RC23306 (W0408-122) August 25, 2004
Computer Science

IBM Research Report

Advanced Clinical Notification System

Rose Williams[1], Yves A. Lussier[2], Srikant Jalan[1],
Tara Borlawsky[2], Jianrong Li[2]

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

[2]Department of Biomedical Informatics
Columbia University
New York, NY

Advanced Clinical Notification System

Rose Williams[1], Yves A. Lussier,MD[2*]*
Srikant Jalan[1], Tara Borlawsky[2], Jianrong Li[2],

1. IBM Thomas J. Watson Research Center, Hawthorne, NY
2. Department of Biomedical Informatics, Columbia University, New York, NY
* Corresponding authors

1. ABSTRACT

In this paper, we describe the notification services, architecture, implementation, and integration of two independent systems, which together form a novel and complex clinical event communication system. While our clinical notifications are delivered over the Nextel cellular network to the Blackberry 7510 device, our system is capable of delivering hl7-xml notifications across a heterogeneous set of mobile gateways supporting mobile devices, pagers, instant messaging and email. Notifications are driven by provider role and hospital policy. Notifications are also driven by policy when message delivery fails.

2. BACKGROUND

Most information requirements for doctors and nurses center on the patient's medication lists, problem lists, laboratory and other test results. Inadequate access to information and ineffective communication among patient care team members are potential proximal causes of medical errors [1]. The frequent occurrence of asynchronous messages in clinical data processing necessitates the inclusion of a general notification system as one of the core components of information management architectures [3]. Clinical event monitors, which generate messages based on clinical events and patient data, improve the quality of and reduce the cost of health care by delivering information to health care providers when and where it is needed [4]. In addition, automated alerting systems can improve provider response time [6] by efficiently detecting and communicating a wide variety of clinical conditions [5]. In general, event monitors should push critical information to providers as well as deliver supporting information to help in decision making [2].

3. INTRODUCTION OF TWO INDEPENDENT SYSTEMS

3.1 Characteristics of systems delivering time-sensitive patient data

There are several crucial features that should be included in a clinical event notification system, including the availability of multiple communication channels and multiple notification criteria (e.g., critical, severe, urgent). Message delivery should also comply with the standards outlined in the Health Insurance Portability and Accountability Act of 1996 (HIPAA), which mandates that the exchange of electronic health data adhere to certain security and confidentiality regulations. Additionally, clinical event monitors should use an industry accepted communication protocol, such as Health Level Seven (HL7), for practical and economic reasons.

3.2 Vigilens Clinical Event Monitor

The New York Presbyterian Hospital and the Department of Medical Informatics of Columbia University have designed and implemented a state-of-the-art, telemedicine-enabled, vendor-independent, multi-institution decision support system (*Vigilens*), which is designed to improve the quality of patient care. Currently, the Vigilens system is used to detect critical laboratory values in real-time based on institution-specific rules regarding upper and lower limits for laboratory results. NYPH tracks more than 60,000 laboratory results daily. For those results determined to be critical, an alert message is posted to an internet-based clinical information system used at Columbia University Medical Center, WebCIS [11]. Physicians use email or wap gateways to "pull" and review patient data.

3.3 WebSphere Everyplace Intelligent Notification Service

Intelligent Notification Services (INS) are included in IBM's WebSphere Everyplace products[14]. INS enables users or enterprises the ability to monitor particular content sources and notify users of related events using user-specified delivery preferences. The current product is shipped with the ability to monitor news, Notes Mail, and Exchange Mail content sources. When a match is found between the information of interest and the data feed, a notification is sent to the user over SMS, SMTP, and Lotus Sametime protocols.

