

Clinical Event Monitoring at the University of Pittsburgh

Michael M. Wagner¹, Marvin Pankaskie¹, William Hogan¹, Fu-Chang Tsui¹,
Stuart A. Eisenstadt^{1,4}, Eric Rodriguez³, and John K. Vries²

¹Center for Biomedical Informatics, ²Office of Biomedical Informatics, ³Division of Geriatrics,
University of Pittsburgh; ⁴Information Services Division, University of Pittsburgh Medical Center

Although the literature on event monitoring is extensive, it does not cover all issues that we encountered while developing an event monitor at our institution. We had to resolve issues related to event detection, scalability, what topics were suitable for asynchronous decision support, and overlap of efforts of other groups at the institution attempting to improve quality and lower cost of care. In this paper, we describe our experience deploying CLEM, the clinical event monitor at the University of Pittsburgh with emphasis on these topics.

INTRODUCTION

The term *clinical event monitor* refers to an embedded expert system in the domain of clinical medicine (e.g., (1, 2)). Clinical event monitoring can be classified into synchronous (when the expert system runs in response to data entered by a clinician in real time as in order entry) or asynchronous (all other circumstances, for example when it is triggered by the arrival of laboratory results).

We designed and deployed an asynchronous clinical event monitor. Our design objectives were: (1) that it be scalable to the enterprise level with a large number of rules, (2) work in an environment that included many legacy systems, (3) support complex care plans as in a disease management application, (4) support just-in-time education, and (5) could also function with response time adequate for synchronous mode.

Although there is a sizable literature on technical and social aspects of clinical event monitoring, we encountered questions for which answers were not available. This paper gives an overview of our work developing CLEM, the clinical event monitor at the University of Pittsburgh Medical Center (UPMC). We give special emphasis to those areas in which our experience differed from previously published experience, or for which little guidance was available.

SYSTEMS ENVIRONMENT

UPMC is representative of modern health systems in the sense that it is a growing amalgam of previously independent practices and hospitals. The information system environment reflects this heritage, comprising a large number of legacy systems. Our current approach to data integration is data warehousing: Each system (e.g., the pharmacy system) sends data in HL-7 format

through a message router to a proprietary data warehouse (Fig. 1).

This architecture poses challenges for clinical event monitoring. Because of limitations in legacy systems, often data come from transaction queues in systems whose data model may not have a one-to-one correspondence with the model needed by an event monitor. In the worst case, a sequence of data over time define an event.

CLEM

A clinical event monitor evaluates events, in the context of everything (ideally) that can be known about a patient, and it communicates its conclusions via some communication channel to a person. The basic infrastructure needed to support event monitoring is a source of events, a source of patient data, and a notification service. The event monitor itself is an algorithm that takes as input events, patient data, and a representation of medical knowledge, and outputs messages (1). We will use this framework to describe CLEM.

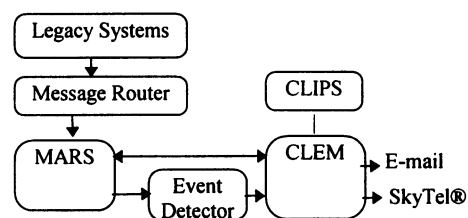


Fig 1. CLEM architecture.

Event detection

We define an **event** as data that triggers the execution of an event monitor. What can be an event is constrained by the requirements of knowledge engineering; that is, domain experts and knowledge engineers need to define events that are natural to the medical problem being modeling. Thus, there may not be a 1-to-1 mapping between data available to an event monitor and the events of interest referred to in a knowledge base.

We classify events into three types—atomic, atomic temporal, and compound temporal—based on the characteristics of the mapping: Our classification differs from that in (1), which is not based on the nature of the mapping problem.

An **atomic event** is an event for which there is a 1-to-1 correspondence with external data. For example, in many systems the storage of a new laboratory result is an identifiable data event that corresponds to an event of interest to domain experts.

An **atomic future event** is an atomic event, usually created by the event monitor itself, that causes the inference algorithm to run at some time in the future (e.g., a check for the existence of a gentamicin level 48 hours after gentamicin is prescribed). To implement atomic future events, an event monitor must have a persistent data store.

