Building a Hospital Information System: Design Considerations Based on Results from a Europe-wide Vendor Selection Process

Klaus A. Kuhn, Richard Lenz, Rainer Blaser
Institute of Medical Informatics, Philipps-University Marburg, Germany

A number of research and development projects in the U.S. and in Europe have shown that novel technologies can open significant perspectives for hospital information systems (HIS). The selection of software products for a HIS, however, is still nontrivial. Generalist vendors promise a broad scope of functionality and integration, while specialist vendors promise elaborated and highly adapted functionality. In 1997, the university hospital Marburg, a 1,250 bed teaching hospital, decided to introduce a new large-scale HIS. The objectives of the project included support of clinical workflows, cost effectiveness and a maximum standard of medical care. In 1997/98 a formal Europe-wide vendor contest was performed. 15 vendors, including several from the U.S., participated. Systems were checked against the hospital's objectives, functionality, and technological criteria. One of the results of both technology and market assessment was the identification of fundamental technological and design aspects strongly influencing functionality and flexibility.

INTRODUCTION

In hospitals, in the U.S. as well as in Europe, the paradigm shift from administrative computing towards clinically centered, new information infrastructures has reached considerable influence and speed. The potential of advanced information processing has been identified. New technologies are being utilized in development projects covering a broad range of aspects. Nevertheless, the process of selecting software products to build a HIS remains a nontrivial task. There are still generalist (total system) vendors and specialist (niche) vendors, both with their advantages and problems. Generalists promise to sell fully functional and complete HISs, but not all real world installations are confirming this. Specialists offer systems with a highly adapted functionality, but they bring their built-in integration problems with them. Specialists argue in favor of the "best of breed" approach, generalists argue with a broad range of functionality and inherent benefits of an integrated approach. Costs are high, and so are the expectations of users - but often disappointment is predictable. Organizational impacts are massive, as medicine, data processing, and organization are closely interwoven. Typically, a hospital or its consultant will have no problems to formulate thousands of selection criteria, but there may be less than 20 vendors to meet them. The university hospital Marburg, a 1,250 bed academic teaching hospital, performed such a formal selection process in 1997/98. Both technology and market were assessed. A central result was the identification of fundamental technological and design aspects influencing or determining functionality and flexibility. After a short look at the market, we will report how key challenges and objectives can be supported by today's software products and by some of the most important current technologies.

PROJECT OBJECTIVES AND METHODS

The university hospital of Marburg performed an analysis of drawbacks of its existing HIS in 1997, which resulted in the detection of typical problems such as heterogeneity, the millenium problem, legacy technologies, and a lack of clinical functionality. The board of directors decided to introduce a new large-scale HIS with software for patient data management (ADT etc.), for business/financial purposes, and for clinical computing (for simplification, we will refer to these as "main parts").

A total quality management project running since 1995 had major influence on the project's objectives, among which a maximum standard of medical care and cost effectiveness were basic goals. The system should support flexible documentation for clinical and administrative/legal purposes, interdepartmental information flow and clinical processes. Organizational impacts have been carefully considered: functions shifting work from one person or personnel group to another should not be realized without "compensatory" functions offering benefits. All potential user groups, including clinicians, nurses, and administrators were intensively involved in the selection process. Business process analyses are being performed.

Systematic contacts to about 30 vendors offering at least one of the three main parts took place in 1997/98. A formal Europe-wide selection process was carried out; 15 vendors, including several from the U.S. participated. Criteria lists, consisting of about 3000 application-specific and about 400 technical items were designed, used, and evaluated. During criteria evaluation and tech-
Technology assessment, central objectives were identified. Currently, the selected software products are being introduced.

TECHNOLOGY AND MARKET ASSESSMENT

The market situation. In our market analysis, we considered 9 generalist vendors offering the mentioned "main parts" of a HIS. Not all of them sell fully integrated solutions: five systems contained internal interfaces. Limited clinical functionality was a typical problem of the generalists. They show a tendency to extend their scope, however, and to offer further functionality which traditionally resided in separate departmental subsystems. Examples are newly integrated modules for laboratories, radiology, clinical workstations, and picture management. The concept of a centralized database (which does not necessarily exclude distribution) and a global database schema play a major role for the generalist vendors.

