored as OWL axioms. Approaches to represent actions like "prescribe fibrate therapy" are discussed next.

Actions: They constitute discrete operations within the context of a clinical guideline. In some cases actions are terminal and inconsequential to the overall progression of the guideline, or the patient state. In such cases, they be represented as a consequent in a rule and executed on the rule side. In other cases, actions could have a direct influence in the future classification of the patient under the precepts provided by the guideline. For example, in the use case above, the action of prescribe a fibrate to a patient may impose a restriction over other concepts in the ontology and could be represented as an OWL-axiom.

Definitions: Due to the intrinsic nature of a definition, for the most part, it will be appropriate to represent them as OWL axioms to be managed by the ontology engine. For example, the concept of cardiovascular disease (CVD) could be described at the intersection of other concepts such as vascular disease and cardiac muscle disease.

```
CardiovascularDisease ≡
CardiacMuscleDisease ∩ VascularDisease
```

OWL axioms can express definitions in terms of their properties and data values. For example, "adult diabetic patient" can be expressed as follows:

```
AdultPatientWithDiabetes
≡ AdultPatient ∩ ∀has.CardiovascularDisease
```

In addition, the use of OWL axioms allows the representation of qualitative restrictions over the possible values of a data type. For example, the definition of "near-normal levels of LDL" can be expressed as an OWL axiom that involves a restriction to the values for LDL within a range of values and is illustrated below:

```
LDLNearNormal
≡ ∀values.LDLNearNormalRange
          ∩ ∀refers-to.LDL
LDLNearNormalRange is defined as an XML
simple type with values > 135.0
```

However, temporal, spatial and sequential definitions cannot be expressed as OWL axioms. For example the concept of "last known observation" shown in **Figure 1** is one of those cases. This concept can be expressed in a rule format as follows:

```
IF observation(t,v)
AND NOT EXISTS t' > t AND
observation(t', v') THEN
lastKnownObservation(v)
```

This can either be executed by the rule engine or could be translated to an SQL query that can be executed on the data repository in an operational context.

ARCHITECTURE

In this section, we present a systems architecture to implement a clinical guideline (**Figure 2**). The components of this architecture are as follows:

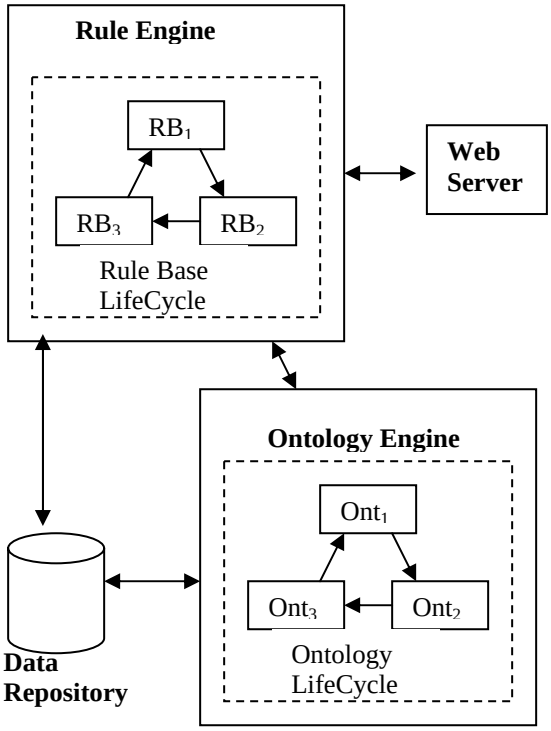

Figure 2: Architecture

- **Data Repository:** This may be implemented as a database management server which stores and responds to queries on the electronic health record (EHR). Both the Rule Module and the Ontology Module are linked to the Data Repository via adapters. At run time, these adapters populate the object model of the Rule Engine and the ontology model of the Ontology Server.
- **Rule Engine:** This may be implemented using a business rule management server (BRMS) such as ILog[14] or Blaze[15]. In the operational context, the rules engine executes production rules that implement decisions or are involved in patient state transitions. In the maintenance context, the rule engine also manages changes in rules that might occur due to changes in the clinical guideline specifications that occur over time.

