

IBM Research Report

Partitioning Knowledge Bases between Advanced Notification and Clinical Decision Support Systems

**Yves Lussier*, Rose Williams, Srikant Jalan, Tara Borlawsky*,
Jianrong Li*, Edie Stern**

IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598

*Departments of Biomedical Informatics and of Medicine
Columbia University
New York, NY



Research Division

Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

Partitioning Knowledge Bases Between Advanced Notification and Clinical Decision Support Systems

Yves A. Lussier MD^{1,3}, Rose Williams^{2,3}, Srikant Jalan², Tara Borlawsky¹, Jianrong Li¹, Edie Stern²

1. Departments of Biomedical Informatics and of Medicine, Columbia University, New York, NY

2. IBM Thomas J. Watson Research Center, Hawthorne, NY

3- Corresponding Authors

Abstract

Ephemeral medical administrative knowledge is substantially different from comparatively enduring clinical knowledge. The functional partition of such administrative knowledge in dedicated decision support systems (DSS), separate from clinical knowledge, provides many advantages. Classically, however, clinical event monitors and the syntax for their logic modules include the coding of notifications (e.g. Arden Syntax). The following paper will describe the rationale for segregating policies, user preferences and clinical monitoring rules. It will detail the implementation of knowledge bases partitioned between an advanced notification DSS and a clinical DSS, which together form a novel and complex clinical event communication service. In this communication service, notification rules are abstracted from care-provider roles, hospital policies are abstracted from alerting mechanisms, and user-defined preferences determine which devices are to be used for receiving alert notifications. Our prototype service operates from a clinical event monitor through a secure notification service proxy to the BlackBerry™ Personal Digital Assistant (PDA)/phone device over the Nextel cellular network. Our design differs from previous notification systems in that it integrates an advanced and versatile notification platform supporting a wide range of mobile devices with a sophisticated XML/HL-7 compliant communication protocol. This protocol includes advanced notification attributes, detail and context in managing provider roles, schedules, institutional policies, personal preferences and acknowledgments.

Introduction

Clinical event monitors are decision support systems that generate alert messages based on events of possible clinical significance and related patient data. They can improve health care quality and reduce costs by supplying personnel with the necessary information at desirable times via preferred routes [2]. Automated wireless alerts have been used to prevent medical errors and reduce the time until appropriate treatments are ordered by immediately alerting clinicians to critical laboratory result values [3], allowing them to intervene more quickly than when the information is delivered using more traditional means [4]. The choice of communication channel, such as telephone, alphanumeric pager, or, most recently, personal digital assistants (PDAs), depends on the urgency of the message, availability of the responsible caregiver and knowledge of the appropriate message recipients [2]. While mobile technologies afford the clinical

setting many advantages for real-time information about a patient's condition remotely, they also add overhead costs in application development, deployment, and technical support [10]. One approach is to extend legacy clinical event monitors by including a notification component using current standards (e.g. Arden Syntax). However, there are significant challenges in minimizing any adverse affects on the organization while the notification component of the clinical DSS application evolves. Additionally, the frequently changing notification protocols and available mobile devices would necessitate a constant evolution of software upgrades to address the emerging technologies.

Furthermore, as many professionals today use an array of devices for mobile communications, what is needed is a configurable technology to address user preferences for delivering alerts. For the clinical setting, this problem is compounded in that physicians rotate ward and patient responsibilities, so that user role and context per patient must also be resolved.

Using middleware technology, one can preserve the existing clinical DSS system while extending its services to include full-featured notification functions. This paper describes the function and implementation of a loosely coupled clinical decision support service (developed by Columbia University Medical Center) with an intelligent notification service (developed by IBM Research) using XML/HL7 as a communication protocol between the two systems.

Benefit of Partitioning Knowledge Bases

The portability of handheld technologies is one of their biggest advantages in a clinical setting. Yet, extending legacy event monitors by increasing the complexity of the notification component of clinical knowledge bases using current standards, such as the Arden Syntax and Guideline Interchange Format (GLIF), significantly increases the total cost of operations. The Arden Syntax for Medical Logic Modules (MLMs) is a language designed for writing and sharing task-specific health knowledge. By easing the sharing of knowledge bases, the Arden Syntax increases the use of knowledge-based systems in the realm of healthcare by enhancing knowledge sharing abilities of independent systems while reducing maintenance issues and costs [16]. GLIF, a common representation for institutional and national standard practice guidelines, facilitates guideline sharing across different healthcare institutions [17]. Combined, the MLMs found in the Arden Syntax and the syntax of GLIF already partially support the codification of knowledge pertaining to clinical alerts. However, the additional attributes of personal preferences and institutional/departmental policies for patient information communicated across varied devices greatly complicates the maintenance of the notification component of event monitors.

Further, as experienced in the last year with the changes imposed by the Health Insurance Portability and Accountability Act (HIPAA) regulations, medico-administrative policies change more frequently than clinical rules. This adds to the cost of maintaining clinical knowledge bases. For HIPAA compliance, it is important that institutions implement corporate policies for mobile devices regarding password

protection, encryption, authenticated synchronization, and physical security of the devices [9]. Finally, securing the integrity of data delivered over mobile channels complicates the event monitor maintenance even further, and requires a completely different implementation skill set.

Overall, there are many advantages to partitioning the clinical knowledge from the administrative, and the broadcasting of that clinical knowledge to mobile devices. By partitioning the knowledge base, it is operationally easier to accommodate the different skill sets, the different rates of change, and to ensure that the more frequent changes associated with policies, and the delivery of individual alerts do not affect critical clinical knowledge systems. With more frequent change comes more frequent potential for user error. By partitioning the knowledge base, and by using a dedicated DSS, the potential extent of damage from such error is limited.

One method for partitioning knowledge is the “publish and subscribe” communication model. Publish-subscribe methodology enables decoupled and asynchronous messaging, allowing systems to exchange data between independent applications in an event-driven manner. A notification system receives a message from a “publisher” and, through the application of a notification algorithm, attempts to send the message to a “subscriber.” Vanderbilt University Medical Center has utilized this methodology of separating the creation and delivery of messages in the implementation of a notification system for new physicians’ orders [15].

