Chapter XI
Care2x in Medical Informatics Education

Andreas Holzinger
Institute of Medical Informatics, Statistics, and Documentation, Medical University Graz, Austria

Harald Burgsteiner
InfoMed/Health Care Engineering, Graz University of Applied Sciences, Austria

Helfrid Maresch
InfoMed/Health Care Engineering, Graz University of Applied Sciences, Austria

ABSTRACT

In this chapter the authors report about their experiences in education of both students of healthcare engineering at Graz University of Applied Sciences, and students of medicine at the Medical University Graz, gained during the winter term 2004. Care2x is an open source Web-based integrated healthcare environment (IHE). It allows the integration of data, information, functions, and workflows in one environment. The system is currently consisting of four major components, which can also function independently: hospital information system (HIS), practice management (PM), a central data server (CDS) and a health exchange protocol (HXP). Although the components are under heavy development, the HIS has reached a degree of stability, where one can use it at least for educational purposes. Various groups also report the usage of enhanced versions of Care2x in real life settings. Our experiences in both—very different—student groups have been very promising. In both groups the acceptance was high and Care2x provided good insights into the principles of a hospital information system. The medical students learned the principal handling of a HIS, whereas the engineering students had the possibility to go deeper into technical details.

INTRODUCTION

In this chapter, the authors report about their experiences in the education of students of healthcare engineering (HCE) at Graz University of Applied Sciences, and students of medicine at the Medical University Graz, gained during the winter term of 2004. Care2x is an open-source Web-based integrated healthcare environment (IHE). It allows the integration of data, information, functions, and work flows in one environment. The system currently consists of four major components, which can also function independently: the hospital information system (HIS), practice management (PM), a central data server (CDS), and a health exchange protocol (HXP). Although the components are under heavy development, the HIS has reached a degree of stability so that one can use it at least for educational purposes. Various groups also report the usage of enhanced versions of Care2x in real-life settings. Our experiences with both—very different—student groups have been very promising. In both groups, the acceptance was high and Care2x provided good insights into the principles of a hospital information system. The medical students learned the principal handling of an HIS, whereas the engineering students had the possibility to go deeper into technical details.

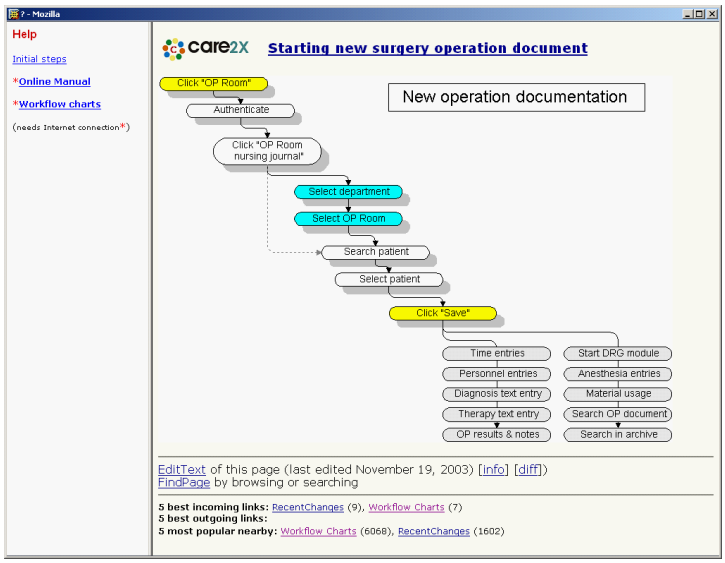
How to prepare both medical and engineering students in the best possible way for their later work with modern HISs is a common question. Whereas students of engineering are rather enthusiastic about IT, students of medicine are skeptical in general about using it. However, HISs are not widely accepted by healthcare professionals; that is, barriers to the use of HIS are primarily sociological, cultural, and organizational rather than technological (Moore, 1996).

