

Information Systems in Healthcare

— State and Steps towards Sustainability

R. Lenz

Chair for Computer Science VI (Data Management), University of Erlangen-Nuremberg, Germany

Summary

Objectives: To identify core challenges and first steps on the way to sustainable information systems in healthcare.

Methods: Recent articles on healthcare information technology and related articles from Medical Informatics and Computer Science were reviewed and analyzed. Core challenges that couldn't be solved over the years are identified.

Results: The two core problem areas are process integration, meaning to effectively embed IT-systems into routine workflows, and systems integration, meaning to reduce the effort for interconnecting independently developed IT-components. Standards for systems integration have improved a lot, but their usefulness is limited where system evolution is needed.

Conclusions: Sustainable Healthcare Information Systems should be based on system architectures that support system evolution and avoid costly system replacements every five to ten years. Some basic principles for the design of such systems are separation of concerns, loose coupling, deferred systems design, and service oriented architectures.

Keywords

Information system, systems integration, medical informatics applications

Yearb Med Inform 2009: 63-70

1 Potential for Process Improvement

Health information technology is generally considered as a fundamental key to improving healthcare quality (e.g. [1;2]). In fact, numerous studies have demonstrated the potential of information technology to improve the quality of healthcare in various ways. The term "Health Information Technology" (HIT) thereby comprises a diverse set of technologies related to transmitting and processing various kinds of information in the context of the healthcare process. The most prominent functionality covered by the term is probably the Electronic Health Record (EHR), which promises to overcome the shortcomings of paper based health records (e.g. [3]). In particular, the idea of ubiquitously available information is intriguing. In addition to providing patient related information at the point of care, various techniques for decision support have been shown to be effective. Some examples have been summarized in a recent review by Chaudhry et al. on the impact of health information technology on quality, efficiency, and costs of medical care [4]. Most studies contained in this review are aimed at enhancing adherence to guidelines through the use of information technology. Computer-generated reminders and alerts are often used to achieve this effect (e.g. [5-7]). Decision support is often combined with Computerized Provider Order Entry (CPOE), which is also a prominent approach to improve guideline adherence and to reduce the probability of medical errors (e.g. [8]). In addition to the general classification

of approaches addressed in the survey of Chaudhry et al. there is a broad range of functionalities IT can provide to effectively support healthcare processes [9].

- So called "Info buttons" can be used for various purposes, like context sensitive information-retrieval in a clinical information system [10;11]. The simple idea is to use information from a clinical information system as a search key for an external knowledge base, and to minimize the effort for this search to a single mouse-click.
- Computer applications can contribute to improve different aspects of data quality, like completeness [12], timeliness [13], etc. thereby improving the information basis for decision making.
- Computer systems can contribute to better monitoring the current status of a patient, e.g. by presenting patient data in a more coherent way, by providing optimized views to patient data for dedicated purposes, or by generating alerts if some parameter or a combination of parameters is developing into dangerous areas [14].
- Computer systems can calculate drug doses from previously entered data (e.g. age, weight, gender), check compatibility with other medications, and check compatibility with allergies [15;16]. Medication is one of the examples for decision support that can be effectively combined with computerized provider order entry: CPOE systems are estimated to reduce medication errors up to 81% [17]. In a recent survey Ammenwerth et al. com-

pared 25 studies on the effects of electronic prescribing and found a reported reduction of relative risk on medication errors of 13% to 99% [18].

- Computer systems can calculate disease probabilities [19].

This list of examples is far from exhaustive. New helpful functionalities of IT are arising with new technologies such as mobile devices and embedded systems. These include smart homes and ambient assisted living [20;21]. The increasing amounts of information collected in electronic medical records are expected to contain valuable treasures that could also be exploited for clinical research [22;23]. Moreover, bioinformatics research increasingly provides new sources of information that can be combined with existing information to gain new functionality. In a recent survey Kuhn et al. predict that genomics and proteomics data will have a significant impact on preventive, diagnostic, and therapeutic measures [24].

These examples, and many other reports, clearly document proven positive effects of HIT and promise even more. However, Chaudhry et al. also found that the generalizability of success stories is limited [4]. This is coherent with the observation of others. Aarts and Berg found that the same system used in different settings produced different results [25]: The system that was a success in one setting failed in another one. Some authors even described cases in which the use of IT in healthcare fostered errors rather than reduce their likelihood [26-28]. So, what is it that makes IT effective in healthcare? Factors that influence success and failure of IT projects in healthcare have been studied in detail, but the results of these studies are not yet sufficiently considered by those who build systems and standards.

In this paper I argue that we need system architectures that better fit with the often demanding socio-technical perspective on healthcare IT. In particular, an IT system in a healthcare setting that actually supports organiza-

tional learning must be capable of continuously adapting to new requirements. This has an impact on system architectures and also on desirable standards.

