Managing the Alert Process at NewYork-Presbyterian Hospital

Gilad J Kuperman, MD, PhD, 1, 3, 4 Rosanna Diamente, RN, MA, 1 Vrinda Khatu, 2 Terri Chan-Kraushar, 2 Pete Stetson, MD, MA, 1, 3 Aurelia Boyer, RN, 2 Mary Cooper, MD, JD 1

(1) Departments of Quality Assurance and (2) Information Systems, NewYork-Presbyterian Hospital, (3) Department of Biomedical Informatics, Columbia University, (4) Department of Public Health, Weill-Cornell Medical College, New York, NY

Clinical decision support can improve the quality of care, but requires substantial knowledge management activities. At NewYork-Presbyterian Hospital in New York City, we have implemented a formal alert management process whereby only hospital committees and departments can request alerts. An explicit requestor, who will help resolve the details of the alert logic and the alert message must be identified. Alerts must be requested in writing using a structured alert request form. Alert requests are reviewed by the Alert Committee and then forwarded to the Information Systems department for a software development estimate. The model required that clinical committees and departments become more actively involved in the development of alerts than had previously been necessary. In the 12 months following implementation, 10 alert requests were received. The model has been well received. A lot of the knowledge engineering work has been distributed and burden has been removed from scarce medical informatics resources.

Introduction

Quality problems in health care have been well-documented. 1 ii iii Fernandopulle and others have suggested that substantial improvement in health care quality will require ii iv (1) health care organizations discovering innovative ways to improve quality, (2) effective professional training and continuing education, (3) increasing the evidence base in health care, (4) assuring that evidence is applied when it should be, (5) aligning reimbursement with quality improvement, and (6) implementing information technology including electronic health record (EHR) systems.

EHRs have the ability to improve quality in a variety of ways. First, EHRs can create efficiencies and opportunities for innovation because they fundamentally alter the ways clinicians do their work. v Specifically, EHRs change (1) clinical results management, (2) orders management, (3) clinical encounter documentation, (4) clinical communication, and (5) interaction with administrative (e.g., billing and scheduling) systems. Second, EHRs can serve as a data source for analyses as health care organizations measure their current level of performance. For example, a clinical documentation application can capture information about the adequacy of pain control or the extent of a pressure ulcer.

A third and important way that EHRs can improve quality is through the use of rule-based clinical decision support (CDS) systems. vi Rule-based CDS systems embedded in EHRs represent a powerful way to improve care because they can help guide the clinician at the time that the individual health care decision (e.g., the order, the documentation of the encounter) is being made. A study estimated that implementation of ambulatory computer order entry nationwide would save the health care system $44 billion dollars annually, including 190,000 admissions due to avoiding preventable adverse drug events. vii These projections were contingent on the presence of sophisticated clinical decision support; at “intermediate” levels of decision support, the projected savings were much less.

There are several prerequisites to the creation of a successful CDS system: (1) clinicians must be active users of the EHR, for example, suggestions for appropriate drug use only make sense if prescribing is being done electronically, (2) requisite data must be present in an automated form, for example, an automated risk stratification algorithm for coronary disease requires that the family history be present in the patient database, (3) automated data must be well structured and coded, i.e., narrative text generally is inadequate for a rule-based CDS
system, (4) there need to be appropriate ways to notify the provider about the results of CDS, including paging and email for asynchronous alerts, and (5) the rules themselves must be created and maintained over time (so called knowledge engineering activities).

The process of knowledge engineering for CDS systems is difficult. Experts must be engaged and their knowledge represented in an explicit and unambiguous manner. Published guidelines can serve as a starting point to some extent, but experts still must resolve inconsistencies and ambiguities. Also, knowledge changes over time and rules must be kept up to date.