A **compound temporal event** is a sequence of data that, when recognized, causes the inference engine to run. Compound temporal events require both persistent storage *and* algorithms that recognize temporal patterns. An example of a compound temporal event in our environment is a medication dosage change. This event cannot be seen directly in the data. Instead, it must be deduced from a sequence of records in which the medication is discontinued then restarted at a different dose. This example illustrates a subtle limitation in compound event detection, and, by extension, decision support from data warehouses fed by legacy systems: The meaning of the “discontinue” record changes from “stop of medication” to “start of edit” based on the arrival of a subsequent record (whose arrival time is not constrained by current protocol). If there is no upper bound on the time when a data sequence may complete, then that compound event cannot be detected.

In our experience, most events in our environment are compound temporal events. For example, even the receipt of an HL-7 transaction of a potassium result from the laboratory system into the warehouse is not an atomic event because it could be a duplicate report. Only after an algorithm checks this datum against a persistent store of transactions to ensure that it has not been processed earlier, will it qualify as an event-of-interest to a knowledge engineer.

For this reason, CLEM, unlike previously described systems, uses a separate module, called an **event detector**, to identify compound temporal events. The input to the event detector is a stream of new patient data and the output is a stream of events that are referred to in CLEM’s rules.

Source of patient data

CLEM gets all patient data from a data warehouse. The data warehouse is a locally developed repository based on a document retrieval model with a highly parallel architecture (3). The data warehouse handles the “vocabulary problem” by mapping from legacy system language to locally defined canonical forms.

The retrieval of patient data is usually a performance bottleneck for event monitors and it is therefore a key

issue for scalability. CLEM does several things to improve efficiency. First, rather than call the database incessantly during rule evaluation, CLEM stores *all* of a patient’s data in memory before evaluation. Temporal inference and temporal pattern recognition are also potential bottlenecks, therefore CLEM stores patient data in time-sorted order in memory as list structures that support efficient temporal inferences. Ultimately, we believe that the requirement for efficiency and scalability will dictate that data warehouses cache patient data in structures suitable for inference engines, and that the API between integrated patient databases and event monitors will provide the transfer of binary images of these structures.

Second, CLEM processes events in batch mode. Between batches, the event detector accumulates events. In each batch (arbitrarily set to run at 30-minute intervals at present), CLEM clusters events by patient and then evaluates all events for the patient together; thus, if there were more than 1 event for a patient during a cycle, at least one database query would be eliminated.

In summary, our experience and analysis suggests that to achieve scalability (both in numbers of patients and in size of rule sets) and bounded response time (in the real-time range) the following requirements must be met: (1) the integrated patient database must maintain a cache for each (or recently active) patient as a binary image that is suitable for the inference engine, (2) the architecture of the integrated patient database must support replication with respect to this service, (3) the architecture of the event monitor should support multiple inference engines, and (4) the load on the inference engines must be distributable by patient.

Inference engine and representation

We based our inference engine on CLIPS, a forward chaining rule-based expert system shell developed by NASA. The rationale for this choice and our experience with it are described in (4).

Notification

CLEM uses e-mail and Skytel® 2-way pagers to communicate alerts to interns. From 7:30 a.m. to 2 p.m. on weekdays, CLEM addresses the alert to the intern responsible for a patient. Between 2 and 6 p.m., CLEM also cc’s the intern who will be cross-covering for the primary intern that evening. Between 6 p.m. and 7:30 am, and all day on weekends, CLEM sends alerts to the cross-covering intern and cc’s the primary intern. Example alerts are in the Appendix.

CLEM determines the identity of the intern responsible for a patient from a database that it maintains. This database is populated with data from another clinical system (Clinipac) that interns use to maintain lists of their patients for sign out and report generation. This source of primary intern coverage is

very reliable with the exception of interns rotating in the ICU who keep all ICU patients on their lists.

CLEM determines the identity of the cross-covering intern from a monthly schedule that is generated by the Department of Medicine. Readers interested in coverage lists should see (5).

INTRODUCTION OF THE SERVICE

There is extensive literature on techniques for successful introduction of physician order entry [e.g., (6, 7)], but less on the introduction of an asynchronous system that uses e-mail or pagers. The general principles that we could glean were (1) find a 'slam-dunk' alert, meaning something that would be a true positive with high utility, (2) solicit support from clinical leadership, and (3) connect your efforts to existing committees and groups involved with quality assurance. We organize this section around our experience in applying this advice.