Subsequently I briefly summarize the research results concerning success factors and barriers for effectively implementing IT systems in healthcare. From that I will draw some conclusions concerning standards for interoperability, interaction paradigms, and architectural approaches, which might serve as steps to more effective and sustainable IT systems in healthcare.

2 Success Factors and Barriers

Strictly speaking, before discussing success factors one should define what “success” for an information system actually means. According to the D&M information system (IS) success model [29] IS success can be measured based on multiple dimensions, including technical success (e.g. system quality), semantic success (e.g. information quality), individual impact (e.g. user satisfaction), and organizational impact (e.g. overall effectiveness). Aarts and Berg proposed a success model for healthcare information systems, which particularly focuses on medical work practices [25], which corresponds to the individual and organizational impact dimensions of DeLone and McLean’s model. However, the different dimensions are not independent, but interrelated: improving the technical basis of a system may contribute to improving information quality (e.g. availability), and improving information quality certainly contributes to improving individual and organizational impact. Thus, it does make sense to focus on overall effectiveness as an indicator for the general outcome, like Aarts and Berg do, and still do research on a broad spectrum of success factors.

Influential factors for success and failure of IT systems in healthcare have been studied in detail for many years (e.g. [30-32]). All these studies under-

line the importance of a socio-technical understanding of healthcare IT. The findings of these studies are not surprising: from a general perspective, the same set of influential factors is mostly valid for any kind of IT project (e.g. [33-35]). Sauer has identified no less than 50 factors for project failure which can be categorized into different factor classes, comprising user involvement, mutual understanding of project participants, benefit/value of outcome, management commitment, technical design quality and system complexity, project management, system and project performance, resource adequacy, situational stability, and implementation process issues including change management and requirements management [33]. These criteria are not specific for the healthcare domain. However, some of the factors seem to be particularly important in healthcare due to the nature of healthcare processes.

Brender et al. have analyzed the importance of different success factors for IT projects in healthcare in a pilot Delphi study [36]. In this study 110 success factors and 27 failure criteria were distinguished and classified into six groups: functional, organizational, technical, managerial, cultural and legal. Interdependencies between different factors were not considered, since this is new territory, but such interdependencies (e.g. trade-offs between different factors) certainly do play a role and they further complicate matters [33]. Aarts and Berg argue that “... identifying factors that determine success or failure is very difficult because implementation decisions that were completely sensible at the time they were made in hindsight often prove to be constraining”. Yet, enumerating success factors is not useless, as they give us valuable hints of what has to be considered, but they cannot be taken as a recipe which will guarantee success. Moreover, prioritizing success factors is extremely difficult, especially when dealing with more than 100 factors.

Many critical success and failure factors are somehow related to a few core

problem areas, which are clearly identified as challenges in various review articles on decision support, guideline implementation, and healthcare information systems. Kuhn and Giuse identified four core challenges in their review article in 2001 [37]: integration and standardization, human-computer interaction and the structure of data, information, and knowledge, socio-technical and organizational issues, and processes in healthcare. In 2006 Kuhn et al. found that “integration, interoperability and interaction design” are still today’s core problems [38]. Obviously, the core problems have not changed much. They are all somehow related to two central problem areas: systems integration and process integration. Subsequently, I will briefly characterize these two main barriers to effective and sustainable systems in healthcare.

Process Integration

The term “*process integration*” is used here as a generic term for all techniques that help to seamlessly embed information technology into routine work processes. In other words – instead of optimizing an IT system, the socio-technical system in a specific setting as a whole should be optimized.

The problem of insufficiently adapted software is not a problem of healthcare IT alone. It is rather a general problem of software engineering. In [39] Sten and Per Sundblad summarize the problem as “the software is often too complex and too unrelated to the business to change as fast as necessary”. They conclude as a consequence that “software architects must improve their business understanding to help solve the problem” and users “must learn to better communicate to software development teams.” Medical informatics researchers have long recognized the problem. The first three of the so called “Ten Commandments” for effective clinical decision support defined by Bates et al. are aimed at process inte-

gration [40] – the third recommendation makes it explicit: “Fit into the user’s workflow” – the rest of the commandments are concerned with change and project management issues. In 2003 Maviglia et al. write in a “lessons learned” paper about guideline implementation: “... the biggest obstacle to implementing complex automated guidelines that we encountered was with presentation and integration into the clinical workflow” [41].