The most significant advantage of specialized subsystems is their high adaptation to the users' needs in the domain they were built for. Typically, they offer built-in domain-specific knowledge, e.g. a specific terminology and direct support for specific processes. There are different kinds of specialist vendors: Very high specialization, e.g. on data management for a specific device or modality, is one extreme, the other is "specialization" on a whole class of functionality, such as document management or the complete functional spectrum of a health care professional workstation. Thus, a HIS can be assembled from a variable number of components. The selection of a "best of breed" product can be performed within a narrow, e.g. a department-specific, or within a broad scope.

The order in which functionality should be introduced is also variable, but to a much smaller degree. Our order will roughly be (1) a patient database with software for ADT, basic organizational tasks, and (legal) documentation, together with business/financial modules, (2a) flexible documentation with data entry, data management, presentation, and data reuse, (2b) broad availability of data, information flow over systems' borders, (3) support of more complex organizational aspects such as scheduling and order entry.

In this situation we found it mandatory to elaborate how autonomy and heterogeneity can be handled, how flexibility and adaptability can be realized, and what technical basis is offered by each system.

Key challenge #1: Heterogeneity and autonomy. As long as the "best of breed" principle for specialized systems is used, as dedicated subsystems are introduced according to the rapidly emerging medical needs, and as medical devices (e.g. cardiac catheter systems) come with their own databases, heterogeneity is a key problem of HIS. It comprises two main aspects: Software products may differ in functionality, presentation, and terminology, and the data may differ in representation and semantics. System autonomy is at the roots of heterogeneity, where at least design autonomy (own data model, query language, semantic interpretation of data, functions etc.) and execution autonomy (execution order of local operations not controlled by foreign system) have to be distinguished.

Key objective #1: Integration of data. Integration covers different aspects, like ubiquitous data access, consistency, and a single system image. We will restrict the discussion to the levels of data, function, and presentation, and start with the integration of data: in order to combine heterogeneous component systems into one coherent system, different database schemas as well as the database contents have to be integrated. Several approaches are possible. The global schema approach, which means a combination of schemas into one integrated schema, is sometimes postulated for HIS, but will rarely be used in its pure form for real world systems. The federated approach, where an export schema is provided by each local database to be shared with other databases, while an import schema describes information from remote nodes, is a looser form of coupling and closer to HIS reality. Multidatabase language approaches, which involve even looser coupling, play no significant role in HIS. The loosest coupling can be subsumed under the term of interoperable systems. Ideally, interoperating systems should be able to use each others' functionality. The approach is often used in a very limited form, where communication is left to standard protocols for data exchange under control of the application.

Integration approaches offered on the market by generalist and specialist vendors typically combine limited forms of interoperability (standardized communication via interfaces) with limited "federation": data from other systems are locally replicated and/or additional results databases are introduced. These approaches have two drawbacks in common: data are always inconsistent to some degree, and functionality among (autonomous) systems differs. Standards such as HL7 and the widely used concept of communication servers relieve the management of interfaces, but, in all cases, consistency, integrity, functional and presentation integration are not directly supported. Instead, communication tends to follow the producer-consumer paradigm: it is considered sufficient that data produced elsewhere are mainly consumed, i.e. read. The data flow in the opposite direction is comparatively small: minimal consistency is achieved by distributing some central patient data among producer systems.
In summary, data integration is possible to a certain degree by means of the principles described above (interfaces, limited interoperability, limited federation). This will result in additional costs: Interfaces have to be created and managed, replication has to be realized and handled. Partially and temporarily inconsistent data may be negligible to some degree in real-world systems. To which degree, however, is not really clear - an inconsistency in medicine will always bear a risk. Delayed reporting of a result may be adequate in one case, but fatal in another one.

Data inconsistencies can be avoided if different applications are built upon a common database. As stated before, some generalist vendors offer such an integrated solution. Having a look at data management within these systems, we found that a careful schema design is of high importance for the evolution of a system. A related aspect is that user-defined changes influencing a database schema are non-trivial to handle, as persistency over new releases has to be granted.

