MEDINFO 2004
M. Fieschi et al. (Eds)
Amsterdam: IOS Press

Design of a Standards-Based External Rules Engine for Decision Support in a Variety of Application Contexts: Report of a Feasibility Study at Partners HealthCare System

Robert A. Greenes[a], Margarita Sordo [a], Davide Zaccagnini[a], Mark Meyer[b], Gilad J. Kuperman[c]

[a] *Decision Systems Group, Brigham and Women's Hospital, Harvard Medical School, Boston, MA, USA;*
[b] *University of Illinois College of Medicine at Peoria, Peoria, IL, USA* [c] *Department of Clinical Practice Evaluation, New York Presbyterian Hospital, New York, NY, USA.*

Abstract

This project explored functional requirements for an institution-wide method, at Partners HealthCare, for interpreting clinical knowledge for decision support. Such knowledge is currently incorporated in a variety of clinical applications, yet the methods of representation and of execution vary and the ability to author/ edit the rules by human experts is limited. We expanded on a 2002 "Knowledge Inventory" at Partners to evaluate feasibility of designing a single representation approach entailing: (a) exploration of specific needs of different applications, in terms of kinds of response required (synchronous/asynchronous, time criticality, etc.), context (e.g., implied patient, time frame, or episode) , and kinds of actions to be triggered; (b) kind of representation of knowledge and feasibility of casting knowledge in the form of if...then statements; and (c) data and knowledge resources used (implied data model, and particular knowledge sources and terminology sources). The result of analysis was to design an architecture to accomplish this goal. We also did preliminary analysis of requirements for authoring for such a representation, and for implementation.

Keywords:

Clinical decision support, knowledge bases, rules engine

Introduction

Electronic, encoded knowledge is used effectively in a variety of highly successful clinical applications and subsystems at Partners HealthCare System, Inc. (Partners) to provide automated decision support. Such knowledge is encoded in actionable form, usually as *if...then* rules, where the *if* portion is a logical condition expression, and the *then* portion is an action – usually to recommend something and carried out via various means of notification (directly on screen for synchronous applications, or, for asynchronous applications, by email, text page, or by scheduling a notice for popup or display during a future login). Some rules may be encoded as tables in which conditions for firing are listed, for example, drug interactions. Those can also be recast in the form of *if...then* rules. Other kinds of knowledge, such as inferencing knowledge (e.g., ontological relationships), or groupings of elements (e.g., order sets, composite findings, or templates of data for forms) were not the focus of this current study.

The actual representation of rules knowledge varies among applications, depending on the design approach, system/programming language platform, as well as other factors, e.g., whether an authoring tool was provided for editing of rules, whether the logic is encoded in tables or in directly executable statements, or whether the knowledge is implicit in the sequence/ flow of data entry applications and their controlling logic. This variety of approaches makes it difficult for those individuals focused on the medical knowledge underlying quality/safety/ efficacy initiatives to assess the current rules in effect, or to make changes in them. It also makes it difficult to implement and manage a common set of rules throughout the integrated delivery system of Partners, across different implementation forms of the same kind of application, such as rules used in computerized physician order entry (CPOE) at different institutions. Lastly, by using dissimilar and non-standard approaches to representing knowledge, it is more difficult to incorporate or adapt externally derived or validated rules, or to export internal rules for external use. This latter capability anticipates the desirability and likelihood of future national evidence-based rules libraries in standard form.

Thus a single rules-representation approach would have much benefit to Partners, for knowledge authoring, editing, and update; for fostering consistency in rules implementation and maintenance in various Partners subsystems; and for both contributing to and using authoritative national-scale knowledge bases.

Having a consistent representation of rules to facilitate authoring and update doesn't necessarily imply that this representation will correspond to a single executable form that can serve the needs of different applications. It may be that considerable differences exist in the means necessary to adapt the representation or to recode rules to implement them in various subsystems.