It seems plausible to not only give students theoretical background about the structure, functions, and common tasks of an HIS, but to also let them work with a fully functional HIS during lectures. This is essential, particularly if students are required to be able to work with possibly any HIS in practice after only a short period of vocational adjustment. However, it depends on many different factors regarding which HIS to choose. One of the most important is whether it is necessary to teach (with) a particular HIS of a certain vendor, for example, if this system is deployed in a network of local hospitals. Another key factor, especially for noncommercial educational institutions, is the economic impact of the introduction of a commercial HIS at the university. Third, for the education of students of medical informatics, it might also be reasonable to teach the process of developing (parts of) a bigger software engineering project. Hence, the need for an open-source system arises if one does not want to start the development of his or her own HIS. Although there are many more factors to consider in general, we chose Care2x as our primary educational HIS for the following reasons.

CARE2X

Care2x is a generic multilanguage, open-source project that implements a modern hospital information system (the Web page of Care2x is located at http://www.care2x.org/). The project was started in May 2002 with the release of the first beta version of Care2x by a nurse who was dissatisfied with the HIS in the hospital where he was working. As of today, the development team has grown to over 100 members from over 20 countries. Care2x is a Web-based HIS that is built upon other open-source projects: the Apache Web server from the Apache Foundation (http://www.apache.org/), the script language PHP (http://www.php.org/), and the re-

Figure 1. Help page describing the clinical path for starting a new surgery-operation document



lational database-management system (RDMS) mySQL (http://www.mysql.com/). There exist several source-code branches that try to integrate the option to choose from other RDBMSs like Oracle and postgreSQL. The latter one is already supported in the current version at the time of this writing (Deployment 2.1). For our investigations, we chose the most feature-rich version that was available from the Care2x Web page in early fall of 2004. This release had the version number 2.0.2. Some minor deficiencies that we report later may already be fixed in the current version, Deployment 2.1.

Care2x is a very feature-rich HIS that is fully configurable for any clinical structure. It is built upon different modules, which include, for example, in and outpatient administration, admission, pharmacy, radiology (including DICOM [Digital Imaging and Communications in Medicine] image uploads), laboratories, ambulatories, nursing, medics, DRGs (diagnosis-related groups), and so forth. Online help is available for some clinical paths. See Figure 1 for an example.

REVERSE ENGINEERING

The reverse engineering of existing complex software packages starting at the source-code level has a higher value for practical education than a new development. Bothe (2001) argues that groups of students will rarely be able to develop a project further than to a prototype stage during a single lecture. Access to the source code is not available for most commercial HISs, which is another advantage of using Care2x as an educational system. In our first lecture, the students of HCE were asked to test all functions and paths of Care2x. They had to set up a small virtual clinic and employ doctors, nurses, and technical stuff. Finally, patients had to be admitted, attended to, and dismissed at all stations. In a second lecture in the upcoming

semester, our students have the assignment to analyze a fully functional HIS at the source-code level. Since Care2x is built upon a modular structure, small teams of programmers have tasks like finding and fixing bugs in the current version, adding simple modules for special functions not included in the official version, or implementing interfaces to other existing information systems or medical equipment. In the spirit of open-source projects, reasonable additions and modifications can and should be published to the Care2x community Web page.

LESSONS LEARNED

Approximately 100 students from medicine and 25 from HCE participated (Figure 2). The whole lecture was built in the following way.

1. Theoretical foundations of HIS in traditional lectures
2. Principles of Care2x explained (HCE group's lecture was more technology orientated)
3. Familiarization with Care2x in practical sessions
4. Practical work, specific work flows
5. Applying reverse engineering (HCE group only in the second part of the lecture)
6. Examination (both theoretical and practical)

During the education, the students were faced with the following strengths and weaknesses of Care2x.

Strengths:

- everyone can make his or her own tools
- work does not have to be done in a strict order
- very flexible
- easy to handle
- continuing design and development
- open source
- lots of different languages
- big community that takes care of Care2x
- easy to select the different departments and stations

Figure 2. Students at work with Crae2x: we assigned groups of 2-3 students with different tasks related to the administration of a virtual hospital

Weaknesses:

- no real standard between the modules
- documentation is only rudimentary
- a few tools are not really easy to interpret
- lack of security measures
- not a state-of-the-art user interface
- there is no global list of patients from which to select one

OBSTACLES IDENTIFIED

During our lectures and trainings, there emerged several problems while using Care2x. There are a lot of small bugs that caused troubles. The biggest problem was that sometimes the browser responded with an "inactivity error" and the session would time out. Most of the time this error message was shown, the last click had not been made but one single minute ago. The next problem with the handling was that sometimes the back button on the Web browser would lead to nowhere because Care2x does not manage this. Much later, we found out that the back button of the browser is unnecessary because the program has included this function. That did not solve the problem completely: Every now and then the integrated back button of Care2x led to nowhere, too. In addition, some pages did not include the Care2x back button (inconsistency), resulting in a blank page. This required the user to restart at the very beginning and click through all the menus once again, which was boring for the students.