**Key challenge #2: Flow of information and interdepartmental support of organization.** Many persons from different personnel groups are involved in a patient’s diagnostic and treatment process. They share information, add and alter data, and they use applications with overlapping functionality. The consumer-producer metaphor does describe major parts of clinical interaction, but an extension to an information workflow metaphor is more realistic: e.g. a document may have to be edited and altered several times at different locations.

Administrative and clinical information flows are dynamic, interdepartmental, and highly related to organization. Therefore, clinical personnel has to be relieved from organizational tasks such as scheduling, phone calls, keeping track of orders etc. As a consequence, applications will have to support organizational aspects (e.g. scheduling, handling of worklists), and information/document flow (e.g. flow of problem lists or progress notes).

**Key objective #2: Functional and process integration.**

The basic aspect of data integration needs to be extended. As soon as computer applications try to support yet simple workflows over systems’ borders, autonomous interfaced systems will have difficulties not to pass data only, but to continue an interaction started in the sending system: functionality of the interacting systems has to be integrated. Functions themselves are embedded in processes. Central aspects of processes are the flow of information and the flow of control: e.g. control may be passed from a ward to a lab and back in a scheduling process, which is accompanied by some flow of information. Business process modeling tools are used to model processes under various aspects such as organization, functions, control and data flow; some tools offer simulation features. Often a formal language is used, which is helpful for a better understanding among persons with different professional backgrounds. The approach is of considerable importance, when processes are to be reorganized, and when new information systems are being introduced. The step from modeling to real workflow support is offered by workflow management systems. Unfortunately, although they are ideal in theory, they still have severe drawbacks for clinical environments.

Examination of the market shows that the ideal solution for functional integration, which could be offered by vendor-independent reusable and generic services, is not yet available. The technological concepts of distributed object management will be helpful, however, and we strongly recommend to check a vendor’s perspectives of utilizing them and thus trying to avoid future legacy systems.

To date, functional integration over systems’ borders is rudimentary, and process integration is even weaker. The necessary flow of data is difficult to realize with today’s approaches of limited federation and limited interoperability. Either a higher degree of database coupling or real software interoperability are needed. The first alternative is more realistic, as it becomes feasible under a centralized database concept offered by some vendors. The second alternative, the utilization of new middleware concepts, is in a very early stage of realization. Typically, the flow of control is implicitly represented, i.e. hidden in the program code, today. Explicit control mechanisms should be provided, which also could become possible with a new software design offering the necessary levels of abstraction (see below).

**Key challenge #3: Adaptability and flexibility.**

The various aspects of data processing in hospitals range from data input, information flow, and adequate ubiquitous presentation to scheduling and order entry. Typically, clinical environments are more complicated than office or industrial environments. Medical personnel needs flexible systems that are highly and easily adaptable. Vendors should offer end user tools which are powerful, terminologically close to the understanding of the end-user, and to be used with a minimum of technical skills.

**Key objective #3: Flexible software design.** Both key challenge #3 and #2 result in the need of a powerful software design. Current informatics concepts like plug-in, adaptation, construction set paradigm, and software-reuse of building blocks have to be assessed for practicability and their role in vendors’ concepts. The conventional approach of providing adaptability in a software product is based on the customizing of parameters. The resulting standard software products
are extremely successful throughout industrial applications. The products are difficult to build and complex, however, and the customizing processes are complicated and time-consuming. Significant system-near know-how is needed, which results in a high dependency on specific vendors. Flexibility is limited to the specified parameters and their domains; adaptation to all needs over the whole range of clinically oriented applications can hardly be provided by this approach.