Slam-dunk alerts

We reviewed the CPMC repository (8), and the literature on reminding (9), (10), (11) for alerts with high expected utility. We initially thought that rules about renally excreted and nephrotoxic drugs would satisfy the 'slam-dunk' desiderata. We also had discussions among the clinicians in our group and a respected division chief. Based on these sources, we implemented the alerts listed in Table 1.

Support of clinical leaders

We promoted the system whenever the opportunity presented itself at seminars and committee meetings, attempting to obtain a top-down mandate for deployment, however it was not forthcoming. Therefore we sought and obtained permission from the Chief of Medicine and the Director of the Medical Residency program to send alerts to medical interns on a voluntary basis, and we kept them informed of progress through weekly reports. As we will discuss, after 4 months of operation with volunteers, the clinical leadership in consultation with the medical chief residents agreed to make CLEM an official service in the sense that interns would be expected to read e-mail and use 2-way pagers when on service.

Involvement of quality improvement teams

At UPMC, the Department of Pharmacy Services is active in quality improvement activities. In particular, Pharmacy had identified 60-70 problem areas and had estimated their financial impact. These analyses were a basis for prioritizing our efforts. Pharmacy already had several initiatives in place, so regular meetings occurred between author MMW and the director to prevent overlap of interventions.

Phase-in of event monitoring

CLEM was introduced on a voluntary basis in February, 1997. We asked 6 interns who were on-

service if they would be willing to volunteer to receive alerts. All agreed to receive e-mail alerts.

Table 1. Alerts and routes. No route means alert is inactive.

Topic	E-mail	Pager
Hct drop > 4.0 pts.	+	+
DC Telemetry, MI ruled out	+	+
Check level of therapeutic drug	+	
Renal Insufficiency and demerol	+	+
Creatinine rise while on nephrotoxic drug	+	
New med/Review dose guideline	+	
Clearance change/Review meds	+	
Positive microbiology culture	+	+
Radiology report impression	+	+
Drug level result report	+	+
Review ceftriaxone dose	+	
Guideline for use of vancomycin levels	+	
Long acting benzodiazepines in the elderly	+	
NSAIDs in elderly	+	
Propoxyphene in elderly	+	
Amitriptyline in the elderly	+	
Famotidine IV-PO conversion	+	
Thrombocytopenia/Med review	+	
Platelet drop while on heparin	+	+
K < 3.0, K > 6.0	+	+
Is K still < 3.0!	+	
On digoxin, ?serum K	+	+
On digoxin, K < 3.3	+	
High K while on supplemental K	+	
K-sparing diuretic/high serum K	+	
Possible warfarin Interaction	+	
Guideline for amphotericin	+	
IV-PO antibiotic conversion	+	
Bactrim and granulocytopenia	+	
Fractional excretion of sodium	+	
Narcotic-MAOI interaction	+	
Potential additive toxicity anticholinergics	+	
Constipating Rx/No bowel med		

Although we solicited feedback (by e-mail), we received only one message in 5 weeks. Therefore, we hosted a lunch at which 6/7 intern volunteers attended. At that time, CLEM was sending a total of 50-60 alerts per week. At the luncheon, we asked for feedback on event monitoring in general, specific alerts, and suggestions for new alerts. The majority of the interns were neither enthusiastic nor critical of event monitoring; one said that the alerts were informative and another useful. They requested all positive cultures and drug levels be sent by e-mail. No suggestions to discontinue specific alerts were received.

We then issued SkyTel® 2-way pagers to 6 interns. This pager has a 20-character by 4-line alphanumeric display and can receive pages of up to 512 bytes in size. The SkyTel® pager can transmit responses to these messages back to CLEM. We expected 2-way pagers to (1) decrease the time-delay inherent in e-mail delivery, and (2) to enable CLEM to have a dialog with the interns. This dialog would support a variety of key features such as fail-safe delivery (acknowledgment of receipt), collection of feedback about the value of alerts, and pushing of journal articles and medical guidelines to network printers.

The initial feedback from interns about 2-way paging was that they liked receiving critical lab values, positive cultures, and reports of levels of therapeutic drugs via 2-way pagers because it increased their work efficiency (they did not have to log into the hospital information system periodically to see if the results were available yet).