Software engineers do have answers to this problem. Roughly summarized the main recommendations are: iterative software development with close end user involvement and short iteration cycles with rapid feedback [42]. Agile programming methodologies such as XP [43] even take these principles as their core message. For user interface design [44] and human-computer interaction [45] it is also recommended to have the end user closely involved, and participatory design has been considered by researchers in medical informatics, in particular, as a key success factor [46-48].

An important observation with process integration is that adapting a system to the user’s needs is not a one-time-effort. In [49] Lehman and Belady describe the inevitable need for software evolution in general. They classify programs into several classes: S-type programs („structured“) dealing with a formally specified problem are correct if they fulfill the specification. P-type programs („problem-solving“) try to approximate the solution of a real-world problem which might change over time, so the program must be adapted to a changing specification. Healthcare Information Systems are so-called E-type programs (“embedded“): The program is part of the world that it models. Software of this kind is inherently evolutionary. Both, changes in the software or changes in the environment affect each other.

Both participatory design and continuous evolution, does not fit well with the interests of healthcare IT vendors who want to sell a single software prod-

uct over and over again. So called COTS (commercial off-the-shelf) systems are affordable but not tailor-made for a specific setting. Of course, today’s commercially available healthcare systems are adaptable to different settings to some degree, but they are not tailor-made. This does not mean that commercial systems cannot improve healthcare quality. Some successful implementations of commercial systems have been described in [50-52]. Evaluating and comparing these systems is difficult [53;54]. The few existing results however, indicate that even the successful commercial systems cannot reach the user satisfaction of tailor-made systems [54]. Chaudhry et al. write in their survey that most of the success stories that fulfilled the criteria of the survey stem from few hospitals where tailor-made systems were adapted with high effort to the specific needs of the specific setting. Interestingly, those hospitals typically also have a large IT staff which put a very high effort in continuously maintaining and adapting the system according to changing demands.

To make the success stories more generalizable, and to achieve both affordable and effective systems, it is most important to reduce the effort needed for demand driven system evolution as much as possible.

Systems Integration

The inability of IT systems to continuously adapt to site-specific and changing requirements is aggravated by the heterogeneity of system components. In the long run, sustainable process integration can only be achieved if IT-systems are actually aimed at supporting processes. Computer applications, however, are typically aimed at supporting certain organizational units or dedicated functionalities rather than healthcare processes, which are interdisciplinary, cross-organizational, and increasingly inter-institutional in their nature. Thus, process support has a lot to do with systems integration.

Ideally, a truly integrated system would comprise a single central database with a carefully designed database schema, which serves as a platform for different applications that share common data. This would help to reduce uncontrolled data redundancy (“*one fact in one place*” [55]) and help to improve data consistency. Healthcare systems, however, are very complex; so, a single database approach would lead to systems that might be difficult to maintain. Adapting a complex database schema with thousands of interrelated tables to new requirements can also become difficult, and software entropy will eventually turn even a well managed system into a legacy system that should be replaced [49;56]. As soon as multiple independent institutions are involved in the healthcare process, a single database approach is completely unrealistic. Therefore, independently developed IT components must somehow be integrated to support cross-organizational and inter-institutional processes. Autonomy of system components and integrity of data are conflicting goals. Neither can be fully achieved without compromising the other one to some degree [57]. Heterogeneity on various levels is a consequence of design autonomy.

The second essential precondition to achieve affordable and effective systems is to reduce the effort needed for bridging these various kinds of heterogeneity. Standards are essential to achieve this. In the next section the current state of standards for healthcare IT is briefly sketched.

3 Standards

Different standards are needed for different kinds of integration requirements. In [58] integration requirements are classified into functional integration and data integration. In addition, technical and semantic integration are distinguished. Technical or syntactic integration issues like bridging different hardware and operating systems or

mapping different syntactic data representations are important in every integration project but they are not specific for the healthcare domain. Thus, without questioning the importance of XML and general integration middleware, only standards for semantic integration are considered in this paper.

Semantic integration requires some consensus on the meaning of commonly used data and functions. Standards for data integration have to be further distinguished in “*type level*” and “*instance level*” standards. System implementers need a semantic reference on a type level, which helps them to design their database schemas to cover all the required information. HL7 and DICOM are the most prominent standards that particularly serve this purpose. Users require a semantic reference on an instance level to fill these databases with commonly used terms. ICD and SNOMED are examples for such standards. The important difference between these two kinds of standards is that type level standards are needed at design time, whereas instance level standards are (primarily) needed at runtime when the system is used and databases are filled with content. So, for building semantically interoperable system components, type level standards such as HL7 are of predominant importance as a semantic reference. HL7 V2.x has been extraordinarily successful in this role. Yet, it has reached its limits [59]: There is no common information model or ontology in HL7 V2.x – instead, the semantic reference in HL7 V2.x is hidden in various message type specifications. This makes the process of developing new message types that are consistent with the existing ones very difficult. Interaction based on messages, which is the fundamental interaction paradigm for HL7 V2.x, also has some disadvantages: messages are transient and they typically need a sender and a recipient, which are more or less directly coupled via appropriate interfaces. Typically, before it is possible to send the first message, it is necessary to explicitly establish a communication

