

Association for Information Systems AIS Electronic Library (AISeL)

AMCIS 2011 Proceedings - All Submissions

8-6-2011

Model-Supported Business Alignment of IT — Conceptual Foundations

Martin Juhrisch

University of Technology Dresden, martin.juhrisch@tu-dresden.de

Hannes Schlieter

University of Technology Dresden, Hannes.Schlieter@tu-dresden.de

Gunnar Dietz

Dongbei University of Finance and Economics, mail@gdietz.de

Follow this and additional works at: http://aisel.aisnet.org/amcis2011_submissions

Recommended Citation

Juhrisch, Martin; Schlieter, Hannes; and Dietz, Gunnar, "Model-Supported Business Alignment of IT — Conceptual Foundations" (2011). *AMCIS 2011 Proceedings - All Submissions*. 206.
http://aisel.aisnet.org/amcis2011_submissions/206

This material is brought to you by AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2011 Proceedings - All Submissions by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Model-Supported Business Alignment of IT — Conceptual Foundations

Martin Juhrisch

TU Dresden

Martin.Juhrisch@tu-dresden.de

Hannes Schlieter

TU Dresden

Hannes.Schlieter@tu-dresden.de

Gunnar Dietz

Dongbei University of Finance and Economics

mail@gdietz.de

ABSTRACT

Business Information Technology (IT) alignment focuses on the efficient support of business processes by IT. Therefore, existing software artifacts are addressed by business process models. When the processes change, however, there is a need to adjust the supporting software systems. Thus, already during the design phase of business process models, IT artifacts should to be considered. The instrument of conceptual modeling gains wide acceptance, especially in the health care sector to describe and manage clinical processes, such as Clinical Practice Guidelines (CPG) and Clinical Pathways (CP). There are no holistic approaches so far that provide the alignment between these two concepts and ensure the quality of treatment and the consistent adaptation of a Hospital Information System (HIS), in particular the hospital's Workflow Management System (WfMS). To link business process models and the WfMS, the Description Kit Approach (DKA) is used to prepare conceptual models to make them automatically analyzable. It is suggested that at an early stage of the modeling process the use of guidelines has a substantial benefit for avoiding integration conflicts in conceptual models. Furthermore, due to the way the approach bridges the semantic gap, changes of business requirements as well as technical implementation restrictions influence each other. This results in an ongoing system development process that can be interpreted as a permanent management of application systems. Our results contribute to model-based management theories that have so far neglected the distributed construction of conceptual models.

Keywords

Business Alignment, Conceptual Modeling, Hospital Information Systems, Clinical Practical Guidelines, Clinical Pathways, Description Kit Approach.

INTRODUCTION

A Hospital Information System (HIS) is a holistic information processing and storage subsystem of a hospital. This term not only refers to the computer-based parts. Amongst other things, it includes the Workflow Management System (WfMS) that allows to support the business processes using information technology. The management of an HIS is a major challenge, especially in hospitals where cost pressure and technological progress demand that the HIS has to be implemented in short time by the available staff.

Conceptual models, in particular process models, are increasingly used in health care for the documentation of medical recommendations and requirements to the process-oriented organization. Clinical Practice Guidelines (CPGs) are created to aggregate medical knowledge; Clinical Pathways (CPs) serve to optimize the clinical processes. Last but not least the integration of CPs into the HIS — particularly the automation — still raises open questions. Both organizational and technical obstacles have to be overcome. Technical barriers arise from the requirement that dynamically changing CPs have to be implemented into the HIS. Organizational challenges are to implement CPGs into CPs and to secure their consistency. A mapping by clinical algorithms under consideration of additional information such as levels of evidence or rules of liability onto a CP is the main concern for a model-based procedure. To support this task methodically, conceptual models are used for the description of CPG and CP.

However, even with these established techniques the implementation remains a huge organizational effort. The models facilitate the communication between the participants in a project, but they are typically not used for further development. To reduce administrative costs, a direct link between the CPGs, the CPs and the HIS, which runs the automated part of a CP, has been implemented with help of the Description Kit Approach (DKA). There are two basic procedures to bridge the gap

between the medical experts and the clinical processes, which mainly result from the orientation of the guideline on medical aspects and the differing degree of abstraction (Schlieter and Esswein, 2010)).

On the one hand, a CPG is taken into account as a source of condensed evidence; on the other hand, a CP is directly derived from a CPG. On the other hand, approaches focus not so much on the derivation but more on the formal preparation of clinical recommendations. Thus, the flow logic of the clinical algorithm is extracted and converted into a machine-readable language (Kaiser et al., 2007).

Actually there are no holistic approaches that provide the adaptation between the concepts of CPG and CP and exploit the full potential for ensuring the quality of medical treatments and the consistency of the HIS configuration — especially the WfMS. Prepare your submissions on a word processor or typesetter. Please note that page layout may change slightly depending upon the printer you have specified.

THEORETICAL BACKGROUND

Clinical Practice Guideline

Since about 20 years, CPG are the central element for the practice of evidence-based medicine. CPG are defined as “systematically developed statements to assist practitioner and patient decisions about appropriate health care for specific clinical circumstances” (Field and Lohr, 1990). Wollersheim et al. characterize the content and contribution of CPG as the inclusion of specific recommendations and sufficiently supporting evidence with a clear structure and an attractive layout (Wollersheim et al., 2005). The term CPG is used differently and is often not clearly distinguished from related concepts like medical protocols, standard operation procedures or clinical pathways. CPG can be developed either locally (internal guidelines) or regionally or nationally (external guidelines) (Thomas, 1999). Hence, several CPGs for same clinical case with different designs and representations exist.

