Process Based Data Logistics: Data Integration for Healthcare Applications

Stefan Jablonski

Chair for Databases and Information Systems, Institute for Applied Computer Science
University of Bayreuth
D-95440 Bayreuth, Germany
Stefan.Jablonski@uni-bayreuth.de

Abstract. Data integration in medical applications is a crucial and sensitive task. It turns out that there are rigid factors like heterogeneous, distributed data sources, security, and complex, dynamically changing application requirements that require a highly sophisticated and flexible data integration concept which – in a final stage – has to cover the whole healthcare landscape. We have developed a process based data logistic approach that is coping with these requirements. It is based on process technology including workflow management and database technology including ontologies and terminology management.

1. Data Integration

There are many different scenarios for data integration. The ideal setting is that all applications access a central data store, preferably a database system. However, not all applications can follow this (simple) model. Healthcare applications, especially clinical applications demand a more sophisticated model. Figure 1 depicts a typical scenario of a clinical application. There are a number of so-called modalities (MOD_x), i.e. technical equipment like a digital camera for making pictures of the papilla, a laser scanning device, or other ophthalmic equipment for taking various measurements of the human eye[1]. Typically, data created at a certain modality (e.g. pictures, diagrams) are of different quality and structure (format). Also, data are firstly stored locally at the modalities themselves; these local data stores are here called DB-MOD_x.

Besides modalities – where findings of investigations are created – various information systems (IS_y) are deployed in clinical applications, like a hospital information system (HIS) and a patient information system. Similarly to modalities, these information systems also maintain their own data management facilities; we call them DB-IS_y. Typically, data created at modalities have to be integrated into these information systems. However, also data exchange among information systems is necessary (e.g. to exchange patient information), and data also have to be transferred from information

[1] We mostly refer to examples from ophthalmology since this research work stems from a project about the glaucoma disease. However, the concepts presented are generally applicable. This work was funded by the German Research Foundation (SFB 539).

systems to modalities (e.g. patient identifications). Thus, from an abstract perspective, data have to be exchanged between and among modalities and information systems, respectively. Hence, we generalize modalities and information systems to "clinical applications" (APPs).

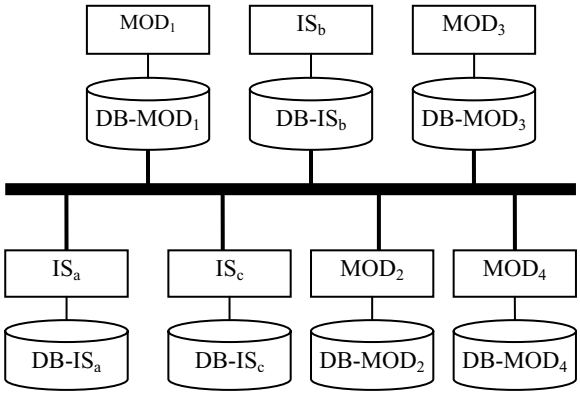

Figure 1: Clinical (healthcare) application scenario

Data exchange between APPs does not always occur synchronously and at once, i.e. when data are entered into the system (e.g. at a particular modality). For example, before such data is passed on to some consuming systems (APPs) they must first be validated.

As an example, we regard a special research process from our glaucoma research project that is collecting data from the diverse ophthalmic modalities and is storing them in a central data base, called glaucoma register [JLMM04]. Besides, patient identifications have to be distributed among all modalities; they originate from the glaucoma register. In some special cases, when patients are already registered in the central HIS, patient identifications have to be incorporated from there into the glaucoma register even.

There are two principal solutions to the task of data integration in such a challenging environment: a first solution would be to use a sophisticated data management concept for the data integration problem. Centralized, distributed or federated data management systems are these three facets of this data base oriented solution. We argue in [Jabl06] that none of the three approaches is feasible. Another solution approach is to deploy process technology. This is the approach we are pursuing. Albeit, the usage of conventional workflow management technology – which seems to be an obvious solution – is also not practicable. In Section 4 this issue will be considered. In order to prepare this discussion and to present our principle solution idea, some most relevant constraints stemming from the clinical application area have to be identified:

- (C1) The data artefacts of the APPs have different formats, use different terminologies, and refer to different ontologies. They must be synchronized [Teic98].
- (C2) Data produced at the different APPs use different identification schemes. These must be coordinated.