3.4 Our End-to-End Prototype

Wireless handheld technology, including devices such as two-way alphanumeric pagers and personal digital assistants (PDAs), offers portability and mobile access to necessary information [1]. However, while two-way pagers can receive messages and transmit responses wirelessly to any email address [8], PDAs offer more functionality, larger screen sizes and better input mechanisms. Additionally, PDA phones, such as the Blackberry handheld device, offer all-in-one solutions by combining the features of a cellular phone and PDA [1]. Our prototype extends the current INS product features by monitoring for clinical event content sent by the Vigilens system, and by extending delivery to include secure data "push" channels to the Blackberry devices. Additionally, we have added the ability to deliver content to medical providers based on their role and hospital policy at the time of clinical event. We have also implemented automated procedures for monitoring and executing message delivery to address network issues.

4. CURRENT IMPLEMENTATION

4.1 Overview of system behavior

Given the variety of user preferences and uniqueness of each clinical scenario, it is difficult for an information system to predict whether a result merits urgent notification for a particular user [9]. It has been long recognized that certain results should be considered "critical" and need to be communicated immediately to clinicians [5]. However, many results do not formally fall into a "critical value" category on the basis of threshold criteria, but may be serious for a particular patient [7]. Individual institutions have specific policies and criteria for defining which alerts are critical and how they should be communicated to the appropriate personnel.

In the NYPH hospital system, for example, the Vigilens decision support system analyzes the patient data against a set of clinical protocols. All data sent by the ancillary clinical systems are converted to a common format by a gateway server, which transmits them to a clinical data repository and to Vigilens. The Vigilens event monitor detects and publishes messages about the patient's current problem. The event monitor sends the message content with corresponding notification attributes that reflect the properties of the message.

Generally, the message delivery takes into account a care provider's personal preferences and their role in an institution when determining how to communicate an alert. Our approach recognizes that allowing mobile users to decide *how* they should be notified is part of an efficient communication solution.

The INS system supports user preferences and simplifies the management and personalization of receiving notifications. INS separates user subscription information, device preferences, and system configuration data from the delivery process. The Trigger Manager (see figure 1 below) accepts content provided by content adapters, which monitor information feeds from email, enterprise applications and other sources. The Trigger Manager filters the input and looks for important keywords in the content. The Trigger Manager's query match engine compares the filtered data with the user's criteria as defined in the user's subscription. If a match is found, the Trigger Manager sends an XML message to the Universal Notification Manager. The Universal Notification Manager checks for device preferences and related information before sending the message to the gateway. Finally, the message is delivered to the user's device. Figure 1 shows these core INS components.

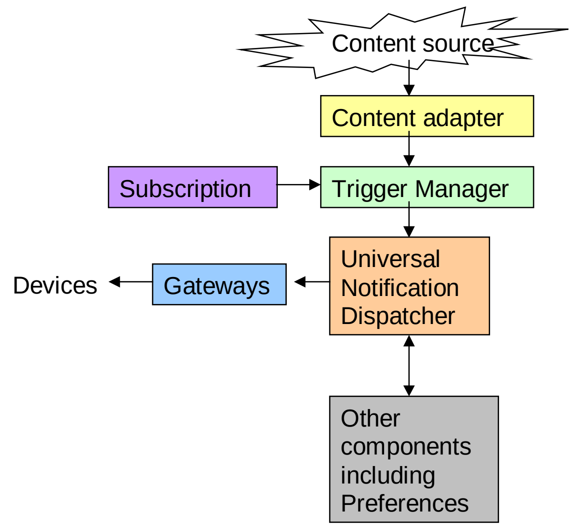

Figure 1: Core INS components

4.2 Medical Information exchange between Vigilens and INS

Healthcare service delivery is an information-intensive effort. Years ago, interest for using information technologies focused on reducing paper and improving hospital administrative processes. Today, interest in information technology centers on improving clinical care and ancillary service deliveries. There is significant focus on the development, implementation, and adherence to standards that support the external data interchange between clinical systems. Today, the industry generally supports Health Level Seven (HL7) as the external data interchange model. Vigilens uses HL7 to communicate clinical messages. Specifically, Vigilens uses Notification Attributes (NAs) to help simplify the management and personalization of the configuration of the notification message [12]. Figure 2 illustrates the notification attributes used.

1)Urgency: The Urgency NA reflects the time interval from the beginning of a message to the time when an injury or an adverse effect may affect a patient.