Our system not only uses a publish-subscribe model for communication, but also provides an additional level of abstraction that enables the identification of “subscriber” based on the provider-role for the patient at the time of messaging event. This function is in addition to the run-time feature of policy access and review to ensure alert protocol compliance, as well as to address mobile communication fault-tolerance mechanics. All features are separated from the clinical knowledge base systems.

Clinical Knowledge Base

Large scale automated monitoring of a considerable number of distinct clinical conditions is complex, and to our knowledge, no sizable projects offer user-customizable communication modalities over all monitored conditions. As both the numbers of clinical conditions and communication modalities increase, the complexity of customizing user preferences amplifies following a geometric progression [13], and it becomes more difficult for an information system to predict whether a laboratory result merits urgent notification to a particular user [14]. Current implementations of composite DSSs systematically use some form of notification attributes in communication protocols to exchange knowledge among the knowledge bases. A DSS containing clinical knowledge needs to be augmented with notification attributes and then communicated to the notification system in order for medico-administrative knowledge to be useful and independent of each specific clinical alert instantiation.

The Arden Syntax allows for encoding urgency, but this was shown to be insufficient and inadequately scalable for decoupling clinical decisions from decisions in a Criteria-Based Notification System [13]. However, notification attributes that describe the urgency and severity of a patient lab result or other clinical data simplify the management of communication modalities (CM) and allow for the separation of clinical and medico-administrative knowledge. This is particularly important for institutions where the manual assignment of a terminal device for every alert is a foreboding task. The following notification attributes (NA) were developed and incorporated into extensions of GLIF and the Arden Syntax: Urgency, Severity, Risk, Evidence and Positive Predictive Value. Urgency refers to the time interval beginning when a message is sent to the time when an injury or adverse event can affect the patient. In the absence of further intervention, the maximum degree of potential injuries to the patient is reflected by the Severity attribute. The Risk NA represents the probability of the occurrence of an adverse event or injury when intervention is absent. Evidence measures the strength of the evidence supporting a rule, and the Positive Predictive Value NA measures the ability of the event monitor to produce a correct message [13]. The methods by which these NAs were developed, using experts opinions and reviewing current practices, their effectiveness for encoding existing clinical conditions and an analysis of two archetypal coding schemes, the Guideline Interchange Format (GLIF) and Arden Syntax, have been previously addressed by our group [2, 13, 16, 17].

The five extended notification attributes provide the clinically relevant information necessary to compute an appropriate target and delivery mechanism for the notification. They afford an institution the advantage of standard interfaces and generic best practice rules, while facilitating institutional-level customization and configuration. These attributes increase the reusability and sharability of institutional event monitor rules across institutions by replacing institution-specific components with generic ones. The message specifications create a standard interface for event monitors to communicate with the message management module. Secondly, standardized message specifications facilitate institutional-level customization and control of event notification by providing the mechanisms to assert these policies. The assignment of alerting mechanisms becomes independent of clinical conditions (CC), removing institution-specific devices and technologies from the knowledge bases of the event monitors and inserting CC-specific and institution-independent NAs. Third, standardized messaging facilitates both institution-level management of policies and simplifies the user-level preference configuration [13], which supports the publish-subscribe communication model used in event notification systems [15]. Events are assigned to combinations of notification attributes rather than to each instance of clinical alerts.

Notification Knowledge Base

User Preferences:

Our design recognizes that allowing mobile users to decide *how* they should be notified is part of an efficient communication solution. As such, we have a notification knowledge base that separates user subscription information, device preferences, and system configuration data from the clinical alert. Our notification component, filtering incoming content, looks for important keywords and compares the filtered data with the user's criteria as defined in the user's subscription. If a match is found, our component checks the user's device preferences and related information before sending the message to the mobile device. The notification knowledge base includes the user specification of many types of preferences and subscription settings. Notification knowledge also includes delivery channel information such as device name, host, port, and time periods when the user does not want messages delivered. Notification knowledge also includes message rules, which allow the delivery of messages per device based on priority levels. For example, an urgent message may get delivered to the user's pager, while a normal message may be delivered to the user's PDA.

Roles and Policies:

Our design also recognizes that in clinical settings, physician schedules rotate responsibility for patient care. A physician ordering a laboratory test and subscribing to the results may no longer be responsible for the patient's care when the results become available. Therefore, our design supports not only how a user is notified, but also who is notified based on the provider's role at the time of message delivery. Thus, our notification knowledge base uses roles to determine the intended recipients of alerts. A role refers to the function or responsibility assumed by a person in the context of a healthcare event. Role information documents a person's association with an identified healthcare activity. Examples include Primary Care Provider, Resident, Intern, Hospitalist, etc. The characteristics of a role are as follows:

- A role association can be based on the patient, task or certain observation values.
- A role can be performed by multiple people.
- A person can be performing multiple roles.
- A role is specified for a specific period of time that is predetermined based on a schedule.
- A notification device can be associated with a role and the person carrying the device implicitly assumes the role.
- A role can be pre-specified, or it can be specified as a part of each notification event.
- A role can be delegated to another person for a specified period.

In our prototype system, we use role information based on the schedule of users. At this time, this information is entered separately and is not integrated with the hospital scheduling system. Future work could obtain this information from the different hospital scheduling systems. Currently the role

information resolves to a single person performing the role at a specified time. Certain role information such as the “Attending Physician” for a patient or the “Ordering Physician” who requested the particular test to be performed is derived from the clinical event data, which has been sent to the notification system as a part of the alert content. Once the alert is sent to the device, our system monitors for user acknowledgement.

We have also implemented the delivery of the notifications based on hospital policies. The characteristics of the policy definitions are as follows:

- Policies define the workflow of the notification events.
- Based on timeouts of positive acknowledgement from the initial recipient, the notification can be cascaded to other recipients using different notification devices.
- The recipient can accept or decline the notification request.
- There is a Catch-All recipient, for notifications that cannot be routed using the pre-defined policies, to ensure that no notification ever gets lost due to errors in the policy definitions. This also ensures that the notification can be delivered to a responsible person in case of no positive acknowledgement from others.
- We must ensure that there are no cyclical loops in the policy definitions.