- (C3) We already mentioned that data exchange between the APPs is very cumbersome and, thus, is a meticulous task. This is mostly due to the fact that the clinical personnel are very flexibly using the modalities during examinations. So it can hardly be predicted when an APP will be used: the physicians decide very spontaneously, e.g. what modality to use in order to get more and detailed data about the patient for making a better diagnosis. Thus, it would not be tolerable if the physicians are totally controlled and supervised by an IT system which dictates the next steps in the examination.
- (C4) Nevertheless, all examinations have to obey certain clinical and/or medical requirements. These are normally derived from clinical pathways [Hind96] and medical guidelines [MZPT03].
- (C5) Our application scenario is continuously changing since it is representing a research process. Due to newly discovered facets, the research process is constantly adjusted. This has an influence on data exchange between APPs.
- (C6) Data exchange between APPs must occur transparently to the clinical personnel – as far as possible. As the personnel already have to handle multiple computer applications, it does not want to be burdened by yet another IT system [Ande97] – especially if this is "just" for data management. Data management is not considered as a "primary" task (like taking a picture of the retina) but it is regarded as an "auxiliary" task that should best be done by the IT system autonomously.

Security also is a strong issue. Personal data must be protected from unauthorized access. However, we do not deal with security in this paper. An analysis of the above requirements shows that they belong to three categories:

- Technical: to cope with different formats, terminologies, ontologies (C1), and identification schemes (C2) requires a technical solution. Among other things, mappings between data contribute to this matter.
- Application specific: clinical application scenarios tend to be spontaneous (C3) but restricted (C4), and dynamic (C5). This can be traced back to the enormous complexity of a patient treatment process.
- Usage: Transparency of data management is a major need, since it is regarded as an auxiliary, technical task, not as a medically motivated requirement (C6).

Our contribution is a process based data logistics mechanism that – transparently to the users (the clinical personnel) – controls the data exchange between the APPs. It is a process based concept since

- processes are regarded as being a suitable means to cope with complex application scenarios (C4),
- processes can easily be modified and adjusted to changing requirements, and (C5)
- processes deliver a sophisticated structural framework to integrate the diverse (technical) tasks of data exchange, i.e. terminology, format, and ontology adjustment, and synchronization of identification schemes (C1, C2).

The above list reveals that still constraints C3 (flexible execution) and C6 (transparency of execution) must be dealt with. These constraints will not be met by processes directly; but our process based data logistics concept will cope with it (Section 2).

In the next section the method for data logistics is introduced. Section 3 discusses related work. Workflow management is reconstructed in Section 4. The data logistics framework is presented in Section 5.

2. The Data Logistics Method

The first step of our method is to model clinical application processes [BKMJ04] [Ires97] [Kory95]. In order to sustain consistency and adequacy these process definitions are based on medical guidelines and clinical pathways as much as possible.

It is well recognized that process modelling is an excellent means to cope with complex and huge applications [JaBu96]. The global view provided by processes supports this; especially the interactions between multiple participants and systems can best be modelled. Thus, constraint C4 is met already. It is also well known from literature [BKMJ04] that process models are best suitable to be adapted for rapid changes. The graphical representation of processes facilitates change operations effectively and efficiently (constraint C5).

Since we choose a perspective oriented process model (Section 4) the technical requirements C1 and C2 can also be met easily. In the perspective oriented process model, data will make up one separated perspective [JaBu96]. Terminology, format, and ontology adjustment, and synchronization of identifications are specific tasks associated with this data perspective. Thus, it can easily be identified where in a process model these tasks have to be reflected.

Still constraints C3 and C6 have to be addressed. They cannot be directly covered by normal process management. In order to cope with them, process management has to be extended by certain concepts, which we refer to as data logistics. Data logistics assumes the following:

- When clinical users are performing an examination, they are not guarded directly by clinical processes as they are defined in the first step of our method. Instead, they can use an APP as spontaneously as without the data logistics having been installed (C3). However, there must be some mechanism that controls the medically correct order of examinations. We will discuss this in Section 5.
- When an APP is used, data will be produced there. As usual, these data are at first stored in the local data stores (DB-MOD_x and DB-IS_y, respectively). According to data logistics specifications – which are derived from the defined clinical processes (Section 5) – these data are transformed and transported to specified sinks. This happens totally transparent to the users (C6).