Currently there are so far no modeling language specifications that can be used for the design of CPGs. Some umbrella organizations, e.g. in Germany, recommend the standard nomenclature of the Society of Medical Decision Making (Barak et al., 1992). Depending on the preference of the development team and the specificity of the symptom complex are used either entirely prosaic descriptions applied or any combination of the text, tables, checklists and conceptual models. Figure 1 depicts the modeled section of the CPG for interdisciplinary treatment of mastocarcinoma.

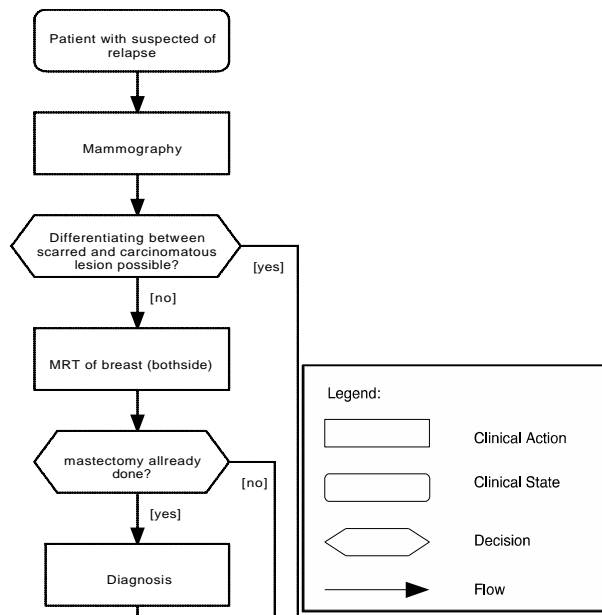


Figure 1. Part of the Clinical Practice Guideline for the treatment of mastocarcinoma — section: Diagnosis of local/locoregional recurrence

Clinical Pathways

CPs are fundamental for a process management with the aim of achieving economical and qualitative objectives of hospitals (Panella et al., 2003; Moy, 1997). They are also an instrument to bring knowledge from clinical practice into evidence (Rotter et al., 2008). The European Pathway Association defines Clinical Pathways as a “methodology for mutual decision making and organization of care for a well-defined group of patients during a well-defined period” (Bleser et al., 2006). Characteristics of care pathways are: “An explicit statement of the goals and key elements of care, based on evidence, best practice, and patient expectations; the facilitation of the communication, the coordination of roles, and the sequencing of activities of the multidisciplinary care team, patients, and their relatives; the documentation, monitoring, and evaluation of variances and outcomes; and the identification of the appropriate resources” (Bleser et al., 2006).

The terms Critical Pathway, Clinical Pathway or Integrated Care Pathway are used for this concept synonymously. But, until now, no real consensus was found in the scientific community. Along the clinical processes, clinical pathways include administrative, nursing, as well as other management processes (Weiland, 1997).

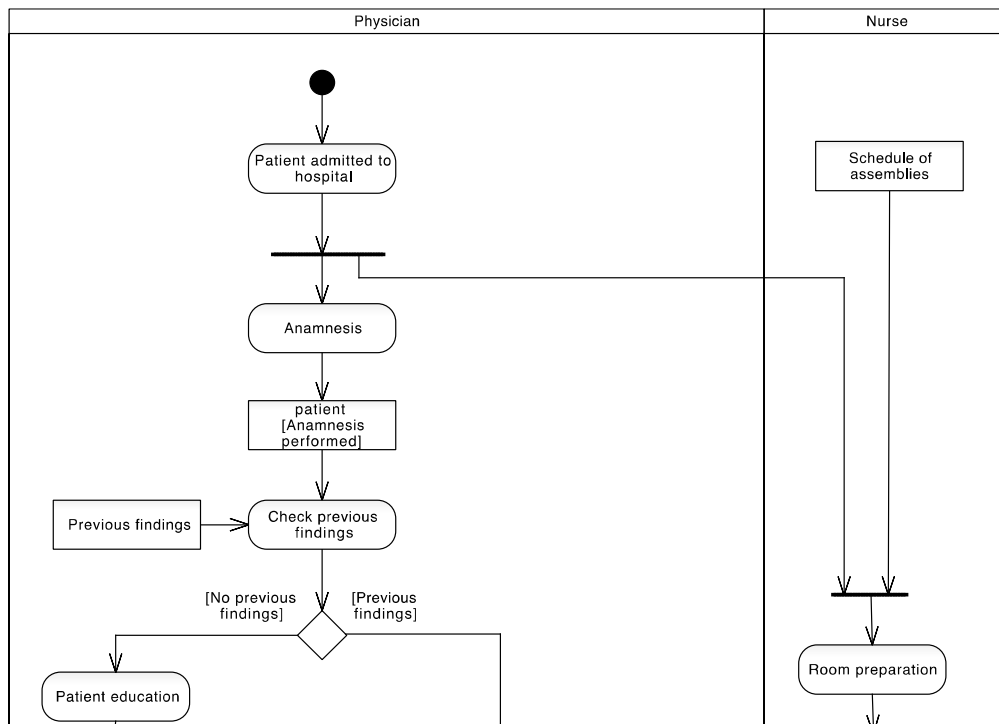


Figure 2. Part of the Clinical Pathway for the treatment of mastocarcinoma as an UML activity diagram

Quality characteristics of clinical pathways are the accuracy, comprehensiveness, actuality and availability. This means that they must include medical evidence. Therefore, they must be compliant as possible to the CPG. The design must be clear and unambiguous to avoid misinterpretation. Furthermore, clinical pathways must be accessible to the users. Typically, the CPs are modeled with the flowchart notation, Business Process Modeling Notation (BPMN), or as Activity Charts of the Unified Modeling Language (UML) OMG (2010), or are proprietarily developed in-house. Figure 2 depicts a CP that corresponds to the CPG for the treatment of mastocarcinoma.