Fault tolerance for network- device communications and message delivery:

In a hospital environment, loss of connectivity to a device is very common. There are network connectivity dead zones within the building, as well as critical care units that prohibit the activation of mobile devices. As such, our notification knowledge base encompasses hospital policies for cascades clinical alerts in case of messages not being delivered due to network outages or device being turned off.. For example, if a user does not respond to an alert within 10 minutes, we retry the message to the same user. If the message has still not been responded to within another 5 minutes, we cascade the message to the next user in the following order:

Intern (Urgent/BlackBerry) → Resident (Urgent/BlackBerry) → Hospitalist (Urgent/BlackBerry) → Hospitalist (FYI/EMail).

If any of the roles cannot be resolved then the message is sent to the specified Intelligent Notification System (INS) user, hospitalist1@ins.realm, as an FYI message using Email.

There is increasing evidence that new communication modalities (e.g., 2-way pagers, wireless telephones, etc.), when linked with a user’s preference configuration, can improve clinicians’ efficiency and efficacy. Flexible and user-configurable clinical notification systems are currently under development, but these systems are limited in their capacity to maintain complex communication modalities with a large number of rules. Few institutions have implemented large scale automated monitoring of a considerable number of distinct clinical conditions, and to our knowledge, none of these

sizable projects also offer user-customizable communication modalities, institutional policies management and advanced notification, and flexible mechanisms for “fault tolerance”.

Communication Protocol between the Clinical and Notification Decision Support systems

The communication protocol between the clinical event monitoring system and the intelligent notification system follows the standard as defined by Health Level 7 (HL7). The data is sent as the HL7 event message type PPR_PC1 (i.e., Patient Problem Record Protocol). We have extended the HL7 standards using some additional attributes to support the extended NAs generated by the clinical DSS.

The PPR_PC1 protocol is divided into multiple segments. The MSH (Message Header) segment includes the date and the time of the message and defines the intent, source, destination and syntax specifics of a message. The PID (Patient Identification) segment contains permanent patient identifying and demographic information that, for the most part, is not likely to change frequently. The PV1 (Patient Visit) segment is used by Registration/Patient Administration applications to communicate information on a visit-specific basis, including the patient’s assigned location. The PRB (Problem Detail) segment contains the data necessary to add, update, correct, and delete the problems of a given individual. The ROL (Role) Segment contains the data necessary to add, update, correct, and delete from the record persons involved, as well as their functional involvement with the activity being transmitted. The OBX (Observation/Result) segment is used to transmit a single observation or observation fragment. There can be multiple ROL segments identifying the different personnel associated with the current task, such as the “Ordering Physician,” “Attending Physician,” etc. In addition, there can be multiple OBX segments that identify different observation values. Because the HL7 Notification Detail Standard (NDS) segment only defines the notification message severity, we also include additional NAs. The segment is extended to provide further information, such as the “Notification Alert Urgency,” “Notification Alert Risk,” “Notification Alert Evidence,” and the “Notification Alert False Positive Rate.” These additional attributes along with the information present in the ROL segment help in determining the appropriate physician who should be notified.

The following three tables provide samples to demonstrate the protocol we developed: a sample HL7 message from Vigilens (in the original HL7 format) (Table 1), the equivalent XML representation of the HL7 message (Table 2), and the extended XML segment of the HL7 Notification Detail Standard (Table 3).

Table 1: Sample HL7 Patient Problem Record Protocol message (HL7 PPR_PC1)

Table 2: Sample of the equivalent HL7 Patient Problem Record Protocol message (HL7 PPR_PC1)

Table 3: Extended XML segment of the HL7 Notification Detail Standard (NDS)

Knowledge about all aspects of the message are contained within the Notification knowledge base, including the acknowledgment of the receipt of the message, the actual user who is notified, the responses from the users, and any cascading of notification that has taken place. The acknowledgement is sent back to the clinical system for logging using the HL7 standard ACK_NO2 message. The MSA (Message) segment contains the details of the acknowledgement as illustrated in Table 4.

Table 4. Sample XML acknowledgement message using the HL7 standard ACK_NO2

Design Implications for disparate Knowledge Management

The previous event monitor of our institution (Columbia Presbyterian Medical Center), which contained hundreds of different MLMs, was decommissioned in year 1999 and allowed for only two notification modalities: a patient-oriented secure web portal and an email. Assignment of each email recipient was written directly in the clinical knowledge base in Arden Syntax, which would inevitably have been time consuming for large user communities with varying roles. We have since designed its successor system, “Vigilens”, which is a server-based DSS that allows for secure telemonitoring of clinical repositories and secure notifications over the internet. Its integration with the Notification System (IBM Websphere Everywhere Suite) is illustrated in Figure 1. Besides decoupling the clinical knowledge bases from the medico-administrative ones, we were also motivated to deploy over a larger variety of pervasive communication devices and to support diverse communication protocols. For the purpose of evaluating the feasibility of the proposed loosely coupled systems, we have implemented, in the test server, forty types of clinical alerts on Vigilens with two types of notification attributes (urgency and severity) and three states (high, low, unknown), leading to six attribute-state combinations. Each policy and user preference can be assigned to one of these six combined states of NAs, which makes them clinically relevant without having to individually deal with each current clinical alert or with future additions. Traditional clinical DSSs either neglect the issues of modalities of communications, policies or user preferences, or address only a small number of combinations because of difficulties with affordably scaling up these aspects and the clinical rules. Indeed, each of these combinations would otherwise have to be implemented in the system.