The content of the second step of our data logistics method was mentioned already: data logistics is derived from clinical processes. We will see in Section 5 that data logistics is nothing but a set of processes. We call them data logistics workflows (DL workflows) in order to distinguish them from the "normal" clinical processes defined in the first step of our method.

The DL processes run permanently in the application system and take care that data exchange between the APPs is performed at the right time between the right APPs. One of the main features of DL processes is their adjacency to clinical processes, i.e. even to medical guidelines and clinical pathways the latter are based upon. So, the very technical character of data logistics is directly connected to the clinical application field. This is a huge advantage since it guarantees the correctness and adequacy of the DL workflows.

After having the data logistics method introduced, three main issues have to be investigated in the next sections: How to specify clinical processes (Section 4)? How to derive DL processes from clinical processes (Section 5)? How to execute DL processes (Section 5)?

3. Related Work: Data Management

Related work in the data oriented field is divided into three parts: the core database field [BKMR03], the field of connection server technology for clinical applications [Hass97], and the field of standards for data interchange in clinical application systems. Due to space limitations the core database field cannot be discussed in detail. Refer to [Jabl06] for a profound discussion.

Often specialized middleware, called connection server or integration engine, is used to establish necessary communication relationships between clinical applications in a standardized format, e.g. Health Level 7 (HL7) [HaMo93]. Connection servers achieve this by translating between system specific protocols and by transporting data between different data bases. However, this approach shows a deficit on the conceptual level. It is more or less implementation and not model driven. This results in the fact that a comprehensive model covering all communication scenarios is missing. Such a model is strongly demanded in our application. Additionally, the connection server concept lacks a close conceptual interconnection with process models. However, this is a major demand of an implementation concept for data logistics.

We do not elaborate on standard formats for data exchange in clinical applications in this paper [With01]. Those standards are relevant for data logistics just in parts. Our goal is to quickly store data created at a specific modality into the glaucoma register. The glaucoma register consolidates all data and integrates them under a common data scheme. Along this task, standards are just relevant to consolidate terminology, formats, and ontologies [JLM+05]. However, the data must not be completely transformed into a standard format in order to achieve this. Only in the case that data from the glaucoma register are transmitted into the clinical practice, i.e. into the HIS, then such data

standards are relevant. In [GMJP05] we show how data can be exchanged between the glaucoma register and an HIS using the CDA standard.

To summarize, from the different fields of data management particular aspects should be adapted for data logistics (e.g. format consolidation). However, none of these concepts is sufficient to cover the requirements of data logistics fully.

4. Related Work: Process Management

The following discussion about process management mostly concentrates on workflow management. Workflow management is regarded as that part of process management that aims at the execution of processes – which is our major interest[2]. We will not provide a comprehensive introduction into workflow management in this section; instead we refer to [JaBu96] for a broad discussion of this topic.

According to the perspective (or aspect) oriented workflow model [JaBu96] a workflow consists of several perspectives which are weaved together. The functional perspective builds the skeleton of a workflow; it describes the tasks a workflow has to accomplish. The organizational perspective identifies the agents (e.g. physicians) who are responsible to perform a task. The operational aspect specifies what applications support the execution of a workflow step. The control flow perspective defines the execution order between workflows. The data aspect defines both the in and out parameters of workflows and how data flow between workflows.

In Figure 2 a very simple workflow model is depicted. Workflow WF1 is executed before workflow WF2 and before workflow WF3. The solid lines describe the data flow between workflows. For instance, WF1 produces a data item OUT1 that becomes the input IN2 of WF2. Additionally control flow is defined between WF1 and WF2 that – for instance – specifies that WF2 may not be executed before a certain time has passed and WF1 has terminated. AGT1 denotes the agents that are responsible to perform WF1; APP1 denotes the applications called by WF1 (e.g. an ophthalmic system). For simplicity we assume in the following discussion that there is just one single agent and application specified; this does not limit the scope of our discussion.