channel. In regional, national or even international healthcare networks, we need loosely coupled components: it should be possible to have an information exchange among system components which have not been previously interconnected via some message based interface. A document-based approach seems much more suitable for this purpose. In contrast to transient messages, electronic documents are assumed to be persistent, they can exist independently from the system that created them, and they carry their own context information. Moreover, the document based interaction paradigm fits well with traditional work practices in healthcare.

HL7 V3 is intended to eliminate these disadvantages of V2. The core of V3 is the Reference Information Model (RIM) which serves as a common information model for all message types. New contents can be consistently derived from a small number of generic core classes, so semantic scalability has been considered from the beginning. The term “*semantic scalability*” has some more important facets: it not only refers to the ability of consistently extending the standard, but also to the ability of RIM-based systems to “understand” even new message types, as the RIM defines an unambiguous semantics for each data element, no matter in which message type it appears. This also offers the opportunity to use the same RIM as the semantic basis for document-based standards, such as the Clinical Document Architecture (CDA) [60;61]. Mentionable at this point is that the three-level-structure of CDA supports another form of semantic scalability: CDA-based document exchange is not limited to fully RIM-based systems, because the header information is already sufficient to contextually describe a document. This makes it possible to have a broad range of systems participate in an exchange of CDA documents in an early stage and then increase semantic interoperability step by step.

So far only data integration issues have been considered. Yet, functional integration is equally important. *Func-*

tional semantic integration requires consensus on the meaningful interaction of system components. The need for standards supporting functional integration is often neglected. Usually, only standards for data interchange are mentioned when semantic interoperability is discussed. However, as soon as multiple independently developed components are supposed to interact in order to cooperatively provide some higher level functionality, a meaningful interaction protocol must be ensured somehow. IHE provides such a framework for the healthcare domain [62]. The standard specifies how different roles interact on the basis of existing standards for data interchange in order to cooperatively provide some higher level functionality. With “cross-enterprise document sharing” (XDS), IHE is expanding its scope on inter-institutional exchange of electronic documents. In its current state, XDS requires a single central document registry, based on ebXML methodology, that mediates accesses stored in multiple document repositories. The necessity of such a central component might be a disadvantage if decentralized environments like national or even international networks are the target. Though the specifications are not yet mature, XDS might be a first step towards a basic infrastructure for distributed document based electronic health records.

4 Steps Towards Sustainability

Affordable and sustainable IT systems in healthcare should prevent the necessity of costly system replacements every five to ten years. This does not mean that we should strive towards systems that last for ages. It means that the necessity to replace or modify some functionality or feature, or component, should not eventually lead to the necessity to replace the whole system. This can only be achieved if both initial goals formulated in section two are considered equally:

1. Process integration requires close end user involvement. So we need system architectures that are flexible enough to support a demand driven system evolution.
2. Heterogeneous system components must be coupled in an adequate way to support inter-organizational healthcare processes. Since autonomous systems are unavoidable, we need ways for loosely coupling these systems that preserve their autonomy.

So, what are the principles for designing evolutionary system architectures? Are the currently available standards for systems integration sufficient to support such evolutionary system architectures or are they rather obstructive in this respect? Generally, standards for semantic interoperability tend to freeze knowledge to some degree – Evolutionary systems need the opposite: flexibility. Some basic principles for evolutionary information systems are briefly sketched below.

Separation of Concerns

Separating different concerns for different responsibilities is an important principle in order to reduce the com-

plexity of a system and to improve its maintainability. Data security is an example for such a concern, which is particularly important for healthcare information systems. An effective security infrastructure is an important precondition for communicating personal patient related data across a network. Yet, like in every other modularized system, the security infrastructure should be kept flexible enough to cope with newly arising security challenges. Thus, generic security services should be provided to be used in arbitrary applications, in order to avoid the need to adapt a systems code for security reasons. In [63] Blobel and Roger-France show how a security infrastructure can be decomposed into a hierarchy of security services.

Such a layered approach with different levels of abstraction is a frequently used technique for modularization and reduction of complexity. The basic idea is that each layer provides functions and services to support the next upper layer. In [64], this principle is applied to the functionality of a hospital information system to motivate a layered architecture for adaptive Healthcare Information Systems. The general idea is described [58] and illustrated in Fig. 1.

