Modeling interdependencies between business and communication processes in hospitals

Birgit Brigl, Thomas Wendt, Alfred Winter

University of Leipzig, Institute of Medical Informatics, Statistics and Epidemiology

Abstract

The optimization and redesign of business processes in hospitals is an important challenge for the hospital information management who has to design and implement a suitable HIS architecture. Nevertheless, there are no tools available specializing in modeling information-driven business processes and the consequences on the communication between information processing tools. Therefore, we will present an approach which facilitates the representation and analysis of business processes and resulting communication processes between application components and their interdependencies. This approach aims not only to visualize those processes, but also to evaluate if there are weaknesses concerning the information processing infrastructure which hinder the smooth implementation of the business processes.

Keywords

Hospital information systems, architectural models, business processes, communication processes

1 Introduction

The optimization and redesign of business processes in hospitals is an important challenge for hospitals in the next years. Although the support of business processes by computer-based information processing tools is not the cure-all [1], the implementation of business processes without them is not imaginable. This is where the hospital information management must decide which application components suit best, which communication interfaces and communication links are necessary, which communication standards and which message types must be supported– in short: what is an optimal HIS architecture to support the business processes and fulfill the resulting information needs. In particular, the introduction of new application components and the replacement of legacy systems [2] needs a detailed specification concerning the communication with other existing application components. This applies especially for hospital information systems following an DB^n architectural style, where we can find a great variety of application components of many different vendors which all have their own database system. This leads not only to distributed but also to redundant data storage which is rather the normality than the exception.

For business process modeling and simulation, there are a lot of useful tools available, which concentrate on the domain layer where conceptual models can be built, considering information processing tools as resources that don't have to be specified any more (see e. g. [3], [4], [5]). There are no tools available which are specializing in modeling information-driven business processes and the consequences on the communication between information processing tools and, thus, can give answers to information management questions arising in this context, like

- On what paths can data, representing needed information, be transported from the storing database system to the processing application component?
- On what paths can data, representing produced information, be transported from the processing application component to the storing database system (or systems in case there is redundant storing) ?
- Does the hospital information system of a certain hospital provide a suitable infrastructure (communication links, interfaces, message types, application components etc.) for transporting the data? Will some additionally planned components be sufficient for constructing the needed infrastructure?

To overcome this problem we will present an approach which does not simply look at the business processes at the domain layer but additionally at communication processes between application components and their interdependencies. This approach aims not only to visualize those processes, but also to evaluate if there are weaknesses concerning the information processing infrastructure which hinder the smooth implementation of the business processes.

2 The three-layer graph-based meta model (3LGM^2) for modeling hospital information systems

The three layer graph-based meta model (3LGM^2) for hospital information systems and a corresponding 3LGM^2 tool were designed to support the hospital information management in its enterprise architecture planning (EAP) (see e.g. [6], [7]) activities. Accordingly, it distinguishes three layers of information systems, which especially provide a framework for describing both business processes at the domain layer and communication processes between application components and their interdependencies:

The *domain layer* describes a hospital independently of its implementation by its enterprise functions. Enterprise functions need information of a certain type about physical or virtual things of the hospital. These types of information are represented as *entity types.* The access of an enterprise function to an entity type can be in a using or an updating manner.

The *logical tool layer* concentrates on *application components* supporting enterprise functions. Application components are responsible for the processing, storage and transportation of data representing entity types. Application components may have a local *database* to store data. *Component interfaces* ensure the communication among application components. A component interface can receive or send messages of a certain message type. For the communication among application components *communication links* can be defined as relations between two communication interfaces, one being the sender of a message, the other one being the receiver. Each communication link is described by the message types which in fact are communicated.

The *physical tool layer* consists of physical data processing components (like personal computers, servers, switches, routers, etc), which are physically connected via data transmission connections (e.g. data wires).

Between concepts of the different layers there exist so-called interlayer relationships, which enable to describe the dependencies between model elements belonging to different layers. In this paper, we use the following interlayer relationships:

- An enterprise function is supported by a set of application components.
- A certain database may be master for an entity type, and thus be responsible for the storage of certain entity types. In case of redundant data storage, this particular database will contain the current data.
- Entity types can logically be represented by a dataset type and a message type to describe how they are stored and communicated. Dataset types describe what information is stored in a database, message types describe what information is transported by a communication link between two application components.

The description of 3LGM^2 is simplified accordant to the concepts needed in this paper. In particular, we focus on the domain layer and that part of the logical tool layer covering computer-based information processing tools. Nevertheless, our approach can easily be expanded to conventional, i. e. paper-based tools. There exists a comprehensive UML-based description of the 3LGM^2 in [8] or [9].

3 Extending the 3LGM^2 to model interdependencies between business and communication processes

3.1 A 3LGM^2 business process

We refer to a business process as a sequence of enterprise functions using and/or updating information about entities of a certain domain. This definition restricts the common interpretation of the concept 'business process' as we just look at the information processing aspects of enterprise functions as well as information driven interdependencies between enterprise functions. Other events like the availability of physical resources or the termination of activities are not considered, because these kind of events are outside the scope of our approach.