In June, 1997, we proposed to the leadership of the Department of Medicine that event monitoring be extended to all of general medicine with the incoming group of interns. After obtaining their approval, and additional approval from the Chief Resident's office, we asked for and received time during the intern-orientation program to brief the interns on CLEM.

Currently, CLEM is sending alerts to 35 housestaff by e-mail and SkyTel® 2-way pager. These housestaff provide care to all patients on the 8 general medical services, the intermediate care service, the medical ICU and the CCU. Our experience with 2-way paging in this cohort is that the interns acknowledge alerts promptly (using the reply capability of the 2-way pager).

DISCUSSION

Event detection

Previous published reports of event detection did not identify the problems we encountered with event detection when working with legacy systems. The explanation may be that previous work has been done in monolithic systems, or in heterogeneous systems whose data models happened to correspond to the event models used by domain experts. Since legacy systems with mismatched data models will be commonplace for the foreseeable future, a separate event detector will be necessary in many environments.

Asynchronous decision support using e-mail

Roughly five institutions have experience using e-mail directly to clinicians as a communication channel for decision support. The state of knowledge about efficacy is limited to a demonstration in 1991 of acceptability and efficacy when used to alert housestaff and attendings at Beth Israel (9).

Our interns almost all read e-mail daily and they did not mind getting alerts this way. However, they tended to read e-mail in the morning or the evening at home; thus, alerts that came in during the day were not seen promptly. A few expressed a preference for alternatives channel of communication for certain alerts. For example, they expressed a preference to be paged by the pharmacist for new medication orders whose dosing did not match guidelines. They were accustomed to this preexisting approach AND the approach had the added advantage that they could rectify the problem by giving a verbal order to the pharmacist.

The principles that govern when to use e-mail for decision support are not well understood. E-mail seems

appropriate when alerts involve a non-life-threatening event (e.g., suggested change from IV famotidine to oral nizatidine or avoidance of unnecessary drug level monitoring), but not when the physician needs to be informed as soon as possible (e.g., excessive dosing of antibiotics, severe electrolyte imbalances, drug interactions and contraindications).

Asynchronous decision support using 2-way pagers

To our knowledge, there is no previous published work on the use of 2-way pagers in alerting. Our experience suggests that 2-way paging is acceptable to interns for certain kinds of information: data that is rate-limiting in the care process or that may significantly change the care of patients (e.g., culture and radiology reports). Further work is need to (1) define additional types of information that should be sent via this communication channel, and (2) to determine whether 2-way paging reduces delays in the correction of critical lab values, or potential errors in patient care.

Redundant alerting systems

It is known that event monitors and clinical laboratories alert clinicians redundantly about critical lab values (10). We found a variation of this type of interaction; between CLEM and clinical pharmacists. At UPMC, pharmacists screen drug orders based on guidelines, some of which were also implemented in CLEM. Since orders are paper-based, and pharmacists screen then *before* inputting them into the pharmacy computer system (which feeds the warehouse), pharmacists are in a position to apply guidelines and contact ordering physicians to discuss orders *before* CLEM sees the orders. The implication of this for CLEM, or any clinical event monitor, is that if a challenged order is deemed valid (e.g., there are clinical indications for it unknown to the pharmacist or not covered in the guideline), then the order will be accepted by the pharmacist, input into the pharmacy system. Subsequently, CLEM will receive the order, apply guidelines, and generate an alert which would be a false-alarm.

This pharmacist-CLEM interaction is an instance of a general phenomenon—serial alerting-system interactions. Such setups increase the rate of false alarms of the downstream system. The previously recognized laboratory-event-monitor interaction represents a second class of interactions—parallel alerting-system interactions. Parallel interactions have unpredictable effects because the reaction of clinicians depends on which system alerts first and other factors (e.g., acceptability of the form of the alert).

The frequency of both types of interactions are likely to increase as more groups within health-care systems institute programs to improve quality of care. One way to prevent such interactions (which can annoy clinicians and impair the efficacy of the interventions themselves),

is to create an organization that coordinates decision-support activities across an enterprise..

Future directions

We are working on several extensions to the current system. Our efforts to improve management of hyperlipidemia and other chronic diseases by processing inter-visit events are described in (12). We will soon use CLEM to push journal articles to physicians in a patient-specific manner. We also plan to extend this feature to support a curriculum-driven educational program that ties didactic material to individual patient cases.