Implementation of Clinical Practice Guidelines

The following section provides the actual work in the field of integrating process models in health care sector. There is an ideal synergy between the CPGs and CPs, which consists in the aggregation of evident knowledge on the one hand, and the dissemination of that knowledge on the other hand (Schlieter and Esswein, 2010; Wollersheim et al., 2005; Priori et al., 2003; Weiland, 1997). Several methodological approaches for the development of CPs based on the CPGs exist. One approach is to formalize the CPG in a computer-interpretable (CI) form (Kaiser et al., 2007; Peleg et al., 2003). Arden-Syntax or XML are

markup languages to prepare the narrative CPGs in a CI format. The aim of these approach is to transfer the non-formal rule-system of the CPG to the decision support systems (DSS) of the HIS (Health Level Seven International, 2010). Such methods are for example ProForma, GLIF, GUIDE oder Prodigy (Peleg et al., 2003). The DSS supplements the CPs within a hospital. Therefore the hospital information systems (HIS) have to be able to interpret the formal language and to combine it with the specific state of the patient documented in electronic health record (EHR). The approach of that transformation of CPG into CI structures is thus rather an extension of the paths than a methodology of a systematic transfer and organizational implementation.

A further approach to take CPGs into account for CPs is the inclusion of the analysis of a CPG in its research phase at an early project stage. Project members need to analyses the existing literature. In order to ensure a high quality of the analysis results several iterations are necessary.

Another approach, which represents an intermediate way between the formal procedure of CI transformation and not a systematic literature review, are model-based methodologies to transfer CPG-knowledge in clinical pathways. Jacobs et al. present this idea for implementation of clinical pathways in treatment of breast cancer (Jacobs et al., 2007). Reference models are the central element of their idea. They are used as an instrument to structure and transfer the CPG content.

Conceptual Modeling

After we introduced the two main instruments which we are aim to integrate. The following section addresses the state-of-the-art in conceptual modeling and model comparison.

CPGs and CPs can be described with the aid of conceptual process models (for example UML activity diagrams). Such models represent the behavior of a system as a sequence of states, transitions and conditional branches. Since these models are characterized by a low level of formal precision, only simple operations can be performed on them automatically, such as consistency checks.

The terminus “modeling” can be understood as reconstructing an expert’s specific problem perception, expressed by using a conceptual modeling language. A conceptual modeling language is the combination of a technical language with a modeling grammar, and results in the creation of conceptual models (that represent “sentences” of the language).

We will assume here that modeling grammars always include a diagrammatic notation, so that a model refers to a graphical representation (Gurr and Toulas, 2000; Frank, 1999; Wand and Weber, 2002).

A domain specific (modeling) language (DS(M)L) is a (modeling) language where termini of the technical language become even part of the (modeling) grammar (Becker et al., 2007; Pfeiffer, 2008). The use of a DSML outside its domain therefore does not make sense. Not only the constructed model has therefore a semantic connection to a certain domain, but also already the modeling language (Pfeiffer, 2007). In contrast to languages like UML or EPC this increases the amount of elements that can be formally evaluated. This increased level of formalization increases possibilities for an (automated) operationalization of such models by semantically meaningful model operations.

Empirical research shows that models that are created by a distributed modeling process differ with respect to the employed vocabulary, the degree of abstraction and the scope of the covered domain (Hadar and Soffer, 2006). The high degree of freedom when modeling and the lack of high standardization when using semi-formal languages causes several integration conflicts when trying to combine distributed developed models (Pfeiffer and Gehlert, 2005; Pfeiffer, 2007).

For a successful comparison it is necessary to detect corresponding parts of the models that should be compared. It is likewise easy to combine two models when semantically equivalent elements within these models already have been identified. As seen, the creation of linguistic communities and the restriction of the freedom of modeling can help to ensure semantic comparability of models. Only by the common use of certain linguistic concepts with a well-understood meaning (i.e. a consensus within a linguistic community), a comparison and therefore mapping of semantically equivalent model parts has a change to be possible in an at least semi-automated way. This includes a normalization of the technical language with help of a communication process between all participating persons.

The current literature always assumes that model comparisons are always done without any transition of the described domain. This means that both models are embedded within the same domain and describe essentially (parts of) the same. However, in a real business context this is rarely the case. Domains can be quite different and reflect for example different business areas, different organizational units, different language, different levels of knowledge, or different culture. Domains can also span different organizations or different languages. The viewpoint on a certain problem depends highly on the domain it is embedded in. One main example from software development is the comparison between requirements models (which are often discusses on a managerial level of a company) and implementation models (which are often highly

technical). This example also shows that one often has to compare models using semi-formal languages with models using formal languages.

Daily business practice is full of decisions that are based on comparing information from different domains. (Example: “Should we invest our money to buy another company or to develop a new product line on our own?”). The necessary data and information for such a decision has to be prepared and analyzed for such decisions. This means that comparability has to be created. Models offer a good way to prepare and structure data, but comparability is still a problem when models come from different domains. They were often created without already having the now defined question in mind. The vision would be a comparability without any need for preparing the data (and therefore change the models) for a comparison. It would therefore be useful to create models already in a way that prepares for a comparison between different domains.

