A Conceptual Approach to an Open Hospital Information System

Kuhn K[a], Reichert M[b], Nathe M[b], Beuter T[b], Heinlein C[b], Dadam P[b]

[a] *Medical University Clinic, 89070 Ulm, Germany*

[b] *Dept. Databases and Information Systems, University of Ulm, 89069 Ulm, Germany*

A concept for an open hospital information system integrating heterogeneous components is proposed. The approach is object-oriented, but is not built directly upon object-oriented database management systems. Basic ideas are a software bus, workflow management with extended transaction concepts, and integrated knowledge-based modules. From the application perspective, conventional and problem-oriented views, support of complex medical and organizational processes including scheduling, and integrated clinical guidelines are important features.

1. Objectives

The design and realization of open systems is of central importance for HIS technology of the 90s. Homogeneous environments have their advantages [1]. In many cases, however, purchased HIS components meet application needs at reasonable costs, and their use results in the introduction of heterogeneity [2]. Therefore, a major challenge is to integrate or at least to interface existing and future applications. A typical solution for an integrated open system today would use a central patient database with a common database interface for the integration of various peripheral systems [2]. The message standard would be HL7 and terms would be controlled by a medical entities dictionary [2]. To date, for the full breadth of a complete integrated academic information management system, no true integration exists [2].

The intention of this paper is to describe a research project funded by the state of Baden-Wuerttemberg in Germany. The project is aimed at the systematic analysis of information systems needs for an integrated open system of the future that is capable of truly integrating heterogeneous components and leaving a maximum of autonomy to the cooperating systems. Current developments in computer science methods are being evaluated under the aspect of usefulness in a clinical environment.

From the computer science perspective, central concepts are a software bus [3, 4] with conventional and "intelligent" agents [5] and an information resource dictionary system [6] in an object-oriented approach [7], extended transaction models [8], and workflow management [9]. From the application perspective, concepts include the computerized medical record [10, 11], medical process automation [12], the combination of conventional and problem-oriented views [13], and the integration of clinical guidelines [14].

The project is set up to result in prototype realizations in 1994. This paper gives an overview of concepts developed and problems identified.

2. Methods

The basic component of the project is a software bus (Fig. 1) defining the integration infrastructure for distributed, heterogeneous systems. It offers services to the application programs, which are intended to

keep local implementation needs minimal; applications are "plugged in" to the bus. At the center of the infrastructure are application-independent services like trading services abstracting from distribution and heterogeneity, object services for the storage, manipulation, and exchange of complex structured objects, and security services.

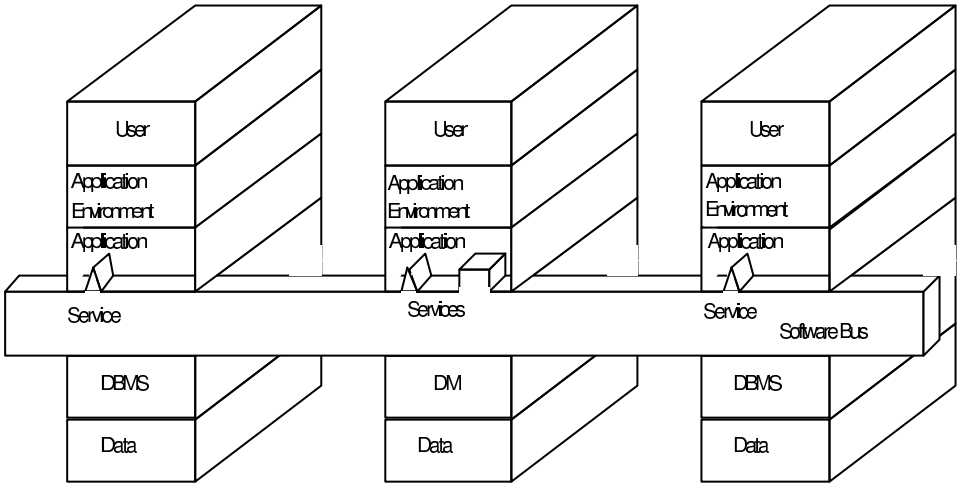

Figure 1: The software bus described in a simple model similar to the proposal of Leguit [15].
Without software bus, three direct integration/ interfacing methods are possible:
1. Interfacing via (parallel) remote logins to different applications: one user at one workstation is confronted with different application environments.
2. Direct connections between applications: development costs are high and bilateral conventions are needed.
3. Distributed DBMS: close to an ideal solution, but applicable only in fairly homogeneous environments.
The software bus has been introduced to integrate heterogeneous systems via level integration.