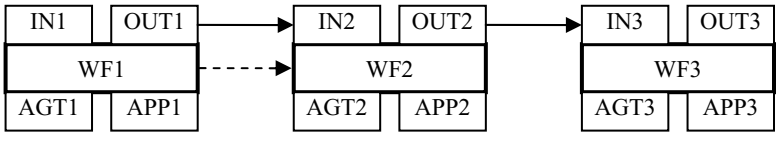

Figure 2: A sample clinical workflow

[2] In this paper we synonymously use the terms process and workflow; however, if the execution aspect should be focused "workflow" is used instead of "process".

The reason why we discuss workflow management in this paper is obvious. In Section 1 we were arguing that clinical processes are major building blocks of data logistics. Thus it seems to be a valid idea to model clinical processes as workflows and to execute them on top of a workflow management system. However, this approach was already rejected in Section 1 (constraint C3); we will elaborate on this in the following.

Figure 3 shows as a simplified (UML) sequence diagram about the processing of workflows. The workflow management system (WfMS) selects a workflow that is ready for execution; in our case this is WF1 (cf. Figure 2). The WfMS notifies the declared agent AGT (here: AGT1) to perform it. The agent calls an application (here: APP1) that has to be executed. Required input data are always provided on the calls.

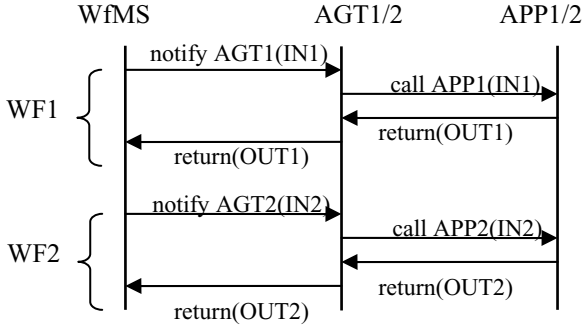

Figure 3: Processing of workflows

Although, both the workflow of Figure 2 and the execution scenario of Figure 3 are very simple one main characteristics of workflow management can excellently be discussed with this example. The WfMS is the part in the scenario that proactively triggers and initiates the work to do. That means the following: not before the WfMS has triggered an agent to perform a certain task, the agent will be able to do something. This behavior is contrary to what we have experienced in clinical application systems and what we have postulated in Section 1 (constraint C3). There, we were assuming that the agents (here mostly physicians) are spontaneously selecting ophthalmic modalities for examinations. That means, they cannot wait for a WfMS to offer such a modality for use, but they want freely – just piloted by medical requirements – to be able to select a modality for an examination. This is an absolute contradiction to the behavior a WfMS would induce. As a marginal note: many trials to use WfMS technology in clinical environments were failing due to this discrepancy [KMA+05]. Nevertheless, this observation does not mean that WfMS are totally inapplicable for clinical applications. It just means that they are not directly applicable in processes like the ones we have described here, namely processes which are characterized by spontaneous decisions and are embedded into an extremely complex medical context.

In order to return to one of the primary goals we have to revisit our application scenario. Originally, data have to be exchanged between applications. This task is not directly fulfilled by the workflow depicted in Figure 2 and Figure 3. Workflow management focuses the coordination of tasks, here WF1 and WF2. Indirectly, through the association

of agents and application systems to those tasks, the latter are also coordinated and connected. For instance in our scenario, data are flowing from APP1 to APP2 finally. However, this requires the interaction of the agents AGT1 and AGT2 before.

For the upcoming discussion of data logistics we need a semi-formal model for workflow management. Therefore, we define the perspectives of a workflow in the following. Control flow CF is a mapping $CF \subseteq WF \times WF$ which connects individual workflows of the set of all workflows WF with each other. Data flow DF is a mapping $DF \subseteq (WF, D) \times (WF, D)$. ($(WF_x, d_x), (WF_y, d_y)$) is element of DF if d_x is an output of WF_x and d_y is an input of WF_y (and both are type compatible, i.e. data item d_x can be mapped onto data item d_y). There might be multiple DF or CF connections between two workflows.