The dilemma that arises from the desire to describe different things, but nevertheless with still the need to describe a mapping (or weaker said some kind of relationship) between these models, is subsumed in the domain conflict; it represents the obstacle that prevents an easy model comparison (Jührisch, 2010; Dietz et al., 2010a).

Domain conflict: A domain conflict results from the desire to compare models out of different domains. The conflict is a consequence of a varying way of modeling in different domains or occurs because different kinds of artifacts are compared.

Domain conflicts have an effect on the way of modeling the artifacts (objects or phenomena) that should be compared, since within different domains quite different viewpoints and therefore different ways to describe artifacts may exist. As a consequence, the artifacts may have been described with different visualization techniques, different terminology, different degrees of abstraction or details, or different degrees of formalization. This is due to the intended use of these models in different domains.

The identification of model elements (artifacts) by technical terms from a natural language holds a potential for conflicts that arises from a change of the domain. The question is, how semi-formal models that describe a phenomenon (or object) in different domains can be used for a model comparison. The description of a real-world phenomenon is done in natural languages (Kaindl, 2005). Natural languages are decomposed into subject, predicate, and object, and are put into a conceptual model in a form that is given by the modeling grammar. A common use of language is not sufficient for a model comparison between different domains. Only if the essence of the phenomenon that is described in the models can be understood, the avoidance of language conflicts and structure conflicts makes sense. The domain conflict forces the modeler to create a conceptual description of an artifact that is not independent from the real world phenomenon (object), but needs a traceable connection between conceptual description and real phenomenon.

Under the assumption that the understanding of a sentence (representing a real world phenomenon) highly depends on an understanding of the subject and the objects, one can say that only by understanding the objects that are involved in a (business) process, together with their status, is an understanding of what happens with these objects possible. Without an understanding of the objects an understanding of the predicate is meaningless.

To resolve the domain conflict the first step therefore to establish a linguistic community with a consensus about the terminology that should be used in the models. Models in different domains therefore should use a common terminology nevertheless, and the semantic of the terminology must be clear to all sides (all roles that are participating in the modeling process). This involves of course a practicability of the commonly used language for each individual domain. In the next chapter we will discuss some ways to avoid the here mentioned conflicts and highlight an approach that especially addresses the domain conflict.

Description Kit Approach

The Description Kit Approach (DKA) attempts to reform the classical modeling process towards a goal-oriented modeling using guidelines. This allows us to create semi-formal models in the sense of solving a specific problem (or to enrich existing models) and algorithms, which enables an — at least — semi-automatic problem solving. Furthermore, the DKA allows cross-domain but still domain-specific modeling. This is achieved through guidelines that help the modeler to describe domain-specific phenomena in a way that these models remain nonetheless algorithmically comparable across their domain boundaries. These guidelines ensure that constraints, in terms of a restricted modeling, are enforced, but do so without limiting the modeler’s freedom to describe whatever he wants to say. Detailed introductions to the DKA can be found at (Jührisch and Dietz, 2010a; Jührisch et al., 2009; Jührisch, 2010; Jührisch and Weller, 2008).

The DKA has similarities to meta-modeling, but introduces an additional intermediary layer on which the guidelines are introduced. These guidelines act as some kind of “glue” between language creation on the meta-modeling layer and language use (the modeling itself) on modeling layer. On this new intermediate layer Description Kits (DK) are located representing the conceptual reflection of the above mentioned guidelines. DKs are domain-specific frameworks in which certain

phenomena have to be described. Describing phenomena then is done by creating a so-called Descriptions (Descs), which are compliant to their corresponding Description Kit.

Classical conceptual models can be enriched by means of Descriptions; or — even more advantageous — models can be created that only consist of Descriptions and relations between them (a so-called “pure Description Kit Language”). The compliance to the guidelines is enforced by specifications that are made by the Description Kits. One big advantage of the DKA is the possibility to embed Descriptions in other Descriptions and therefore the ability to create hierarchic descriptions of phenomena with different levels of detail. The Description Kits include specifications for embeddings that are allowed on the Description level, but also constraints for parameters and values.

However, the Description Kits are not constituting the modeling language itself — in contrast to domain-specific modeling languages —, but represent merely the guidelines for using the language. The language itself is constituted “above” the Description Kits in form of the so-called Description Kit Types. They represent the actual concepts — on this level still domain-unspecific — that should be available for the modeling.

As for classical modeling the actual modeling (language usage) is done after the language has been created (language creation), which here means the creation of the Description Kit Types. Unique about the DKA is now, however, that the adaptation of the language to a corresponding domain is not done anymore during the language creation, but is done by introducing guidelines for describing within this domain. These guidelines are still adaptable during the modeling process (language usage) without any need — as for classical meta-modeling — for adapting the modeling language itself. This allows an adaptability of the language usage to a certain domain without invalidating existing models by changes of the languages itself. For an explicit description of the procedure model for using the DKA and its role model we refer to Juhrisch and Dietz (2010b).

The DKA ensures that the modeling is done with help of guidelines that ensure that a domain specific description of a certain phenomenon still has a link to the concept it represents. This allows that models created with the DKA are understandable even outside their domain, in which they were created, and therefore are comparable in a domain spanning way. For such comparisons the DKA includes algorithms that can perform such a mapping. This mapping is done in several steps: A 1:1-mapping-algorithm allows to compare two single Descriptions — which however already includes the whole embedding hierarchy. To compare two complete models and parts of models (“sub-models”) a second algorithm, called “convolution” is added. This algorithm allows to “fold” whole (sub-)models into a single (artificial) Description that includes all necessary information of the folded part. This convolution is done along the relations and uses the 1:1-mapping algorithm for calculation the resulting Description. The results of this convolution process then can again be compared by the 1:1-mapping algorithm. Further algorithms also allow the generation of ratios for (sub-)models that represent a rating of the model with respect to certain criteria.