The goal of this study was to determine the ways in which rules are used by various major Partners subsystems, to understand their requirements for representation of the rules knowledge. One aspect was to make explicit the contexts that various applications provide for a rule, sometimes implicitly, such as the fact that they are referring to a specific patient, episode of care (e.g., an abnormal lab result triggering a rule, or a medication that may have adverse effects being selected when contraindicated given a patient's health status) or time frame (whether an episode of care refers to a patient in intensive care, or an ambulatory patient). Another was to identify dictionaries, terminology sourc-

es, or specific knowledge bases used (such as laboratory normal ranges, drug formularies, or drug interactions). Yet another was the implied data model (particularly, which patient data are referenced). If these and related issues could be accommodated by a common representation, then uniform approaches to authoring could be considered. Further, if the common representation could be used to encode sufficiently precise information needed for execution, the task of implementing the knowledge would be facilitated. Note that a variety of approaches to implementation is possible (translation/ compilation, development of rules interpreters in various platforms, use of APIs, etc.) In this project, we focused primarily on identifying the authoring/editing and execution requirements, not the strategy for implementing them, since those tasks were beyond the scope of this feasibility study. We undertook the study with the understanding that if the feasibility project led to a conclusion that a single representation scheme meets a large proportion of needs of various applications, subsequent work would address authoring and implementation tasks.

While Partners-based implementation was a focus, issues of having an integrated approach to knowledge management and decision support across diverse applications within an institution, and among cooperating institutions are not unique to Partners. Further, safety and quality initiatives in health care depend on robust approaches that can be widely disseminated. Therefore, we believed it was important to adopt a standards-based approach that could, if successful, serve as a model for similar work elsewhere or for multi-institutional cooperative initiatives.

Past/Related Work

There is a large literature on event-condition-action rules in the database literature [1]. Arden Syntax is the only current standardized approach to representation of clinical decision logic, used in some information systems for encoding of Medical Logic Modules (MLMs), or single-step *if...then* rules [2]. However, Arden Syntax has a number of well-recognized limitations, primarily in its data model (not object-oriented, limited ability to express temporal conditions, etc.), and the need for each MLM to specify how data elements used in the logic are obtained from the host environment (known as the "curly braces problem", because these specification are enclosed in curly braces) [3].

The Decision Systems Group (DSG) participants in this feasibility study have worked with colleagues at Stanford and Columbia in the InterMed project, aimed at developing a common approach to representing clinical guidelines known as GLIF (Guideline Interchange Format) [4,5]. In 2000, InterMed helped to establish a Clinical Decision Support Technical Committee (CDS TC) in Health Level Seven (HL7), with a Clinical Guidelines and Arden Syntax (already part of HL7) becoming Special Interest Groups (SIGs), under it. The DSG focus in the CDS TC has been on refining the object-oriented expression language for query statements and calculations and decision rules used in GLIF, known as GELLO, as a potential standard for decision support [6].

Regarding rules in Partners systems, the DSG worked with Partners IS on a Knowledge Inventory (KI). The report [7], completed in September, 2002, assessed ways in which various applications at Brigham and Women's Hospital (BWH) utilize knowledge, and provided a starting point for this project.

Although Partners has developed a variety of applications that are at the forefront of health information systems in demonstrating ability of decision support to improve safety, quality, and efficacy of health care, the KI report documented the lack of a formal approach to knowledge management, and the plethora of ways in which knowledge is authored, encoded, and updated. Although the report focused on BWH, the problems were recognized as pertaining across Partners. As a result of the study, the need for an organization-wide approach to the above has been recognized, as well as the need to embrace emerging standards.

Methods

Following the criteria described in [7] we analyzed and re-defined the clinical knowledge embedded in six computerized clinical applications at BHW [8,9] and Massachusetts General Hospital (MGH). The studied applications were: 1)BWH CPOE, which allows physicians to enter orders (e.g., medications, laboratory and radiology tests) interactively, providing real-time decision support as part of the ordering process; 2) automatic alerting, which identifies serious clinical conditions and notifies the patient's attending physician while suggesting potential treatments for patient's condition; 3) adverse drug-events monitor (ADE), which reviews patients' medication profiles for pairs of interacting drugs (the program considers physiological changes, reflected in abnormal laboratory results, that may occur as a result of an adverse drug-drug interaction); 4) outpatient reminders; 5) results manager, an application to help clinicians review and act upon test results in a timely manner; and 6) MGH CPOE.