A workflow $WF_x \in WF$ is a structure $(DESC_x, AGT_x, APP_x)$ where $DESC_x$ is its description, AGT_x is the agent responsible to execute the workflow and APP_x is the application called within WF_x. Consequently, $WF \subseteq DESC \times AGT \times APPP$ where DESC is the set of all activity descriptions, AGT is the set of all agents, and APPP is the set of all available applications. Without loss of generality, we assume that there are normally just one agent and one application associated with a workflow.

With this simple workflow model at hand we now are able to model clinical workflows. Especially as we use the customizable process (workflow) model of iPM [ProD06] we are able to model clinical processes appropriable. In [Meil05] it is broadly shown how the iPM workflow model can effectively be tailored to describe clinical processes. Having accomplished this, the first task formulated at the end of Section 2 is fulfilled.

The simple workflow model also facilitates a formal derivation of DL workflows from clinical processes (Section 5). This was the second open task formulated at the end of Section 2. The basic idea is that data exchange requirements between modalities (formulated in a DL workflow) can be derived from clinical processes as the example at the beginning of this section is revealing: by combining the workflows WF1 and WF2 as shown in Figure 2 a data flow between APP1 and APP2 is specified.

5. Data Logistics

In this section we present the data logistics approach. We start with an example that illustrates the basic idea (Section 5.1). Section 5.2 demonstrates the formal transformation of clinical processes into DL workflows. How to wrap clinical applications and how to implement data transformation are two more critical technical issues. Together with an architecture of our DL system they are elaborated in [Jabl06].

5.1 An illustrating example

Data logistics aims at the organization of data exchange between APPs. What does that mean with respect to the examples of both Figure 2 and Figure 3? When APP1 is executed (under the cover of WF1) then the output data OUT1 has to flow from APP1 to

APP2 (here it becomes the input data IN2). This is equal to the fact that after the use of application APP1, the created data are distributed to APP2. Without loss of generality, in our example there is just a simple data flow between the two applications. It is no conceptual problem to extend this to more receiving APPs.

Data logistics focuses on data exchange between APPs. Thus, we now change perspective. Instead of observing the clinical processes (workflows, respectively) (Figure 2 and Figure 3), we look onto the DL workflows derived from them. A DL workflow needs the following special components:

- DL-APP: This is a special purpose application that is capable to shipping data from data sources to sinks, i.e. to performing data transport between APPs. In order to function it needs the following parameters: the participating APPs (data source and data sink), the corresponding output and input data, and the process context.
- DL-AGT: This is a special purpose agent who is associated with an APP, i.e. each APP maintains a specific DL-AGT. The DL-AGT is triggered when the corresponding APP is used. The DL-AGT then initializes the local DL-APP, i.e. it initializes data logistics.
- DL-WF: This is a DL workflow that is associated with a DL-APP and a DL-AGT.

The above components are sufficient to describe DL workflows. Figure 4 exposes the two DL workflows which are derived from the clinical process of Figure 2. The general rule is to create one DL workflow for each data connection of a clinical process (here: data flow between APP1 and APP2, and between APP2 and APP3; see Section 5.2).

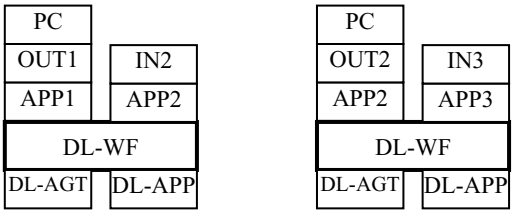

Figure 4: A DL workflow (derived from the clinical process of Figure 2)

In Figure 5 the execution of a DL workflow is illustrated. The following sequence of operations takes place:

- A user is calling an APP, i.e. the corresponding wrapper.
- The wrapper of the called APP recognizes this and informs the WfMS about the usage.
- The WfMS notifies the DL-AGT of that APP to be prepared for initializing data exchange; this normally happens after the user ceases to use the APP.
- The DL-AGT calls the DL-APP – with the corresponding parameters (see above) – to initiate data exchange.

