ENABLING EXISTING LABORATORY INFORMATION MANAGEMENT SYSTEMS (LIMS) WITH INTEROPERABILITY FEATURES BASED ON THE HL7 STANDARD

T. Tagaris*, S. Marinos*, S. Pavlopoulos*, S. Kostomanolakis** and D. Koutsouris*

* National Technical University of Athens/Department of Electrical and Computer Engineering, Biomedical Engineering Laboratory, Athens, Greece
** Center for Medical Informatics and Health Telematics Applications (CMI-HTA), Institute of Computer Science (ICS), Foundation for Research and Technology – Hellas (FORTH), Greece

tassos@biomed.ntua.gr

Abstract: Information and Communication Technologies are today more mature than ever, to provide all the means in order to enable existing information systems to be interoperable. A few years before, one could claim that proprietary operating systems, lack of networking and other issues, prevent software originated from different vendors to communicate with each other. But today there is no excuse. Apart from technology itself, standards and protocols concerning interoperability are both mature and stable. The different versions of the HL7 standard (v2.x and v3.0) have already been used with great success in order to enable intercommunication in healthcare institutions. One of the major needs of interconnection we find when we look at a hospital is the so called HIS – LIS communication. Laboratory Information Management Systems (LIMS) may operate in an autonomous environment but the real benefits for both the hospital and the patient appear when they integrate mainly with Hospital Information Systems. This paper presents the experiences gained trying to create an "adapter" to enable an existing LIMS to intercommunicate with other systems using the HL7 standard.

Introduction

Our experience in integrating systems in healthcare organizations starts in the middle 80s. At that time, windows applications –especially in healthcare organizations– were not that popular and most installations were in MS-DOS and Novell Environments. Moreover, there was no use of any Standards as far as application interconnection is concerned. When it comes to standards regarding coding of exams, diagnosis etc, the situation is the same until recently. Nowadays, healthcare software companies in Greece, under the public domain (government) guidance, started to adopt international standards (mainly HL7[1]) to face interconnection needs. Issues such as how introduction of standards altered the way companies have to design their systems, what are the benefits of adopting international standards and what are the steps needed to be done in order that Greece makes out the maximum benefits of adopting interoperability standards such as HL7 are discussed in this article.

Materials and Methods

What happens when you have to install a LIMS in a Hospital? There are two possibilities. Either it will have to exchange data with a pre-existing system (HIS, ERP etc), or if this is not the case… some time in the near future this will be a necessity! When dealing with many different hospital installations you have to find a standardised way to intercommunicate.

When trying to enable a proprietary system with interconnection capabilities, first of all you have to define the use case scenarios i.e. *Preconditions, Flow of Events, and Post Conditions.* In this article we deal with LIMS systems the HL7 standard ver. 2.3 and the scenarios involved in the intercommunication with ordering systems (HIS etc). The core of the discussion has to do with a messaging interface, so let's see what we mean with that:

A Messaging Interface identifies the part of an application that is responsible for[2]:

- ⇨ Receive, decode and process of messages
- ⇨ Store the data elements of the message
- ⇨ Retrieve data elements from db, encode, compose and transmit a new message

The development process of a messaging interface for an application generally constitutes of three phases:

- ⇨ Requirement Analysis
- ⇨ Interface Design
- ⇨ Implementation

In our case, which is the HIS – LIS communication the *Requirement Analysis* starts with the identification of the use cases for data interchange. As messages were to be used to exchange data between applications, the integration requirements had to be identified and described. This resulted to the identification of event triggers and the definition of the scope of data that had to be exchanged. The scope of data required determines the minimum dataset that has to be sent by the interface,

ISSN: 1727-1983

while the scope of data sent by other applications determines the maximum dataset that can be imported by the interface. In the first scenario we deal with (i.e. the HIS system has to place an order to the LIS) the dataset consists of three groups of information: Patient Demographics and Visit Information Order Identification – Origination Information and Order Analysis. If we want to translate this in HL7 terminology, we need the PID and PV1 Segments, the ORC segment to identify the order and finally the OBR segment for the analysis of the order.

Getting into more detail some PID and PV1 Segment fields are: Patient First and Last Name, Patient ID, Gender, Date of birth, Physician ID (placing the order – if this is the case), Ward ID as well as Bed (if the order regards an Inpatient), Phone Number, the Patient Visit Number and any Comments (concerning the patient). If patient is a woman Flags are needed to indicate Pregnancy and/or Menopauses as well as a date to identify LMP. These data are needed by the LIMS in order to precisely identify the normal range of the requested exams. Normally, an HIS System does not have the "knowledge" of the value that this kind of data have and modifications are needed in order to include such data on the patient file.

