Guarded Process Spaces (GPS): A Navigation System towards Creation and Dynamic Change of Healthcare Processes from the End-user's Perspective

Claudia Reuter[1], Peter Dadam[2], Stephan Rudolph[1], Wolfgang Deiters[1], Simon Trillsch[3]

[1] Fraunhofer Institute for Software and Systems Engineering, Emil-Figge-Str. 91, 44227 Dortmund, Germany
{claudia.reuter, stephan.rudolph, wolfgang.deiters}@isst.fraunhofer.de
[2] Ulm University, Albert-Einstein-Allee 11, 89081 Ulm, Germany
peter.dadam@uni-ulm.de
[3]University Hospital of Giessen and Marburg, Baldingerstr., 35033 Marburg, Germany
trillsch@students.uni-marburg.de

Abstract. Efficient process management becomes more and more crucial for hospitals to survive on a competitive market. Therefore, process management must comply with individual conditions of patients and quickly react to changing requirements and organizational parameters. With Guarded Process Spaces (GPS) we developed a formally based concept that makes it possible to enable end-users to create and flexibly change processes from their point of view. Our approach makes use of existing BPM technology while abstracting from technical interfaces and modeling paradigms. In this way, it provides the basis to gain user acceptance and to achieve technological independence.

Keywords: Healthcare process, clinical pathway, process flexibility

1 Motivation

Today, healthcare providers are facing the challenge of delivering high-quality services while coping with increasing costs due to demographic change and medical progress. In response to this, hospitals start with the introduction and deployment of standard processes (so called "clinical pathways") that are bound to organize the treatment of patients according to a common set of symptoms, a diagnosis, or a therapy. In principle, modern BPMS (Business Process Management Systems) could help to support clinical pathways in practice and to reduce administrative workload by instantiating pathway templates for patients, documenting and monitoring their progress, and managing work lists for doctors and nurses. However, in spite of their potential benefit, BPMS are barely in use in healthcare environments until now.

Originally, BPMS have been developed to support production processes in industry. In such settings processes typically shall be executed exactly as preplanned to ensure that the goals in terms of quality and cost are met for all the products. In clinical pathways, however, the focus must be put on the patient as individual being. Therefore, that approach to use BPMS to support clinical pathways must be able to solve the conflict between standardization of treatment processes on the one hand and flexible deviation from standards due to case-based considerations on the other hand.

The fact that healthcare processes pose challenges to traditional BPMS is not new to the scientific community; frequently, they even serve as motivation for researchers to investigate new approaches [1, 2]. In fact, flexibility and adaptivity during process execution are broadly addressed in BPMS related research in the meanwhile [3]. Therefore, in the near future, we can expect that BPMS come onto the market, which allow for more process flexibility at runtime. However, supporting flexibility at the BPM system level and making this feature usable by end-users are two different stories. A direct interaction with a BPMS, e. g., to insert or to postpone a task requires profound technical skills, which the medical staff is not able and willing to acquire. In addition, as there is no commonly accepted approach how to handle flexibility at the BPM system level, the process modeling and flexibility concepts offered by such BPMS vary to a great extent. This means, that the acquired knowledge would be related to one specific BPMS and could not be easily applied to another one. And even worse, when replacing an existing BPM solution by a new system (e. g., due to technological advances), the end-users not only would have to get familiar with another system philosophy and other interfaces, but it could also mean that all the clinical pathways have to be modeled once again. Facing these problems, it is easy to understand why hospitals are still very reluctant to introduce BPMS for supporting clinical pathways.

Based on this analysis we can draw two conclusions: Firstly, one must find a solution, which enables end-users to flexibly adjust clinical pathways according to the individual demands of a patient, but does not force end-users to acquire deep system near skills in order to perform this task. Secondly, as BPM technology is still undergoing significant changes, one must find a way to make the implementation of clinical pathways, ad-hoc modifications, and related things independent from a specific BPMS; i. e., a comprehensive formal methodology with execution semantics is needed, which on the one hand completely covers the modeling of clinical pathways as well as all operations relevant to their execution (e. g. to perform choices, iterations, ad-hoc deviations). On the other hand it must provide all necessary information such that its models and operations can be mapped or translated to process templates and operations to any target BPMS, which satisfies a certain set of basic requirements.

