Pervasive Adaptive Data Acquisition Gateway for Critical Healthcare

Sérgio Oliveira[1], Filipe Portela[1,2], Manuel F. Santos[1], José Machado[1], António Abelha[1]

[1]Algoritmi Research Centre, University of Minho, Portugal
[2]ESEIG, Porto Polytechnic, Porto, Portugal
sergiomdcoliveira@gmail.com; {cfp, mfs}@dsi.uminho.pt; {jmac, abelha}@di.uminho.pt;

Abstract. The data acquisition process in real-time is fundamental to provide appropriate services and improve health professionals decision. In this paper a pervasive adaptive data acquisition architecture of medical devices (e.g. vital signs, ventilators and sensors) is presented. The architecture was deployed in a real context in an Intensive Care Unit. It is providing clinical data in real-time to the INTCare system. The gateway is composed by several agents able to collect a set of patients' variables (vital signs, ventilation) across the network. The paper shows as example the ventilation acquisition process. The clients are installed in a machine near the patient bed. Then they are connected to the ventilators and the data monitored is sent to a multithreading server which using Health Level Seven protocols records the data in the database. The agents associated to gateway are able to collect, analyse, interpret and store the data in the repository. This gateway is composed by a fault tolerant system that ensures a data store in the database even if the agents are disconnected. The gateway is pervasive, universal, and interoperable and it is able to adapt to any service using streaming data.

Keywords: Gateway, Pervasive Data, Vital Signs, Ventilation Data, Medical Devices, Real-time, Data Streaming, Data processing, Sensors, Adaptability.

1 Introduction

In critical healthcare environments the patients are typically connected to medical devices by sensors. There are a set of sensors able to monitoring the patient condition to the monitors (e.g., vital signs and ventilation). However in many cases these data only are shown in the monitors and they are not stored in database. They are considered temporary values. Having conscience of the problems associated to a delay recording and data loss and how the services can benefit with the introduction of this data in the database (clinical data available to be consulted in real-time), a universal gateway was designed and implemented. This gateway is able to collect patient data in real-time, independent of the medical device used. A report from the Institute of Medicine [1] presents a set of new concerns arising from the increase of information technologies in healthcare and the needs of it support the patients. One of the concerns is related to data quality. This problems can be overcame by implementing systems able to automate some clinical workflow as is the introduction of autonomous system to acquire the data monitored. In summary, integration of information systems in the healthcare environment has provided the opportunity for improvements in work, being the tasks performed faster, more consistently and with less cost [2].

The main goal of this work is developing a pervasive universal and adaptive system able to collect the monitoring data in real-time. The system should be prepared to collect data from a set of medical devices / sensors connected to the patient (eg. ventilator and vital signs monitor) and using an ethernet network approach to provide data streaming. The proposed system complements the existing architecture in Intensive Care Unit (ICU) of Centro Hospitalar do Porto. Actually the ventilators are only used to consult the values in a single moment. The monitored data are not stored in the database. In this context there is the need to create a system capable of collecting these data. In order to overcome this problem, it was developed a pervasive universal gateway that was integrated in the data acquisition subsystem. It is able to collect all patient data (being ventilated) in real-time, making them available for consult and analysis. This gateway presents as main features: adaptability, universality, autonomy, pervasiveness and fault tolerance, data streaming from sensors, data acquisition and data processing in real-time.

This paper is organized in five section. In the second section, it is covered a background of the work. The gateway data acquisition requirements and their characteristics are presented in the third section. The third section also presents the interaction between Client-Ventilator and Client-Server, it is presented the algorithm used to create the Health Level Seven (HL7) messages, the failure mechanism and the system to control the Client versions. In section four is made a discussion of the work and finally the concluding remarks are made in the fifth section.