Especially the possibility to nest Descriptions creates a good foundation to not only create detailed descriptions of certain phenomena, but also a much better comparability of such two descriptions (within two different domains and with probably different levels of detail) than classical methods (parameter and values, free text, ontologies). A detailed descriptions of the algorithms can be found in Dietz et al. (2009); Juhrisch and Dietz (2010a).

The combination of the guidelines and the algorithms allows the solution of certain modeling conflicts (especially the domain conflict) that normally complicate or even prevent a comparability. Many approaches (like ontological approaches) try to repair these conflicts when it is “too late”, i.e. when they already happened. This means often quite high additional efforts (like the creation of ontologies that also have to be maintained). The DKA allows to prevent these conflicts before they happen without the need of creating a language that is only valid in one domain. Only by the high quality of models created with the DKA — due to the guidelines — it is possible to at least semi-automize the evaluation of models, and therefore to allow a model-based configuration of application systems. The DKA has been used already for the configuration of service-oriented architectures (Dietz et al., 2010b) (in different scenarios — especially the service identification) as well as the configuration of identity management systems (Dietz and Juhrisch, 2010).

INTEGRATION FRAMEWORK

In this section, based on the methodology of the DKA, medical processes of the HIS are validated. The basic idea is to actively adapt the processes of the hospital to CPGs, while ensuring the support of the automated part of the HIS. Models that represent CPGs or CPs must be considered as models from different domains, despite their common embedding into the medical context. They are characterized by different ways of representation, but especially by the fact that CPGs focus almost purely on medical aspects and CPs include the context of a specific medical facility and organizational aspects. Moreover, some contents of a CPG can be implemented differently depending on their configuration in a concrete hospital. A

comparison of such models is therefore not trivial. The DKA is well-suited for such a modeling purpose due to its domain-crossing nature.

At first, Description Kit Types must be created for the present use case that reflect the actual language. Essential are the DKTs “Patient”, “Therapy”, “Examination”, “(organizational) Activity”, and “Record”. These DKTs embed other DKTs, e.g. “Status”, “Diagnosis”, “Organ System”, “Institution” (see figure 3). For these DKTs, Description Kits must be derived that represent guidelines for the modeling. These guidelines specify, how a particular concept reflected by the above-mentioned DKTs should be used on the modeling level. The above concepts are concretized and each case contains information about what is needed for a description of that concept and how it should be described. This includes pre-settings and constraints for parameters and values and (other) guidelines for the embedding. See Figure 4 for an exemplary list.

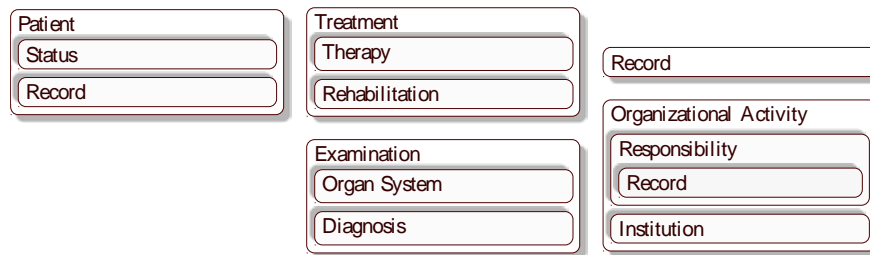


Figure 3. Description Kit Types

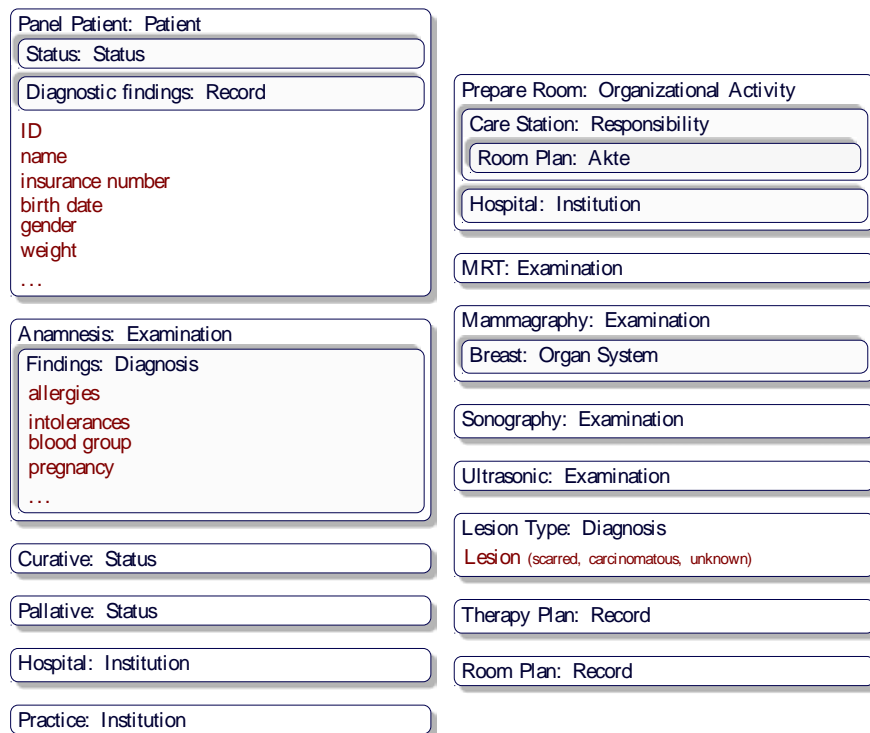


Figure 4. Description Kits

Especially the hierarchical approach here provides excellent conditions to describe facts in any level of detail by embedding more explicit Descriptions into coarser Descriptions — without risking (when comparing) a detail conflict, since it is still one Description.