2)Severity: The Severity NA represents the maximal degree of potential injuries befalling a patient according to the specific event being monitored, in the absence of intervention.

3)Risk: The Risk NA is the probability of the occurrence of adverse effect or injury in the severity specified in the Severity NA when remedy is absent.

4)Evidence: The Evidence NA measures the strength of the evidence that supports the rule. It is an index for a rule's reliability.

5)Positive Predictive Value: The Positive Predictive Value(PPV) NA is a measurement of the efficacy of the Event Monitor to produce a correct message. It is a function of input parameters' error rates, which are parts of input data. In case the message is used by another rule as input, the error rates can be aggregated to get the final PPV. Even when the PPV is poor (e.g., 20%), communicating the alert information to the user remains valuable in certain cases (for example, when the urgency, the severity, the risk and the evidence are all high).

Figure 2: Notification Attributes

Our design goals included a "loose coupling" of the Vigilens Event Monitor system and IBM's Intelligent Notification system. As such, we employed HL7 2.4 as the business-to-business protocol. HL7 messages consist of data fields (described by data types) separated by field delimiters. Data fields are combined into logical groupings called segments. Each segment begins with a three-character literal value that identifies it within a message [13]. Figure 3 is an example of a patient problem report in HL7 2.3 version as found in http://www.medinfo.rochester.edu/hl7/v2.3/ch120026.htm#E11E132. To assist the reader, the three-character segment literal is marked in red.


```
MSH|^~\&|PCIS|MEDCENTER|REPOSITORY|MEDCENTER|||PPR^PC1| <cr>
PID||0123456-1||ROBERTSON^JOHN^H||||||9821111|<cr>
PV1|1|I|2000^2012^01||||004777^LEBAUER^SIDNEY^J.|||SUR||||ADM|A0|<cr>
PRB|AD|199505011200|04411^Restricted Circulation^Nursing Problem List||
||199505011200|||IP^Inpatient^Problem Classification List| NU^Nursing^Management Discipline
List|Acute^Acute^Persistence List| C^Confirmed^Confirmation Status List|A1^Active^Life Cycle Status List|
199505011200|199504250000||2^Secondary^Ranking List|HI^High^Certainty Coding List||1^Fully^Awareness
Coding List|2^Good^Prognosis Coding List|||| <cr>
ROL|1^Diagnosing Provider^Role Master List|AD|^Edwards^John^H^^MD| 199505011200||||<cr>
ROL|45^Recorder^Role Master List|AD|^Smith^Ellen^^^^|199505011201||||<cr>
OBX|001|TX|^Peripheral Dependent Edema|1|Increasing Edema in lower limbs|<cr>
GOL|AD|199505011200|00312^Improve Peripheral Circulation^Goal Master
List||||199505011200|199505101200|Due^Review Due^Next Review List|| 199505021200||QAM|||ACT^Active^Kaiser
Internal|199505011200| P^Patient^Kaiser Internal||<cr>
ROL|12^Primary Nurse^Role Master List|AD|^Wilson^Jane^L^^RN| 199505011200||||<cr>
```


Figure 3: Example of a patient problem report (PPR_PC1) in HL7 syntax

While HL7 2.x is an internationally accepted standard for electronically exchanging medical information, it is not easy to read or manipulate. XML encoding for HL7 v2.x messages makes an HL7 message easier to read, easier to transform, easier to validate, and easier to present. To exchange patient information data between our two independent systems, we use an API library that creates the XML representation of HL7 v2.4.We have also added xml extensions to include the additional notification parameters used by Vigilens(as shown in figure 2) for message alerting content. See figures 4a and 4b for the Notification Attribute xml and dtd examples. See figure 5 for an example of the communication between Vigilens and INS systems used to build the xml string from the Vigilens data to HL7 format, and send to the IBM system.