In context of the SPOT project (Service-based and Process-oriented Orchestration Technology)[1] we developed a concept and a prototype demonstrating that the vision of such a methodology, which fulfills both requirements – ease of use for end-users as well as BPMS independence – can be realized. The concept is based on the notion of “Guarded Process Spaces” (GPS). The analogy to GPS navigation devices is intentional, because like such systems, which can answer the question “Which roads are available to me now?”, Guarded Process Spaces provide maps of possible directions a process can take and guide the user’s decision making as to which paths

[1] See http://www.spot.fraunhofer.de

they can follow to reach a valid goal. Of course, the final goal is a technical process template that can be executed by a chosen BPMS.

In this paper, we will focus on the benefit of GPS from the end-user's perspective. After a discussion of related work in chapter 2, we will introduce a novel navigation paradigm towards process modeling, which is realized based on GPS. In section 3 we will explain the technical implementation of GPS by way of a practical example from the healthcare domain. After that, we will discuss requirements on BPMS in terms of process flexibility and show, how these demands can be fulfilled by GPS in combination with existing approaches from chapter 2. Finally, we will give a short summary and an outlook on our future work.

2 Related Work

Process modeling languages used by process experts are usually too complicated for end-users. Therefore, domain specific languages (DSL) are developed to facilitate process modeling for end-users in their application domain, like, e. g., public administration [4], workflow-based web applications [5], integrated care [6], or medical guidelines [7]. Another example is the feature modeling approach for modeling variability in product families [8], which has already been applied to process management as well [9]. However, the provision of DSLs alone is not sufficient without ensuring that the technical process templates which are derived from such DSLs can be correctly executed by a BPMS. This means, that DSLs without a proper and suitable formal basis are not very helpful to achieve this goal. Due to the lack of formality of DSLs, they are often transformed into formal languages, like e. g. Petri Nets [10]. However, this doesn't prevent the creation of erroneous process templates, which have to be corrected afterwards. This, of course, delays the whole development process, decreases user acceptance, and is certainly not acceptable in case of ad-hoc change.

In recent years, the scientific community has made great technical advances especially with regard to dynamic process management. Several authors suggest maintaining the standard way of proceeding together with its variations within the same process template [11-13]. Using respective workflow patterns, placeholder activities or variation points, it is possible to indicate the positions, where alternative routes may be chosen or even created at runtime. The advantage of these approaches is that end-users don't have to comprehend the process template and the usage of change operations to deviate from the standard proceeding. The disadvantage is that such approaches require that the positions where alternative routes may be chosen have to be fixed in advance, which does not reflect the reality in healthcare. E. g., certain conditions, such as infections, can occur at different points of time. Therefore, these simple solutions are falling short of covering these demands. End-users must be able to flexibly change the process structure at runtime. I.e., there is no way around that they must understand the modeling paradigm and the applicable change operations. The question is which kind of modeling paradigm is suited for such users.

So far, business process modeling relies on two alternative paradigms. According to procedural languages, process designers are responsible for specifying the activities, which have to be performed, the order of execution as control flow, the data objects that are passed between activities and must conform to the technical implementation, and further aspects. The adaptive BPMS ADEPT2 provides such a procedural modeling language together with a set of well-defined change operations, which can be applied to add, move, or postpone activities [1, 14-15]. In contrast to many other adaptive BPMS, ADEPT2 ensures the structural and behavioral soundness of processes even after ad-hoc modifications at runtime [16]. In contrast to this, with BPMS like Declare [17], which pursue a declarative approach to process modeling, it is not required to determine an explicit control flow. Instead, the (implicit) control flow is the set of all possible execution orders which obey the specified constraints. As a side-effect, this also leads to more alternatives how to execute a given process at runtime. At first glance, the declarative paradigm seems to be superior and also much easier to understand. In essence, however, it just substitutes the control flow by constraint relations between activities and is even more complicated in case of non-trivial processes. Therefore, regardless which language is selected at the BPMS level, end-users cannot be expected to act as process designers according to one of these paradigms; they are by far too complex for them.