The modeling is done with help of the obtained DKs, which are instantiated to concrete Descriptions. These can be used to annotate the existing models. In the future, a modeling purely by such Descriptions may be desirable.

During the modeling activity, the level of the DKs can be adapted at any time, for example when new DKs (e.g. for new therapy) are added. Each DK is characterized by its DKT (not necessarily visible for the modeler). That makes it possible to decide immediately whether a model element is an examination, a therapy or an organizational activity.

The effect of guidelines is (among other things) as follows: The modeler is forced to precisely describe each model element according to the guidelines. This means that he has to specify the meaning of model elements (e.g. a decision node) by providing all necessary information for this modeling element (e.g. what is necessary to do the decision at a decision node). See figure 5 for an example of both an annotated CPG and CP.

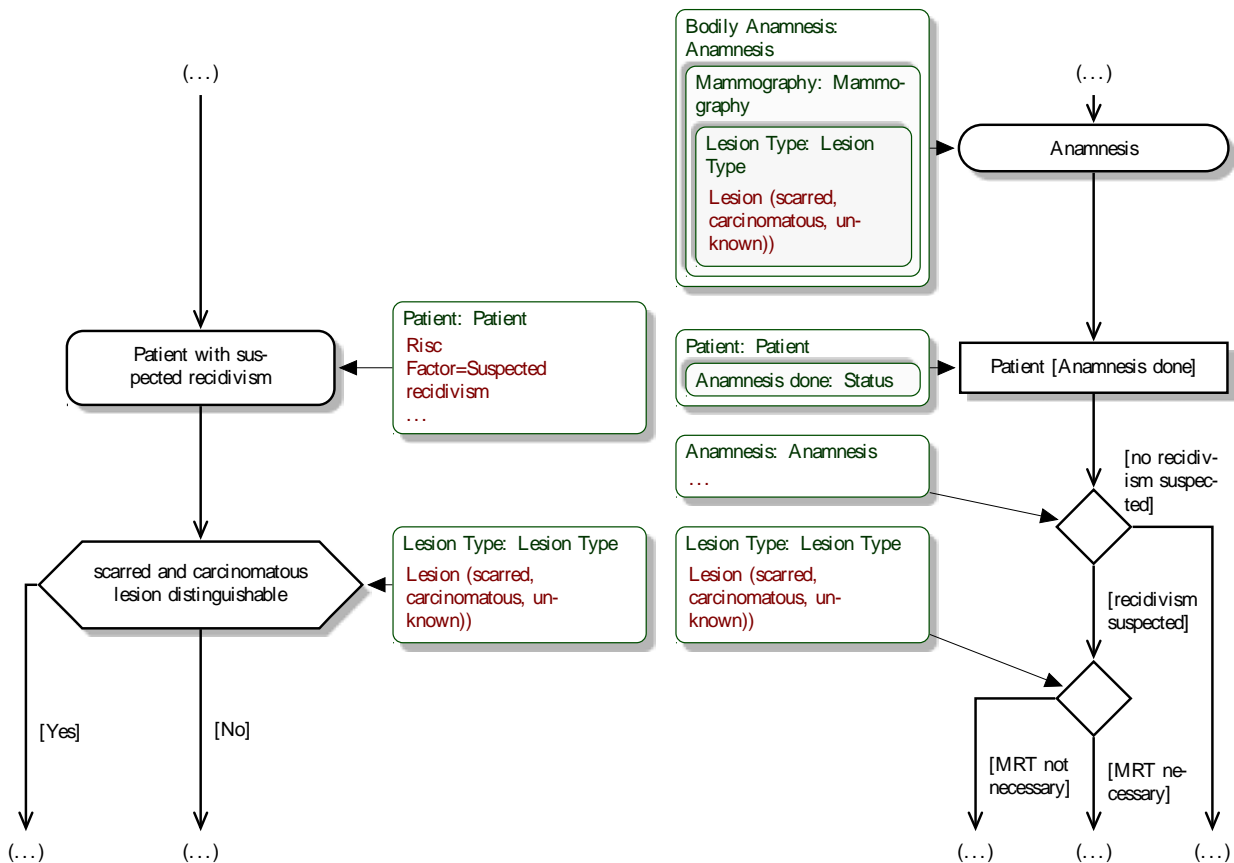


Figure 5. Clinical Practice Guideline and Clinical Pathway annotated by Descriptions

Now, the created models allow a comparability check between CPGs and CPs with the aid of the algorithms of the DKA. Since the algorithms are adaptable at DKT level (Juhrisch, 2010), they can be controlled to ensure the following: Any information in the CP is evaluated as insignificant for the comparison operation, if this information is of no relevance in the CPG (in particular all organizational activities). More generally, some aspects in the models can be evaluated as more important, other as (rather) unimportant for the comparison. This possibility to compare can now be used in different scenarios.

