Collaborative Patient Centred eHealth
E. De Clercq et al. (Eds.)
IOS Press, 2008

doi:10.3233/978-1-58603-922-6-139

Open Source Electronic Health Record and Patient Data Management System for Intensive Care

Jacques MASSAUT[a,1], Pascal REPER[a]
[a]*Intensive Care, Brugmann Hospital, Université Libre de Bruxelles, Belgium.*

Abstract. Background and objectives: In Intensive Care Units, the amount of data to be processed for patients care, the turn over of the patients, the necessity for reliability and for review processes indicate the use of Patient Data Management Systems (PDMS) and electronic health records (EHR). To respond to the needs of an Intensive Care Unit and not to be locked with proprietary software, we developed a PDMS and EHR based on open source software and components.Methods: The software was designed as a client–server architecture running on the Linux operating system and powered by the PostgreSQL data base system. The client software was developed in C using GTK interface library. The application offers to the users the following functions: medical notes captures, observations and treatments, nursing charts with administration of medications, scoring systems for classification, and possibilities to encode medical activities for billing processes. Results: Since his deployment in February 2004, the PDMS was used to care more than three thousands patients with the expected software reliability and facilitated data management and review processes. Communications with other medical software were not developed from the start, and are realized by the use of the Mirth HL7 communication engine. Further upgrade of the system will include multi-platform support, use of typed language with static analysis, and configurable interface. Conclusion: The developed system based on open source software components was able to respond to the medical needs of the local ICU environment. The use of OSS for development allowed us to customize the software to the preexisting organization and contributed to the acceptability of the whole system.

Keywords. database management system, electronic health record, software, intensive care, HL7.

1. Introduction

The electronic health record (EHR) becomes increasingly important in modern health care systems. Numerous clear advantages over paper records are demonstrated. Any form of medical record needs to be accurate, consistent, legible, complete, and simply presented [1]. The use of information technology in health care records allows the user to improve the quality of information, conveys accurate information quickly, meets

[1] Corresponding Author: Prof. J.Massaut, Soins Intensifs, CHU Brugmann, 4 palce Van Gehuchten, 1020 Bruxelles, Belgium ; E-mail : jacques.massaut@chu-brugmann.be.

specific needs, access a patient's data whenever it is needed, and enable the rapid extraction of data to improve overall patient care[2,3].

These advantages of EHR applied also in intensive care were the amount of data to be processed, the turn over of the patients and the necessity for reliability and review processes indicate the use of Patient Data Management Systems (PDMS) [4]. Commercial PDMS exist for intensive care but they lock the users with proprietary software as opposed to Open Source Software (OSS). By publishing source code, OSS allows sharing of software resources and experience [5]. They can also be used in intensive care [6].

To respond to the needs of our intensive care unit and benefit of resources from OSS we developed an EHR system and PDMS based on open source software and components. The aim of that development was also to avoid to be locked with proprietary software and be dependent of costly licenses.

2. Materials and Methods

2.1. Software Design

The software used by the system was designed as a client-server architecture running on the Linux operating system (SUSE Linux Enterprise Server 8.0) and access the PostgreSQL relational database (v 7.2). The development of the system was based on the workflow and dataflow observed in our unit and the procedures and documentations already in use. The data were mapped on the relational database. The database structure can be found at: www.openosiris.org/download/OSDMS poster.pdf. The client software was developed in C with the GTK interface library, using a bottom-up approach.

The software offers the following functions: (Figure 1)

1. Medical notes captures with patient's history, observations and treatments,
2. Nursing charts for vital signs, IN-OUT balance, ventilation parameters and settings,
3. Functionalities for administration of medications,
4. Scoring system possibilities for patient's classification. (APACHE II [7], SAPS II [8]).
5. Reporting at the end of hospitalization in Intensive Care.
6. Encoding of medical activities for administrative and billing processes.

