Detailed Clinical Models for Sharable, Executable Guidelines

Craig G. Parker MD[1], Roberto A. Rocha MD PhD[1,2], James R. Campbell MD[3],
Samson W. Tu MS[4], Stanley M. Huff MD[1,2]

[1]Intermountain Health Care, Salt Lake City, UT
[2]Department of Medical Informatics, Univiersity of Utah, Salt Lake City, UT
[3]University of Nebraska Medical Center, Omaha, NE
[4]Stanford Medical Informatics, Stanford University School of Medicine, Stanford, CA

ABSTRACT

The goal of shareable, executable clinical guidelines is both worthwhile and challenging. One of the largest hurdles is that of representing the necessary clinical information in a precise and sharable manner. Although standard terminologies and common information models, such as the HL7 RIM, are necessary, they are not sufficient. In addition, common detailed clinical models are needed to give precise semantics and to make the task of mapping between models manageable. We discuss the experience of the SAGE project related to these detailed clinical models.

INTRODUCTION

The SAGE project is a NIST funded multi-institutional effort to create standards-based, sharable, executable clinical guidelines. The project envisions a system that enables the authoring, localization and execution of significant clinical guidelines in a vendor independent manner.

The idea of creating sharable, executable decision support mechanisms is not new with the SAGE project. The Arden syntax represents a significant effort to define a sharable representation for medical logic modules. However, since Arden uses implementation specific code (inside the curly braces) to reference data items, the task of sharing is still difficult. The issue in not one of simply mapping concepts from one terminology to another, but also one of the higher-level organization of information[1].

Within SAGE, we face similar challenges in creating sharable executable guidelines. Not only do terminologies vary between institutions, but the manner in which terminologies are used also varies. In the simplest and most inflexible implementations, the information model is defined by the names of the columns in database tables and terminologies serve to provide values for these slots. For example, a table in a clinical database may have a column for systolic blood pressure. In such a system, information associated with a blood pressure (e.g. patient position or a timestamp) is limited to what other columns exist in the database to hold this information.

More robust solutions such as the HL7 Reference Information Model (RIM)[2] or the Clinical Event model[3] implement more flexible information models. In these models, constructs for representing clinical information more resemble data structures in high-level programming languages than relational tables. In addition, they have more flexibility in the partitioning of knowledge between the information model and the terminologies it uses. In this paper, we discuss the difficulties we face in reconciling the differences between information models, we discuss some of the possible solutions to these problems, and we give rationale for SAGE's solution.

CLINICAL MODELS

We need to reconcile the differences in information structure between systems to enable sharable, executable guidelines . The fundamental feature of this reconciliation is the preservation of semantics between systems. Standard terminologies are a necessary component of the solution to this problem, but alone they are not sufficient. Standard terminologies provide the most atomic concepts we need for expressing clinical information. They often provide a mechanism for composition – allowing the creation of compositional concepts from atomic ones[4,5].

Different terminologies are created for different purposes and address different clinical domains. We can use terminologies together to create more expressive data representations. For example, the LOINC® terminology enumerates types of things that can be observed about a patient such as laboratory

tests. However LOINC® does not attempt to create concepts for the results of the coded observations. For this we can use a terminology such as SNOMED-CT®. Below is an example of how these terminologies work together. The example is presented as a snippet from an XML document.

```
<observation>
  <cd code="21840-4"
      codeSystemName="LOINC"
      displayName="Gender"/>
  <value code="248152002"
         codeSystemName="SNOMED-CT"
         displayName="Female"/>
</observation>
```

This example demonstrates the synergistic use of two terminologies. The LOINC® coding system provides a code for the item of interest, or in other words, what we looked for. SNOMED-CT® provides the code for the value of this item, or what we saw.

The framework in which we use these terminologies is our information model. Slots in the concept structure defined by the information model are filled with concepts from appropriate terminologies. An information model is similar to the compositional tools of a terminology. It allows us to combine more atomic terms to describe higher level concepts. This similarity is demonstrated in the following examples, which are presented in a simplified XML style that shows textual representations of the concepts, but not the specific codes.

```
<observation>
  <cd>Supine Systolic Blood Pressure</cd>
  <value>120 mmHg</value>
</observation>

<observation>
  <cd>Systolic Blood Pressure</cd>
    <qualifier>
      <cd>Patient Position</cd>
      <value>Supine</value>
    <qualifier>
  <value>120 mmHg</value>
</observation>
```

Both of these observations are meant to convey that the patient's systolic blood pressure in a supine position is 120 mmHg. The concept of "systolic blood pressure in a supine position" is a composite of three, more atomic, concepts: blood pressure, systolic phase, and supine position. In the first example, all three concepts are represented as a single pre-coordinated term from a terminology. In the second example, two of the concepts are pre-coordinated in the term

"systolic blood pressure" however the third is related via a post-coordination using the information model.