Figure4a: Additional alert parameters extending HL7

```
<PPR_PC1>
 <Existing Elements>
  …
 </Existing Elements>
 <NDS>
  <NDS.3 LongName="Notification Alert Severity" Type ="CE">
   <CE.1>2</CE.1>
   <CE.2>Moderate</CE.2>
  </NDS.3>
  <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>
```

```
FIELD NDS.5
-->
<!ENTITY % NDS.5.CONTENT "%CE;">
<!ELEMENT NDS.5 %NDS.5.CONTENT;>
<!ENTITY % NDS.5.ATTRIBUTES
  "Item CDATA #FIXED '1901'
   Type CDATA #FIXED 'CE'
   LongName CDATA #FIXED 'Notification Alert Urgency'">
<!ATTLIST NDS.5 %NDS.5.ATTRIBUTES;>
<!-- Item='1901' Type='CE' LongName='Notification Alert Urgency' -->

<!--
    FIELD NDS.6
-->
<!ENTITY % NDS.6.CONTENT "%CE;">
<!ELEMENT NDS.6 %NDS.6.CONTENT;>
<!ENTITY % NDS.6.ATTRIBUTES
  "Item CDATA #FIXED '1902'
   Type CDATA #FIXED 'CE'
   LongName CDATA #FIXED 'Notification Alert Risk'">
<!ATTLIST NDS.6 %NDS.6.ATTRIBUTES;>
<!-- Item='1902' Type='CE' LongName='Notification Alert Risk' -->
<!--
```

Figure4b: Sample DTD for additional alert parameters

Figure 5: Sample of the xml transcoding and the communication API.

```
public String sendToIBM() {
        …..
// >>>  Build eventXML from Vigilens data and  HL7 APIs <<<

// Setting Patient Problem Report Message attributes (MSH)
pprpc1.setSendingApplication("Vigilens");
pprpc1.setSendingFacility("Test");
pprpc1.setReceivingApplication("IBM-IMM");
pprpc1.setReceivingFacility("ER");
pprpc1.setControlID(tri_ALT_ts);

// Setting PID
pprpc1.setPatientID("4141414",....)
pprpc1.setPatientname("ABCDEFG",....)

//Setting Notification Attributes
prpc1.setNotificationUrgency("1", "60 seconds", null);
pprpc1.setNotificationSeverity("1", "Moderate", null);
pprpc1.setNotificationRisk("1", "90 - 100", null);
pprpc1.setNotificationEvidence("1", "Probable", null);
pprpc1.setNotificationFPR("1", "90 - 100", null);
        …..
eventXML = pprpc1.getXML("");
eventXML = eventXML;
a.writeforIBMLog(eventXML);
a.writeLABLog("Sending event. ");
String ack = client.sendEvent(eventXML);
a.writeLABLog("Get ACK from IBM: ");
        …..
```

4.2 Flow of System Behavior

Figure 6 illustrates the flow of alert that is generated in Vigilens, and is ultimately delivered to the attending physician's device. Vigilens generates an alert based on the laboratory values. The alert data is converted using the HL7-XML API library. The XML data is sent to the INS system using the communication API. INS acknowledges the receipt of the alert and returns the XML response (see figure 7). INS checks if there are any notification policies defined for the alert. If yes, INS determines the role (e.g. notify Intern, Resident, other) and the urgency (Urgent, Normal or FYI) of the notification. The user may be notified on different devices based on the urgency of the alert (e.g. Urgent messages on BlackBerry devices and FYI messages using Email). If the role does not resolve to any specific user, then the catch-all user for the specified urgency is notified. There can be a different catch-all defined for Urgent or Normal alert. There is no catch-all user for FYI alerts.