2 Background

2.1 INTCare

INTCare is a research project which gave origin to a Pervasive Intelligent Decision Support System (IDSS) with the same name to Intensive Care Units. It is able to support physicians and nurses decision process in real-time [3]. INTCare take advantages of data streaming, intelligent agents and an autonomous Extract, Transformation and Loading (ETL) process. INTCare is able to support the decision making process by providing new knowledge in real-time [4]. INTCare is composed by semi-autonomous agents responsible for several tasks in real-time, as example: Data acquisition from sensors and medical devices; data streaming; data processing and transformation and induction of data mining models. These tasks do not require human intervention. The Data acquisition subsystem is based in the sensory processing tasks using intelligent agents. When a patient is admitted to an ICU, the clinical staff connects several sensors to the body. These sensors are prepared to collect patient clinical data as is vital signs and ventilation values. In the case of ventilation system, the sensors are connected to the ventilator which is prepared to receive all the signs sent by the sensors. The data received are shown in the ventilator monitor. INTCare interferes in this phase by developing a gateway able to process and collect these data to a database. The process starts when the data arrives to the ventilator. The data are received through a gateway and they are packed into HL7 messages. Then the data are stored on the database by another agent. Automatic data acquisition and consequently data streaming eliminates transcription errors, improves the records quality and allows the assembly of large electronic archives of data [5]. The system characteristics were defined taking into account the environment, the information needs and Data Mining requirements. These characteristics are fundamental for the development of a universal and adaptive gateway [6] providing data acquisition in real-time.

2.2 Interoperability and Pervasive Health

The biggest challenge of this work is to ensure the interoperability of consistent communication of information between all the hospital system and platforms. It should allowing a data streaming process provided by heterogeneous systems [7]. To facilitate the process is necessary defining a communication protocol / language.

The Health Level 7 (HL7) is a language composed by a set of patterns where the interoperability perspective is evident. HL7 help to improving the communication processes that handle with information [8]. Given this context the Agency for Integration, Archive and Diffusion of Medical Information (AIDA) were used [9] as a gateway support. One of the Interoperability change is Pervasive HealthCare. Pervasive in Healthcare (PH) can be defined as a "system for providing healthcare to anyone, anytime and anywhere by removing restraints of time and place, while enhancing the quality of health care" [10]. This approach is based on the information that are stored and available online [11]. In the field, the implementation of PH solution has technical and administrative obstacles, such as resistance to significant changes in the area of technology and information systems [12]. As a solution it is recommended to put the necessary information available electronically, complemented by predictive models that can help clinicians make better decisions in real time [13].

3 Gateway

The first requirements and the interaction between the client and server was already presented [14]. This section makes a short introduction of the system requirements and the interaction existing between client and server. Then the new features are presented.

3.1 System Requirements

In this section functional and technical requirements for developing the gateway are presented. As functional requirements: Client communication with ventilator must be done through the RS-232 port; The server must be able to receive messages from different clients (data acquired through client communication with the monitors ventilation); Store the data into a database; Create HL7 messages for communication between the systems; Efficiently treatment of exceptions; Log and alert system (mail sending when an error occurred); Secondary mechanisms for controlling data missing.

As technical requirements: Low disk space and restricted memory usage. The processes of data acquisition and data storing are supported by intelligent agents. These agents are programed to make their tasks autonomously and in real-time. The gateway was developed in python taking advantage from the several libraries existing in this field.

3.2 Interaction between Client and Server

The Clients are installed in computer associated to the ventilator. Each Client is waiting by the ventilator message. These messages are produced after a signal be measured by the patient sensors. Then the Client receives the message and performs some changes (formatting) in order to send it to the Server in a universal format (HL7) through ethernet network. In order to provide a deeper understanding of the Gateway, a use case diagram, according to the modelling language UML (Unified Modelling Language), is presented in Figure 1.

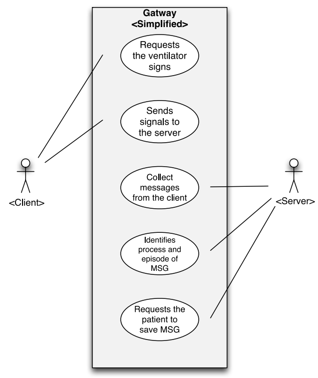

Fig. 1. Use Case Gateway

As mentioned in the first version already published [14], In Figure 1 is possible observe the interactions between the system and the user. "However and since there are not human interactions presented in this functional unit, the interlocutors (client and server) presented are an integral part of the system. The client and server are two integral parts, each of which performs a set of tasks necessary to provide all the functionalities of the gateway. To demonstrate the interactions between objects and scenarios performed by the methods or operations through two operations, a sequence diagrams were depicted.