Figure 1: Clinical event monitoring system (Vigilens) loosely integrated, via secure HL7 protocol over the internet, with an advanced notification system (Intelligent Notification)

A Notification Protocol for Comprehensive Decision Making with a PDA

In our prototype, alerts are displayed on a PDA device (BlackBerry™ 7510) as a message list in color-coded HTML. This specific device was originally selected because its server could be implemented behind the hospital’s firewall and could generate a true end-to-end secure communication between the

device and the event monitor. To our knowledge, no other end-to-end push secure protocol is available for PDA-type devices (alternative push protocols basically encrypt messages to the carrier, where the message is decrypted and re-encrypted in another communication protocol). Most implementations for PDA communications push an unsecured URL and then pull secure communications. These former implementations do not optimize the efficient secure display of laboratory data, as the user is bound to “pull” all secure results. In addition, the notification protocol has been developed to maximize the immediate decision making process by the clinician. Thus, the protocol includes the usual critical alert (patient name, medical record number, critical value) as well as additional contextual clinical and medico-administrative information. The first panel displays a list of alerts, one per line. As the physician recipient selects an alert on the first panel, the screen prompts the details of the specific message. Message details include patient name, medical record number, date of birth, room location, the urgency and severity of the alert, contact information (Figure 2) and additional laboratory results related to the decision making process (Figure 3). The physician can scroll down to view additional data, accept the receipt of the alert, or provide additional comments and feedback about the alert. In this same screen, the clinician can also view other associated laboratory values from the test panel. The Notification History page (Figure 4) displays the list of users to whom the notification was sent (due to multiple recipients at the same time or due to cascading policies). The action taken by each recipient and the comments by each physician are also displayed on this screen.

Figure 2. Display of a critical alert showing the clinical notification attributes.

Figure 3. Display of Laboratory Panel Associated to a Critical Alert Message.

Figure 4. The Notification History page (figure 4) displays the list of users to whom the notification was sent.

Discussion

Decoupling the recipient assignment from the clinical knowledge base was deemed essential to reduce the complexity of the maintenance of clinical knowledge in the context of increasingly varying medico-administrative policies subsequent to the HIPAA regulations. Indeed, the escalating potential changes in the former clinical knowledgebase were directly related to the product of an increasing number of dimensions related to assigning the correct recipients for an alert (show in equation 1 below with an example). In contrast, the proposed model does not increase geometrically with every dimension, but rather linearly (additive rather than multiplicative) as shown in equation 2. Since the number of recipients (IoR = thousands in our hospital) and therefore of possible user preferences (Pr) are vastly larger than the number of notification attributes (NA=2 in this example), of communication modalities (CM = PDA,

paggers) or of roles (R) in an institution, the term NP in equation 1 is far greater than the term NP in equation 2a).

Equation 1. Complexity Notification in a Common Clinical and Administrative Knowledge Base

$$NP = CL * Po * Pr * IoR * R * CM$$

Where:

NP = Number of parameters to be assigned in a “ common” logical rule

CL = Clinical logic (K>7 meq/l)

Po = Policies

Pr= User preferences

R = Number of Roles

CM= Communication modalities (e.g. PDA, Secure Portal, Pager, Secure email, etc.)

IoR = instance of recipient (physician “ JD” , nurse “ EL” , etc)

Equation 2. Simplicity of Notification in Decoupled Clinical and Administrative Knowledge Bases

$$2a) NP = NP_C + NP_{Po} + NP_{Pr} = NA * (CL + Po * \{R * CM\} + Pr * \{R * CM * IoR\})$$

$$2b) NP_C = NA * CL$$

$$2c) NP_{Po} = NA * Po * R * CM$$

$$2d) NP_{Pr} = NA * Pr * R * CM * IoR$$

Where:

NP_C = Number of parameters to be assigned in the clinical knowledge base

NP_{Po} = Number of parameters to be assigned in the policy knowledge base

NP_{Pr} = Number of parameters to be assigned in the a preference knowledge base

NA = number of notification attributes

One limitation of this study is that only a subset of the previously published notification attributes has been implemented (“severity” and “urgency”). As the quantity of clinical and medico- administrative rules increase, it is likely that the implementation of additional notification attributes is required to increase the value, particularly the “clinical evidence” in support of the alert generated. This is particularly useful for “ calculated” events such as drug-drug interactions, where the evidence in the literature varies from anecdotal to well established. Additionally, implementation of additional modalities of communications would allow for studying the value of the implementation for managing user preference choices; currently our experience has focused on implementation of policies and clinical alerts.

We have described notification attributes to increase the scalability, development, and maintenance of decoupled and distributed clinical, and notification knowledge bases. We have demonstrated an implementation that incorporates roles, a publish-subscribe model and a HIPAA compliant notification mechanism involving the development of a communication protocol between a clinical decision support system and a notification DSS. We have then examined in a research environment the feasibility and functionality of the implementation. Additionally, we have also provided a metric to assess the value of the proposed methods in other environments. In summary, we have demonstrated in this proof-of-concept study that clinical and notification rules can be decoupled via notification attributes, honoring stringent HIPAA-compliant security criteria, and that this decoupling enables an increase in flexibility of notification with a potential for significantly reduced management cost of notification rules such as policies and user preferences.

Acknowledgements

The team acknowledges the partial funding from the Columbia Center for Advanced Information Management (NYSTAR), as well as the managerial contribution of Guruduth Banavar and Lorraine Herger.

References:

1. Chen ES, Mendonca EA, McKnight LK, Stetson PD, Lei J, Cimino JJ. PalmCIS: A Wireless Handheld Application for Satisfying Clinician Information Needs. *J Am Med Inform Assoc.*; 11(1): 19-28 Jan-Feb 2004.
2. Hripsack G, Clayton PD, Jenders RA, Cimino JJ, Johnson SB. Design of a Clinical Event Monitor. *Comp Biomed Res.* 1996; 29: 194-221.
3. Carroll AE, Christakis DA. Pediatricians' Use of and Attitudes about Personal Digital Assistants. *Pediatrics.* 2004 Feb;113(2):238-42.
4. Warner HR Jr, Miller S, Jennings K, Lundsgaarde H, Pincetl P, Robinson EN Jr, Sommers J, Childress C. Clinical event management using push technology--implementation and evaluation at two health care centers. *Proc AMIA Symp.* 1998;:106-10.
5. Eisenstadt SA, Wagner MM, Hogan WR, Pankaskie MC, Tsui F-C, Wilbright W. Mobile Workers in Healthcare and Their Information Needs: Are 2-Way Pagers the Answer? *Proc AMIA Symp.* 1998;:135-9.
6. Fischer S, Stewart TE, Mehta S, Wax R, Lapinsky SE. Handheld computing in medicine. *J Am Med Inform Assoc.* 2003 Mar-Apr;10(2):139-49.
7. Rothschild JM, Lee TH, Bae T, Bates DW. Clinician Use of a Palmtop Drug Reference Guide. *JAMIA* 9(3): May/Jun 2002.
8. Al-Ubaydli M. Handheld computers. *BMJ.* 2004 May 15;328(7449):1181-4.
9. Tooley MJ, Mayo A. Handheld technologies in a clinical setting: state of the technology and resources. *AACN Clinical Issues.* 14(3):342-9, 2003 Aug.
10. Tschopp M, Lovis C, Geissbuhler A. Understanding Usage Patterns of Handheld Computers in Clinical Practice. *Proc AMIA Symp.* 2002:806-9.
11. Kafeza E, Chiu DK, Cheung SC, Kafeza M. Alerts in mobile healthcare applications: requirements and pilot study. *IEEE Trans Inf Technol Biomed.* 2004 Jun;8(2):173-81.