Looking at 3LGM^2, a business process describes the dynamical aspects of the domain layer. The meta model of a 3LGM^2 business process looks as in figure 1.

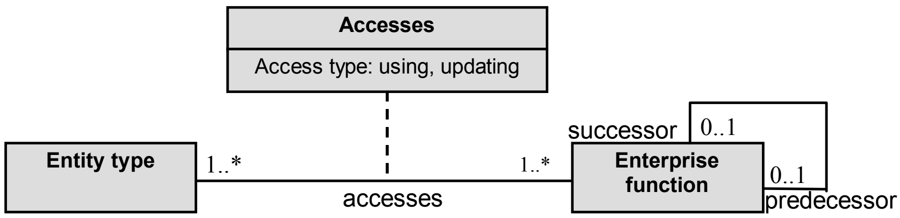

Figure 1: A UML-based meta model for 3LGM^2 business processes.

A 3LGM^2 business process consists of a *sequence of enterprise functions* and the *entity types accessed* by those enterprise functions in an using or updating manner. Each enterprise function may have at most one predecessor and one successor. The following additional constraints have to be obtained:

- There exists exactly one enterprise function which has no predecessor. This enterprise function is the starting point of our business process.
- There exists exactly one enterprise function which has no successor. This enterprise function is the ending point of our business process.
- For each enterprise function one of its successors must at least have one using access to an entity type which is accessed by that function in an updating manner. This condition reflects the inner connectivity of a business process.

At this point we deliberately restrict us to rather elementary business processes to reduce the complexity at the logical tool layer. If we want to examine more complex business processes we just have to decompose them in to simpler ones. Figure 2 shows an elementary example of a 3LGM² business process.

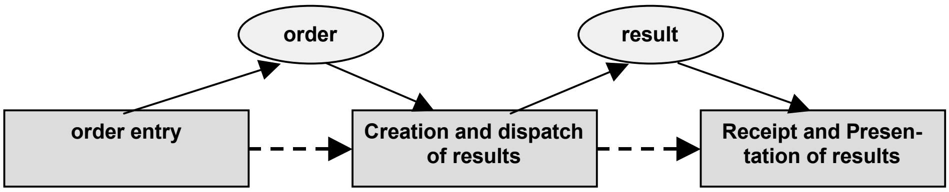

Figure 2: Example of a 3LGM² business process. (Ovals: entity types; rectangles: enterprise functions). The enterprise functions are taken from [10].

3.2 A 3LGM² communication process

We refer to an communication process as a sequence of communication links between application components necessary to satisfy the information needs given by business process. Looking at 3LGM² a communication process describes the dynamical aspects of the logical tool layer. The meta model of a 3LGM² communication process looks as in figure 3.

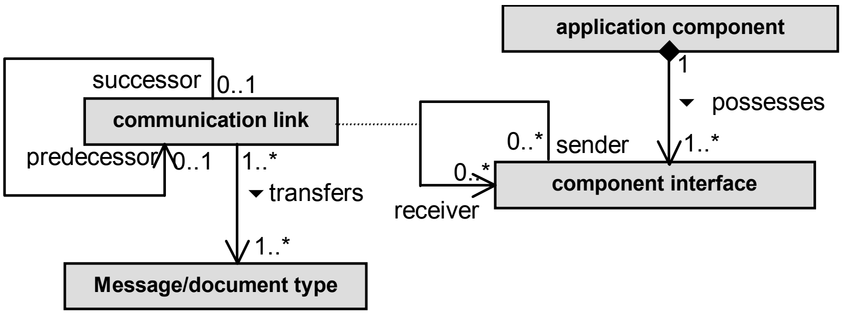

Figure 3: A UML-based meta model for a 3LGM² communication process.

A 3LGM² communication process consists of a sequence of communication links and the appertaining application components and their component interfaces. Within the sequence, a communication link may occur multiple times.

The following additional constraint has to be obtained: For each pair of communication links cl_1, cl_2 where cl_1 is the direct predecessor of cl_2, the sender of cl_1 must be possessed by the same application component as the receiver of cl_2. This condition ensures that there is also a sequence of communicating application components. Figure 4 shows an elementary example of a communication process.

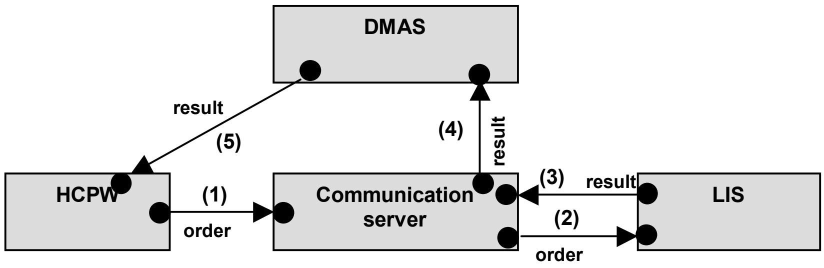

Figure 4: An example of a communication process (HCPW: healthcare professional workstation; LIS: laboratory information system; DMAS: documents management and archiving system). Black circles represent communication interfaces; arrows represent communication links; numbers in round brackets represent the sequence of communication links, the labels of the arrows represent the message types transmitted.