The general software problems that did not concern the running process were not severe. However, there is a serious problem when it is possible to admit a patient to more than one station, or when it is possible to alter a patient's record after his or her death.

A severe problem that has to be solved is that patients have to be discharged and then hospitalized again when we just want them to be transferred from the ambulatory to a station.

There are some translation errors and missing notes. For example, if a new patient record is being applied, there are red stars above some properties. Although this is an obvious sign for experienced users, it is not noted anywhere why these stars appear. The students found out that these stars show the minimum amount of

Figure 3. Example of a graphically embedded complex form, the diagnostic test order

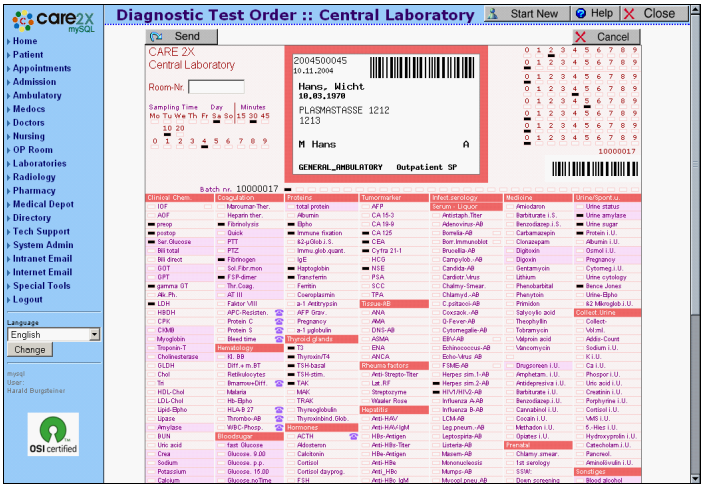
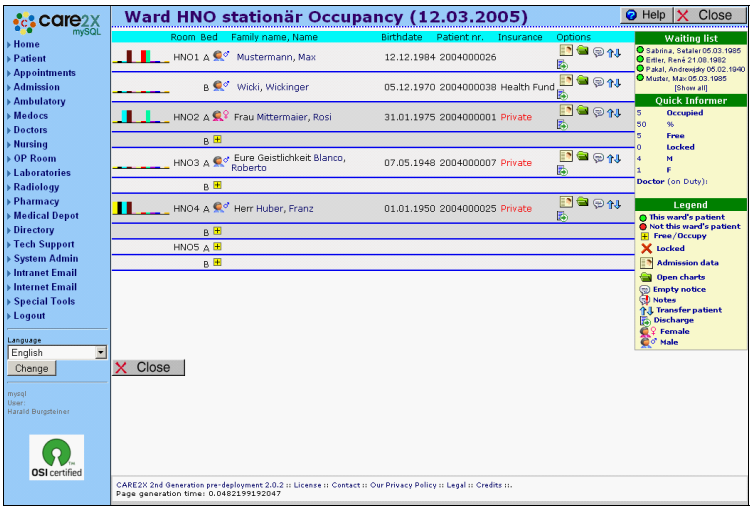


Figure 4. Nursing information about stationary occupancy for one of our virtual wards



data that is required to create a patient record, but how would the students of medicine with little experience in IT know this fact?

It is also sometimes annoying that bits of information are hidden behind a link. For example, if you want to hospitalize a patient, you have to remember the social insurance number because it is not shown in the place it is needed. This is due to the fact that Care2x works with only one window. Sometimes there might just be too little space to provide all the information needed, and then the user has to write this information down or remember it; this cannot be the aim of an HIS.