12. Wagner MM, Tsui FC, Pike J, Pike L. Design of a clinical notification system. Proc AMIA Symp. 1999:989-93.
13. Tao Y, Wang D, Shortliffe EH, Lussier YA. Extended attributes of event monitor systems for criteria-based notification modalities. Proc AMIA Symp. 2002:762-6.
14. Poon EG, Kuperman GJ, Fiskio J, Bates DW. Real-Time Notification of Laboratory Data Requested by Users through Alphanumeric Pagers. J Am Med Inform Assoc. 2002 May-Jun;9(3):217-22.
15. Geissbuhler A, Grande JF, Bates RA, Miller RA, Stead WW. Design of a General Clinical Notification System Based on Publish-Subscribe Paradigm. Proc AMIA Annu Fall Symp. 1997;:126-30.
16. Hripcsak G, Ludemann P, Pryor TA, Wigertz OB, Clayton PD. Rationale for the Arden Syntax. Comput Biomed Res. 1994 Aug;27(4):291-324. Review.

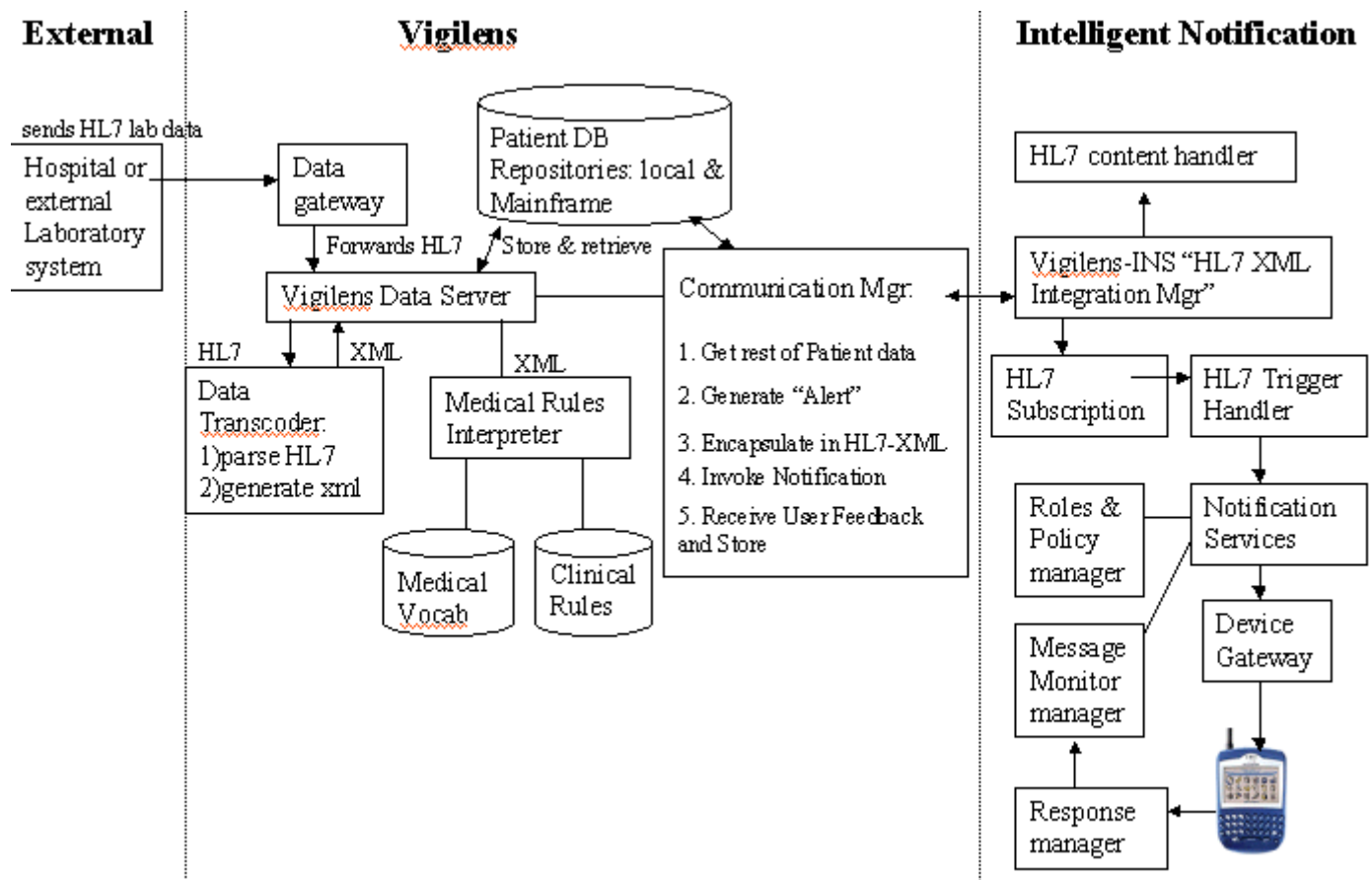


Figure 1: Clinical event monitoring system (Vigilens) loosely integrated , via secure HL7 protocol over the internet, with an advanced notification system (Intelligent Notification)