3 Guarded Process Spaces: Applying the Navigation Paradigm to Process Modeling

Our investigations during the SPOT project were driven by the following objectives:

- It must be possible for end-users to design process templates from their business point of view and to automatically execute them using a chosen BPM system.
- End-users must be enabled to change processes both at modeling time and at runtime on a case to case basis.
- The user interface for the creation and the modification of processes must remain the same even if the underlying technology changes.

Due to the fact that both the procedural and the declarative process modeling paradigm are too complicated for end-users, we have selected a different approach: We apply a navigation paradigm to process modeling which we call *Guarded Process Spaces* (GPS). Due to lack of space, we cannot describe this approach in detail in this paper. Instead, we will introduce GPS informally and point out their usage by the clinical staff. A detailed and formal description can be found in [18, 19], where GPS correspond to SPF type graphs.

As already indicated in the introduction, from a user's point of view a GPS acts like a navigation system, which uses a given set of streets to offer routes and alternative routes from the current location to the desired destination. At technical level a GPS consists of a set of nodes and has a tree-like structure. The nodes represent navigation points, which are used to implement rules on the selection of process activities. The root of the GPS represents the starting point, from where "travels" can be planned. Like a GPS system, the Guarded Process Space indicates all potential traveling options. By selecting an option, a user is travelling to the next navigation point within the coordinate network; there, further traveling options are available. With every step that a user takes, the amount of traveling options to approach the final destination decreases. The selected route from the starting point to the destination defines an executable clinical pathway.

We illustrate process modeling from the user's perspective by means of a practical example. During their inpatient stay, some patients experience shortness of breath, the reason for which can be a bacterial pneumonia or a left heart failure. The clinical staff shall develop clinical pathways that coordinate the diagnostics of one of these diseases as well as both of them together. Fig. 1 illustrates how this is done. According to the GPS, the pathway for all clinical diagnostics is divided into different categories, such as "Radiology" or "Laboratory". These categories represent the navigation points within the GPS and may contain further specialized navigation points. E. g., the navigation point "Radiology" comprises a chest X-ray examination and encapsulates further radiological activities within navigation point "Additional radiology". Now, the modeler team can decide whether or not additional examinations should be scheduled within the clinical pathway by default.

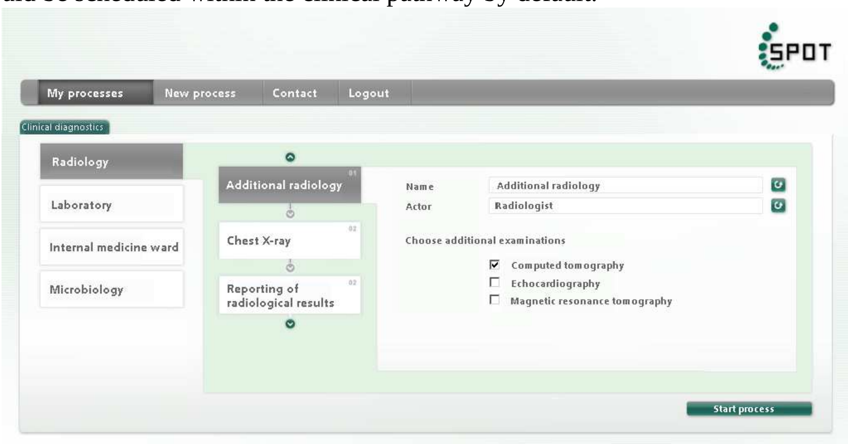

Fig. 1. Modeling clinical pathways from the end-users' perspective.