3.3 Mapping 3LGM² business processes on 3LGM² communication processes

If we understand a 3LGM² business process as a kind of requirements specification for the logical tool layer, it is not sufficient just to describe business and communication processes independently. The more important aspect we have to look at is what kind of communication between application components is necessary to enable the execution of a business process. The structure of the resulting communication process depends on the individual architecture of the logical tool layer, i.e. where data about entities are stored, processed and communicated.

Given a 3LGM² model of a hospital information system and a business process whose enterprise functions and entity types are part of the domain layer of the 3LGM² model, the following algorithm will lead to a communication process at the logical tool layer. For each enterprise function `f` of the given business process, the following algorithm must be carried out:

(1) For each entity type `et` updated by `f` and used by one of its successors $s_1, \dots, s_n$: determine all communication links which transport `et`.

(2) For each pair (f, s_i): Determine the set of application components $\underline{AC}_f$ supporting `f` and the set of application components $\underline{AC}_{si}$ supporting s_i.

(3) Find a shortest path of communication interfaces over which `et` can be transported between the component interfaces belonging to elements of $\underline{AC}_f$ and the component interfaces belonging to elements of $\underline{AC}_{si}$ using the determined communication links of (1). Result is a sequence of communication interfaces.

(4) Transform that sequence of communication interfaces into a sequence of communication links.

According to the sequence of enterprise functions coming from the given business process, all sequences of communication links found by that algorithm will be combined to a communication process supporting the business process. If we again look at the examples given in figure 2 and figure 4, it is the sequence of numbers which is derived. Applying that algorithm, the following situations may occur:

For an entity type, there is no shortest path to be communicated from one application component to the other. This is a very serious hint, that the infrastructure at the logical tool layer is insufficient to support the considered business process.

There are several (shortest) paths available to communicate an entity type from one application component to the other, indicating that the logical tool layer may contain redundancies. It should be clarified if these redundancies are deliberate or not.

The shortest path to communicate an entity type from one application component to the other is very long. In this case it should be asked, whether the infrastructure at the logical tool layer does really sufficiently support the business process. It may be useful to introduce additional component interfaces and communication links.

4 Discussion

In this paper we presented an approach which can support information managers modeling the architecture of their information system, focusing on the interdependencies between business processes and communication processes. For a given 3LGM^2 model of a hospital information system and a given business process, this approach enables us to evaluate if there are weaknesses at the logical tool layer, which hinder the implementation of a business process. In this respect, it covers new features for information systems modeling. Even if this approach originates in the scope of the 3LGM^2 research activities it may be applicable to other approaches if the necessary concepts can be modeled.

Particularly, for the answering of questions like those mentioned in the introduction, our approach may be useful because the necessary analysis of the hospital information system can be done automatically, provided that there is a tool available which implements the algorithm. For this reason, the 3LGM^2 tool which supports modeling of hospital information systems will be extended.

In the future, the presented approach will be further extended considering also more complex business processes, and the physical tool layer of information systems.

References

[1] Hammer M, Champy J (1994): Reengineering the corporation. New York: Harper Business.

[2] Brodie M.L., Stonebraker M. (1995). *Migrating legacy systems*. Morgan Kaufmann, San Francisco.

[3] Oberweis A., Scherrer G., Stucky W. (1994). INCOME/STAR: Methodology and Tools for the Development of Distributed Information Systems. Information Systems 19(8), 643-660.

[4] Ferstl OK, Sinz EJ (1997): Modeling of Business Systems Using the Semantic Object Model (SOM) - A Methodological Framework. In Bernus P., Mertins K., Schmidt (eds) (1997). *Handbook on Architectures of Information Systems. International Handbook on Information Systems.* Springer,

[5] Scheer A-W (1994). ARIS Toolset: A Software Product Is Born. *Information Systems* 19(8), 607-624.,

[6] Spewak SH, Hill SC (1992): *Enterprise Architecture Planning. Developing a blueprint for data, applications and technology*. New York: Wiley.

[7] Zachman JA (1999): A framework for information systems architecture (Reprint). IBM systems journal 38(2&3), 454-470.

[8] Winter A, Brigl B, Wendt T (2001): A UML-based Ontology for Describing Hospital Information System Architectures. In Patel VL, Rogers R, Haux R (eds) (2001). MEDINFO 2001. IOS, Amsterdam. S. 778-782.

[9] Winter A, Brigl B, Wendt T: Modeling Hospital Information Systems (Part 1): The Revised Three-Level Graph-Based Meta model 3LGM^2. Submitted for publication.

[10] Ammenwerth E, Buchauer A, Haux R (2002). A Requirements Index for Information Processing in Hospitals. Meth Inf Med 41 (4): 282-8.