By analyzing current rules and contexts in which they appeared, we defined a common data model and a set of common rule forms. Each rule is described as Boolean combinations of simpler conditions or 'primitives' representing similar medical concepts within different contexts. We define context as a set of facts and/or circumstances surrounding an event. For example, the event: *new lab result indicating level of serum* represented by the primitive *'patient serum lab > Y'* may occur as part of an adverse drug-event *'patient on medication X AND patient serum lab > Y'*, or on its own, as part of automatic alerting. Hence the event (a new lab result) may occur 'inside' CPOE when a physician is about to order a medication (context: CPOE, medication about to be ordered) for a patient, or 'inside' automatic alerting (context: automatic monitoring for abnormal lab values).

Description of applications: A review of applications providing clinical decision support was done as part of the KI report [7]. Both BWH and MGH CPOE were designed by information teams at each hospital to allow physicians and other clinicians to enter all patient orders into the application. Each application contains about 95 rules categorized as follows: Medication (64 rules), Chemotherapy (5 rules), Radiology (15 rules), General (6 rules) and Laboratory tests (7 rules). Most of the medication rules are linked to medication dictionary systems (knowledge bases), where, for example, a single drug-drug interaction rule can check for 1631 distinct chemical combinations in the drug-drug interaction table (e.g., "cyclosporine and nelfinavir"). A similar mechanism applies for allergies, where a single generic

rule warns about 1,111 possible allergies stored in a table in the medication dictionaries. In other words, a single rule, linked to contextual information tables can provide extensive interventions with a significant impact in health care quality.

The automatic alerting application consists of 8 panic lab alert rules and 24 drug-lab interaction rules. It identifies serious clinical conditions and notifies patients' physicians about critical conditions reflected in abnormal laboratory results and drug-lab interactions. When a rule is triggered either by a possible drug-lab interaction or an abnormal change in lab results, a notification action is sent to the providers in a Coverage list [10].

The ADE monitor consists of 54 rules. The ADE monitor identifies clinical events that may indicate a possible occurrence of an adverse drug event. The rules in the application check for medical conditions, new medication orders, lab results above or below pre-defined threshold values, and medication orders associated with possible changes in lab results [11].

The outpatient reminder application consists of 25 active rules in four categories: health maintenance (reminders for screening tests), expensive medication reminders (suggests less expensive alternatives), diabetic care (reminders for annual routine tests) and therapeutic recommendations.

The results manager is an application in the outpatient setting that enables physicians to review, acknowledge and act upon results of chemistry, hematology, radiology, and cytology tests. It contains rules in the following categories: 24 rules for critical-results thresholds (lab results), 2 blood chemistry rules, 3 hematology rules, 14 pap smear rules and 5 mammography rules.

From all applications, we 1) identified active rules, 2) determined data elements involved, 3) determined implicit and explicit context to which each rule referred, and 4) identified data dictionaries, terminology sources and knowledge bases used.

Results

Proposed System Architecture: Figure 1 shows a proposed Client-Server architecture for an external rules engine. The goal is to have a centralized rule base and rule engine, so rules can be shared by different applications/entities interacting in the system, and can be authored and maintained/updated more readily by domain experts. Our approach supports integration and sharing of data and resources through an architecture oriented to entities and processes, where the former generate events to be processed by the latter. This architecture supports access to data sources for patient information, as well as to knowledge bases (e.g., formularies or drug interaction tables, medication dictionaries), and a rule base. To have a common representation of rules, the contexts where such rules appeared were identified and modeled: where (e.g., CPOE, automatic alerting), when (e.g., a test is overdue, a test is being ordered before the recommended time) and why the rule is triggered (e.g., medication is being ordered and there is interaction with a current medication, laboratory result indicates a medical condition, overdue test has not been performed).

On the client side, clinicians and front-end applications (e.g., CPOE, ADE, Automatic Alerting) access the server to store and retrieve information. These activities can be carried out in two modes: 1) Synchronously, e.g., a physician orders a medication or procedure, and the physician's actions are directly coupled with actions/responses from the system. 2) Asynchronously, e.g., a lab result is generated by the corresponding entity with no actual interaction with a user, or a background process is running to look for potential ADEs.