The pathway modeler team navigates through the GPS in order to develop a clinical pathway for the diagnostics of bacterial pneumonia. All tasks they select will be included in the clinical pathway. After having made all these choices the clinical pathway is determined and an executable process template can be generated. As we will see later, this does not mean that the resulting clinical pathway is now completely inflexible: If needed, the end-user (e.g., the ward physician) can repeat tasks or can "reactivate" deselected tasks in the context of ad-hoc deviations at run-time to adjust the clinical pathway to the individual needs of a specific patient.) By performing process modeling this way, the end-users can mentally completely stay in their "world" and just select the tasks to be performed. All the other aspects like setting up the resulting control flow, the data flow, deadlock avoidance and other things are handled at GPS system level and do not bother him. In addition, the GPS also "knows" which tasks depend on each other or, just the opposite, exclude each other. This means that certain kinds of mistakes are automatically avoided.

4 Implementation of Guarded Process Spaces

Fig. 2 illustrates how the support for this navigation and decision process is implemented at the technical level. The graph on the left side corresponds to the GPS and the graph on the right side represents the currently developed clinical pathway. The nodes in the GPS graph represent either tasks or logical operators like, e.g., AND, XOR, OR, and OPT (for optional). In Fig. 2, the root node "Clinical diagnostics" is connected with an AND-operator, i. e., all child nodes have to be selected. According to the OPT-operator at "Additional radiology", it is possible to choose an arbitrary number of child nodes or none at all.

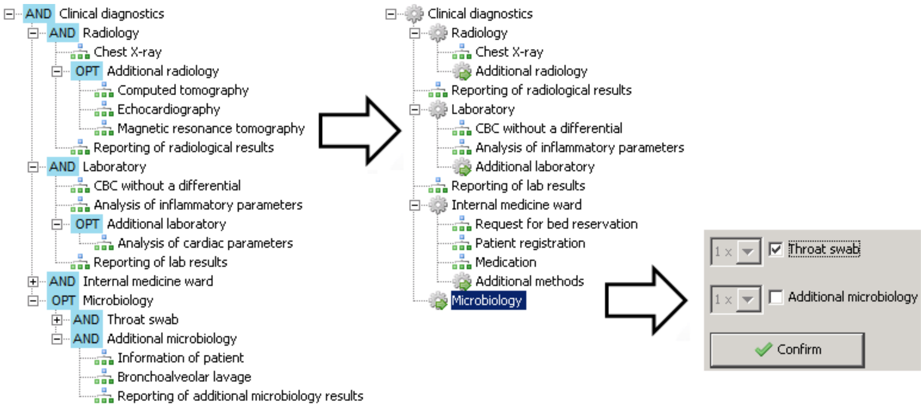

Fig. 2. Modeling of a clinical pathway for diagnostics of bacterial pneumonia based on GPS.

Since AND-operators don't leave many options, a big part of the clinical pathway can be automatically derived from the GPS in this case. E. g., a chest X-ray always includes some laboratory examinations and reporting to the ward physician. With regard to the possible variations "Additional radiology" and "Additional laboratory" the modeler have to decide which ones (if any) shall occur in the clinical pathway. For diagnostics of pneumonia a throat swab is required. Therefore, the modeler team selects "Throat swab" from the navigation point "Microbiology". After that, the clinical pathway for bacterial pneumonia is complete. In order to obtain the pathway for diagnostics of left heart failure, the modelers must only take a slightly different route with respect to some navigation points.

As indicated above, the GPS can also contain constraints. Thus, it can be expressed, for example, that the activity "Reporting of radiological results" must not be scheduled as long as "Chest X-ray" and optionally "Additional radiology" are not finished. One could also state that some nodes may require or exclude other nodes. In addition, it is possible to specify constraints for node cardinalities. I.e., one can define the maximum number of times that a clinical pathway may contain a specific navigation point or a process activity. This feature can be used to model cyclic treatments. The more general process knowledge a GPS captures, the easier becomes the creation of clinical pathways and – finally – executable process templates for the clinical staff.