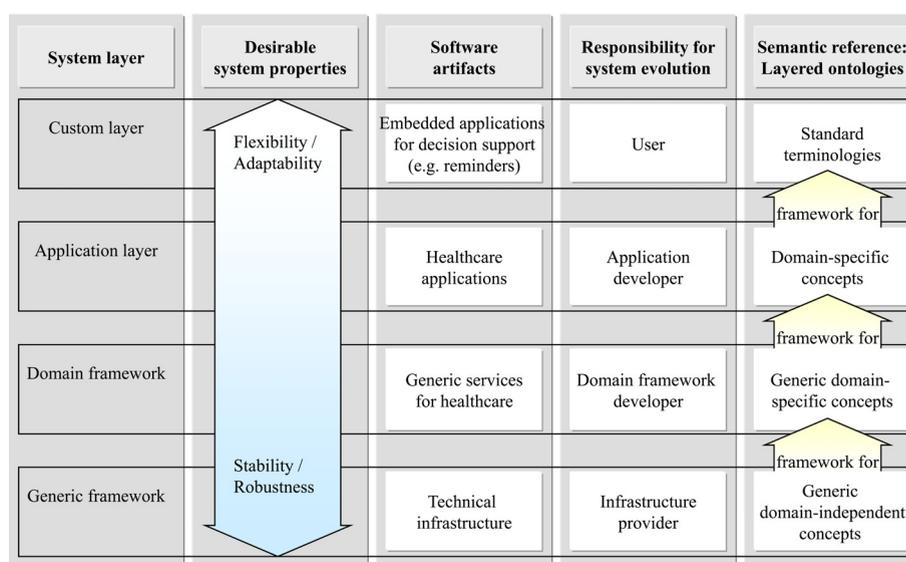


Fig. 1 A layered approach for evolutionary IT systems in healthcare [58].

The different layers can be seen as a consequence of the fact that different groups of people are the drivers for different aspects of system evolution. The evolution of healthcare domain specific concepts should be in the hands of domain experts rather than IT experts. This principle already motivated the concept of *archetypes* that has been developed in the context of openEHR to separate the definition of semantic constraints for electronic health records from the definition of the core database schema for health record systems [65;66].

Deferred Systems Design

Another fundamental principle for evolutionary systems is *deferred systems design* [67]. Semantic decisions that are frozen in a database schema are hard to revise. So decisions that can be deferred from design-time to deploy-time do improve adaptability. Decisions that can be even further deferred to run-time improve continuous adaptability. The vision behind this is the idea of action based IS development, where the system is shaped during social action. A layered approach, as sketched above, supports deferred systems design, since semantic decisions on higher levels are independent from semantic decisions on lower levels. The lower levels just provide a generic semantic framework that helps to reduce semantic heterogeneity on the higher levels.

Loose Coupling

Tightly coupled system components are difficult to replace and they do not fit well into a distributed environment with autonomous organizational units. Loose coupling, in contrast, could improve scalability and maintainability. In this position paper, the term “loose coupling” does not stand for a specific type of system architecture. It rather represents a number of desirable properties for evolutionary IT-systems:

- It should be possible to add an application to the system without the need to modify other applications in any way.
- It should be possible to remove an application from the system without affecting other applications.
- Information interchange between two components should be possible without previously interconnecting these systems explicitly.

In healthcare networks, these properties are highly desirable in order to enable ad hoc communication between arbitrary, previously unconnected parties.

Service Oriented Architectures

Layered system architectures require very clearly defined interfaces for each layer. Actually, the specification of a layer is primarily a functional specification of the interface rather than a specification of data semantics (which is also important, though). Thus, independent evolution of these layers would require a standardized functional specification of each layer. Service oriented architectures support the decoupling of layers. Yet, the concept would require a standardization of services. Such functional frameworks are rare in the healthcare domain so far. IHE is an example for a functional specification, but it is unclear whether XDS is actually suitable as a low level domain layer for a decentralized document based healthcare information system, because it requires a central registry. An enhanced approach supporting federated clinical affinity domains is suggested in [68].

Another example is OMGs healthcare domain taskforce, formerly known as CORBAmed, who have defined healthcare related basic services like Person Identification Service (PIDS), Terminology Query Service (TQS) and Clinical Observation Access Service (COAS). Such services, however, require a shared middleware. It is unclear how this fits into a large scale distributed and decentralized environment with loosely coupled components.

Generally speaking, semantically rich service interfaces do not fit well with the idea of loosely coupled components. It seems to be a good idea to reduce procedural interfaces to a minimum set of necessary services. A new architectural style, based on this idea, called REST (Representational State Transfer) [69;70] seems to be particularly suitable for emergent system design. Yet, it is difficult to predict whether such a technique will prove flexible enough to meet the challenges of evolutionary information systems in healthcare. It is the task of informatics and medical informatics research to find out.