Both representations may be valid in a given information model. However, automatically determining the semantic equivalence of the two observations is difficult. The problem is in the partitioning of knowledge[6]. Since the terminology and the information model have their own compositional mechanisms, compositions done by one are not evident to the other. In other words, we use different tools to analyze the compositions of a terminology than to analyze the compositions of the information model.

The reasons for this division are both theoretical and practical. First, while information models may specify which terminologies to use in specific slots, they do not define terms. On the other side of the problem, terminologies do not generally define terms for things like real numbers such that a lab result or a blood pressure measurement could be defined by composition of a name and a value. Rather, they rely on information models to define numeric value slots and to place appropriate constraints on those slots. In addition, terminologies often allow compositions of terms in ways that do not make clinical sense. Finally, we would argue that due to the way systems have been implemented in the past, people are more accustomed to name-value pair thinking than to compositional sentence building.

While the information model used in a clinical decision support system may recognize both observations in the previous example as valid, it may not recognize them as equivalent. Consider a clinical guideline for the workup of syncope. Abnormal orthostatic blood pressure measurements suggest a diagnosis of orthostatic hypotension and therapy based on this diagnosis. The following examples demonstrate how orthostatic blood pressure measurements may be represented in the information models of different systems. These examples are stylized for clarity and brevity.

```
Observation:
   Orthostatic Blood Pressure:
      Supine Blood Pressure:
         Systolic Blood Pressure
         Diastolic Blood Pressure
      Standing Blood Pressure:
         Systolic Blood Pressure
         Diastolic Blood Pressure
```

```
Observation:
   Orthostatic Blood Pressure:
      Blood Pressure:
         Systolic Blood Pressure
         Diastolic Blood Pressure
         Patient Position = Supine
      Blood Pressure:
         Systolic Blood Pressure
         Diastolic Blood Pressure
         Patient Position = Standing

Observation:
   Orthostatic Blood Pressure:
      Supine Systolic Blood Pressure
      Supine Diastolic Blood Pressure
      Standing Systolic Blood Pressure
      Standing Diastolic Blood Pressure
```

Each of these examples is capable of representing the information needed by the guideline. The differences in the representations lie in the partitioning of concepts between the terminology and information models. In the first model, the orthostatic blood pressure event is composed of two blood pressure events. These events are pre-coordinated with the patient's position. In the second model, the orthostatic blood pressure is similarly composed of two blood pressure events. However, instead of pre-coordinating the observation with the patient position, each blood pressure event has an explicit attribute for patient position, which is constrained to a specific value. Finally, in the third model, the orthostatic blood pressure event is composed of four, more granular, blood pressure events. These events are each pre-coordinated with both the patient position and the cardiac phase. Not only may each of these representations be valid in their own systems, they may all be valid instances created in conformance with a common information model such as the HL7 RIM.

SOLUTIONS

For a clinical guideline to be executable it must have a representation for the concepts that it is concerned with. If our guideline needs to make a decision based on orthostatic blood pressure measurements it needs a model for them. However, selecting any one of the models listed above makes the executable guideline incompatible with systems using the other models. To overcome this, either 1) the guideline must understand all possible representations, or 2) at some point during the implementation of an executable guideline at an institution, the model of that institution must be mapped to the model used by the guideline.

The first option is untenable since it would be impossible to foresee all of the ways that an institution
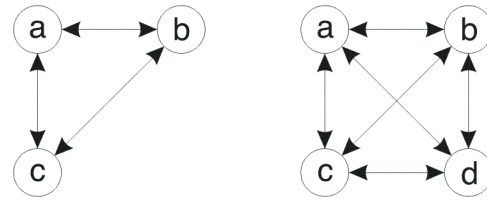

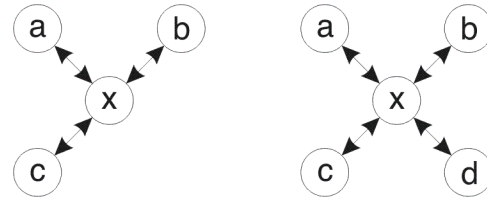Figure 1: Many-to-many mappings.

Figure 2: Many-to-one mappings.