ACKNOWLEDGMENTS

This work was supported by grants 5 T15 LM07059-10 and 1 R29 LM06233-01A1 from the National Library of Medicine and by SkyTel®. We thank Charles P. Friedman for helpful discussions.

REFERENCES

1. Hripsak G, et al. Design of a clinical event monitor. *Comp Biomed Res* 1996;29:194-221.
2. Glaser J, Teich J, Kuperman G. Impact of information events on medical care. *HIMSS*, 1996.

3. Yount R, Vries J, Council C. The Medical Archival System: Information retrieval system based on distributed parallel processing. *Inf Proc Manag* 1991;27(4).
4. Pankaskie M, Wagner M. Use of CLIPS for representation and inference in a clinical event monitor. *JAMIA-Symposium Supplement* 1997.
5. Hiltz F, Teich JM. Coverage List: A provider-patient database supporting advanced hospital information system services. *Proc 18th Ann SCAMC* 1994:809-13.
6. Sittig DF, Stead WW. Computer-based physician order entry: the state of the art. *JAMIA* 1994;1(2):108-23.
7. Massaro TA. Introducing physician order entry at a major academic medical center: I. Impact on organizational culture and behavior. *Acad Med* 1993;68(1):20-24.
8. <http://www.cpmc.columbia.edu/arden/listmlm.cgi>.
9. Rind DM, et al. Effect of computer-based alerts on the treatment and outcomes of hospitalized patients. *Arch Int Med* 1994;154:1511-7.
10. Kuperman G, et al. Detecting alerts, notifying the physician, and offering action items: A comprehensive alerting system. *JAMIA-supplement* 1996:704-9.
11. McDonald C. *Action-Oriented Decisions in Ambulatory Medicine*. Year Book Medical Publishers, Inc., 1981.
12. Tsui FC, Wagner M, Thompson M. Implementing NCEP guidelines in a web-based disease-management system. *JAMIA-Symposium Supplement* 1997.

APPENDIX

Date: Tue, 11 Mar 97 21:50:07 -0500 (EST)
From: CLEM <clem@thor.cbmi.upmc.edu>
To: XCOVER_INTERN@med.pitt.edu
Subject: ALERT[39]: Vancomycin monitoring

DOE JOHN A mrn:123-45-6789 location:10S rm:S1043.
CLEM sent this alert to you because you were covering for Dr. PRIMARY INTERN.

This patient was recently started on VANCOMYCIN

Pharmacy Services guidelines do not recommend routine vancomycin blood level monitoring UNLESS the patient is on dialysis, has compromised renal function, or will be receiving vancomycin for > 10 days. In these cases, trough vancomycin levels (10-15 mcg/mL) should be used to monitor for drug accumulation. Additional information on vancomycin dosing and monitoring is available at:

<http://www.cbmi.upmc/dsr/vancomycin.html>

-or-

CLEM will send you a copy of this information by email if you reply to this alert with 'GUIDELINE' as the message

----- About alerts ... -----

This alert was generated by a computer program using laboratory, pharmacy, and coverage data that are generally accurate but not perfect. Therefore, you should not construe this alert to constitute definitive medical advice or use it to replace or overrule your clinical judgment or diagnosis. If you have questions or suggestions, please reply to the sending address, or page Dr. Wagner (2788).

(A)

clem@cbmi.upmc.edu
DC TELEMTRY, MI NEG
DOE JANE
123-45-6789 rm:E1018

SERIAL CPK RESULTS:
76.0 05/15/97 10:41
138.0 05/15/97 20:00
126.0 05/16/97 04:15

Based on these results if MI has been ruled out, and patient is on...

PLEASE REPLY TO THIS MSG:
* HEADED
* IGNORED

(B)

Fig. A1. E-mail and pager alerts. (A) This alert was sent via e-mail to the cross-covering intern with a cc: to the primary intern. The alert is based on a Pharmacy Service guideline. The intern can obtain the full-text of the guideline by replying to the alert, or on a WWW site. (B) This alert was sent via pager. The four rectangles represent sequential screens of information that the intern toggles through. The last screen elicits feedback from the intern about whether the alert influenced the management of the patient.