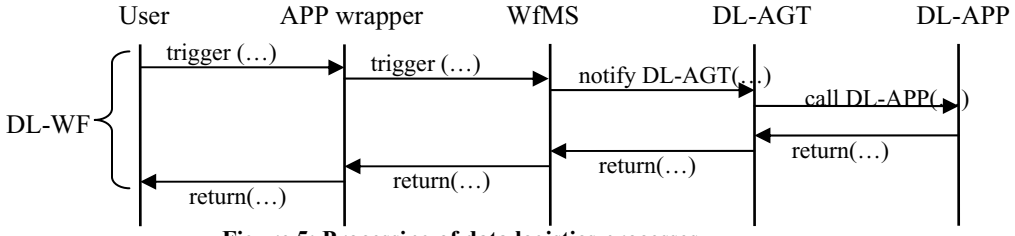

Figure 5: Processing of data logistics processes

There are a couple of observations that must be discussed here in detail:

- There is only one DL workflow to be performed, namely DL-WF. Depending on the provided parameters it is controlling data exchange between corresponding APPs. Similar, there is only one specific agent (DL-AGT) and one specific application (DL-APP) required.
- Since there is just a "normal" workflow, a "normal" agent and a "normal" application involved, a conventional WfMS is suitable as execution platform.
- Although Figure 5 is simplifying the processing it nicely demonstrates that now the initializing actor is the user of an APP. This corresponds to requirement C3. Also, data exchange takes place transparently; so, constraint C6 is also fulfilled.

Aside from implementation details to be explained later, the demonstrated DL workflow is fulfilling all the requirements identified in Section 1: it is derived from clinical processes (C4, C5), it allows spontaneous user interaction (C3), it organizes data exchange transparently (C6).

However, the advantages identified in this section do not come for free. There are a number of tasks to accomplish in order to enact data logistics. First of all, clinical processes must be available. Although the definition of clinical processes is regarded as absolutely necessary in modern clinical environments in order to optimize processing, it is a task that requires meticulous effort. Also, a couple of technical features must also be implemented: e.g. application wrappers, specific DL components.

5.2 Formal derivation of data logistics workflows

One of the most important advantages of our data logistics approach is that it has both a robust content related origin (the clinical processes) and a unique formal foundation (the unique derivation of DL workflows from clinical processes).

The transformation from clinical processes to DL workflows focuses on data flows between process steps of a clinical process. Thus, for any two workflows of a clinical process WF_i and WF_j connected by data flow (from WF_i to WF_j) the following DL workflow DL-WF can be derived: $DL\text{-}WF = (DESC_{DL\text{-}WF}, DL\text{-}AGT, DL\text{-}APP)$

$DESC_{DL\text{-}WF}$ is the description of the data logistic workflow, DL-AGT is the general data logistics agent, and DL-APP is the general data logistics application (Section 5.1). These

three aspects of the DL workflow will be the same for all clinical processes. The input and output parameters of DL-WF are more interesting than the above aspects. For WF_i and WF_j with input data IN_i, IN_j, output data OUT_i, OUT_j, applications APP_i, APP_j, respectively, the input data for DL-WF are (PC, OUT_i, APP_i). The output data for DL-WF are (PC, IN_j, APP_j). The parameter PC indicates the process context in which a data logistics workflow is performed. Through this context information the two workflows WF_i and WF_j are determined, and thus the correct data exchange can happen. This context information is necessary, since the APPs can be re-used in various clinical processes for different purposes. So the data logistics system must recognized not that just a particular APP is used, but it must also determine in what context (i.e. clinical process) this APP is used. This determines where to transmit the data created at the APP.

6. Conclusion

In this paper we have presented a process centred data integration concept. It can optimally be used in applications where data produced in independent data sources must be integrated. Database oriented integration concepts are not applicable in such contexts due to the heterogeneity of the system environment. An important advantage of the data logistics approach is its formal foundation since it is completely derivable from the specification of clinical processes.