Figure 3: Common model (no mappings).

may choose to combine their terminologies and information models to represent their clinical information. While the second option is possible, it places a large burden on institutions desiring to implement the executable guideline. In addition, there is no guarantee that another guideline with the need to represent orthostatic blood pressure measurements would choose the same representation.

Without a common representation of the detailed clinical models needed for decision support a separate mapping may need to be created for each combination of clinical guideline and implementing institution. For example, consider three hospitals, each of which has their own way of representing orthostatic blood pressure measurements, and each creating different types of guidelines that rely on these measurements. To implement the guidelines, each institution must map their model to all others. When a fourth hospital enters the picture, they must create mappings to other three models, and the other three hospitals must make mappings to a new model. As the number of models increase, the number of mappings grows exponentially (Figure 1).

However, if the institutions agree on common model to be used in the guidelines they create, then each institution only has to map to that model. As new institutions enter, they only need to create mappings to the common model. The number of mappings needed grows linearly with the number of models (Figure 2). If the common models are useful enough we

could eventually migrate to the situation represented in Figure 3 where no mappings are required because each institution has adopted the common model as their internal representation.

THE SAGE APPROACH

A number of "standard" representations for clinical models have been proposed[2,7,8,9]. The SAGE approach is based on various activities currently under development by HL7. On ultimate hope is for a robust and practical model to emerge from HL7's Template special interest group. Many of the people involved in creating earlier models are contributing to this effort. However, such a standard is not yet available.

For our immediate purposes we have adopted the strategy of defining detailed clinical models as restrictions on a virtual medical record (VMR). In turn our VMR is based on HL7 RIM derived artifacts (e.g. HL7 version 3 message types). A RIM based VMR is consistent with other HL7-related VMR efforts. This VMR allows us to specify a broad set of classes of information that are of interest for clinical guidelines such as orders, observations, and goals.

We use the Protege-2000 knowledge authoring tools as the authoring environment for our system. Our VMR is defined by a set of classes in Protege. In creating detailed clinical models based on these classes, we identify groups of clinical items within a class that can be represented in a common style. We then use Protege to create a detailed clinical model for this group by constraining the appropriate VMR class. For example, many laboratory observations are similar enough in structure to be described by a single model. Since our models are in effect restrictions on the HL7 RIM, we hope to be able to convert them to an HL7 template formalism when one is available.

CONCLUSION

In the arena of clinical messaging similar needs for common representations have given rise to standards such as HL7 version 2. It is clear that similar standards are needed to enable executable guidelines and clinical decision support in general.

ACKNOWLEDGEMENTS

SNOMED is a registered trademark of the College of American Pathologists.

LOINC is a registered trademark of the Regenstrief Institute, Inc.

REFERENCES

1. Hripcsak G, Pryor TA, Wigertz OB. Transferring medical knowledge bases between different HIS environments. In: Prokosch HU, Dudeck J, editors. Hospital Information Systems: Design and Development Characteristics; Impact and Future Architecture. Amsterdam: Elsevier Science B.V.; 1995. p. 241-64.
2. Schadow G, Russler DC, Mead CN, McDonald CJ. Integrating Medical Information and Knowledge in the HL7 RIM. Proc AMIA Symp; 2000. p. 764-8.
3. Huff SM, Rocha RA, Bray BE, Warner WR, Haug PJ. An Event Model of Medical Information Representation. Journal of the American Medical Informatics Association. 1995, 2 (2):116-134.
4. Spackman KA, Campbell KE. Compositional Concept Representation Using SNOMED: Towards Further Convergence of Clinical Terminologies. Proc AMIA Symp; 1998. p. 740-4.
5. Dolin RH, Spackman KA, Markwell D. Selective Retrieval of Pre- and Post-coordinated SNOMED Concepts. Proc AMIA Symp; 2002. p. 210-4.
6. Rector AL. The Interface between Information, Terminology, and Inference Models. In: Patel V, et al, editors. MEDINFO 2001. Amsterdam: IOS Pess; 2001. p. 246-50.
7. Rector AL, Nowlan WA. The GALEN project. Comput Methods Programs Biomed 1994 Oct; 54 (1-2): 75-8.
8. Beale T. Archetypes: Constraint-based Domain Models for Future-proof Information Systems. OOPSLA 2002 workshop on behavioural semantics; 2002 [cited 2003 Mar 12]. Available from: URL: http://www.deepthought.com.au/it/archetypes.html.
9. CEN/TC251/WG II: Mori AR, Rector AL. Short Strategic Study on International Cooperation on Issues of Terminology. Final Version. WGII/N00-14; 2000 May 20.