Figure 2 elucidates how the data is acquired through Client and from the ventilator. The Client (Ventilator (X) - where X is an individual ventilator) performs a set of messages exchanged between two objects: Port RS-232 and Ventilator. Initially the Client has to identify if there is any Serial port installed that supports Client execution. If yes, it sends a message (MSG) identified by the ventilator port. Then, the Ventilator sends a message with the respective values of the fields requested by the Client. Finally, the Client receives the message sent by the Ventilator and chooses the set of fields that will be introduced, creates a message according VL syntax and sent it to the Server."

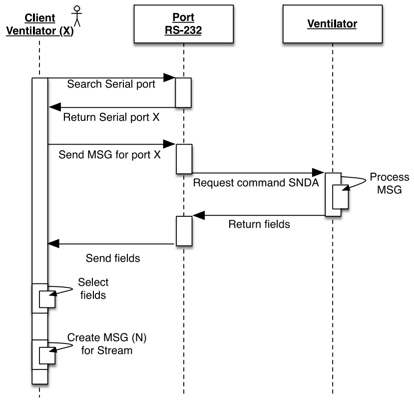

Fig. 2. Client Sequence Diagram

Figure 3 presents the sequence diagram of the operations made during the data streaming process from the Client to the server through Agent HL7. This diagram is a sequence of the diagram presented in figure 2. The Server System receives the message and then it identifies the Client IP and the patient Bed associated to this message. Then a request is sent to the database to identify which is the patient that is being monitored in those bed. Then, the collected data are correctly identified (there is a match between the data collected and the combination of patient identification (PID) and patient bed in the Electronic Health Records). At the end, the Server System updates the message (adding the PID) and sends the data collected from the patient for Agent HL7.

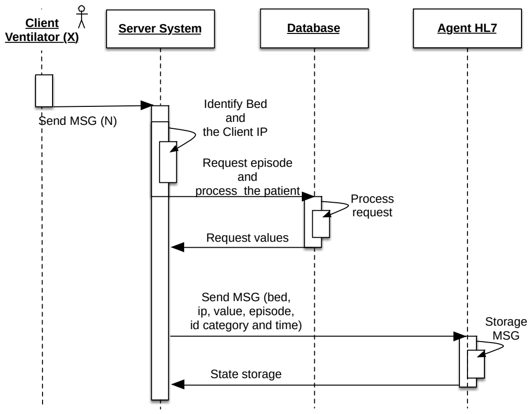

Fig. 3. Server Sequence Diagram

Sequence diagrams of Figures 2 and 3 represent the operations performed by the Gateway developed. The Client and the Server System are responsible to ensure an integral communication between heterogeneous systems parts.

3.3 Server parallel processing

The Server is always listening for Client messages. When a message is received by the server, it is instantiated one high-level threading class. This message implicitly gives origin to a new task. Each task is responsible for the creation of all HL7 messages and sent it to the multi-task server. In case of the server is receiving a new message but if it has not yet finished running (is processing the last message received) it creates a new task. Always this situation is verified the Server creates a new task with parallel processing. The developed Server is a distributed processing system.

3.4 Create and sending Health Level Seven messages

When the Server creates a new task, the database is queried to know who owns the received message. Then it proceeds to the process of creation and sending of HL7 messages. The next algorithm presents all the process.

Algorithm 1 - Create and send HL7 message to the Agent HL7

Require: date_number, date_value, process, name, bed and fields

Function create HL7 message (date number, process, name, bed, value_fields, num_fields, service)
  For all fields **Do**
    MSH=Create segment MSH with (name, date number, generateID)
    PID=Create segment PID with (process, name)
    PV1=Create segment PV1 with (bed)
    OBR=Create segment OBR with (num_fields, name, date_number, service, bed)
    OBX=Create segment OBX with (field)
    MSGHL7= Concatenate (MSH, PID, PV1, OBR, OBX)
    Call function SEND_MSGHL7(MSGHL7)
Function SEND_MSGHL7(MSG=”empty”)
  If MSG **!=** "empty" **Then**
    Send MSG for Agent HL7
      Wait for Ack
  Else
    Print “MSG is empty”
  If Ack == -1 **Then**
    Call booking engine
  Else
    Print “Successfully received message”