However, as we all know: "There ain't no such thing as a free lunch". To make the creation of clinical pathways by end-users as easy as described above, somebody has to pay the price, and that are the implementers of the GPS graph in this case. The degree to which this graph covers an application area with all its possible tasks, choices and constraints is crucial for the ease of use, the range of applicability, and the acceptance of the GPS based modeling environment as well as the resulting executable clinical pathways. The clinical staff can (and must) help to develop the initial hierarchical structure of a GPS and to identify relevant process activities. The implementation of the GPS graph itself, with all its choices and constraints, the implementation of activities by executable application components, user interfaces, and task-specific control and data flow aspects will require IT-specialists. To decrease the complexity of this task one can take a stepwise refinement approach by first modeling the GPS graph rather coarsely and verify it with end-users using the modeling environment. Then one refines one or several nodes and verifies it again, etc.

5 Enabling Process Change from the End-user's Perspective

With GPS, end-users are now in the position to create various clinical pathways using the offered navigation paradigm. However, in healthcare it is often not possible to plan the complete treatment process in advance. Instead, the treatment process develops depending on further insights gained during the execution of the process. Under certain circumstances, it may even become necessary to abandon the original plan, to return to a specific point, and to choose a different option. Therefore, in this chapter we describe healthcare-specific requirements on process flexibility and discuss how they can be addressed using GPS in combination with existing BPMS.

Flexible extension of pathway instances. Although, the modelers of clinical pathways determine medical treatment to a large extent, some decisions can only be made by the doctor in charge of the patient. With regard to our pathway example, each patient who is suspected to suffer from bacterial pneumonia will undergo a throat swab. After that, the further proceeding depends on whether the finding is positive or negative. Assuming, the finding is positive, then the GPS specifies the available options; i. e., to carry out a bronchoalveolar lavage (cf. Fig. 2). When selecting this choice, the GPS may enforce an application constraint, e.g., that a bronchoalveolar lavage is only possible if the patient has been transferred to the isolation ward first in order to prevent the spreading of the disease. If this case is not rare to occur, the pathway modelers will have anticipated this additional examination along with the associated tasks and may offer it in terms of placeholder activities or variation points.

The GPS-based approach opens several options how to map such a case to the BPMS layer: One could "materialize" such conditional tasks as conditional branches. As these branches (whether they are likely to get executed or not) contain all necessary activities, this may lead to rather complex process templates, however. Other options, in case the BPMS is offering these features, are late binding of activities/subprocesses and dynamic modifications of the process instance. Then, the pathway can start with the “standard case” and would only be extended if necessary. By using GPS it is possible to choose different realization alternatives on a technical level, while the interface for the end-users remains the same.

Minor deviations from clinical pathways. Clinical pathways specify the standard way of proceeding. Accordingly, the pathway for diagnostics of bacterial pneumonia only schedules a chest X-ray in the course of radiological examinations. However, in certain cases, it can become necessary to perform additional procedures. For example, the ward physician examines the chest X-ray and the lab results of her patient. As it is not possible to confirm the diagnosis on the basis of these findings, she decides on scheduling a computed tomography. This examination is not part of the clinical pathway by default. Therefore, in the context of radiology a deviation from the pathway occurs. After changing the configuration of the navigation point “Additional radiology”, the new task is added to the work list of the responsible radiologist. Fig. 3 indicates how end-users can handle minor deviations from clinical pathways with respect to the given GPS.