This could me modeled and enforced as rule-base lifecycle. The Rule Engine also checks for conistency conditions such as cycles in the rule base or the same condition triggering inconsistent actions.

- **Ontology Engine:** This is implemented using the Network Inference Cerebra Server[16]. In the operational context, the server performs classification inferences on patient data to determine if a patient belongs to a particular category (e.g., high risk patient). In the maintenance context, the ontology engine manages changes in concept definitions over time. This could be modeled and enforced as an ontology lifecycle. The Ontology engine also checks for inconsistency of some concepts or terminological cycles that might be created as a result of changes in the definitions.
- **Web Server:** The Web server is in the presentation layer and is responsible for display of application content and services to a web browser.

From the maintenance viewpoint, the above figure represents various lifecycles that reflect changes in definitions and guidelines. However these lifecycles need to be integrated with each other as changes in the definition can lead to changes in guidelines and vice versa. This illustrated in the figure below:

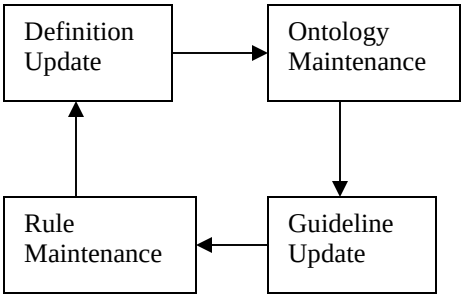

Figure 3: Integration of Rule/Ontology Lifecycles

Let's assume that the definition of cardiovascular disease is changed by dropping the condition that it also be a Vascular Disease.

`CardiovascularDisease≡CardiacMuscleDisease`

This would lead to a change in the semantics of the guideline as more patients would come under the purview of the guideline. This would require a change to the guideline requiring the addition of a conjunct which checks whether a patient has vascular disease as well. On the other hand, let's assume that the guideline in the use case has changed as follows:

```
IF   DiabeticWomanWithHigherRiskforCVD   AND
goal(∀presents.TriglyceridesUnder180)
THEN "Prescribe fibrate therapy"
```

This results in a creating a new goal state (with triglycerides < 180) that needs to be added to the ontology.

The approach and architecture makes the guideline maintenance process more tractable. Typically guideline execution environments have been implemented using rule engines or using special purpose guideline execution engines. From the maintenance viewpoint, the combinatorial explosion of the 4 types of changes illustrated in **Figure 3** will have to be implemented in the guideline engine. Our architecture isolates the various changes described above in the rule and ontology engine modules, thereby improving the tractability of the maintenance process.

DEFINITION MANAGEMENT

The facts that are used to establish a given definition may change based on the occurrence of new evidence or new practice criteria. The change in definitions which also affects the clinical guidelines based on them. Thus definition management is of critical importance.

For the ontology engine to support definition management, one has to have expressions that it can perform inference on. For example, one could define Cardiovascular Disease as a list of possible ICD9 codes or to represent it as an axiom as discussed in the section on Use Case. Such an approach will allow adaptation of definitions in a flexible way as they change over time. For instance, the ontology engine could propagate any change in the definition of the concept Cardiovascular Disease to all other concepts and guidelines that depend on it.

Representing definitions using OWL axioms help us leverage the ontology engine to manage the consequences of changing definitions such as:

- **Recognition of inconsistent definitions:** Whenever a definition is changed, the ontology engine can detect it.

- **Identification of cycles:** Consider the following OWL axioms which characterize the following definitions.
  - *Diabetes patients are likely to suffer from CardiovascularDisease*
  - *All patients that are likely to suffer from Cardiovascular disease may have abnormal LDL*
  - *People with abnormal LDL are likely to have Diabetes*

  The ontology engine can perform inferences to discover circular definitions as illustrated above.
- **Discovery of missing definitions:** Whenever definitions are changed from one version of the

ontology to another, the ontology engine can infer the differences in the definitions.

GUIDELINE MANAGEMENT

A critical component of change in guidelines is to check whether that introduces cycles in the rule base which could result in an infinite loop during run time. Some of these cases could be detected by a compile time analysis of the rule base. Consider the following rules:
IF patient's LDL level is Near Normal THEN patient is at a low risk for Cardiovascular disease.
IF a patient is low risk for Cardiovascular disease THEN patient has Normal LDL Level
If the definitions of "Near Normal" and "Normal" overlap then there is a cycle in the rule base which the rule engine should be able to detect.