The first function of the represented algorithm generates an HL7 message. The number of messages generated it is directly dependent of the number of fields collected. For example if the ventilator collects 10 fields then the respective task will send 10 HL7 messages for Agent HL7. Then, the second function is created and it is identified whether the Agent HL7 is receiving messages. Table 1 describes the segments used in the HL7 message created:

```
MSH|^~\&|NAME_IDNET|MSG_ID1||ORU^R01|MSG_ID2|P|2.3.1
```

PID|1||PATIENT_ID||NAME^PATIENT
PV1|1|U|BED_NUMBER
OBR|1|||NAME_IDNET|||MSG_DATE|SERVER_DATE
OBX|1|SM|VARIABLE_ID||VARIABLE_VALUE||||||R|||MSG_DATE
|ID_SERVICE^BED_NUMBER||^AUTOMATIC

Table 1 –HL7 Segments and Description

Segment	Description
MSH	Message Header
PID	Patient Identification
PV1	Patient Visit
OBR	Observation Request
OBX	Observation/Result

All the messages sent to HL7 agent must have the presented structure. The five represented segments must be sent in the same message. If some segment is missing the HL7 agent did not process the message properly. Another important aspect for the message to be processed by the Server it is the message ID. The ID of each message cannot be repeated, in this case an automatic ID is created based in message date (timestamp number), the service ID and a continuous and non-repeatable number.

3.5 Reserve Engines

The reserve engines is a failure tolerant system able to ensure a continuous data storing, even if there is some problems in the HL7 process. The server receives messages from Clients and it has as priority handling the messages and sending them to the Agent HL7. However if for some reason the Agent HL7 is down, a set of mechanisms are automatically activated. With goal to avoid losing messages, two extra mechanisms able to store the messages not stored by the agent were developed. The first extra mechanism (second mechanism) is triggered in the case of Agent HL7 fails. This mechanism sends the message directly to the database. In fact it gets and processes all the received message and creates several messages in SQL. Then the Server insert each of the messages into the database. The SQL message data are the same presented in HL7 messages, only the shape of segments differs. In extreme cases when the database is down it is need to activate the third mechanism (second extra). The third mechanism is only called when the first and second storing mechanisms are unable to store the data sent by the Clients. The third mechanism does not send data to any server or agent. It stores the data locally on files. The files are generated in two formats (.sql and .hl7). The messages generated (in HL7 and SQL) in the previous mechanism are not lost, they are stored locally on files for future processing. Each file has the messages generated by each mechanism, i.e., it has all the information provided by a Client in the message. Later it is possible to take the files generated and insert the information by the HL7 Server or put it directly into the Database. Once they are generated two files with the same information only it is possible to select a format to store the data in the database. One flowchart was designed to represent the process above described.

Figure 4 is focused in demonstrating the mechanism developed, their conditions and server actions. Briefly when the first message is received, the number of fields (variables) are counted. Then it is verified if the message received can be converted in a HL7 message. If yes, the message is created and it is sent to the server agent. Then and after the agent storing the message in the database it sends an acknowledged (ACK) message. This process is repeated for all the message fields. In case of ACK failure, the agent will convert the first message in Sql code and he will store it directly in the database. In case of database connection fails, the data will be stored in a file. This is a three steps process and only if one steps fails the following task is activated.