Interoperability between all modules of the software is realized through access to the PostgreSQL database and not the use of local memory in the interface. The software was developed to be open source in all its components and is interfaced with Open Office for reporting. To track run-time errors and achieve sufficient software reliability, we systematically tested all the software with Valgrind, a suite of debugging and profiling tools.

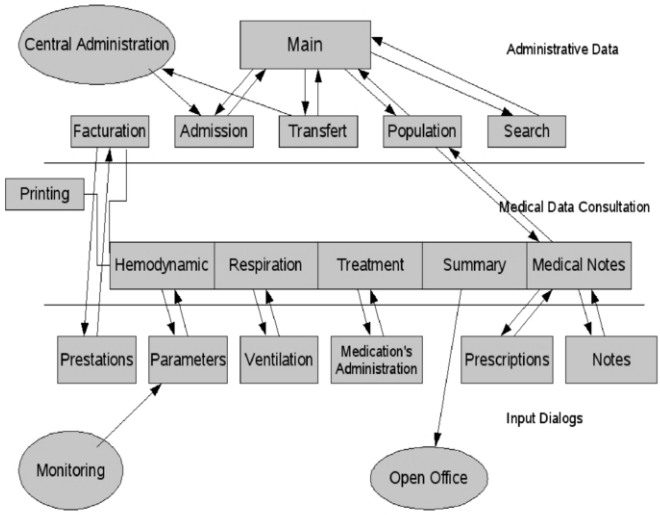

Figure 1. Interface and Software Design

2.2. Implementation and software use

All the above functionalities were implemented but only the medical part of the application was used.

For every new patient, on admission, data have to be introduced in the software using classical medical observations: relevant medical history, previous treatment and chronological history of the actual problems justifying the ICU admission, as well as clinical findings and complementary examinations with specific description of the medical diagnostic. For every ICU day, clinical data on patient evolution can be described with the most important elements of the day: biological and bacteriological results, respiratory and hemodynamic status and results of the last performed complementary examinations. Daily treatment and therapeutic strategy may be prescribed and updated using an integrated care provider order entry.

Description of ICU population is necessary to evaluate the patient prognosis and the severity of illness. Comparison between patients needs patient's evaluation and stratification for study or clinical purpose. It is therefore necessary for the ICU management to score the patients with well admitted ICU scores. For that purpose, APACHE II and SAPS score could easily be determined for every patient on admission and during the ICU stay permitting to stratify ICU population and giving important informations for the patient follow-up and the ICU management.

Reporting at the end of hospitalization in ICU is necessary for communication with referring hospital specialists or general practitioners. This allows transferring important follow-up information and enhancing better collaboration between the different patient care teams: in ICU, in the other hospital wards and also at home when the patient leaves the hospital. The automatic help on reporting at the end of the hospitalization enhances collaboration and information transmission of all who are involved in the care of acute patients.

Encoding of medical activities for administrative and billing processes can be done with the system. Complete administrative information is important for the general

management of care institutions where ICU structures are known to use a lot of personal and financial means: the software allows obtaining, after a minimal time, various information to answer to the questions of the medical authorities inside or outside the hospital.

Scientific studies are also enhanced by the possibility of extracting of the database complete data about a specific population and its ICU evolution.

2.3. Hardware

Minimum hardware configuration for the system consists of one x86 server with windows PC running virtual network connections (VNC) viewers, but such a configuration does not allow backup and replication. The hardware we used consists in two Intel x86 servers with uninterrupted power supply, one master and one slave to assure the integrity of the database by replication, 14 medical grade panel PCs (Advantek PPC-153m) connected via RS232 medical bus to the patient's monitoring devices and to the servers via dedicated local Intranet network. The master server runs the postgreSQL database and the client-software that can be accessed through VNC. The slave server is used for replication of the database. The 14 medical grade panels PCs run the client software at the bedsides.