- Find a CPG to a given CP: “What are the guidelines for certain procedures in our department?”
- Find a CP to a given CPG: “Do we consider a certain guideline in our hospital?”
- Compliance checking: Does a given CP comply with the related CPG(s)? Where are inconsistencies? If something is missing, what is missing?

Thus, in a semi-automated process, the creator of a CP will be guided to produce a pathway that complies with the CPG(s). The modeler still can describe situations in his usual manner: The mapping algorithm is fuzzy and it can handle Descriptions

with completely different level of details because of the hierarchical refinement. (The convolution algorithm further contributes to this fact and allows even one-to-many comparisons beyond the embedding). Because of the higher informational content of CPGs (which are characterized with help the DKA), model transformations are possible that create almost a “skeleton” for a CP automatically. These have to be further enriched and must be adapted to the needs of a particular institution; especially some additional facts (in particular organizational) have to be added.

The Descriptions in a CP are available for an electronic access through the HIS. Thus, we created the basis for a flexible architecture for the management of HIS by using conceptual models. The HIS is either being parameterized by the model information directly or adapted indirectly by a derivation of a workflow from the automated part of the CP.

CONCLUSION

The paper has shown that the fuzziness in the description of CPGs and CPs leads to inconsistencies and conflicts when attempting to automate a model comparison or transformation. The further use of models of one domain in another domain requires to resolve these conflicts subsequently or respectively to be able to resolve them at all. This is often highly complex, so that no further use of the models occurs — as it is currently the case for the implementation of CPG to CP and CP to the WfMS.

Therefore the restriction of the freedom of modelling is often discussed in the literature to counter the missing standardization and to enhance the comparability of models. However, this is only possible at the expense of a decreased usability of these models in their domains. In this study we prefer a more recent position that cannot directly be deduced from semiotics. This approach “philosophically” bases on the Use Theory (Wittgenstein, 2000). To be able to express certain concepts embedded within one of the three domains — medical association, hospital or HIS — models have to be domain specific on the one hand, but comparable to models in other domains on the other hand. The DKA offers a framework for restricted modeling without destroying the adaptability to certain different domains. The methodology includes algorithms for comparing models in different domains and is therefore capable to dissolve not only certain standard comparability conflicts but also the domain conflict. The work extends the previous application field of DKA of software development to clinical process models. We assume that the approach is also useful to assure process compliance in virtual organizations like flexible supply chains. This will be a direction we will investigate in the future.

If CPG and CP are now described under the influence of guidelines, the model information can be evaluated automatically, which not only promises a more effective development process but also simplifies to ensure a maximum consistency of CPG, CP and the WfMS. The usefulness of the method was demonstrated in a proof-of-concept conducted at the University Hospital Carl Gustav Carus at the University of Technology in Dresden. In the further research we will carry out a greater empirical analysis of the effectiveness and the efficiency of the approach.

The discussion of the ability of to improve clinical processes has two sides. One is the technical part which is addressed by the paper. The other side is the subjective dependency of the acceptance and the use of such concepts. The effectiveness of the proposed approach depends on the subjective attitude of the physicians. The majority of physicians apparently believe that CPGs and CPs are helpful to improve the clinical decision making (Gaddis et al., 2007). However, there are a minority of the physicians, who do not accept these instruments. Subjective barriers must be addressed individually by measures to promote acceptance.

REFERENCES

1. Barak, N., Margolis, C., and Gottlieb, L. (1992). Proposal for clinical algorithm standards: Society for medical decision making committee on standardization of clinical algorithms. *Medical Decision Making*, 12(2):149–154.
2. Becker, J., Pfeiffer, D., and Räckers, M. (2007). Domain specific process modelling in public administrations — the picture-approach. In *Proceedings of the 6th International EGOV Conference*, pages 68–79, Regensburg, Germany.
3. Bleser, L. D., Depreitere, R., Waele, K. D., Vanhaecht, K., Vlayen, J., and Sermeus, W. (2006). Defining pathways. *Journal of Nursing Management*, 14(7):553–563. PMID: 17004966.
4. Dietz, G. and Juhrisch, M. (2010). Model-based management — design and experimental evaluation. In *Proceedings of the 14th Pacific Asia Conference on Information Systems (PACIS 2010)*, Taipei, Taiwan.
5. Dietz, G., Juhrisch, M., and Esswein, W. (2009). On the restriction of conceptual modeling — outlining an approach to enable business driven soa. In *Proceedings of the 15th Americas Conference on Information Systems (AMCIS 2009)*, pages 1–14.