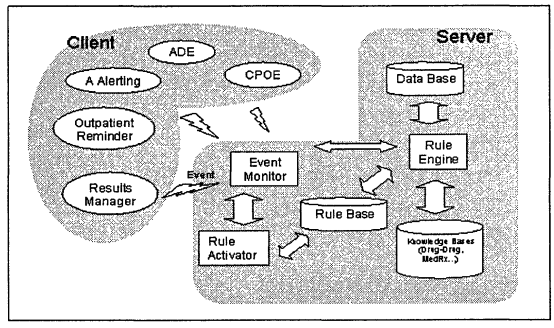

Figure 1 - System Architecture – schematic view

The server side comprises a rule base, databases with patient information, knowledge bases, rule engine, event monitor and rule activator. Whenever an entity on the client side generates an event, the event monitor receives a message indicating the source (where/who/what is generating the event), the reason of the event (new lab result, medication being ordered, etc.), the intended receiver, and, of course, the new information. The event monitor 'contacts' the rule activator. The rule activator selects the potential rules from the rule base that could be executed given the information contained in the message. Such rules are executed by the rule engine by 1) retrieving any additional information from knowledge and data bases, 2) evaluating the primitives and Boolean combinations thereof in the logic part of the rule, 3) triggering the corresponding actions, e.g., screen notification, paging, storing derived data and, 4) sending a message acknowledging completion of execution.

Common data model: Figure 2 shows a simplified view of the data model used in this proposed system. The design considers the implicit and explicit contexts of rules in determining the nature of references data elements and knowledge sources. It stores data from patients: general information, problem list, medications currently taken, lab test status or results, and allergies. It also contains information about healthcare staff (role, activity status, availability, etc.). It shows links to knowledge bases: medication databases (e.g., RxNorm), alternative medications, drug-drug interactions, food-drug interactions, vocabulary (e.g., SNOMED, ICD9, RxNORM, LOINC). The design of our proposed system is flexible in that it can reference whatever vocabularies are desired for particular classes of data. Similarly, the whole data structure can be mapped into the HL7 RIM standard data model. Table 1 shows how tables in our model map into HL7 RIM (v 3.0) classes [12]. The left column in Table 1contains classes used in the proposed data model, while the right column contains the RIM classes required to represent such data.

Table 1: The Data model and the equivalent in RIM

Class in Data Model	Equivalent Classes in RIM
Allergy	Act/Observation
Patient	Person/LivingSubject/Role/Participation
LabResult	Act/Observation
ProblemList	Act/Observation
Medication	Act/SubstanceAdministration
Staff	Person/Role/Employee

Primitives: shows that from about 250 rules in all 6 systems, we identified a total of 51 primitives which we modeled by translating them into SQL queries. Of these 51, only 41 primitives were unique.

Table 2: Modeled Primitives

System			Modeled Primitives	
Name	No. Rules	No. Primitives	No.	%
Automatic Alerting	32	9	9	100
Outpatient Reminders	25	14	14	100
MGH CPOE	95	9	9	100
BWH CPOE				
Results Manager	48	9	9	100
ADE	54	10	10	100

Handling events. An event is a notification occurring in response to an action, such as a change in state, or as a result of the user clicking the mouse or pressing a key. An event handler receives control when the corresponding event occurs. For our data model, an event handler 'catches' any event generated by entities in the system, e.g., a lab result that becomes available, a physician using CPOE selecting a medication, or a time elapse signaling an overdue lab test. The event handler then triggers the execution of whatever rules may apply given the type of event, data items and the context where such event occurred. Event handling is centralized. Events generated by clients are handled on the server side. The proposed model for handling events is message-driven transactions.

Entities in the system can communicate between themselves through messages, sent/retrieved to/from a message queue. A message header contains information about the sender and intended receiver. Entities create/consume messages from the queue without the need to tightly couple sending and receiving of data. Communication is guaranteed by reliability of the messaging system. In Figure 1, an event monitor handles all incoming/outgoing messaging from and to sources.