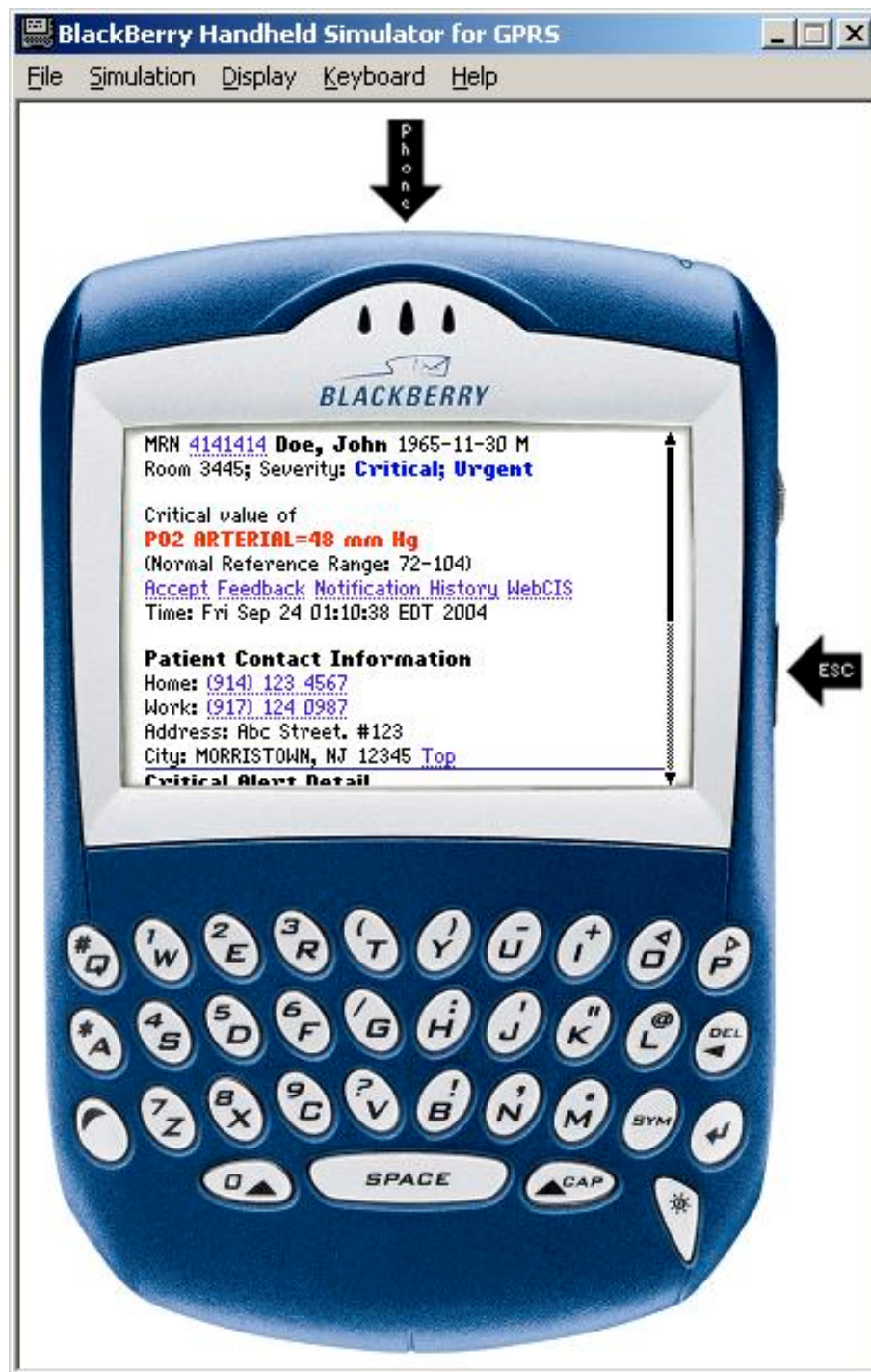


Figure 2. Display of a critical alert showing the clinical notification attributes (Critical, Urgent) on the second topmost row, the critical message and providing the contact information of the patient. Notifications criteria are used by the notification system to route the message according to roles, policies and user preferences to the right recipient(s) using the right modality(ies) of communications.