2.4. Communications with monitoring devices and other medical software

We planned first to access the data from the monitoring devices of our unit through the RS232 interfaces of the Philips Intellivue monitoring devices at every bedside. That needed complex and specific communication software to be developed and implemented at every bedside. Central acquisition of data using HL7 communication standard with the Mirth HL7 communication engine installed on servers, is an easiest process and a better solution. The data can then be transferred to the postgreSQL database. The Mirth HL7 communication engine can also be used to send clinical data from the database to other medical software or export clinical summary at the end of hospitalization in intensive care.

2.5. Identification, authorization and security

Identification and authorization are to be done at the database level with encrypted passwords. Several levels of authorization must include read access only, write access with privilege for prescription or not, and administration of medications. Identification must be repeated at every important access like note writing, prescriptions and administration of medications. Except for local access on private network, database access must be done through secure connections. VNC on windows does not provide such secure access and supplementary software are needed.

2.6. License

The code developed for the system was published under the General Public License version 3 at https://savannah.nongnu.org/projects/freeosiris.

2.7. Upgrade

Before extending his use to 30 IC beds of our institution the system is being upgraded. Principal upgrades includes multi-platform support including Windows Operating System, separation of clinical data from patient identification and administrative data on two separate databases to facilitate extraction of anonymous data for clinical review, research and privacy preservation as proposed in the OpenEHR model[9] (http://www.openehr.org). New development will use more secure languages: Ada and SparkAda in place of C.

3. Results

The PDMS was used in our unit, from February 2004, for the care of more than three thousand patients. The system is accessible at desks or offices through VNC viewers on windows PCs. Its design allowed an access to the database's functionalities with a high availability level (less than 5 hours of interruption over one year). The system is used on an every day basis for staff discussion using central display and for every patient's notes and treatments. Indicators were also developed to follow the activity of the unit and are used at regular intervals for evaluation as well as database queries to answer specific clinical question. The use of the system at every bedside through panel PCs was less successful. The panels PCs with fans are noisy for the patients and less robust.

The use of open source resources was however effective to customize the solution to ICU medical request and contributed to the acceptability of the software. The PostgreSQL database largely contributed to the overall efficacy and robustness of the system. The use of the C language permits to obtain small response times but limits the portability of the system and complicated the debugging process in this critical environment. For that reason, Valgrind software was used during development period to systematically track run-times errors. The system was well accepted locally for medical activity, but was harder to interface with the information system of the hospital and is to be completed to include all the work at the bedside.

4. Discussion

We developed the present software to respond to the needs of our surgical unit, with the hope that this will enhance quality in our unit. In a review of Clinical Informatics in Critical Care, G. Martich describes several reasons to implement information system in intensive care [4]. The first one is that information systems could reduce medical errors and first of all medications errors. The second reason is that information overload is present at point of care in intensive care units. Clinical informatics at the bedside can help to better manage this load. Other reasons are described like necessity to achieve and assess compliance to guidelines and accreditation rules. To be effective to improve outcome, databases should be developed and controlled at the level were change is to occur [10]. Several studies have demonstrated the effectiveness of using relational databases to improve care of intensive care units patients and specially infected patients [11, 12].

We decided to base our development work on OSS for three main reasons, first to benefit of the large OSS library and resources, second to avoid to be locked into proprietary software and third to be able to adapt the software to the manual procedures preexisting in our unit. Economical reasons were also present. These reasons are similar to that described by Douglas Carnal. That author described in 2000 that open collaboration over the Internet is changing development methods and that OSS will be a significant part of the Medical Software's Future [5]. However, Martich's review of Clinical Informatics in Critical Care in 2004 does not mention any OSS used in Intensive Care and we founded only one OSS specific to Intensive Care well described in the literature [13]. 292 medical projects are available for download on sourceforge.net but only one is directly related to intensive care and available only for the German language. A community of users of OSS in Intensive Care is clearly to be created. Several factors limit adoption of OSS, like limited support and some times insufficient quality [14]. For our application, we used external support for the hardware and internal support for the software. We tried to respect, during development and implementation, a level of quality corresponding to the needs of our environment.

Intensive care environments require systems with high availability. The system described here was able to respond to these requirements by the use of dedicated and duplicated servers and the use of dedicated local network. Robustness however was lower at the bedsides on the Panel PCs.