Synchronous and asynchronous events: Synchronous events require sender and receiver to actually communicate and respond to each other's requests. Synchronous messaging is modeled by forbidding the intended receiver of a message to perform any activity until it acknowledges reception of the message. Although asynchronous events do not require actual interaction with the recipient, 'timers' for receiving an acknowledgement can be attached to the message. Hence, an asynchronous message may be sent to the message queue without requesting immediate acknowledgement from the intended recipient, however, if after the predefined waiting time an acknowledgement has not been received, an acknowledgement request will be sent to the receiver.

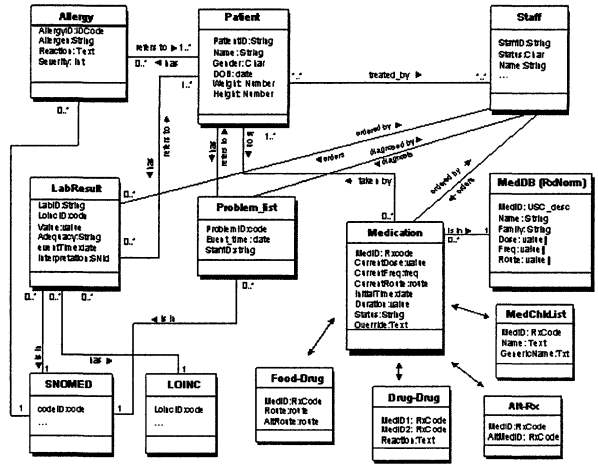

Figure 2 - Data Model – simplified view

ActionModel: The action model depicted in Figure 3 shows a proposed action taxonomy. The model is by no means exhaustive; although it contains the major classes in the action taxonomy, the current version focuses primarily on notification actions. The main class is the *Action* class. It contains four attributes: the type of event that triggers the action (*TriggeredBy*), the recommended action, a list of recipients (staff personnel) sorted in order of importance (first recipient in the list receives the first alert notification, and so on) and a message string displayed on the computer screen. *TriggeredBy* is split into three major classes indicating the type of event that triggered the action.

ActionType indicates the action to be carried out by the recipient(s) of the message For notification actions, the recipient is a staff member. The ActionType splits into five major classes: *LabOrder*, the system suggests laboratory tests should be ordered/cancelled; *SubstanceAdministration* indicates that a medication may be ordered/cancelled; *Cancel* gives the option to cancel an action carried out by the recipient which triggered the alert message; *Referral* indicates that the patient may be referred to other health care providers; and *GetApproval* indicates that the action carried out by the recipient requires further approval.

RecipientList is a list of recipients of the alerting message sorted in order of importance; the first recipient in the list is notified first. The *Recipient* class provides the recipient ID, the allotted time for receiving a response from the recipient before considering the notification as unacknowledged and sending a new notification to the next recipient in the list. The method of notification is also specified, e.g., pager, POD screen, email, floor staff. *Message* is the alert message sent/displayed.

We developed a prototype template-driven author interface to facilitate selection of rule primitives and filling in of the relevant parameters. The authoring tool automatically generates SQL statements implementing the rule against a sample database, showing feasibility. Regarding implementation, the small number of data classes referenced and action services invoked show the tractability of doing the data mapping and interfacing to services to utilize a common rules engine in most host platforms.