5 Conclusion

IT systems in healthcare are increasingly required to support inter-institutional processes. Autonomous system components will be part of these systems. Core challenges in this context are to preserve systems autonomy and still achieve systems integration and process integration, which are aggravated by continuously changing requirements. Standards that fit with the idea of evolutionary information systems are required.

Current standards for system integration in healthcare have improved a lot, as they are beginning to consider various aspects of semantic scalability. Also document standards for information interchange, such as CDA, are increasingly supported. To support a continuous system evolution, however, a layered architecture with stable and robust basic services would be highly desirable. It is still unclear whether current developments are actually suitable for this or if new standards are needed that explicitly consider such a layering. It is also unclear how inter-institutional processes are best supported on the basis of a fully decentralized document-oriented information exchange.

Continuous user-centred adaptation requires flexibility. Thus, process integration is supported by IT architectures

which allow flexible and ad hoc specification of process context and information interchange. A first idea to combine this with the decentralized document based approach is to include process related information into CDA documents, which means passive containers of information are turned into active documents which are essential parts of inter-institutional processes. The idea of process aware document oriented information systems is motivated out of the core challenges process integration and systems integration. Yet, many problems remain unsolved and a lot of research has to be done before an idea turns into a mature technology.

Acknowledgements

I would like to thank Christoph Neumann and Frank Lauterwald for their valuable hints to improve this manuscript. I also thank the anonymous reviewers for their helpful comments.