References

[AaJa00] Aalst, W.v.d.; Jablonski, S.: Dealing with workflow change: identification of issues and solutions. International Journal of Computer Systems Science & Engineering (CSSE), Vol. 15 (2000), No. 5

[Ande97] Anderson, J.G.: Clearing the way for physicians' use of clinical information systems. Communications of the ACM, 40(8):83–90, 1997

[BKMR03] Blazewicz, J.; Kubiak, W.; Morzy, T.; Rusinkiewicz, M. (Eds.): Handbook on Data Management in Information Systems, Springer, 2003

[BKMJ04] Beyer, M.; Kuhn, K. A.; Meiler, C.; Jablonski, S.; Lenz, R.: Towards a Flexible, Process Oriented IT Architecture for an Integrated Healthcare Network. Special Track on "Computer Applications in Health Care" (COMPAHEC 2004). Proceedings of the Annual ACM Symposium on Applied Computing (SAC'04), 2004

[Hind96] Hindle, D.: Clinical pathways: a serious business. In: Health Management Bulletin. Ausgabe 1(1), 1997, S. 87-96

[GMJP05] Gerdsen, F.; Müller, S.; Jablonski, S.; Prokosch, H.-U.: Standardized Exchange of Medical Data between a Research Database, an Electronic Patient Record and an Electronic Health Record using CDA/SCIPHOX. In: American Medical Informatics Association's Annual Symposium (AMIA 2005), 2005, Washington D.C., 2005,

[HaMo93] Hammond, W. E., Moore, G. J. E. De (Eds.) et al.: “HL7: A Protocol for the Interchange of Healthcare Data.” Progress in Standardization in Health Care Informatics. IOS Press, 1993.

[Hass97] Hasselbring W., 1997. Federated integration of replicated information within hospitals. Int J Digit Libr.

[Ires97] Ireson, C.L.: Critical pathways: effectiveness in achieving patient outcomes. Journal of Nursing Administration, vol. 27, no. 6, 1997

[Jabl06] Jablonski, S.: Process Based Data Logistics: Data Integration for Healthcare Applications. Internal Report, University of Bayreuth. 2006

[JaBu96] Jablonski, S., Bußler, C.: Workflow management - modeling concepts, architecture and implementation. London. International Thomson Computer Press, 1996

[JLMM04] Jablonski, S.; Lay, R.; Meiler, C.; Müller, S.: Process Based Data Logistics: A Solution for Clinical Integration Problems. International Workshop on Data Integration in the Life Sciences (DILS 2004) March 25-26, 2004, Leipzig, Germany

[JLM+04] Jablonski, S., Lay, R., Meiler, C., Müller, S., Hümmer, W.: Process Based Data Logistics in Clinical Environments. Proceedings of Multikonferenz Wirtschaftsinformatik (MKWI 2004), Track on Knowledge Supply and Information Logistics in Enterprises and Networked Organizations, 2004

[JLM+05] Jablonski, S., Lay, R., Meiler, C., Müller, S., Hümmer, W.: Data Logistics as a Means of Integration in Healthcare Applications. Proc. SAC, Santa Fe, New Mexico, 2005

[KMA+05] Koppel, R.; Metlay, J.P.; Cohen, A.; Abaluck, B.; Localio, A.R.; Kimmel, S.E.; Strom, B.L.: Role of Computerized Physician Order Entry Systems in Facilitating Medication Errors. JAMA. 2005 Mar 9;293(10):1197-203

[Kory95] Korycan, T.: Achieving cost and quality benefits in critical pathway implementation. Quality Management in Healthcare, vol. 3, no. 2, 1995

[Meil05] Meiler, C.: Modelling, Planing, and Execution of Clinical Paths. PhD Thesis, University of Erlangen-Nuremberg, 2005

[MZPT03] Maviglia, S. M.; Zielstorff, R. D.; Paterno, M.; Teich J. M.; Bates, D.W.; Kuperman, G.J.: Automating complex guidelines for chronic disease: lessons learned. In: Journal of the American Medical Informatics Association. Ausgabe 10(2), 2003, S. 154-165

[ProD06] ProDatO GmbH, i>ProcessManager, http://www.prodato.de/software/processmanager, Retrieved 2006-03-26

[ReDa97] Reichert, M., Dadam, P.: A Framework for Dynamic Changes in Workflow Management Systems. DEXA Workshop, 1997

[Teic98] Teich, J.M.: Clinical information systems for integrated healthcare networks. Proc. AMIA. Symp., 19–28, 1998

[With01] Withrow, S.C.: Managing Hipaa Compliance: Standards for Electronic Transmission, Privacy, and Security of Health Information. Health Administration Press 2001