Concerning the Order Origination - Identification we need the Placer Order Number, the Physician ID (the one who ordered the order), the Ward ID (the one who ordered the order), the Order Date and Time and any Comments (regarding the order).

Finally, concerning the Order Analysis then for each service included in the order the following information needs to be transmitted: Service ID, Sample Type ID (depends on how the services are defined), Sample Serial Number (if it is HIS driven), Flag to indicate STAT, ROUTINE etc, Comment (regarding the specific service).

The dataset exchanged when results of the order are returned to the ordering system, is quite straightforward: Order ID, Service ID, Result (Observation) ID, Result Value, Normal Range, Units, Result Status and Comments.

Following the requirement analysis, comes the second step which is the *Design of the Interface*. The main question one has to answer here is what will be the message format and the transfer protocol. In the HIS-LIS scenario the answer is easy as HL7 has dominated the specific domain. We decided to implement two alternatives regarding the transport protocol (TCP/IP and use of the File System) as in many cases we have to deal with legacy systems that can only exchange information using files. The Monitoring interface has also to deal with transfer protocol issues such as: System and interface restarts, recovery, re-synchronization etc. We will come back to that later on the analysis of the HL7 Monitoring interface (see Figure 2).

The final step is the *Implementation of the Interface*. At that step we had to either use a message parser to facilitate complex messaging or develop our own solution. As you will see we did something that stands in the middle of the two. We used an activeX component which is free on the internet and comes from the Orion's product called Symphonia[3]. This component has all the HL7 ver. 2.3 messages implemented and exposed as objects with the appropriate methods and properties. We used this component along with Microsoft's Visual Basic ver. 6.0 Environment to implement the messaging interface. In the discussion section we will explain the reasons that made us follow this approach and the alternatives we could have used.

The overall architecture of the system (LIMS with the HL7 Interface included) is presented in Figure 1:

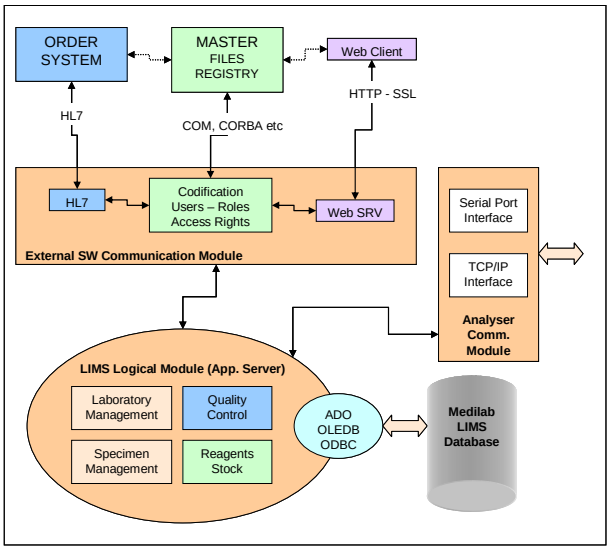

Figure 1: Overall Architecture of the LIMS and the HL7 Communicator

In the external SW communication module (i.e. the messaging interface) we have developed:

1. the TCP/IP communication (i.e. sending – receiving HL7 messages) as well as the file system based communication
2. Parsing of the HL7 messages (ver. 2.3) and mapping to the business model objects (i.e. patient, order etc)

In the LIMS Logical module (the application server) the business model objects are mapped to the corresponding database tables. We used the Oracle's PL/SQL and stored procedures to create the Business Rules of the system and map the objects to the database. As there are installations where the database is not ORACLE but Microsoft SQL Server, we have also transfer the same stored procedures into SQL Server's corresponding code.