References

- Institute of Medicine. *To Err Is Human. Building a Safer Health System*. National Academy Press; 2000.
- Institute of Medicine. *Crossing the Quality Chasm: A New Health System for the 21st Century*. National Academy Press; 2001.
- Shortliffe EH, Cimino C. *Biomedical Informatics - Computer Applications in Healthcare and Biomedicine*. 3rd ed. Springer; 2006.
- Chaudhry B, Wang J, Wu S, Maglione M, Mojica W, Roth E, et al. Systematic review: impact of health information technology on quality, efficiency, and costs of medical care. *Ann Intern Med* 2006 May 16;144(10):742-52.
- McDonald CJ. Protocol-based computer reminders, the quality of care and the non-perfectability of man. *N Engl J Med* 1976 Dec 9;295(24):1351-5.
- McDonald CJ, Hui SL, Smith DM, Tierney WM, Cohen SJ, Weinberger M, et al. Reminders to physicians from an introspective computer medical record. A two-year randomized trial. *Ann Intern Med* 1984 Jan;100(1):130-8.
- Kucher N, Koo S, Quiroz R, Cooper JM, Paterno MD, Soukonnikov B, et al. Electronic alerts to prevent venous thromboembolism among hospitalized patients. *N Engl J Med* 2005 Mar 10;352(10):969-77.
- Teich JM, Spurr CD, Schmitz JL, O'Connell EM, Thomas D. Enhancement of clinician workflow with computer order entry. *Proc Annu Symp Comput Appl Med Care* 1995;459-63.
- Lenz R, Reichert M. IT Support for Healthcare Processes - Premises, Challenges, Perspectives. *Data & Knowledge Engineering* 2007;61(1):39-58.
- Cimino J. Infobuttons: anticipatory passive decision support. *AMIA Annu Symp Proc* 2008;1203-4.
- Collins BJ, Speedie SM. Attaching Context Sensitive Infobuttons to an EHR Options and Issues. *AMIA Annu Symp Proc* 2008;914.
- Hogan WR, Wagner MM. Accuracy of Data in Computer-based Patient Records. *Journal of the American Medical Informatics Association* 1997;4(5):342-54.
- Scannapieco M, Missier P, Batini C. Data Quality at a Glance. *Datenbank Spektrum* 2005;14:6-14.
- Chen HT, Ma WC, Liou DM. Design and implementation of a real-time clinical alerting system for intensive care unit. *Proc Amia Symp* 2002;131-5.
- Ambrisko TD, Nemeth T. A computer program for calculation of doses and prices of injectable medications based on body weight or body surface area. *Can J Vet Res* 2004 Jan;68(1):62-5.
- Muller T. Typical medication errors in oncology: analysis and prevention strategies. *Onkologie* 2003 Dec;26(6):539-44.
- Koppel R, Metlay JP, Cohen A, Abaluck B, Localio AR, Kimmel SE, et al. Role of computerized physician order entry systems in facilitating medication errors. *JAMA* 2005 Mar 9;293(10):1197203.
- Ammenwerth E, Schnell-Inderst P, Machan C, Siebert U. The Effect of Electronic Prescribing on Medication Errors and Adverse Drug Events: A Systematic Review. *J Am Med Inform Assoc* 2008 Sep;15(5):585-600.
- Hejlesen OK, Olesen KG, Dessau R, Beltoft I, Trangeled M. Decision support for diagnosis of lyme disease. *Stud Health Technol Inform* 2005;205-10.
- Demiris G. Electronic home healthcare: concepts and challenges. *Int J Electron Healthc* 2004;1(1):4-16.
- Demiris G, Hensel BK. Technologies for an aging society: a systematic review of "smart home" applications. *Yearb Med Inform* 2008;33-40.
- Prokosch HU, Ganslandt T. Perspectives for medical informatics. Reusing the electronic medical record for clinical research. *Methods Inf Med* 2009;48(1):38-44.
- Haux R. Health information systems - past, present, future. *Int J Med Inform* 2006 Mar;75(3-4):268-81.
- Kuhn KA, Giuse DA, Lapao L, Wurst SH. Expanding the scope of health information systems - from hospitals to regional networks, to national infrastructures, and beyond. *Methods Inf Med* 2007;46(4):500-2.
- Aarts J, Berg M. Same Systems, Different Outcomes. *Methods Inf Med* 2006;45(1):53-61.
- Ash JS, Berg M, Coiera E. Some unintended consequences of information technology in health care: the nature of patient care information system-related errors. *J Am Med Inform Assoc* 2004 Mar;11(2):104-12.
- Campbell EM, Sittig DF, Ash JS, Guappone KP, Dykstra RH. Types of Unintended Consequences Related to Computerized Provider Order Entry. *J Am Med Inform Assoc* 2006;13(5):547-56.
- Ash JS, Sittig DF, Dykstra R, Campbell E, Guappone K. The unintended consequences of computerized provider order entry: Findings from a mixed methods exploration. *Int J Med Inform* 2009 Apr;78 Suppl 1:S69-76.
- DeLone WH, McLean ER. The DeLone and McLean Model of Information Systems Success: A Ten-Year Update. *Journal of Management Information Systems* 2003;19(4):9-30.
- Tierney WM, Overhage JM, McDonald CJ. Computerizing guidelines: factors for success. *Proc AMIA Annu Fall Symp* 1996;459-62.
- Ash JS, Stavri PZ, Dykstra R, Fournier L. Implementing computerized physician order entry: the importance of special people. *Int J Med Inf* 2003 Mar;69(2-3):235-50.
- Lorenzi NM, Smith JB, Conner SR, Campion TR. The Success Factor Profile(c) for Clinical Computer Innovation. *Medinfo* 2004;1077-80.
- Sauer C. Deciding the future for IS failures: not the choice you might think. In: Currie WL, Galliers B, editors. *Rethinking Management Information Systems*. Oxford University Press; 1999. p. 279-309.
- Al-Mashari M, Zairi M. BPR implementation process: an analysis of key success and failure factors. *Business Process Management Journal* 1999;5(1):87-112.
- Luftman JN, Papp R, Brier T. Enablers and Inhibitors of Business-IT Alignment. *Communications of the Association for Information Systems* 1999;1(11).
- Brender J, Ammenwerth E, Nykanen P, Talmon J. Factors influencing success and failure of health informatics systems—a pilot Delphi study. *Methods Inf Med* 2006;45(1):125-36.
- Kuhn KA, Giuse DA. From hospital information systems to health information systems. Problems, challenges, perspectives. *Methods Inf Med* 2001;40(4):275-87.
- Kuhn KA, Wurst SH, Bott OJ, Giuse DA. Expanding the scope of health information systems. Challenges and developments. *Yearb Med Inform* 2006;43-52.
- Sundblad S, Sundblad P. Business Improvement Through Better Software Architecture. *Microsoft Architecture Journal* 2007;10(January 2007):36-41.
- Bates DW, Kuperman GJ, Wang S, Gandhi T, Kittler A, Volk L, et al. Ten commandments for effective clinical decision support: making the practice of evidence-based medicine a reality. *J Am Med Inform Assoc* 2003 Nov;10(6):523-30.
- Maviglia SM, Zielstorff RD, Paterno M, Teich JM, Bates DW, Kuperman GJ. Automating complex guidelines for chronic disease: lessons learned. *J Am Med Inform Assoc* 2003 Mar;10(2):154-65.
- Versteegen G. *Projektmanagement mit dem Rational Unified Process*. Springer; 2000.
- Beck K. *Extreme Programming*. Addison-Wesley; 2000.
- Preece J, Rogers Y, Sharp H. *Interaction Design - Beyond Human Computer Interaction*. Addison-Wesley; 2002.
- Carver L, Turoff M. *Human-Computer Interaction: The Human and Computer as a Team in Emergency Management Information Systems*. Communications of the ACM 2007;50(3):33-8.
- Timpka T, Sjoberg C, Hallberg N, Eriksson H, Lindblom P, Hedblom P, et al. Participatory design of computer-supported organizational learning in health care: methods and experiences. *Proc Annu Symp Comput Appl Med Care* 1995;800-4.
- Sjoberg C, Timpka T. Participatory design of information systems in health care. *J Am Med Inform Assoc* 1998 Mar;5(2):177-83.