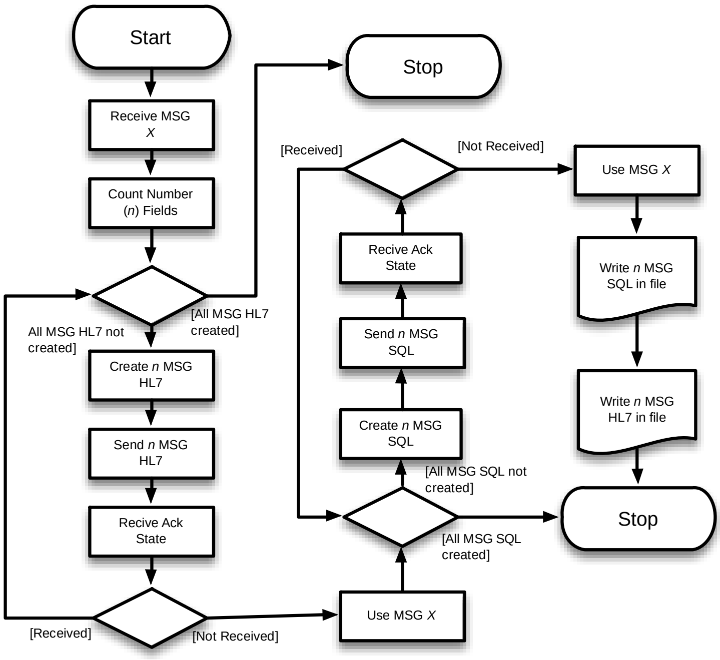

Fig. 4. Flowchart of Server mechanisms

3.6 Starting and Alerting Systems

The Server which are receiving the messages from ventilators and Agent HL7 has a restart mechanism. This mechanism was developed in a file Batch to automate the tasks. Each Agent is associated to a Server and it is composed by a set of commands. This Agent can start the Servers always they are not running. The agent checks periodically (5 min) if the Servers are running or not. Additionally, it was implemented a warning system to inform the state of Servers and Agent HL7. This system sends an email periodically to inform the state of the gateway. So it is possible to know whether the Servers and the Agent HL7 failed / are down. In case of the down time be long, it is sent an email informing that the servers and the Agent HL7 are still off.

3.7 Control gateway versions

The development of an interoperable gateway will provide greater ease of use in the future. It can be applied to other ventilation system monitors since it was developed a module capable of controlling clients' versions. This module was developed for three reasons: The ventilation systems may change; It is need to change the fields that are collected and there were changes in system settings: values, IP and Port. A Client is always waiting to receive a message with an order to update the software. If the client do not receive any message to make an update, it continues to make their work. When the Client receives a message, it identifies if it is an update message and if the message was written in a local file. Then the Clients software version are updated. At the end, all the Ventilator Clients will be with the most recent version and up-to-date.

4 Discussion

The development of a gateway system to collect data from ventilators and the sensors connected to the patients provides a new way of consulting and view clinical values. The spectrum of values collection provides a better understanding of changes in the patients values. Thus, the system has demonstrated to be useful, due to be interoperable with many data acquisition devices and it can be easily adaptable to other environments, providing a data streaming in real-time. The fact of being used a universal language (HL7) makes this gateway a universal solution, because it can connect to others clinical system and the data can be used for several other goals. The gateway developed ensures that there are not data loss, due to the development of two extra acquisition mechanisms. The alert system can be used to restart the HL7 Agent and Server and sending emails containning their status. All the development and implementation process was followed by health professionals. Their contribution was valuable to develop a most sensitive system. The data collected are then presented to the medical staff using interfaces already developed, as is INTCare Monitoring System.

5 Conclusion

This paper presented the development of a pervasive and adaptive data acquisition system. The gateway was developed and incorporated in a real hospital system: INTCare. The data streaming is made in real-time and the data acquisition system allows for greater data collection capabilities, giving useful information in order to, for example, predict clinical condition. In order to develop a viable system, a high number of functional and technical requirements that demonstrates to be essential for the proper functioning of the system were taken into account. The system configuration was also important to make all systems interoperable enabling the communication between different systems.

The gateway has already collect around 2 million of ventilation data (in the last three months) from the sensors / ventilator. This situation would not be possible with manual registration. These data are providing to healthcare professionals a greater insight into the state of patients regarding to the respiratory system. It has also provided the development of Data Mining models to make predictions related to barotrauma and patient extubating [15, 16]. The fact of being pervasive, all the technicians can access to their settings anywhere and anytime and the information provided by the ventilator are always available. Additionally it sends emails when something is not working well.

All of this development are an important pointing to the researchers in order to have an interoperable, universal and pervasive system able to ensure a full data acquisition in critical healthcare environments. For the future it is expected improving the gateway by adding security flaws and deploying the gateway in other hospital services.