Figure 2 presents the architecture of the HL7 interface engine developed in more detail:

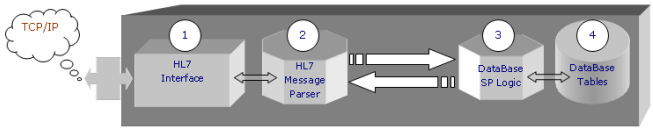

Figure 2: Architecture of the HL7 Interface Engine

We can identify four main components:

ISSN: 1727-1983

1. The HL7 Monitoring Interface
2. The HL7 Message Parser
3. The Business Logic
4. The storage system (database tables)

We will present these four modules one by one:

The HL7 monitoring interface is the interface of the system with the administrator of the system (see figure 3). The exchanged HL7 messages are monitored and there are several parameters that the administrator may define such as:

1. Whether the interface will accept incoming messages or not.
2. The frequency that the engine will check the database to find pending results and create the appropriate messages to send them to the ordering system
3. Whether these results will be send automatically or only after request (solicited or unsolicited update)
4. The administrator may manually decide to resent the results of an order
5. The identification of the systems that take part on the communication as well as the communication parameters (whether the communication is based on TCP/IP or file system, the IP address and port of each system etc)
6. The way that the engine will handle the specimen identifiers

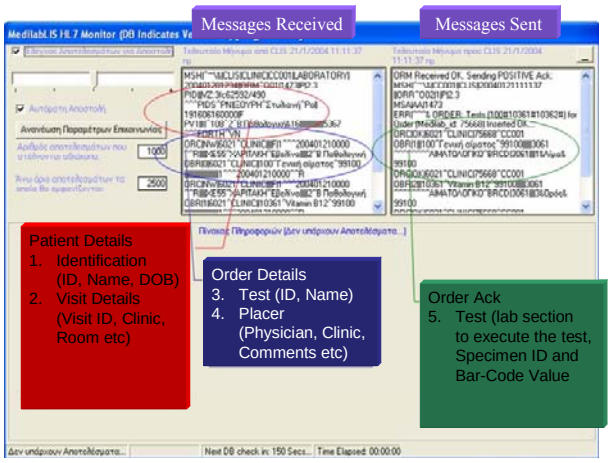

Figure 3: The monitoring interface of the HL7 communicator

The HL7 message parser handles the following messages of the protocol:

- ⇨ New Order to the Lab (ORM O01)
- ⇨ Acception or rejection of the Order (Acknowledgement) (ORR O02). With the same message the system may return to the ordering system the required barcode labels to identify the specimens needed for the order.
- ⇨ Query about an order (QRY R02)
- ⇨ Answer to the above query (QRF R04))
- ⇨ Request to cancel an order (ORM O01)
- ⇨ Acception or Rejection of the cancel request. (Acknowledgement) (ORR O02). The system may accept to partially cancel an order depending on whether the processing of the corresponding specimen has started or not.
- ⇨ Unsolicited update (ORU R01)
- ⇨ Unsolicited update acknowledgment (ACK R01)
- ⇨ Solicited Update Query (QRY R02)
- ⇨ Answer to the Solicited Update Query (QRF R04)

Before even parsing the messages, the interface stores each one of them in order to be able to either trace any malfunction or to handle network problems and restart the communication in cases it was lost for any reason. After logging the messages, the parser maps the information of the message to the appropriate business objects and the control passes to the *business logic module.* This is a very important module as it includes all the knowledge of the LIMS logic and procedures. Several decisions are made by this module having to do mainly with the numbering of the specimens needed to execute the order. A very big issue that most of the LIMS systems have to deal with is the specimen identification.

An example of specimen handling procedure is presented below:

- ⇨ Patients are referred to the Laboratory either electronically, through the HIS system, or by telephone/fax.
- ⇨ Non-electronic referrals are entered into the laboratory system by the Laboratory receptionist. The Patient Number is entered and stored in all referrals.
- ⇨ The patient will either attend in person to have the required specimen taken or a specimen will be taken externally and sent to the laboratory for processing.
- ⇨ External specimens will be stored in an appropriate container and this will be placed in a sealed plastic bag along with a completed request form. The form will include the hospital number if it exists, selected patient demographics (name, address, date of birth, contact telephone number), referrer details as well as details of the required examinations. The outside of the bag will be labeled with a standard hospital label which will include the Patient Number in plain text, the patient name and date of birth and the hospital number in barcode format.
- ⇨ When a patient arrives at the Laboratory reception or a specimen is received, the existing order details are retrieved by entering the Patient Number by keying or by scanning the barcode on a patient letter, wristband or the specimen bag label. If no order exists then the order is created.
- ⇨ If no patient exists then the patient is created in Alert and the LIMS.
- ⇨ Laboratory (LIMS) format labels are printed by the receptionist to attach to the sample container or to pass on to the Laboratory Doctor to attach when the sample is taken from the patient.