Figure 6: Notification based on Roles

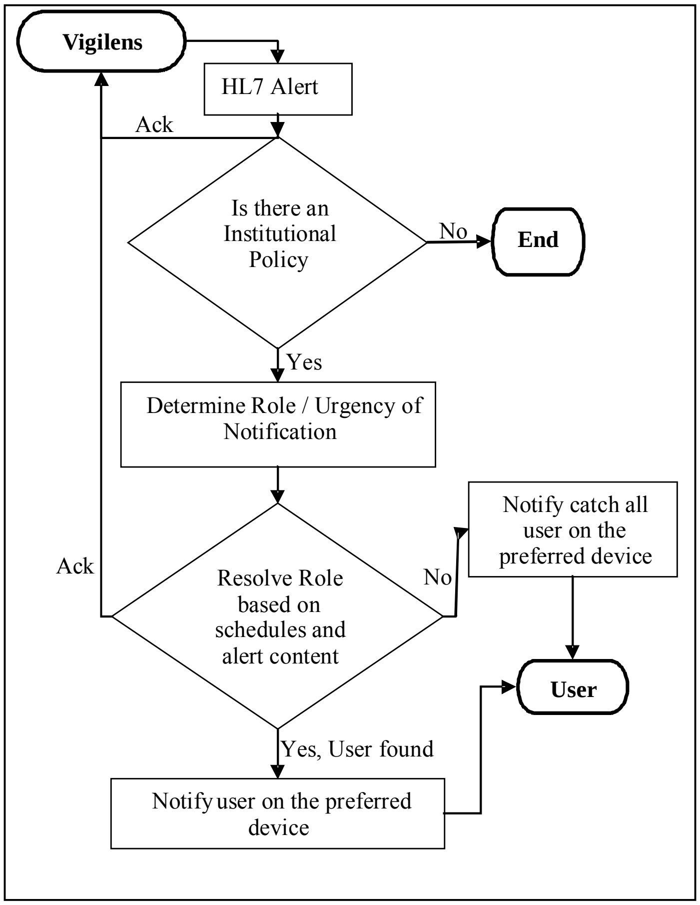

Figure 7: XML response from receipt of

<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>

Figure 8 below illustrates the flow of alert after the user responds to the previous alert, or in the case of a timeout where the user does not respond to the alert in the specified time period. Once the user is informed on the preferred device, INS notifies the cascading engine to monitor for the responses from the user. The user may respond to the alert or may decide to ignore the alert. It is also possible that the user never received the alert due to other network issues. INS checks for the policy to decide whether the notification needs to be cascaded based on the response or timeout. Based on the cascading definition, the next user is notified on his/her preferred device. Timeout for FYI messages are ignored. Timeout for messages sent to catch-all users are ignored. For all other messages for which no policies are defined, the message is forwarded to the catch-all user.