Acknowledgements

This work has been supported by FCT - Fundação para a Ciência e Tecnologia within the Project Scope UID/CEC/00319/2013 and the contract PTDC/EEI-SII/1302/2012.

References

1. A. Berry and Z. Milosevic, "Real-Time Analytics for Legacy Data Streams in Health: Monitoring Health Data Quality," presented at the Enterprise Distributed Object Computing Conference (EDOC), 2013 17th IEEE International, 2013, pp. 91–100 (2013).
2. A. Dwivedi, R. Bali, and R. Naguib, "Building New Healthcare Management Paradigms: A Case for Healthcare Knowledge Management," presented at the Healthcare Knowledge Management Issues, Advances, and Successes, (2006).
3. F. Portela, M. F. Santos, J. Machado, A. Abelha, Á. Silva, and F. Rua, "Pervasive and Intelligent Decision Support in Intensive Medicine – The Complete Picture," in Information Technology in Bio- and Medical Informatics, M. Bursa, S. Khuri, and M. E. Renda, Eds. Springer International Publishing, 2014, pp. 87–102 (2014).
4. F. Portela, M. F. Santos, and Á. Silva, "Towards Pervasive and Intelligent Decision Support in Intensive Medicine – A Data Stream Mining Approach.," AIM Journal (2016 (accepted).
5. M. F. Santos, F. Portela, M. Vilas-Boas, J. Machado, A. Abelha, J. Neves, A. Silva, and F. Rua, "Information Architecture for Intelligent Decision Support in Intensive Medicine," W. Trans. on Comp., vol. 8, no. 5, pp. 810–819, Maio (2009).
6. F. Portela, P. Gago, M. F. Santos, Á. Silva, F. Rua, J. Machado, et. al, "Knowledge discovery for pervasive and real-time intelligent decision support in intensive care medicine," KMIS (2011).
7. P. Gibbons, N. A. Phd, T. Flewelling, T. Jepsen, D. K. Md, J. Larson, J. Ritter, M. R. Md, S. Selover, and J. Stanford, Coming to Terms: Scoping Interoperability for Health Care Health Level Seven EHR Interoperability Work Group. (2007).
8. B. G. M. E. Blobel, K. Engel, and P. Pharow, "Semantic interoperability--HL7 Version 3 compared to advanced architecture standards," Methods Inf Med, vol. 45, no. 4, pp. 343–353, (2006).
9. L. Cardoso, F. Marins, F. Portela, M. Santos, A. Abelha, and J. Machado, "The Next Generation of Interoperability Agents in Healthcare," International Journal of Environmental Research and Public Health, vol. 11, no. 5, pp. 5349–5371, May (2014).
10. H.-A. Park, "Pervasive Healthcare Computing: EMR/EHR, Wireless and Health Monitoring," Healthc Inform Res, vol. 17, no. 1, pp. 89–91, Mar. (2011).
11. M. Mikkonen, S. Väyrynen, V. Ikonen, and M. O. Heikkilä, "User and Concept Studies As Tools in Developing Mobile Communication Services for the Elderly," Personal Ubiquitous Comput., (2002).
12. U. Varshney, "Location Management for Mobile Commerce Applications in Wireless Internet Environment," ACM Trans. Internet Technol., vol. 3, no. 3, pp. 236–255, Agosto (2003).
13. F. Portela, "Pervasive intelligent decision support in critical health care," PhD Thesis, Dec. (2013).
14. Oliveira, S., Portela, C.F., Santos, M.F.: Pervasive Universal Gateway for Medical Devices. Recent Advances in Electrical Engineering and Education Technologies 205-210 (2014).
15. Sérgio Oliveira, F.P., Manuel Filipe Santos, José Machado, António Abelha, Álvaro Silva and Fernando Rua: Intelligent Decision Support to predict patient Barotrauma risk in Intensive Care Units. In: Elsevier (ed.) Procedia Technology - HCIST 2015 - Healthy and Secure People, (2015).
16. Sérgio Oliveira, F.P., Manuel Filipe Santos, José Neves, Álvaro Silva and Fernando Rua: Feature selection for detecting patients with weaning failures in Intensive Medicine. In: CPS (ed.) Mathematical Methods and Computational Techniques II, (2015).