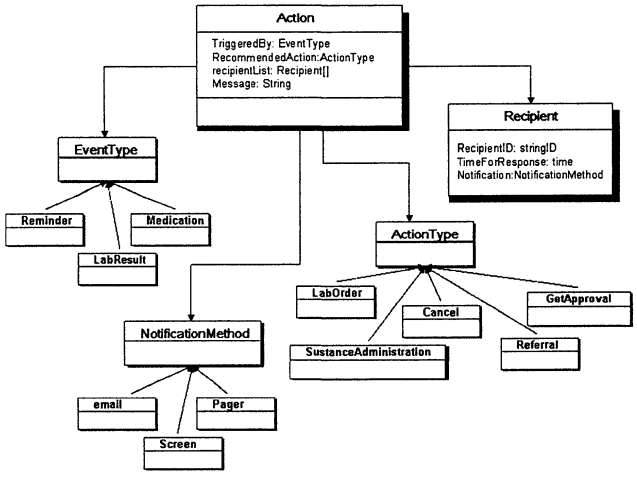

Figure 3 - Action Model – simplified view

Discussion

We have evaluated the feasibility of the above design by verifying through our analysis that all of the rules in the reviewed applications can be executed. We have shown that only a relatively small set of primitives are needed to represent a wide variety of rules. We have further shown that a data model with a small number of classes represents the data and knowledge referenced. Further, we have shown that the classes can be mapped to the HL7 RIM. The primitives as well as compound logic expressions can be readily expressed in the object-oriented GELLO expression language currently under review in HL7.

We have shown that the above parsimony lends itself to development of authoring tools that use templates or wizards, and that implementation of a rules engine is feasible. Future work involves pilot implementation and establishment of a knowledge management approach at Partners that incorporate this model.

Acknowledgment.

Supported in part by a grant from the Partners Information Systems Research Council.

References

[1] Paton N, ed. Active Rules in Database Systems. Heidelberg: Springer Verlag, 1999.

[2] Hripcsak G, Cimino JJ, Johnson SB, Clayton PD. The Columbia-Presbyterian Medical Center decision-support system as a model for implementing the Arden Syntax. Proc Ann Symp Comput Appl Med Care. 1991; 248-252.

[3] Jenders RA, Sujansky W, Broverman CA, et al. Towards improved knowledge sharing: assessment of the HL7 Reference Information Model to support medical logic module queries. Proc AMIA Ann Fall Symp. 1997; 308-312.

[4] Ohno-Machado L, Gennari JH, Murphy SN, Jain NL, Tu SW, Oliver DE, Pattison-Gordon E, Greenes RA, Shortliffe EH, Barnett GO. The guideline interchange format: a model for representing guidelines. J Am Med Inform Assoc. 1998 Jul-Aug; 5(4):357-372.

[5] Peleg M, Boxwala AA, Ogunyemi O, Zeng Q, Tu S, Lacson R, Bernstam E, Ash N, Mork P, Ohno-Machado L, Shortliffe EH, Greenes RA. GLIF3: the evolution of a guideline representation format. Proc AMIA Symp. 2000; 645-649.

[6] Sordo M, Ogunyemi O, Boxwala AA, Greenes RA. Software specifications for GELLO: An object-oriented query and expression language for clinical decision support. http://www.hl7.org/Library/Committees/dss/GELLOspecsJan041%2EPDF, 2004.

[7] Boxwala AA, Denekamp Y, Greenes RA, Kuperman GJ, Middleton BL,. Survey and evaluation of knowledge bases for clinical decision support. Proc MEDINFO2004, IMIA, 2004; in press.

[8] Kuperman GJ, Teich, JM, Bates DW, McLatchey J, Hoff T. Representing hospital events as complex conditionals. AMIA 1995; 137-141.

[9] Kuperman GJ, Teich, JM, Gandhi TK, Bates DW. Patient safety and computerized medication ordering at Brigham and Women's Hospital. Journal on Quality Improvement 27 (10), 2001; 509-521.

[10]Kuperman GJ, Teich, JM, Bates DW, Hiltz FL, Hurley JM, Lee RY, Paterno MD, Mus B. Detecting alerts, notifying the physician, and offering action Items: A comprehensive alerting system. AMIA 1996; 704-711.

[11]Jha AK, Kuperman GJ, Teich JM, Leape L, Shea B, Rittemberg E, Burdick E, Seger, DL, Vliet MV, Bates, DW. Identifying adverse drug events: Development of a computer-based monitor and comparison with chart review and stimulated voluntary report. JAMIA, 5(3), 1999; 305-314.

[12]HL7 RIM http://www.hl7.org/Library/data- model/RIM/modelpage_mem.htm

Address for correspondence

Robert A. Greenes, M.D., Ph.D., Brigham & Women's Hospital, 75 Francis Street, Boston, MA 02115, email: greenes@harvard.edu