Instead, new software design principles are beginning to appear on the market, that offer the perspective of high functional adaptation combined with workflow support. A central technical concept is a component-based software design[17], which involves a clear conceptual separation or encapsulation of different levels of abstraction (e.g. data, functions, forms, flow of control). In order to support the creation or modification of forms, workflows, and software modules, applications are built by assembling components. The development process is guided by an environment providing a toolkit tailored for the end-user who needs only limited system know-how. It is helpful to distinguish between "material" and "tool" objects: The "tool" part holds the formal description of forms, interactions, and processes, the "material" part consists of the objects on which these tools operate[18]. The component-based software design should help to enable software reuse as well as fast and easy development and reconfiguration. Distributed objects middleware such as CORBA or DCOM support the development of distributed component-based applications by providing mechanisms for a platform-independent interoperability[19]. This is relevant for the future flexibility of products, and vendors are beginning to consider it. The analysis of the market shows that there is an increasing number of tools enabling the generation of forms or even workflow-like interactions. The technical approaches are different, the parametrization approach is predominant. Component-based concepts are appearing, however.

Because of its impact and rapid development, the web-based approach deserves attention. Actually, it provides a platform independent graphical user interface, location transparency, and a common interface style. The built-in capabilities primarily support the user interface level. Security problems, a high programming overhead for the realization of session oriented applications based on the stateless, and somewhat limited interaction capabilities have to be considered, however. In terms of integration, the web-based approach has to be combined with technologies solving the application and the data problem, e.g. by following an interoperability approach; the perspectives of CORBA and DCOM are remarkable in this respect[17,19]. On the market, however, web-based approaches are still mainly offered for information/knowledge distribution and for result servers with local replicas.

**CONCLUSION AND DISCUSSION**

When Clayton et al. described their information system architecture[20] in 1992, they pointed out that there is more than one "right" way to build a good system. This is certainly still true. Our Europe-wide formal vendor selection process gave us a chance of having a closer look at functionality and technical concepts, and we have come to a series of statements.

*First*, the trade-off between autonomy and integration is central for HIS design. The main question is, how much generalization is possible for an integrated system, and how much specialization is needed for sufficient local functionality. As soon as autonomous systems have to be connected, important aspects of integration are insufficiently supported by today's products, which results in limitations concerning consistency, clinical communication, information flow, and organizational support. In comparatively homogeneous environments offered by some generalist vendors, information flow is easier to realize, which results in a better support of interdepartmental functionality.

*Secondly*, what we consider extremely important, is that tools for the generation of forms and workflows are becoming available, which opens new perspectives for clinical computing. Clayton et al. stated in 1992, that "no single commercial vendor can supply even 10% of the total information applications desirable..."[20]. We can say today that products are becoming available that can help to realize significant parts of the applications needed most.

*Thirdly*, we agree that technology changes rapidly[20], but clarification of concepts and an examination, how sound design principles of software products are, are extremely important in order to understand the potential of a system. We explicitly mentioned the database schema, distributed object oriented middleware and a component-based software architecture in this context. Altogether, we found that a logically central database, together with a component-based architecture, a concept for schema evolution, and a modern toolkit for the generation of forms for data input and organizational aspects, is a possible and promising concept for a HIS architecture. Bleich and Slack[21] predicted the increasing importance of an integrated approach, although they saw no real vendor perspective for it in 1992; in 1998 we found that this is beginning to change. Specialized subsystems will exist for a long time. Middleware products may be helpful in the integration of these systems, but design and execution autonomy are limiting factors. A promising approach would be
to minimize the number of interfaces, to introduce a logically central database concept, and to gradually replace legacy systems. Many projects have contributed to this perspective, although non-trivial tasks remain to be solved. The usefulness of distributed object oriented technologies is much better understood, and products are available. New and powerful standards (like CORBAmed) are evolving. Database research helped to clarify terms, and the consistency-autonomy tradeoff has become clearer. Different types of semantic heterogeneity are better understood; medical vocabulary projects are working on this issue. A synthesis of knowledge representation, database and software engineering research seems possible. The advances in software design principles may result in the most important change. Especially component-based approaches show the potential of supplying toolboxes for the development of clinical applications. Actually, in hospital information systems, novel technologies have been successfully introduced.

In another early paper of 1991, Stead argued in favor of a new generation of centrally planned systems sharing a logically common database. We see our results as a technology and market counterpart of his perspective, which has come closer to reality.

References