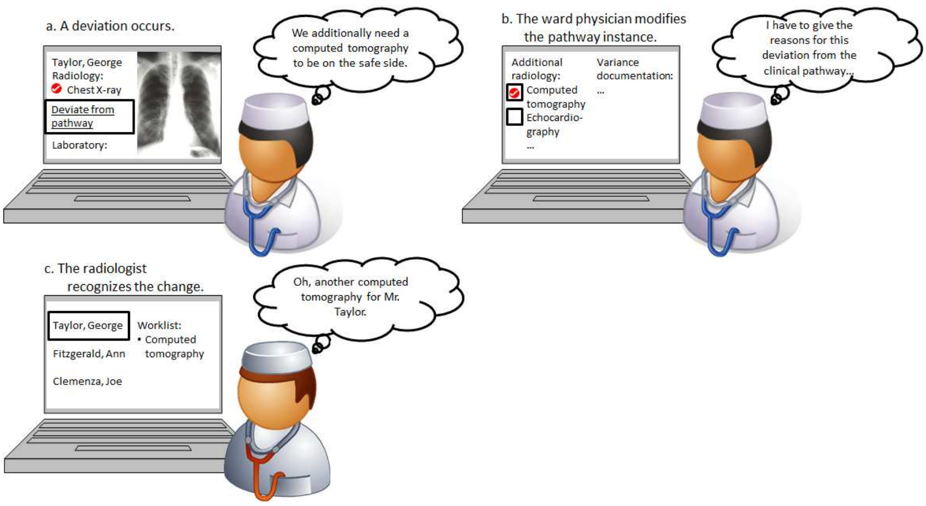

Fig. 3. Ad-hoc change of pathway instances from the end-user’s perspective.

Provided that the position where such a deviation may happen is known in advance, it is sufficient to use conditional branches, placeholder activities, or variation points. Frequently, it is not so clear at which point of time a variance arises, however. Regarding the GPS, it is not important when the physician decides to deviate from the pathway. The GPS may define constraints that determine the position where a computed tomography should be performed at best. However, if the execution flow has already passed this position, the examination may be scheduled at the next possible place.

At the BPMS level, constructs like conditional branches seem to be right choice, at first glance. However, a solution based on statically defined conditional branches would require to integrate such alternatives or placeholders for them at all possible positions within the process template to achieve the necessary degree of flexibility. This, in turn, would lead to extremely complex process models. Therefore, if a deviation can occur at different points of time, a BPMS is required that either allows for more flexibility in scheduling activities at runtime, like, e. g., Declare [17], or one that provides an adaptation API, which allows for structural changes, like, e. g., ADEPT2 [1, 14-15] .

Complex deviations from clinical pathways. The detection of second diagnoses or complications may result in more complex deviations from clinical pathways. E. g., a physician chose the pathway for diagnostics of bacterial pneumonia for her patient. During the treatment she discovers symptoms that indicate a left heart failure. She decides to modify the process in a way that it also covers the procedures for this second diagnosis. So, changes can not only occur at isolated spots, but may affect several regions of a process. In our example the doctor would have to insert the activities "Analysis of cardiac parameters" within "Laboratory" and "Echocardiography" within "Additional radiology". Such complex deviations may significantly increase the effort of physicians to perform the change of the pathway and thus raise the probability of errors. Therefore, if the pathway modelers have anticipated such a situation, they could have provided *process variants*, which encapsulate all the changes that have to be made to perform a pathway modification of this kind. In this way, it is even feasible to specify standard ways of proceedings in case of deviations from clinical pathways.

Since process variants automate the execution of change operations, it must be ensured that they are in conformance with the medical treatment and the procedures, which have been undertaken so far. As a GPS already determines all the routes that processes may take, possible variations, and relations between certain activities, it can also be used to verify that changes in the context of a process variant do not contradict previous treatment. On a technical level, BPMS supporting variation points can be a good choice to deal with variants, because in contrast to conditional branches or placeholders it is possible to put different changes in relation to each other. However, there is still the problem that the regions of change have to be determined in advance; i. e., if the execution of a process has already passed a region, it would not be possible to apply the modifications any more. Therefore, in [19] we show how process variants can be more flexibly used in the context of GPS.