At NewYork-Presbyterian Hospital, we have had a clinical information system with results management, order management, clinical documentation and clinical decision support in place since the late 1990’s. By late 2002, several CDS rules active in the production system, however, we were starting to notice inefficiencies in our rule management processes. For example, we had a long list of requests for new alerts. Some of the requests had come in months or years earlier and it was not clear whether the request still was relevant, who had been the original requestor or who should be the contact person for clarification of issues. Also, some active alerts were becoming out of date (e.g., renal dosing rules for antibiotics) and it was not clear who should be responsible for updating those rules. In early 2003, we decided to re-evaluate how we were managing rules in our clinical information system. This paper is a description of the issues we addressed and the new processes we put into place.

Setting

NewYork-Presbyterian Hospital (NYPH) is an academic medical center in New York City, NY. It was formed in 1999 as the result of a merger between New York Hospital, which was affiliated with Cornell University Medical School, and Columbia-Presbyterian Hospital, which was affiliated with Columbia University Medical School; NYPH now is affiliated with both medical schools. NYPH has 2400 beds spread over 5 sites, has over100,000 discharges annually, provides over 1 million outpatient and ED visits annually. At the time of this project, the Eclipsys Sunrise Clinical Care (SCC) application was in place at 3 of the 5 sites and was being used for results review, order management (including CPOE) and clinical documentation by nurses and some physicians.

Between 1999 and 2002, about 20 alerts had been developed. Some of these are shown in Table 1. An Alert Committee had been created in 1999 and was responsible for coming up with ideas for alerts and overseeing their management. The Committee was multidisciplinary and included representation from nursing, pharmacy, laboratory, radiology, IS, quality, medical house staff and medical attending staff. One individual (an informatics-trained MD) did the bulk of the knowledge engineering, project management, and even some technical work for the alerts. Work for the Alert Committee occupied a majority of the MD’s time.

By 2003, even though several alerts were in production and were working well, some important problems with the overall alert management process were beginning to emerge. First, data were not being collected routinely about how often the alerts were firing. Second, due to personnel changes some institutional memory had been lost and there was no record of the detailed functioning of some of the alerts (i.e., no explicit specifications had been documented for the alerts). Third, when suggestions were made for changes to alerts, it was not clear who had final authority over detailed alert functionality. Fourth, there was a long list of requests for more alerts. Many of these were one-line requests, e.g., “lab-based ADR detection”, “positive blood cultures”, etc. In many instances it was lost to history who had originally requested these and who would be the contact person for clarification of intent and issues of knowledge engineering.

Table 1. Example alerts developed at NYPH

- Dosing critique for antibiotics in renal insufficiency
- Suggestion to alter hypoglycemic when make patient NPO
- Pregnancy screening for radiology
- Lab trend alerts
- Jehovah’s witness transfusion warning
- Display of relevant labs at time of medication order
- Suggested alternative therapies
- Suggest rehab consult based on nurse charting data
The Alert Committee set about to create a new model for alert management that would help resolve these issues.

**New model for alert management**

**Overarching principles.** There were two key guiding principles as the new alert management model was created. First, the Alert Committee recognized that good ideas for alerts might come from anywhere in the organization, not just from members of the Alert Committee. Indeed, one of the strengths of the academic medical center is the clinical expertise that resides broadly throughout the institution. The Committee felt its role was to create the processes to allow the broad knowledge of the enterprise to be encoded as rules. The Alert Committee felt it did not have to be the primary source of ideas, though it may advocate for some that it believes would be worthwhile. Second, the Alert Committee wanted to create a model that would assign clear responsibilities for alert creation and alert maintenance. The Committee believed that a rule is a statement of what the organization believes is best care and that only Committees that have authority to create medical policy should be allowed to sponsor rules. By distributing knowledge engineering activities throughout the institution, the Committee hoped to create an alert management process that was scalable and sustainable.

The distinction between the model that existed prior to 2003 and the subsequent one is shown in Figure 1. Whereas under the earlier model, the Alert Committee was the source of ideas for rules and alerts, the new model sought to elicit ideas for alerts from throughout the institution.