Application-dependent services are offered by conventional and "intelligent" agents. Typical application services are order processing, scheduling, transmission of medical results, ADT, billing, access to remote data and knowledge servers. The interfaces of application services are kept in an interface repository [cmp. 6]. The same service may be offered by various agents on different hosts with different implementations; e.g. each department might run its own scheduling agent. A service manager acts as a trader and request broker, hiding distribution and heterogeneity details from clients. Service managers also introduce fault-tolerance into the systems by maintaining backup-services and masking server errors from clients. Our cooperation model could be called client-service, as the server topology involved in a given service is invisible for clients. The approach can be described as object-oriented with concepts like encap-

sulation, abstraction and polymorphism [7]. Application programmers will be supplied with implementations of typical services. Interfaces to the software bus will be realized for purchased applications; it is expected that these systems will typically come with HL7 interfaces.

Workflow management methods are used to model and guide medical processes. Processes consist of a number of steps, where each step can be a service or a process itself, i.e. processes may be nested. Typically, multiple parties are involved in performing a process. The steps of a process are executed sequentially or in parallel, according to predefined protocols. The data and control flow is controlled by a process manager.

To guarantee consistency of these complex processes, extended transaction concepts are needed [16]. As it may be necessary to see intermediate results and to make the semantics of single steps transparent, the isolation and atomicity of conventional transaction protocols (including 2-phase commit) are sometimes too restrictive. In such cases, compensation mechanisms are used for problem, error and exception situations; e.g. whenever a step fails within a particular process, preceeding steps of the process can be compensated by calling compensation methods. The actual compensation method chosen may depend on the nesting level.

Services will be implemented on top of a common object model, which introduces generic object-oriented concepts for dealing with complex structured medical objects. Service implementers work with predefined medical object classes and methods rather than directly on the SQL level. The object methods themselves can be implemented by SQL or some other data manipulation language.

The data model described in terms of the entity-relationship methodology is basically divided into three portions: patient-specific medical information, patient-independent medical knowledge, and administrative and billing information.

It is not intended within the project to develop an own medical data dictionary or to work on medical terminology. Instead, we hope and expect that the results of integrated efforts in the US and in Europe will become available for the medical and medical informatics community in the near future.

From the technical viewpoint, OSF/ DCE [7], C++ and (distributed) relational DBMS are basic components. A prototype has been realized to explore the technical concepts. Example processes have been implemented as distributed DCE applications utilizing DCE Remote Procedure Calls. The process status is managed under the RDBMS INGRES. Service requests are handled by a Service Manager using the DCE cell directory service. While services like remote data retrieval may be executed immediately following an RPC, other services may require a high amount of local user interaction. Therefore, as local instances of the Service Manager, Task Managers for local and persistent request handling have been introduced. A Compensation Manager has been built, which also uses INGRES. Authorization and authentification are based on the DCE security service. The knowledge-based component is being realized on top of the classification-based knowledge representation system LOOM [17].

3. Applications

In a series of meetings with experienced physicians of different disciplines, and after a survey of physicians in our university hospital, the following application fields have been identified as important:

- Several tasks performed by medical personnel are of both medical and organizational nature; this is especially the case for scheduling and order entry. Phone calls to schedule examinations and therapies are considered highly time-consuming by medical personnel. The system will support order entry and automatic scheduling, taking into consideration patient-specific information and general medical knowledge. Here, the option is included to schedule activities that are medically necessary before a major intervention is carried out. Thus, an overview of the current status of the process initiated by a medical order is always available. The underlying workflow plans represent an important form of medical knowledge. Methods of workflow management rsp. process automation will be used.

- Integration of knowledge in the form of clinical guidelines for diagnostic, therapeutic and nursing activities. Here, systematization of medical processes under the aspects of quality and economy is an important objective.
- Combination of conventional with problem-oriented views of the data.
- Complete integration of organizational, administrative and clinical perspectives and tasks.

4. Discussion

The aims of the project described cover a comparatively broad range, and it has not been intended to implement a fully functional HIS within this project. The basic idea, instead, was to analyse current hospital information problems and to relate them to recent computer science research. Results so far, are mainly on the level of system analyses and concepts. In addition, a first prototype has been realized, and further prototypes will appear in 1994. Our research addresses major challenges in open distributed clinical computing: development of reliable and flexible distributed applications, handling of heterogeneity and different versions of services within a multi-server/ multi-client environment, use of automated medical processes with extended transaction concepts, and integration of distributed medical logic modules.

Several basic ideas could be identified, which are promising and will be incorporated into prototypical implementations: the software bus into which applications can "plug in", workflow management with extended transaction concepts, and integrated knowledge-based modules. The idea to use a software bus has been taken from current trends in computer science and from the HELIOS project [3, 4]. An object-oriented approach to distributed systems with encapsulation and abstraction seems useful in a clinical environment. We did, however, decide to base our system not directly upon an object-oriented database management system for several reasons (e.g. lack of performance and lack of a sound underlying theory). Instead, we will provide similar capabilities on top of relational DBMS.