CONCLUSION

Care2x is flexible open-source software. Although there are some bugs, it has the potential to become functional software to support work flows within a (real) hospital. We think the biggest problems are the documentation and the deduction of treatments. Working with Care2x as a beginner is not very comfortable, and the software is not very intuitive. However, if one trains with Care2x, the work flows become clearer and more logical. The online help of Care2x should be better and more comprehensive. Working with the software was very fun because you really can play with a virtual hospital. Care2x is a very good possibility for training with work flows in a hospital. Further improvement of Care2x will open new areas to work with this software.

REFERENCES

Alpay, L., & Murray, P. (1998). Challenges for delivering healthcare education through telematics. *International Journal of Medical Informatics, 50*(1-3), 267-271.

Bemmel, J. H. v., & Musen, M. A. (2000). *Handbook of medical informatics.* Berlin, Germany: Springer.

Bothe, K. (2001). Reverse engineering: The challenge of large-scale real-world educational

projects. *14th Conference on Software Engineering Education and Training*, 115.

Fieschi, M. (2002). Information technology is changing the way society sees health care delivery. *International Journal of Medical Informatics, 66*(1-3), 85-93.

Haux, R. (1998). Health and medical informatics education: Perspectives for the next decade. *International Journal of Medical Informatics, 50*(1-3), 7-19.

Haux, R. (2002). Health care in the information society: What should be the role of medical informatics? *Methods of Information in Medicine, 41*(1), 31-35.

Haux, R., Hasman, A., Leven, F. J., Protti, D. J., & Musen, M. A. (1997). Education and training in medical informatics. In J. Bemmel, J. V. Bemmel, & M. A. Musen (Eds.), *Handbook of medical informatics* (p. 537ff). Heidelberg, Germany: Springer.

Haux, R., & Knaup, P. (2000). Recommendations of the International Medical Informatics Association (IMIA) on education in health and medical informatics. *Methods of Information in Medicine, 39*(3), 267-277.

Hovenga, E. J. S. (2004). Globalisation of health and medical informatics education: What are the issues? *International Journal of Medical Informatics, 73*(2), 101-109.

Moore, M. B. (1996). Acceptance of information technology by health care professionals. *Symposium on Computers and the Quality of Life*, 57-60.

KEY TERMS

Care2x: An open-source HIS available from http://www.care2x.org/. Care2x is a quite mature and stable product that can be used at least for educational purposes for both students of medicine and students of medical informatics. Some groups report the deployment of enhanced and adopted versions in real hospitals.

Diagnosis-Related Group (DRG): The DRG system is an inpatient classification system based on several factors: the principal diagnosis, secondary diagnosis, surgical factors, age, sex, and discharge status. Under the Medicare prospective payment system, hospitals are paid a set fee for treating patients in a single DRG category, regardless of the actual cost of care for the individual.

Digital Imaging and Communications in Medicine (DICOM): The DICOM image format is commonly used for the transfer and storage of medical images. Visit Chris Rorden's DICOM page for information about the format and free software to view and manipulate it.

Hospital Information System (HIS): It is the central medical information system in most hospitals in which most healthcare-related data (e.g., personnel, stations, patients and their medical history, etc.) are stored.

Medical Informatics: The rapidly developing scientific field that deals with biomedical information, data, and knowledge: their storage, retrieval, and optimal use for problem solving and decision making. The emergence of this new discipline has been attributed to advances in computing and communications technology, to an increasing awareness that the knowledge base of medicine is essentially unmanageable by traditional paper-based methods, and to a growing conviction that the process of informed decision making is as important to modern biomedicine as is the collection of facts on which clinical decisions or research plans are made. (Shortliffe, 1995)

Open Source: The idea of sharing the source code of applications or tools for free. Other people are invited to elaborate on future extensions and improvements. Most open-source projects are committed to one of the Gnu public licenses (see http://www.gnu.org/licenses/licenses.html).

RDBMS (Relational Database Management System): A software package that manages a relational database, optimized for the rapid and flexible retrieval of data. It is also called a database engine.

Reverse Engineering: Taking apart an existing system to analyze smaller or single parts. The reduced complexity simplifies the process of enhancing or understanding its functions.