

ORIGINAL PAPER

Electronic Health Record for Intensive Care based on Usual Windows Based Software

Arnaud Reper, Pascal Reper

Intensive Care, Brugmann University Hospital, Free University of Brussels, Belgium

Corresponding author: Reper Pascal, MD. Critical care department. University Hospital Brugmann, Brussels, Belgium. Tel 32 2 477 23 15. Fax 32 2 477 34 58. E-mail:prmdr@gmail.com

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 an EHR based on usual software and components. **Methods:** The software was designed as a client-server architecture running on the Windows operating system and powered by the access data base system. The client software was developed using Visual Basic 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 September 2004, the EHR was used to care more than five 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 basic functionalities 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 usual software components was able to respond to the medical needs of the local ICU environment. The use of Windows for development allowed us to customize the software to the preexisting organization and contributed to the acceptability of the whole system.

Key words: database management system, electronic health record, software, intensive care.

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 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 Usual Windows Software (UWS) which allows sharing of software resources and experience (5). They can also be used in intensive care.

Open source software could also be some alternative solution (6) but are more sensitive to software or operating systems upgrades which can complicate data transmission or recovery.

To respond to the needs of our intensive care unit and benefit of resources from UWS we developed an EHR system (and PDMS) based on Windows Based 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 an architecture running on the Windows operating system (Windows Server) and access the Access relational database (v Office 2000). The development of the system was based on the workflow and data flow observed in our unit and the procedures and documentations already in use. The data were mapped on the relational database. The client software was developed in Visual Basic using a bottom-up approach.

The software offers the following functions: (Figure 1)

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

Interoperability between all modules of the software is realized through access to the Access database and not the use of

local memory in the interface. The software was developed to be compatible in all its components and is interfaced with Windows for reporting. To track runtime errors and achieve sufficient software reliability, we systematically tested all the software elements .

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.

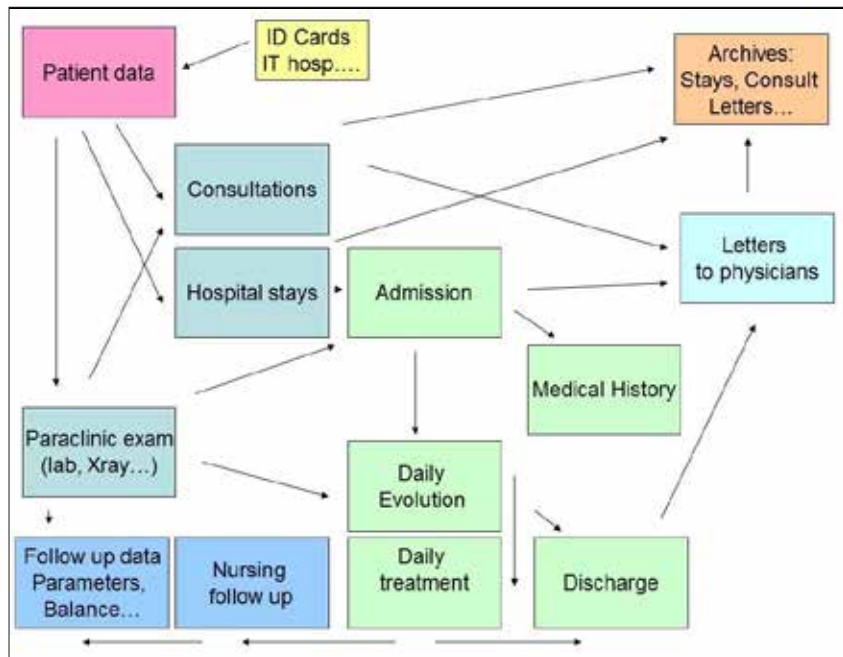


Figure 1. Interface and Software Design

2.3. Hardware

Minimum hardware configuration for the system consists of windows PC running available intrahospital network connections, such a configuration allowing backup and replication through the existing hospital systems. The hardware we used consists in classic Intel PC with uninterrupted power supply, and connection with the hospital network to assure the integrity of the database by replication..

2.4 Communications with other medical software.

We planned to access the data from the general network of the hospital available in every wards. That didn't need complex and specific communication software to be developed and implemented. The data can then be transferred to the ICU database and to the database of the institution. The Mirth HL7 communication engine could 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 but macro use in Windows based software was also a easily used solution.

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.

2.4. License

The code developed for the system was published under the General Windows License existing, bought and widely used in the hospital.