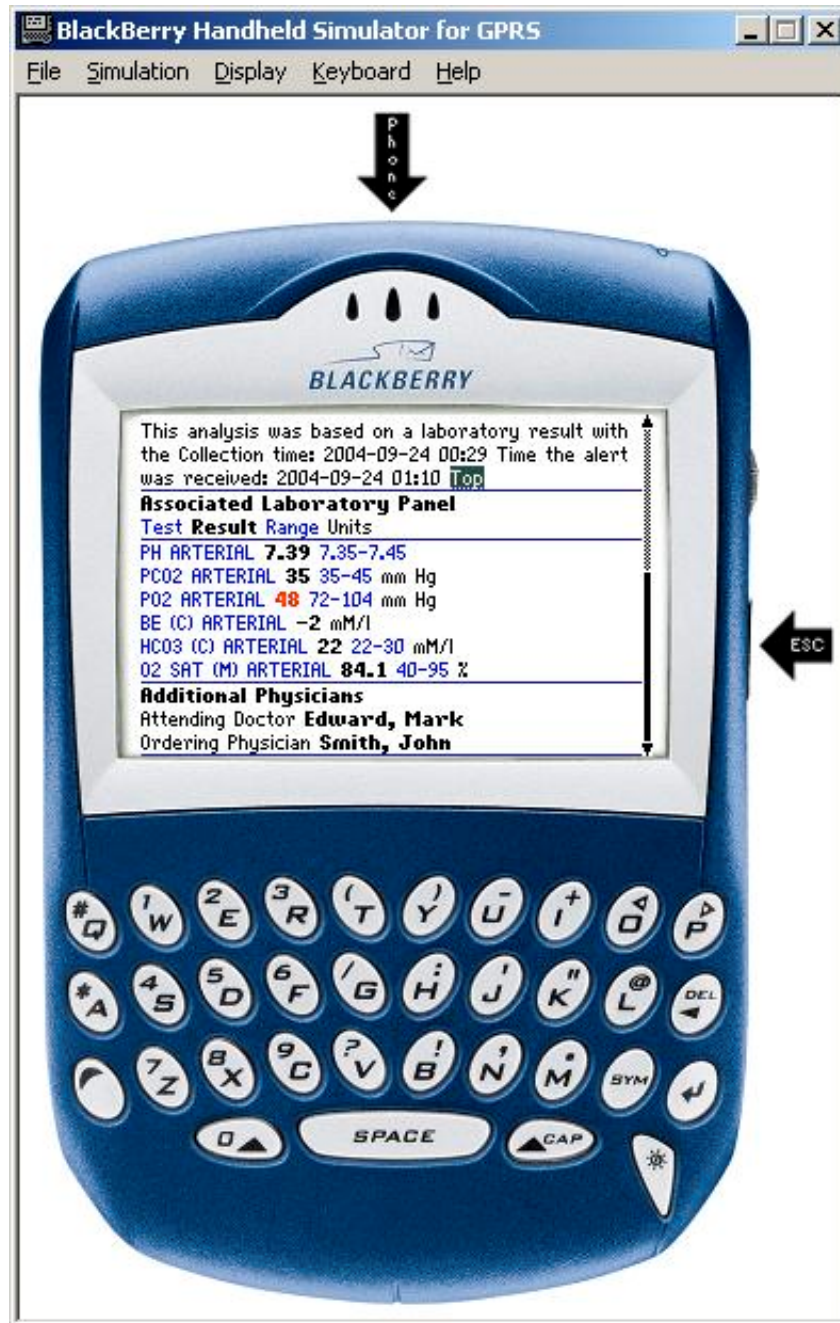


Figure 3. Display of Laboratory Panel Associated to a Critical Alert Message.