48. Lenz R, Blaser R, Beyer M, Heger O, Biber C, Bäumlein M. IT-support for clinical pathways - lessons learned. *Int J Med Inf* 2007;76:397-402.
49. Lehman MM, Belady LA. *Program Evolution - Processes of Software Change*. Academic Press; 1985.
50. Lechleitner G, Pfeiffer KP, Wilhelmy I, Ball M. Cerner Millennium: the Innsbruck experience. *Methods Inf Med* 2003;42(1):8-15.
51. Gell G, Schmücker P, Pedevilla M, Leitner H, Naumann J, Fuchs H, et al. SAP and partners: IS-H and IS-H* MED. *Methods Inf Med* 2003;42(1):16-24.
52. Kuhn KA, Lenz R, Elstner T, Siegele H, Moll R. Experiences with a generator tool for building clinical application modules. *Methods Inf Med* 2003;42(1):37-44.
53. Ammenwerth E, de Keizer N. A viewpoint on evidence-based health informatics, based on a pilot survey on evaluation studies in health care informatics. *J Am Med Inform Assoc* 2007 May;14(3):368-71.
54. Blaser R, Kuhn KA, Overath M, Dietz F, Opitz E, Lenz R. Towards responsive IT-infrastructure - assessment of a health information system. *Medinfo* 2004;2004:1178-82.
55. Date CJ. *An Introduction to Database Systems*. 6 ed. Addison-Wesley; 1995.
56. Parnas DL. *Software Aging*. Proceedings of ICSE, Sorento, Italy 1994;16:279-87.
57. Lenz R, Kuhn KA. Intranet meets hospital information systems: the solution to the integration problem? *Methods Inf Med* 2001 May;40(2):99-105.
58. Lenz R, Beyer M, Kuhn KA. Semantic Integration in Healthcare Networks. *Int J Med Inform* 2007;76(2-3):201-7.
59. Mead CN. Data interchange standards in healthcare IT—computable semantic interoperability: now possible but still difficult, do we really need a better mousetrap? *J Healthc Inf Manag* 2006;20(1):71-8.
60. Dolin RH, Alschuler L, Beebe C, Biron PV, Boyer SL, Essin D, et al. The HL7 clinical document architecture. *J Am Med Inform Assoc* 2001 Nov;8(6):552-69.
61. Muller ML, Uckert F, Burkle T, Prokosch HU. Cross-institutional data exchange using the clinical document architecture (CDA). *Int J Med Inform* 2005 Mar;74(2-4):245-56.
62. Vegoda P. Introducing the IHE (Integrating the Healthcare Enterprise) concept. *J Healthc Inf Manag* 2002;16(1):22-4.
63. Blobel B, Roger-France F. A systematic approach for analysis and design of secure health information systems. *Int J Med Inf* 2001;62(1):51-78.
64. Lenz R, Kuhn KA. Towards a continuous evolution and adaptation of information systems in healthcare. *Int J Med Inf* 2004;73(1):75-89.
65. Beale T. Archetypes and the EHR. *Stud Health Technol Inform* 2003;96:238-44.
66. Beale T. Archetypes: Constraint-based Domain Models for Future-proof Information Systems. OOPSLA 2002.
67. Patel N. *Adaptive Evolutionary Information Systems*. Idea Group Publishing; 2003.
68. Dogac A, Laleci GB, Aden T, Eichelberg M. Enhancing IHE XDS for federated clinical affinity domain support. *IEEE Trans Inf Technol Biomed* 2007 Mar;11(2):213-21.
69. Fielding RT. *Architectural Styles and the Design of Network-based Software Architectures* University of California; 2000.
70. Khare R, Taylor RN. Extending the Representational State Transfer (REST) Architectural Style for Decentralized Systems. *International Conference on Software Engineering* 2004;428-37.

Correspondence to:

Prof. Dr.-Ing. Richard Lenz
 Universität Erlangen-Nürnberg
 Department Informatik
 Lehrstuhl für Informatik 6 (Datenmanagement)
 Martensstraße 3
 D-91058 Erlangen
 Germany
 Tel: +49 9131 85 27899
 Fax: +49 9131 85 28854
 E-mail: Richard.Lenz@informatik.uni-erlangen.de