Software development and testing for Intensive Care need to achieve high reliability. The C language used to develop the software is unfortunately not by itself a safe language. For that reason, we systematically tested the software with Valgrind, a suite of simulation based debugging and profiling tools. The uses of static analysis of the C code with tools like Splint [15] early in the development process and before compilation, or the use of safer languages like Ada or SparkAda [16] are however better solutions and are used to develop secure systems. Ada and SparkAda, with static analysis, will be used for further developments. Finally, to better respond to the need of the work at the bedsides and better integrate nursing work, the system will be upgraded with flexible, evolving and configurable interfaces.

Initially, the lack of module designed to communicate with other medical software and applications was a limitation of the system. The development of communications using the HL7 standard with the Mirth HL7 communication engine can solve this problem and facilitate the integration of the PDMS with other medical software and applications.

5. Conclusion

The developed system based on open source software components was effective and able to respond to the medical needs of the local ICU environment. The use of OSS allowed us to customize the software to the preexisting medical organization of the unit at low cost and contributed to the acceptability of the whole system. The system needs however further design and development to better integrate the work at the bedside and communication with other medical software and applications.

References

[1] Bradbury A, Computerized medical records: the need for a standard, *J. Am. Rec. Assoc.* 19(3) (1990), 25-37.

[2] Collins B, Wagner M, Early experiences in using computerized patient record data for monitoring charting compliance, supporting quality initiatives and assisting with accurate charging at Allina Hospitals & Clinics. *Int. J. Med. Inf.* 74(11-12), (2005), 917-925.

[3] Elliot B, To computerize or not to computerize the patient care record: that is the question. *Del. Med. J.* 74(11) (2002), 435-441.

[4] Martich G, Waldmann C, Imhoff M, Clinical Informatics in Critical Care. *J. Intensive Care Med.* 19 (2004), 154-63.

[5] Carnall D, Medical software's free future. *BMJ,* 321(7267) (2000), 976.

[6] Massaut J, Reper A, Reper P, Open source software can also be used in intensive care. *Intensive Care Medicine,* A 20070 S50 0180 (2007)

[7] Knaus WA, Draper DP, Wagner DP, Zimmerman JE, APACHE II: A severity of disease classification system. *Crit Care Med*, 13 (1985), 818-829.

[8] Le Gall JR, Lemeshow S, Saulnier F, A new Simplified Acute Physiology Score (SAPS II) based on a European/North American multi center study. *JAMA*, 270 (1993), 2478-2486.

[9] Kalra D, Beale T, Heard S, The openEHR Foundation. *Stud Health Technol Inform,* 115 (2005), 153-73.

[10] Clemmer P, Monitoring Outcomes With Relational Databases: Does It Improve Quality of Care? *Journal of Critical Care,* 19, No 4 (2004), 243-247.

[11] Evans R, Pestonik S, Classen D, et all, A computer assisted management program for antibiotics and other anti-infective agents. *N Engl J Med,* 338 (1998), 232-238.

[12] Burke J, Pestonik S, Antibiotic use and microbial resistance in intensive care units: impact of computer-assisted decision support. *J. Chemother,* 11 (1999), 530-535.

[13] Kropyvnystskii I, Sauders F, Schierek P, Pols M, A computer system for continuous long-term recording, processing, and analysis of physiological data of brain injured patients in ICU settings. *Brain Inj,* 15(7) (2001), 577-83.

[14] Sfakianakis S, Chronaki C.E, Chiarugi F, Conforti F, Kateakis D.G.: Reflections on the Role of Open Source in Health Information System Interoperability. *Methods Inf Med,* 46 (2007) Suppl1:50-60.

[15] Evans D, Larochelle D, Improving Security Using Extensible Lightweight Static Analysis. *IEEE Software.* January/February (2002), 42-51.

[16] Hall A. and Chapman R, Correctness by construction: developing a commercial Secure System. *IEEE Software.* January/February (2002), 18-25.