It should be mentioned that the change in model in no way was meant to malign the original model. In fact, the earlier model probably was appropriate to an institution that is becoming familiar with the nuances of alert implementation. At a certain point however, the original model did not scale and a new model was needed.

**Details of the model.** The Alert Committee put forward a more detailed set of principles that would govern the creation and management of alerts. The components included:

1. Alerts must have a sponsoring body that is a hospital committee or department
2. Alerts also must have a named requestor. The requestor is the individual who can help refine the details of the alert logic and the alert message. If the requestor leaves the organization, the sponsoring committee is obliged to name a new requestor.
3. The sponsoring committee and the requestor are responsible for reapproving the alerts on a regular basis (two years was chosen as the reapproval period)
4. Alerts must be requested in writing using an “alert request form” (ARF). The request form is not intended to be a bureaucratic hassle but rather a tool to support the concept of sponsorship and avoid the one-line e-mail request for an alert.

**Alert request form.** The alert request form consists of 7 fields and fits on one page (though when submitted, it may be augmented by additional material such as screen mock-ups and/or logic flow diagrams). The fields on the alert request form are:

1. Alert title
2. Sponsoring department
3. Requestor
4. Description of the alert
5. Rationale for the alert
6. Evaluation metric
7. Applicable on all units or just some (e.g., pediatric/adult only)

![Figure 1. Comparison of old and new models of alert management](image-url)
Process from alert request to software. In addition to creating the alert request form, the Alert Committee identified the process for converting the request to software.

1. The requestor completes the alert request form and sends it to the Chair of the Alert Committee.
2. The Alert Committee chair reviews the form for completeness and forwards it to the manager of the technical team. If the Alert Committee Chair has any questions about the request, he will communicate with the requestor.
3. The technical team lead develops a work estimate. If there are any questions about the details of the alert, the technical team lead can contact the requestor directly. The Chair of the Alert Committee will get involved if any complex issues arise as part of the clarification.
4. The Alert Committee prioritizes the development of the alert along with other alert requests. Generally, alerts to address important safety concerns are prioritized highest. The IS department has resources dedicated to alert development.

Implementation of the model

The NYPH Clinical Alert Committee agreed in March of 2004 to adopt this process. Implementation of the process has proceeded well but has taken some time. Some experiences encountered in the implementation are described.

Educating the organization. The new policy was presented to three committees at NYPH that historically frequently had generated alert requests – the Significant Events Committee, the Medication Safety Committee, and the Formulary and Therapeutics Committee. Other relevant committees (e.g., Laboratory Committee, ICU Committee) were informed indirectly because they have members who also sit on the Alert Committee.

By and large, the clinical committees understood the rationale for the policy and agreed to abide by it even though they realized it would take more work on their part.

Alerts developed under the new policy. In the 9 months following approval of the new policy, 10 alert request forms were received by the Alert Committee. The list of alerts is shown in Table 2. In almost all the cases, the request form was filled out correctly and completely. In about half the cases, the alert request was specified well enough so that it could be passed directly to the technical team for a work estimate. In the other half of cases, the Alert Committee chair had to have a conversation with the requestor to clarify some details of the request.

Once the technical team leader had received the request form, she had sufficient information to create a work estimate in about two-thirds of the cases. The rest of the time, some clarifying information was required from the requestor.

Discussion

Our model has several strengths. It has created explicit sponsorship for alerts and has reduced the bottlenecks in knowledge engineering. Previously, the informatics physician had to do all the knowledge engineering; now much of the work is pushed out to domain experts and scarce medical informatics resources reserved for complex situations. Alert Committee meetings are not design sessions and instead can be used to address more global alerting issues, such as whether alerts are achieving their intended purposes, how to improve user satisfaction with alerts, and how to achieve enterprise-wide sponsorship of alerts. The alert request form assures that the logic of the alert is specified and documented. Information Systems staff appreciate having a well-defined process for receiving requests and written specifications; their satisfaction with the system is high.