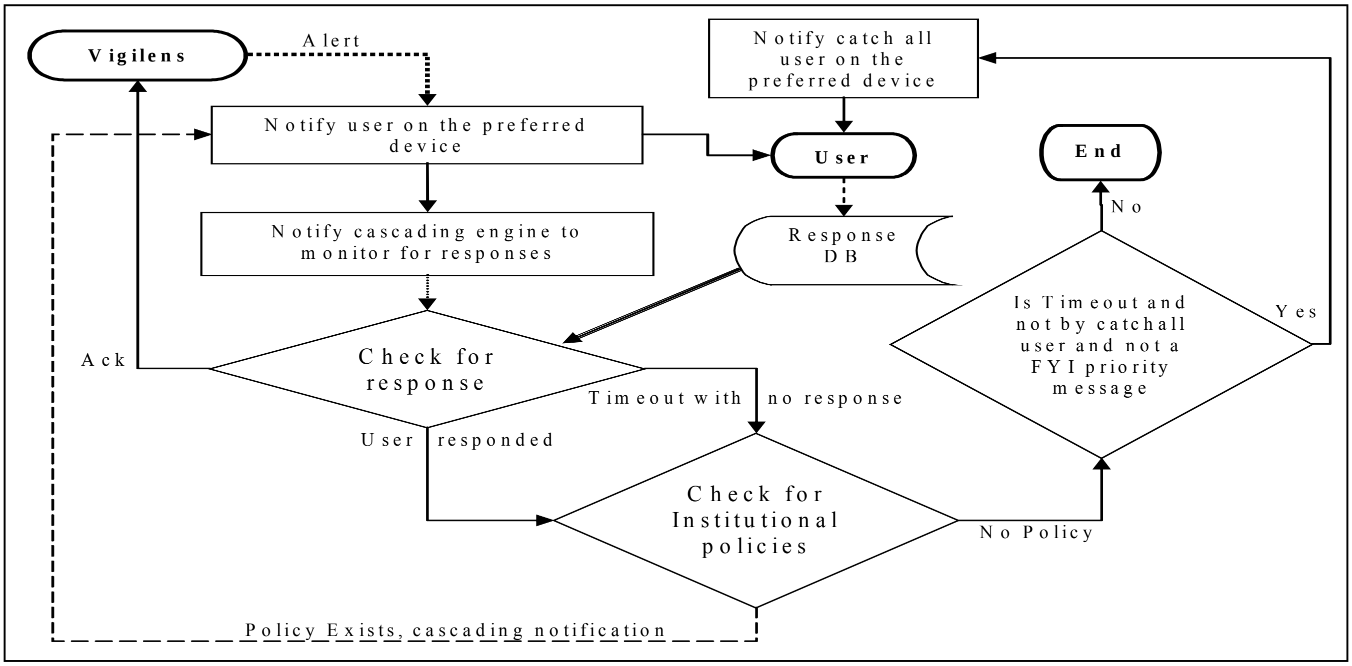

Figure 8: Alert flow illustrating system function for a user's response or message timeout

4.3 System Architecture

Figure 9 illustrates our current architecture. It is a multi-tiered architecture composed of an integration layer and notification service components.

- HL7-XML Integration Manager: Exchanges HL7-XML patient data between Vigilens and INS
- HL7 Subscription Manager: The initial subscription rule for the HL7 XML patient problem report
- HL7 Content Handler: Reformats the data into the structure required by INS.
- HL7 Trigger Handler: Processes the incoming handler message and resolves the user to be notified using the roles definition. Dispatches the data to the Core INS for delivery to user's device.
- Roles and Policy Manager: DB2 repository and access for medical provider roles and notification policies

- Message Monitor/Response Monitor: The HL7 Trigger handler notifies this process by updating appropriate DB2 database tables whenever a notification is send out. The monitor process then waits for responses by the user. The user responds by accepting, rejecting or specifying another user to whom the message should be forwarded, or the user may not respond to the message within a specified period. The monitor process then evaluates the institutional policies and cascades the message to another user in the system using the specified policy priority. This process also removes the existing messages from the system for which the expiry period is over.