ISSN: 1727-1983

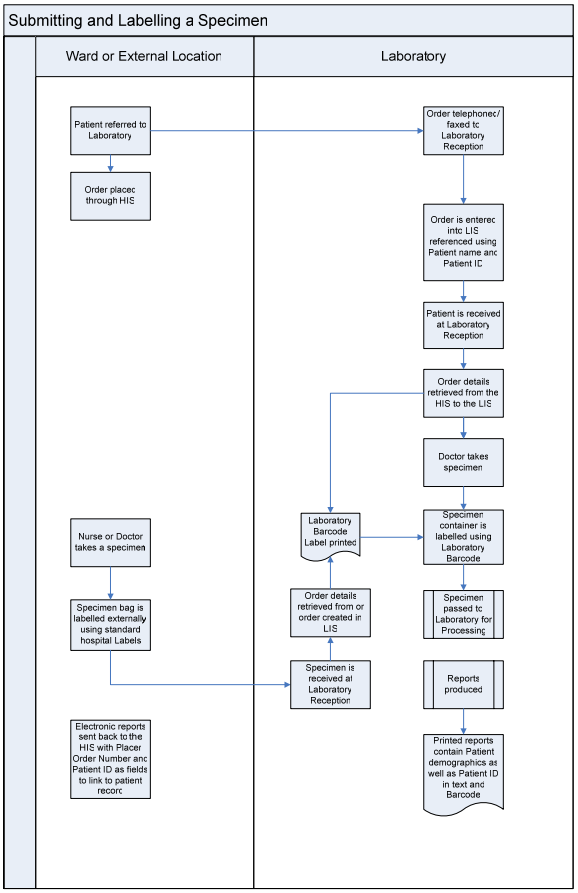

Figure 4: Specimen handling procedure

⇨ The labeled samples are then passed into the laboratory for processing.

This procedure is presented in figure 4 and is handled by the business logic module.

Depending on the abilities of the HIS system (is it able to create id numbers or barcode labels to identify the specimens? How the combined tests are handled? etc) the business module places the order to the LIMS system in the appropriate way.

Another significant task that the application module deals with is the management of the master files or reference tables creation and maintenance. Reference tables maintain identifiers concerning:

⇨ Locations
  - o Wards
  - o Clinics
  - o Healthcare providers

⇨ Services – Results (Tests/Exams)
⇨ Order set definitions
⇨ Physicians (and/or Users)
⇨ Healthcare Providers

All this logic is included in the database engine using stored procedures.

Apart from the technology issues that need to be addressed in order to develop such interface engines there are several issues that need to be discussed and agreed in order to achieve intercommunication. The two parts of the communication have to discuss and agree on issues such as:

1. Who will create the Order IDs? What will be the format? Who guarantees that order ids are unique
2. How is the patient identified? Is there a master patient index in the hospital?
3. Will barcodes will be used to identify Orders, specimens etc? Who creates these labels and what personnel assigns the labels on the physical objects (paper order or specimen itself)
4. Is there a central place where the samples are collected and labelled, or each lab does this on its own?

Questions like these are always there to be discussed and answered. The problem is that the answers are not always the same nor can they be. The interface engine has to be flexible enough in order to be able to handle every possible answer on questions like the above. This was one of the most difficult task we had to solve. The solution we adopted is based both on parameters used to identify and map the answer to the above questions and on using in some cases different (and the appropriate) stored procedures (ie business logic) to handle the situation. The truth is that the questions and answers are a finite number, so the main problem is to try to find all of them along with their valid combinations.

Results

The approach we followed, lead us to a very cost effective solution compared to the HL7 middleware and messengers available in the market. In the main article we discuss the different use cases we dealt with in the different hospitals (4 hospitals in Greece and 1 hospital in Cyprus) and the solutions given to each case. There are now 5 stable and up and running real life installations based on the HL7 communicator developed.