Due to changes in the definition of concepts, two previously non-equivalent concepts may become equivalent or overlapping. Consider the following:
IF patient's LDL level is Abnormal THEN patient is at high risk for Cardiovascular disease.
IF a patient's LDL is Normal THEN patient is at low risk for Cardiovascular disease.
If the definitions of "Abnormal" and "Normal" overlap, then the we have the same rule which could fire and give inconsistent conclusions. The rule engine can detect this and other inconsistencies, providing us with the functionality to partially automate the guideline management process.

CONCLUSIONS AND FUTURE WORK

We have presented an approach of teasing apart a clinical guideline into decisions, patient state transitions, definitions and actions. A modular architecture that enables tractable creation and maintenance of guidelines was presented. We approached the problem from both operational and maintenance aspects and discussed design issues and trade-offs that arise in that context. An important design issue that we addressed is what parts of the guideline is representable as an OWL axiom as opposed to as a production rules. Various design choices were discussed and associated issues were presented.

This is part of an ongoing project at Partners HealthCare to build a Knowledge Management infrastructure for creation and maintenance of clinical guidelines and other knowledge assets. We are currently piloting a rules engine (ILog) and an OWL-based engine for creating an execution environment for clinical guidelines. As we move towards deploying these in a production environment, we will investigate the following issues going forward:

- What happens when a guideline or concept definition changes? How do we "undo" some decisions that were taken in the past or on patients whose treatment is currently in progress?
- Is it possible to handle temporal reasoning within an OWL reasoner?
- What will be the impact of genomic and personalized medicine on clinical guidelines? Will our architecture be able to manage knowledge related to genomic medicine?

REFERENCES

1. http://www.jcaho.org
2. http://www.ncqa.org/Programs/HEDIS
3. Peleg, M, Boxwala AA, Tu S, Zeng Q, Ogunyemi O, Wang D, Patel VL, Greenes RA and Shortliffe EH, The Intermed Approach to Sharable Computer-Interpretable Guidelines: A Review. JAMIA, 2004;11(1):1-10.
4. Clinical Decision Support & Arden Syntax Technical Committee of HL7, inventor Arden Syntax for Medical Logic Systems, version 2.0 , USA, July 7, 1999.
5. Tu SW and Musen MA. A Flexible approach to Guideline Modeling. Proc. AMIA Syposium; 1999:420-424.
6. Sugden B, Purves IN, Booth N, Sowerby M. The PRODIGY Project – the Interactive Development of the Release One Model, Proc. AMIA Symposium; 1999:359-363.
7. Fox J and Rahmanzadeh A. Disseminating medical knowledge: the PROforma approach. Artif. Intell Med 1998;14:157-181
8. Shahar Y, Mikseh S and Johnson P. The Asgaard Project: A Task-Specific Framework for the Application and Critiquing of Time-Oriented Guidelines. Artif Intell Med 1998;14:29-51
9. Quaglini S, Stefanelli M, Lanzola G, Caporusso V and Panzarasa S. Felgible Guideline-based Patient Careflow Systems. Artif Intell Med 2001;22:65-80
10. Gordon C, Veloso M. Guidelines in Healthcare: The experience of the Prestige project: Medical Informatics Europe; Ljubljana, Slovenia; 1999:733-738
11. GLIF3 Technical Documentation. http://www.smi.stanford.edu/projects/intermed-web/guidelines/GLIF_TECH_SPEC.pdf
12. Greenes RA, Peleg M, Boxwala A, Tu S, Patel V and Shortliffe EH. Shareable computer-based clinical guidelines: rationale, obstacles, approaches and prospects. MedInfo. 2001;10(Pt 1):201-5

13. http://care.diabetesjournals.org/content/vol28/suppl_1/
14. http://www.ilog.com
15. http://www.fairisaac.com/Fairisaac/Solutions/Product+Index/Blaze+Advisor/
16. http://www.networkinference.com