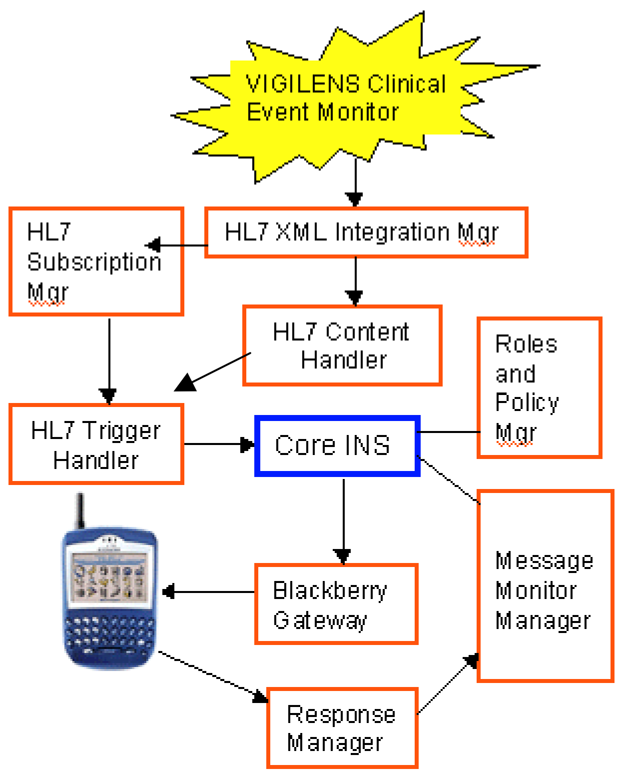

Figure 9: System Architecture for our End-end prototype

5. CONCLUDING REMARKS

5.1 Limitations

In general, our system is capable of delivering notifications across a heterogeneous set of mobile gateways that support mobile devices, pagers, instant messaging and email. However in this project, our design centered on the delivery of the clinical notification using the new Blackberry “mobile data push” technology. As such, our system delivered secure, time-sensitive patient data to the “right person” at the right time – as long as the device remained powered on and in-network. As we began our preliminary clinical trials, we encountered numerous challenges associated with disconnection and middleware failures that resulted in lost mobile-delivery laboratory results. In our trials, we realized that at this time the current BlackBerry middleware solution is more optimized for “Email push” to the device rather than the “message push” to the device. While the device is in network coverage, the messages do get pushed to the device. However, when the device is out of coverage or is powered off, not all clinical messages are delivered to the device. Additionally, in the case of successful delivery of the messages, there is no acknowledgement to the application sending the push message. The current BES solution also treats the push messages as an “Unreliable Push”. The future versions of BES are supposed to rectify this situation and make the message push using MDS a “Reliable Push” with guaranteed delivery acknowledgement to the device.

5.2 Our cascading policy for message delivery: one solution addressing the limitations

We have implemented cascading policies in our solution to provide fault tolerance in case of messages not being delivered due to network outages, device being turned off, or due to the “Unreliable Push” BES problem. An example of a “policy delivery” follows:

Retry the message to the same user if there is no response within 10 minutes. This is to take care of the device being temporarily out of network coverage. If the message is still not responded within another 5 minutes, 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 send to the specified INS user hospitalist1@ins.realm as an FYI message using Email.

5.3 Summary

Decision support tools such as event monitors are increasingly considered an essential requirement for modern medicine, primarily for improving the quality and safety of healthcare delivery. 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”. In this paper, we describe the architecture and implementation of an Advanced Clinical Notification prototype, which is a novel joining of two mature independent systems that generate alerts for clinical care monitoring, and deliver time-sensitive data to the right clinician based on his/her current provider role and institutional policy. This paper also describes our experience in providing adaptable, cascading “fault tolerant” policies to drive notification escalation when messages timed-out or delivery failed ensuring delivery of critical, time-sensitive patient laboratory results.

6. 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.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.

3.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.

4.Hripsack G, Clayton PD, Jenders RA, Cimino JJ, Johnson SB. Design of a Clinical Event Monitor. Comp Biomed Res. 1996; 29: 194-221.

5.Kuperman GJ, Hiltz FL, Teich JM. Advanced alerting features: Displaying new relevant data and retracting alerts. Proc AMIA Annu Fall Symp. 1997;:243-7.

6.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.

7.Kuperman GJ, Teich JM, Tanasijevic MJ, Ma'Luf N, Rittenberg E, Jha A, Fiskio J, Winkelman J, Bates DW. Improving Response to Critical Laboratory Results with Automation. J Am Med Inform Assoc. 1999 Nov-Dec;6(6):512-22.

8.Shabot MM, LoBue M, Chen J. Wireless clinical alerts for physiologic, laboratory and medication data. Proc AMIA Symp 2000: 789-93.

9.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.

10.Wagner MM, Tsui F-C, Pike J, Pike L. Design of a Clinical Notification System. Proc AMIA Symp. 1999;:989-93.

11. Csak G, Cimino JJ, Sengupta S. WebCIS: large scale deployment of a Web-based clinical information system. Proc AMIA Symp. 1999;:804-8.

12 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.

13.XML Special Interest Group Committee, :XML Encoding Rules for HL7 v2 Messages, v2.XML Revision 2.361, 2002 http://www.hl7.org/

14.Simon J, Patterson L. WebSphere Everplace Access Intelligent Notification Service, 2003, http://www-106.ibm.com/developerworks/websphere/library/techarticles/0311_patterson/patterson1.html