6. Dietz, G., Juhrisch, M., and Leyking, K. (2010a). Convolution as the key for service-identification in complex process models. In Proceedings of the ISD 2010 (to appear), Prague, Czech Republic.
7. Dietz, G., Juhrisch, M., and Leyking, K. (2010b). Convolution as the key for service-identification in complex process models. In *Proceedings of the ISD 2010*, Prague, Czech Republic. Springer Verlag.
8. Field, M. J. and Lohr, K. N. (1990). Clinical practice guidelines: directions for a new program. National Academies Press.
9. Frank, U. (1999). Conceptual modelling as the core of the information systems discipline - perspectives and epistemological challenges. In Haseman, D., Nazareth, S., and Goodhue, D., editors, Proceedings of the Fifth America's Conference on Information Systems (AMCIS 99), pages 695–697, Milwaukee.
10. Gaddis, G. M., Greenwald, P., and Huckson, S. (2007). Toward improved implementation of evidence-based clinical algorithms: Clinical practice guidelines, clinical decision rules, and clinical pathways. *Academic Emergency Medicine*, 14(11):1015–1022.
11. Gurr, C. and Toulras, K. (2000). Towards the principled design of software engineering diagrams. Proceedings of the 2000 International Conference on Software Engineering, pages 509–518.
12. Hadar, I. and Soffer, P. (2006). Variations in conceptual modeling: Classification and ontological analysis. *Journal of the Association for Information Systems*, 7(8):568–592.
13. Health Level Seven International (2010). Arden syntax. <http://www.hl7.org/implement/standards/ardensyntax.cfm>.
14. Jacobs, B., Oberhoff, C., and Stausberg, J. (2007). Ableitung von klinischen pfeiden aus evidenzbasierten leitlinien am beispiel der behandlung des marmakarzinoms der frau. *GMS Medizinische Informatik, Biometrie und Epidemiologie*, 3:1–10.
15. Juhrisch, M. (2010). Richtlinien für die modellgetriebene Integration serviceorientierte Architekturen in Analysemodellen. Phd thesis, Technische Universität Dresden.
16. Juhrisch, M. and Dietz, G. (2010a). Constraints in conceptual modelling — outlining an approach to business driven web service composition. *Int. J. Internet and Enterprise Management*, 6(3):248–265.
17. Juhrisch, M. and Dietz, G. (2010b). Context-based modelling: Introducing a novel modeling approach. In Esswein, W., Turowski, K., and Juhrisch, M., editors, *Modellierung betrieblicher Informationssysteme (MobIS 2010)*, Dresden, Germany. Gesellschaft für Informatik.
18. Juhrisch, M., Dietz, G., and Esswein, W. (2009). Perspectives on semantic business process modeling — a generic approach. In *Proceedings of the 13th Pacific Asia Conference on Information Systems (PACIS 2009)*.
19. Juhrisch, M. and Weller, J. (2008). Connecting business and it: A model-driven webservice based approach. In *Proceedings of the 12th Pacific Asia Conference on Information Systems (PACIS)*, pages 1469–1479.
20. Kaindl, H. (2005). Is object-oriented requirements engineering of interest? *Requirements Engineering*, 10(1):81–84.
21. Kaiser, K., Akkaya, C., and Miksch, S. (2007). How can information extraction ease formalizing treatment processes in clinical practice guidelines? *Artif. Intell. Med.*, 39(2):151–163.
22. Moy, J. (1997). Development of clinical guidelines. In *Introduction to Clinical Skills*, pages 449–465. Springer.
23. OMG (2010). UML 2.3 infrastructure. <http://www.omg.org/spec/UML/2.3/>.
24. Panella, M., Marchisio, S., and Stanislao, F. D. (2003). Reducing clinical variations with clinical pathways: do pathways work? *International Journal for Quality in Health Care*, 15(6):509–521.
25. Peleg, M., Tu, S., Bury, J., Ciccicarese, P., Fox, J., Greenes, R. A., Hall, R., and Johnson, P. D. (2003). Comparing computer-interpretable guideline models: A case-study approach. *Journal of the American Medical Informatics Association*, 10:52–68. PMID: 12509357 PMCID: 150359.
26. Pfeiffer, D. (2007). Constructing comparable conceptual models with domain specific languages. Proceedings of the 15th European Conference on Information Systems (ECIS2007), pages 876–888.
27. Pfeiffer, D. (2008). Semantic Business Process Analysis - Building Block-based Construction of Automatically Analyzable Business Process Models. Phd thesis, Westfälische Wilhelms-Universität Münster.
28. Pfeiffer, D. and Gehlert, A. (2005). A framework for comparing conceptual models. In Desel, J. and Frank, U., editors, *Enterprise Modelling and Information Systems Architectures: Proceedings of the Workshop in Klagenfurt, number P-75 in Lecture Notes in Informatics*, pages 108–122, Bonn. Köllen Druck + Verlag GmbH.

29. Priori, S. G., Klein, W., and Bassand, J. P. (2003). Medical practice guidelines: separating science from economics. *European heart journal*, 24(21):1962.
30. Rotter, T., Kugler, J., Koch, R., Gothe, H., Twork, S., van Oostrum, J., and Steyerberg, E. (2008). A systematic review and meta-analysis of the effects of clinical pathways on length of stay, hospital costs and patient outcomes. *BMC Health Services Research*, 8(1):265.
31. Schlieter, H. and Esswein, W. (2010). From clinical practice guideline to clinical pathway - issues of reference Model-Based approach. In Camarinha-Matos, L. M., Boucher, X., and Afsarmanesh, H., editors, *Collaborative Networks for a Sustainable World: 11th Ifip Wg 5.5 Working Conference on Virtual Enterprises*, page 886, St. Etienne, France. Springer.
32. Thomas, L. (1999). Clinical practice guidelines. *Evidence Based Nursing*, 2(2):38–39.
33. Wand, Y. and Weber, R. (2002). Research commentary: Information systems and conceptual modeling—a research agenda. *Information Systems Research*, 13(4):363–377.
34. Weiland, D. E. (1997). Why use clinical pathways rather than practice guidelines? *The American Journal of Surgery*, 174(6):592–595.
35. Wittgenstein, L. (2000). Philosophische Untersuchungen. In *Sprachwissenschaft: Ein Reader, Volume 2. verb. Aufl.*, pages 72–78. Walter de Gruyter Verlag, Berlin.
36. Wollersheim, H., Burgers, J., and Grol, R. (2005). Clinical guidelines to improve patient care. *The Netherlands Journal of Medicine*, 63(6):188–192. PMID: 16011009.