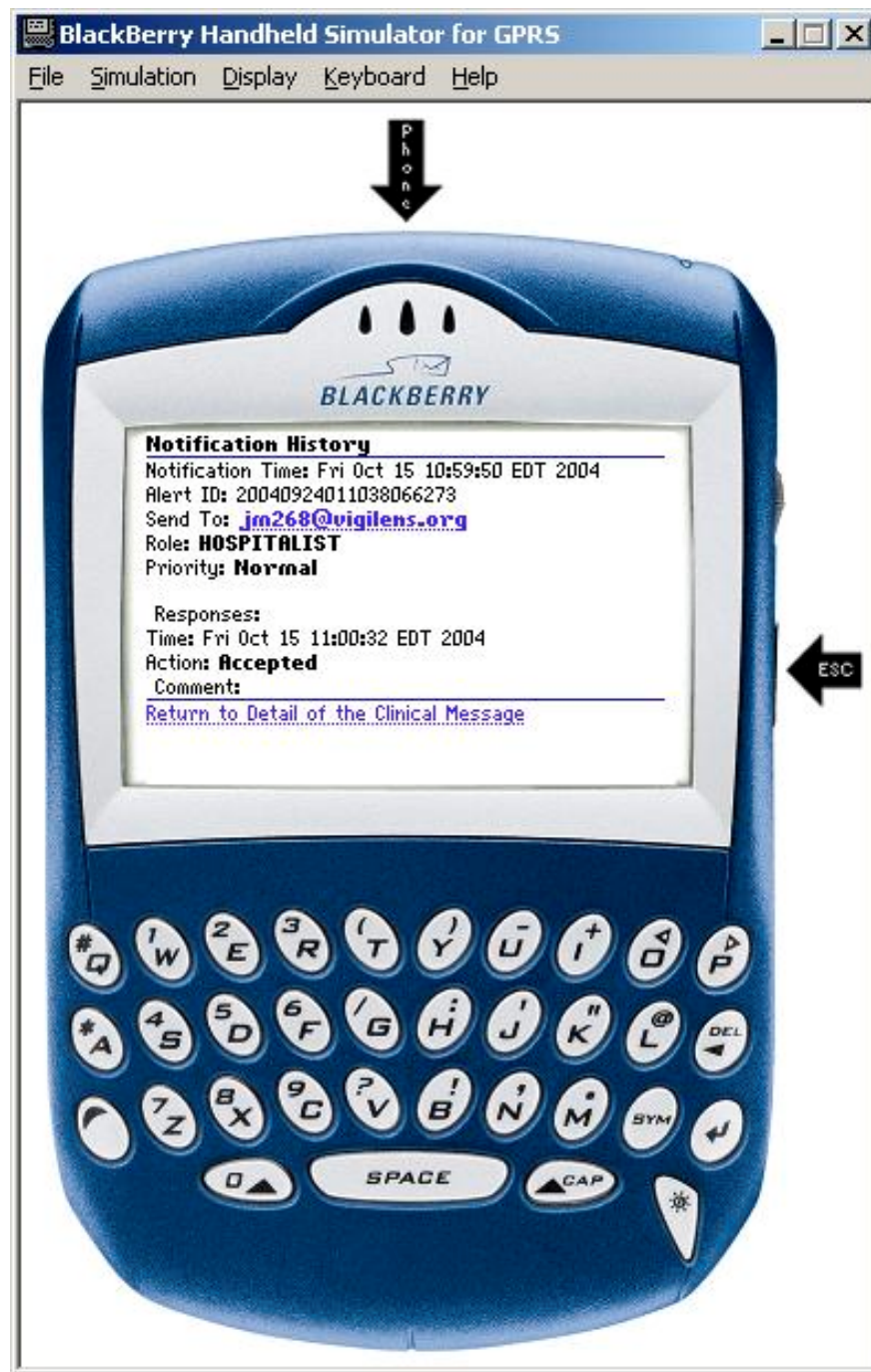


Figure 4. The Notification History page (figure 4) displays the list of users to whom the notification was sent.


```

MSH|^~\&|VIGILENS|TEST|IBM-
IMM|ER|20040527154359||PPR^PC1|20040527154359320|P|2.4
PID|||123456789^^^^MR^COLUMBIA UNIVERSITY
HOSPITAL||SMITH^JOHN^J^III^DR^PHD^L||19670515183020|M|||1234
EASY STREET^STE 1A^HAWTHORNE^NY^10532^USA^B||(914) 123-
4567|||S
PV1||E|GENERAL MEDICINE^101^A^GENERAL MEDICINE^^^MERCER
PAVILION^2^|E|||SJALAN@INS.REALM^ELLEN^ANDY
PRB|AD|20031107112830|112233445566^PANIC HCT^ARDEN MLM
LIST|||IP^INPATIENT^PROBLEMCLASSIFICATIONLIST
ROL|AD|^DAYTIME CLERK^ROLE MASTER LIST|JEDWARDS^EDWARDS^JOHN
ROL|AD|^SUBINTERN^ROLE MASTER LIST|MSMITH^SMITH^MARK
ROL|AD|^INTERN^ROLE MASTER LIST|BMILLER^MILLER^BILL
ROL|AD|^RESIDENT^ROLE MASTER LIST|MPARKER^PARKER^MARY
ROL|AD|^SENIOR RESIDENT^ROLE MASTER LIST|JWILLIAMS^WILLIAMS^JASON
ROL|AD|^ATTENDING PHYSICIAN^ROLE MASTER LIST|SJALAN@INS.REALM^ELLEN^ANDY
ROL|AD|^PCP^ROLE MASTER LIST|EJOHNSON^JOHNSON^ED
ROL|AD|^ORDERING PHYSICIAN^ROLE MASTER LIST|MPETERSON^PETERSON^MIKE
OBX|NM|2479^HCT|1|15|^%|35.4-44.4|||
OBX|NM|1564^CBC|2|27|^%|10.2-20.5|||
OBX|TX|^KNEE FRACTURE|3|MULTIPLE FRACTURE OF KNEE JOINTS|||
NDS|||2^MODERATE||1^60 SECONDS|1^90-100|2^PROBABLE|1^90-100

```

Table 1: Sample HL7 Patient Problem Record Protocol message (HL7 PPR_PC1).

```

<PPR_PC1 xmlns="urn:hl7-org:v2xml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:hl7-org:v2xml PPR_PC1.xsd">
  <MSH>
    <MSH.1 Item="1" LongName="Field Separator" Type ="ST">|</MSH.1>
    <MSH.2 Item="2" LongName="Encoding Characters" Type
    ="ST">^~\&amp;</MSH.2>
    <MSH.3 Item="3" LongName="Sending Application" Table="HL70361"
    Type ="HD">
      <HD.1 LongName="namespace ID" Table="HL70300" Type ="IS">CU-
    Vigilens</HD.1>
    </MSH.3>
    <MSH.4 Item="4" LongName="Sending Facility" Table="HL70362" Type
    ="HD">
      <HD.1 LongName="namespace ID" Table="HL70300" Type
    ="IS">Radiology</HD.1>
    </MSH.4>
    <MSH.5 Item="5" LongName="Receiving Application" Table="HL70361"
    Type ="HD">
      <HD.1 LongName="namespace ID" Table="HL70300" Type ="IS">IBM-
    IMM</HD.1>
    </MSH.5>
    <MSH.6 Item="6" LongName="Receiving Facility" Table="HL70362" Type
    ="HD">
      <HD.1 LongName="namespace ID" Table="HL70300" Type
    ="IS">INS</HD.1>
    </MSH.6>
    <MSH.7 Item="7" LongName="Date/Time Of Message" Type ="TS">
      <TS.1 LongName="time of an event" Type
    ="ST">20030827192206</TS.1>
    </MSH.7>
    .....
  </MSH>
  <PID>
    .....
  </PID>
  .....
</PPR_PC1>

```

Table 2 : Sample of the equivalent HL7 Patient Problem Record Protocol message (HL7 PPR_PC1)

```

<NDS>
  <NDS.1 LongName="Notification Reference Number" Type="NM">
    20040813122025456875
  </NDS.1>
  <NDS.2 LongName="Notification Date/Time" Type="TS">
    <TS.1 LongName="time of an event" Type
      ="ST">20030827192206</TS.1>
    </NDS.2>
  <NDS.3 LongName="Notification Alert Severity" Type="CE">
    <CE.1 LongName="identifier (ST)" Type="ST">2</CE.1>
    <CE.2 LongName="text" Type="ST">Moderate</CE.2>
    </NDS.3>
    <NDS.4 LongName="Notification Code" Type="CE">
      <CE.1>1</CE.1>
      <CE.2>EHC</CE.2>
    </NDS.4>
  <NDS.5 LongName="Notification Alert Urgency" Type="CE">
    <CE.1>1</CE.1>
    <CE.2>60 seconds</CE.2>
  </NDS.5>
  <NDS.6 LongName="Notification Alert Risk" Type="CE">
    <CE.1>1</CE.1>
    <CE.2>90 - 100</CE.2>
  </NDS.6>
  <NDS.7 LongName="Notification Alert Evidence" Type="CE">
    <CE.1>2</CE.1>
    <CE.2>Probable</CE.2>
  </NDS.7>
  <NDS.8 LongName="Notification Alert False Positive Rate" Type
    ="CE">
    <CE.1>1</CE.1>
    <CE.2>90 - 100</CE.2>
  </NDS.8>
</NDS>

```

Table 3 : Extended XML segment of the HL7 Notification Detail Standard (NDS)

```

<ACK_N02>
  <MSH>
  .....
  </MSH>
  <MSA>
    <MSA.1 Item="18" LongName="Acknowledgement Code" Table="HL70008"
Type ="ID">AA</MSA.1>
    <MSA.2 Item="10" LongName="Message Control ID" Type
="ST">20031216110302460</MSA.2>
    <MSA.3 Item="20" LongName="Text Message" Type ="ST">
      <Source>CU</Source>
      <INSNotificationID></INSNotificationID>
      <INSResponseID></INSResponseID>
      <AckType>A</AckType>
      <AckTime>20031216110128</AckTime>
      <RecipientRole></RecipientRole>
      <RecipientName></RecipientName>
      <RecipientDevice></RecipientDevice>
      <RecipientPriority></RecipientPriority>
      <RecipientAction></RecipientAction>
      <RecipientMessage></RecipientMessage>
      <ForwardToRole></ForwardToRole>
      <ForwardToName></ForwardToName>
      <ForwardToDevice></ForwardToDevice>
      <ForwardToPriority></ForwardToPriority>
      <ErrorCode>0</ErrorCode>
    </MSA.3>
  </MSA>
</ACK_N02>

```

Table 4: Sample XML acknowledgement message using the HL7 standard ACK_NO2