The results of using the system at the of the ONASEIO Cardiologic Surgery Centre in Greece are the following:

⇨ Average time to order a simple lab test: ~1,05 min – max 3 min
⇨ Average time to order a group of lab tests: ~1,2 min – max 3 min
⇨ Average time of lab response with the results of a requested order: ~ 140-160 min
⇨ Average time for the results to become available to the HIS after they are signed out by the lab: ~2,5 min
⇨ Number of tests executed from 1/1/2001 to 31/3/2005 from the Blood center and Biochemical lab: 2.185.778 (about 500.000 tests per year)
⇨ Errors in the communication of results: 0
⇨ Uptime of system's communication: 99,95% (2002 – 2005)

Discussion

As already mentioned, technology is already here ready to provide the required solutions in order to enable intercommunication between software systems in the healthcare domain. The problem seems to be more on the side of persuading the key players to adopt

ISSN: 1727-1983

international standards than on finding the right technological solutions. The major problems we had to deal with, in order to persuade people in healthcare to use HL7 were:

- ⇨ Managers have no idea of what standards exist and where each of them can help their organization
- ⇨ The most disappointing situation was when we had to deal with people from other companies who did not have an open mind to accept (and invest) on such solutions.
- ⇨ Of course this can easily be explained as the investment in time, people and money is at no means insignificant.
- ⇨ But if you see the overall cost you realize that the investment is done once, but the benefits of the standards last for the whole lifetime of your product.

A major question was: Should we invest on using a message broker (Middleware) or develop a Peer – to – Peer solution? Both scenarios have their pros and cons. As presented in Figure 5 a solution using a message broker becomes very effective as the systems that need to communicate become more. In that case each system only needs one interface (the one to communicate with the middleware) no matter how many other systems are taking part in the communication. The only disadvantage in that scenario is the cost of the solution. That cost may be affordable for the healthcare enterprise when the need for communication involves 4, 5 or even more systems but, this it is not the case (on the contrary) when there is need for only two systems (HIS – LIS) to intercommunicate.

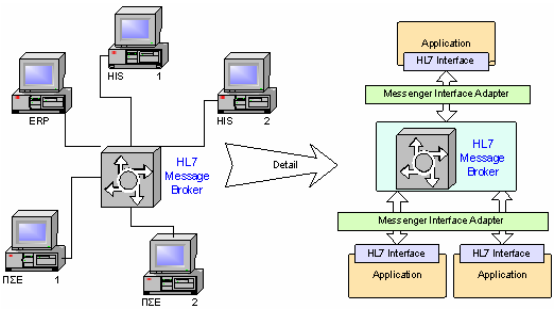

Figure 5: HL7 communication using only one HL7 interface per system.

In cases where there is only two or at most three parts needing to interconnect, there are other much more cost effective solutions to follow, than investing on heavy and expensive message brokers. Peer-to-peer communication may need one interface per system per connection (see figure 6), but when we have the HIS-LIS communication we only need one interface per system (just as it would be the case of using a message broker) and at the same time we avoid the investment needed to install a message broker.

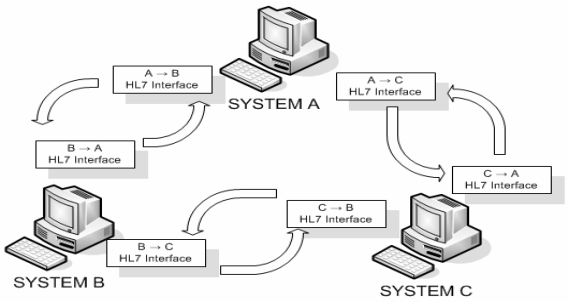

Figure 6: Peer-to-peer communication using HL7. Each system needs one interface for every connection

Conclusions

The problems one will have to face when trying to create a solution based on an international standard but having to deal with particularities of a specific country (and the corresponding way of working) are not trivial but when overcome we end up with an international standard based solution. At this point, other systems need to plug into ours using a conformance statement that has to be clearly defined.

The truth is that more Healthcare solution providers need to start to adopt Standards such as HL7. We also have a big lack of services to support the master files or dictionaries using National or International Standards. These include and are not limited to:

- ⇨ Services (Tests, Medical Acts etc)
- ⇨ Protocols, Treatment Plans
- ⇨ Diagnosis
- ⇨ Charge Descriptions
- ⇨ Pharmaceutical products
- ⇨ Unique patient identification number (PIDS?)
- ⇨ Healthcare Providers
- ⇨ Master Files CODIFICATION

As we already mentioned above, technology is already here. What we need is a big effort to create and maintain national or international standards both for codification and the communication in the healthcare domain.

References

[1] http://www.hl7.org
[2] http://www.ringholm.de/docs/00200_en.htm
[3] http://www.hl7.com

ISSN: 1727-1983