2.5. Upgrade

Before extending this use to more IC units in our institution the system has not been upgraded which is a big advantage. Principal upgrades include multi-platform support including Windows Operating System, separation of clinical data from patient identification and administrative data on separate databases for each new IC unit to facilitate extraction of anonymous data for clinical review, research and privacy preservation. New developments will use more secure performances compatible with Windows upgrades policy.

3. RESULTS

The EHR was used in our unit, from September 2004, for the care of more than five thousand patients. The system is accessible at desks or offices through windows PCs. Its design allowed an access to the database's functionalities with a high availability level (no interruption over the years). 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 questions. The use of the system at every bedside represents the future developments.

The use of Windows resources was however effective to customize the solution to ICU medical request and contributed to the acceptability of the software. The Access database largely contributed to the overall efficacy and robustness of the system. The use of the Visual Basic language permits to obtain small response times and doesn't limit the portability of the system with complicated debugging process in this critical environment. For that reason, Windows environment has to remain stable to avoid version related problems and run-time errors. The system was well accepted locally for medical activity, and was not harder to interface with the information system of the hospital but has to be completed to include all the work at the bedside.

Despite more and more technicians in ICU patients, more frequent elderly patients and higher severity confirmed by ICU scores the mortality in our ICU seems to diminish.

The use of HER could be an interesting complementary part to implement in ICU to enhance quality and security in severely ill patients

4. DISCUSSION

We developed the present software to respond to the needs of our intensive care unit, with the hope that this will enhance quality in our unit. In a review of Clinical Informatics in Critical Care, G. Daniel Martich describes several reasons to implement information system in intensive care (5). 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 where 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 Windows based systems for three main reasons, first to benefit of the large 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 collaboration over the Internet is changing development methods and that open source software (OSS) will be a significant part of the Medical Software's Future [6]. However, Martich's review of Clinical Informatics in Critical Care in 2004 does not mention any OSS used in Intensive Care and we found 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 but 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 based on usual and widely used Windows software, 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 running on the existing network of our institution.

Software development and testing for Intensive Care need to achieve high reliability. The VB language used to develop the software is fortunately by itself a safe widely used language. For that reason, we didn't systematically had to test the software with a suite of simulation based debugging and profiling tools like for example Valgrind. The uses of static analysis of the code with specific tools like Splint (15) early in the development process and before compilation, or the use of safer languages like Ada or SparkAda (16) recommended when OSS are developed were not necessary to obtain better solutions which are used to develop secure systems. Ada and SparkAda, with static analysis, were also not to be used for further developments with OSS. Finally, to better respond to the need of the work at the bedside 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 but simple macros used within the Windows software could resolve this problem without complicated interventions.

5. CONCLUSION

The developed system based on usual source software components was effective and able to respond to the medical needs of the local ICU environment. The use of Windows based systems allowed us to customize the software to the preexisting medical organization of the unit at low cost and con-

tributed 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 or devices and applications.

CONFLICT OF INTEREST: NONE DECLARED.

REFERENCES

1. Bradbury A. Computerized medical records: the need for a standard, *J. Am. Rec. Assoc.* 1990; 19(3): 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.* 2005; 74(11-12): 917-925.
3. Elliot B. To computerize or not to computerize the patient care record: that is the question. *Del Med J.* 2002; 74(11): 435-441.
4. Martich G, Waldmann C, Imhoff M. Clinical Informatics in Critical Care. *J Intensive Care Med.* 2004; 19: 154-163.
5. Carnall D. Medical software's free future. *BMJ.* 2000; 321(7267): 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.* 1985; 13: 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.* 1993; 270: 2478-2486.
9. Kalra D, Beale T, Heard S. The openEHR Foundation. *Stud Health Technol Inform.* 2005; 115: 153-573.
10. Clemmer P. Monitoring Outcomes With Relational Databases: Does It Improve Quality of Care? *Journal of Critical Care, Vol 19.* 2004 Dec; 4: 243-247.
11. Evans R, Pestonik S, Classen D, et al. A computer assisted management program for antibiotics and other anti-infective agents. *N Engl J Med.* 1998; 338: 232-238.
12. Burke J, Pestonik S. Antibiotic use and microbial resistance in intensive care units: impact of computer-assisted decision support. *J Chemother.* 1999; 11: 530-535.
13. Kropyvnytskii 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.* 2001; 15(7): 577-583.
14. Sfakianakis S, Chronaki CE, Chiarugi F, Conforti F, Kateakis DG. Reflections on the Role of Open Source in Health Information System Interoperability. *Methods Inf Med.* 2007; 46 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, Chapman R. Correctness by construction: developing a commercial Secure System. *IEEE Software.* January/February 2002: 18-25.