From the idea of process automation, a significant effect upon clinical routine work can be expected. A comparable approach has been presented recently [12], but there is still significant research work to be done, especially in the field of appropriate transaction concepts. The integration of knowledge-based modules is a common research topic at present, but it has not yet been sufficiently understood in the context of distributed cooperative environments.

From the application view, the importance and benefits of computer-based medical records [10] and of clinical guidelines [14] are widely accepted. The described open distributed approach will support the integration of data and information from heterogeneous sources. In addition, it supports networked knowledge servers. Thus, the system could be used as a tool to establish guidelines and systematic ways to economic and efficient diagnostics and therapies; an objective we consider extremely important. The "intelligent" support of scheduling is expected to result in considerable time savings; the workflow management component helps to implement even complex sequences of medical processes. Typically, the organizational and medical activities in these sequences are based on implicit knowledge which will be explicitly represented in our approach.

5. References

[1] Bleich HL, Slack WV. Designing a Hospital Information System: A Comparison of Interfaced and Integrated Systems. In: *MEDINFO 92, Proc 7th World Congress on Medical Informatics 1992*. Lun KC, Degoulet P, Piemme TE, Rienhoff O (eds). Amsterdam: North-Holland, 1992:174-177.

[2] Clayton PD, Sideli RV, Sengupta S. Open Architecture and Integrated Information at Columbia-Presbyterian Medical Center. *MD Computing* 1992, 9:297-303.

[3] Adomeit R, Deiters W, Holtkamp B, Schülke F, Weber F. K/ 2_R: A Kernel for the ESF Software Factory Support Environment. In: *Systems Integration '92, Proc 2nd Intl Conf Sys Integration*. Ng PA, Ramamoorthy CV, Seifert LC, Yeh RT (eds). Los Alamitos: IEEE Comp Soc Press, 1992:325-336.

[4] Jean FC, Jaulent MC, Coignard J, Degoulet P. Distribution and Communication in Software Engineering Environments. Application to the HELIOS Software Bus. In: *Proc 15th Symposium on Computer Applications in Medical Care 1991*, Clayton P (ed). New York: McGraw-Hill, 1992:506-510.

[5] Brodie ML. The Promise of Distributed Computing and the Challenges of Legacy Systems. In: *Advanced Database Systems: Proc 10th British National Conference on Databases*. Gray PM, Lucas RJ (eds). New York: Springer, 1992.

[6] Goldfine A. The Information Resource Dictionary System. In: *Entity-Relationship Approach. Proc 4th International ER Conference*. Chen PP (ed). New York: IEEE Comp Soc Press, 1985:114-122.

[7] Nicol JR, Wilkes T, Manola FA. Object Orientation in Heterogeneous Distributed Computing Systems. *IEEE Computer* 1993, 26:57-67.

[8] Elmargarmid AK (ed). *Database Transaction Models for Advanced Applications*. San Mateo CA: Morgan Kaufmann, 1992.

[9] Ellis CA, Nutt GJ. Modeling and Enactment of Workflow Systems. In: *Application and Theory of Petri Nets 1993*. Marsan MA (ed). New York: Springer, 1993:1-16.

[10] McDonald CJ, Tierney WM. Computer-Stored Medical Records. *JAMA* 1988, 259:3433-3440.

[11] Board of Directors of the AMIA: Standards for Medical Identifiers, Codes, and Messages needed to Create an Efficient Computer-stored Medical Record. *JAMIA* 1994, 1:1-7.

[12] Gangopadhyay D, Wu PYF. An Object-Oriented Approach to Medical Process Automation. In: *Proc 17th Symposium on Computer Applications in Medical Care 1993*. Safran C (ed). New York: McGraw-Hill, 1993:507-511.

[13] Weed LL. *Medical Records, Medical Education and Patient Care: the Problem-Oriented Medical Record as a Basic Tool*. Cleveland OH: Case Western Univ Press, 1969.

[14] Audet A-M, Greenfield S, Field M. Medical Practice Guidelines; Current Activities and Future Directions. *Ann Intern Med* 1990, 113:703-714.

[15] Leguit FA. Interfacing Integration. In: *Hospital Information Systems: Scope-Design-Architecture*. Bakker AR, Ehlers CTh, Bryant JR, Hammond WE (eds). Amsterdam: North-Holland, 1992:141-148.

[16] Special Issue on Workflow and Extended Transaction Systems. *Data Engineering* 1993, 16.

[17] Brill D. *Loom Reference Manual*. Los Angeles: Information Sciences Institute, University of Southern California, 1993.