The model still requires a fair amount of administration. Incoming alert request forms must be tracked and routed to the appropriate parties and the status of alerts in development must be tracked. At our organization, a quality improvement specialist manages this process. Because responsibility for the creation of alerts has been distributed, the quality of the alert

<table>
<thead>
<tr>
<th>Table 2. Alert request forms received since new process was implemented</th>
</tr>
</thead>
<tbody>
<tr>
<td>• IV-&gt;PO suggestion in CAP</td>
</tr>
<tr>
<td>• Duplicate potassium order</td>
</tr>
<tr>
<td>• Potassium order and no recent low K</td>
</tr>
<tr>
<td>• Pediatric excess K from K phos</td>
</tr>
<tr>
<td>• Pediatric extubation NPO alert</td>
</tr>
<tr>
<td>• Fall risk injury documentation reminder</td>
</tr>
<tr>
<td>• Start time field change</td>
</tr>
<tr>
<td>• Last aminoglycoside admin. Display</td>
</tr>
<tr>
<td>• Darbopoeitin-&gt;erythropoeitin substitution</td>
</tr>
<tr>
<td>• Transfusion appropriateness alerts</td>
</tr>
</tbody>
</table>
messages has been uneven – some are clearer than others. We are putting into place a process whereby physician end users will review alert messages for clarity prior to the implementation of the alert. The pre-release alert review will add a step to the alert creation process, but should improve user satisfaction. Also, about 4 weeks after alerts go live, a note will be sent to house staff asking for feedback on the functioning of the alert. This “post-marketing surveillance” will allow us to refine and improve the alert.

One weakness of the model is that design changes made after the alert request form has been completed may not be documented reliably. Also, sometimes no sponsor steps up to champion a domain where the Alert Committee feels there is clear opportunity for clinical benefit from clinical decision support systems. For example, the Alert Committee had to cajole a group into sponsoring alerts for laboratory trends and drug-lab interactions.

Other efforts at alert management have achieved some of the same benefits documented here. For example, the Arden Syntax\(^\text{a}\) assures that alert logic is well documented and provides “library” and “maintenance” slots to facilitate sponsorship. However, the Arden syntax is a tool for the technical team; its optimal use still would require a process that involves domain experts and a knowledge engineering process.

Geissbuhler and Miller described a distributed knowledge management model implemented at Vanderbilt University Medical Center (VUMC) that is very similar to the one we implemented at NYPH\(^\text{a}\) The model at VUMC handled other knowledge components such as documentation templates and order sets. The VUMC model allows individual end users (as opposed to medical committees and departments) to create and manage knowledge. This may be appropriate because the orders sets and templates at VUMC were largely for one individual’s use; the NYPH model is designed more for institutional implementation. At NYPH we have not extended our model to order sets or documentation template, although this might be possible in future.

HIMSS has published a comprehensive guide which reviews all the facets of implementing clinical decision support.\(^\text{a}\) The HIMSS guide outlines options for organizational structures and processes for the creation of CDS which would need to be tailored at an individual organization.

Future work at NYPH will include experiencing the re-approval process. Because we still are relatively early in our work, we have not yet had to re-approve any rules. According to our model, a rule would be “turned off” if we cannot get re-approval. We also are moving to a new version of the vendor’s software. We expect that the processes we have developed will transfer seamlessly to the new technical environment, but we need to confirm that.

We also have yet to make full use of the evaluation metric aspects of our request form.

**Conclusion**

Clinical decision support systems can improve the quality of care, but the burden of knowledge management is high. Successful models will need to distribute knowledge management tasks across the enterprise. We have implemented one such model and have documented several strengths and some weaknesses of the model. We will need to continue to evaluate this model as the number of decision support rules in our system grows over time.

**References:**