6 From GPS-based Clinical Pathways to Executable Processes

The clinical pathway conforms to the tree-like structure of the GPS graph, but contains only those process activities, which have been selected for the pathway under consideration (cf. Fig. 2). To obtain independence from a specific BPMS, the GPS graph as well as this "clinical pathway graph" serve as a neutral representation which is mapped to process templates of the chosen BPMS to achieve executable processes. As mentioned above, at present there exist a large variety of BPMS which differ significantly with respect to their process models, interfaces, and functionalities. Not all of them seriously qualify as a target BPMS because the effort to implement the mapping would be very high or missing functionalities would have to be implemented "on top" and provided by the mapping layer (as far as this is possible, all). For our proof of concept prototype we selected ADEPT2 [1, 14-15] resp. its commercial version, the AristaFlow® BPM Suite[2] as the target BPMS because it provides a lot of the functionality needed already today.

In [19] one can find the formally defined transformation rules according to which the compilation to GPS-based clinical pathways to the ADEPT2 process model takes place. Fig. 4 shows in which way the process activities within the context of "Radiology" (cf. Fig. 2) can be mapped to executable ADEPT2 process templates. As radiology comprises both default and optional activities, there are two mapping alternatives in ADEPT2: Either conditional branches are used or the additional activities can be inserted into the process instance on demand.

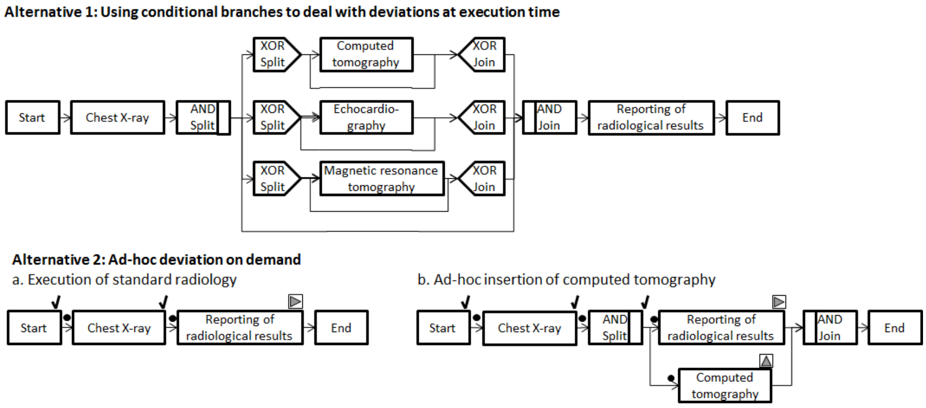

Fig. 4. Mapping the "clinical pathway graph" to ADEPT2 process templates.

According to the constraints of the GPS, the reporting activity should follow the examinations, whereby the chest X-ray has to be performed first. As several or none of the additional examinations can be selected, in alternative 1 we have to insert a complex construct consisting of a parallel as well as three conditional branches. In alternative 2, we only schedule the default activities and perform ad-hoc insertions on demand depending on the current state of the execution. In Fig. 4.a, the reporting activity has already started as a deviation occurs. Thus, it is not possible to insert the computed tomography directly after the chest X-ray, anymore. Instead, it can be added in parallel to the running activity as illustrated in Fig. 4.b. This example shows that the mapping between pathway graph and process template is much simpler than realizing complex workflow patterns. Consequently, BPMS providing comprehensive support for ad-hoc changes at the API level like, e.g., ADEPT2, are ideal candidates for this approach. Other BPMS can be supported as well, but one is faced with limited flexibility and with more complex implementations of the mapping and run-time layer to compensate the missing BPMS functionality.

[2] See http://www.AristaFlow.com

7 Summary

In spite of its potential benefits, BPM systems are not broadly used in healthcare settings yet. In order to be accepted by end-users, the technology has to fulfill the following requirements: Clinical staff must be enabled to model executable process templates by themselves. Moreover, end-users must have the possibility to flexibly adapt running process instances according to the individual demands of their patient. With Guarded Process Spaces (GPS), we presented an approach which uses a "navigation paradigm" to guide end-users in modeling clinical pathways as well as to assist them to perform ad-hoc deviations at run-time for a patient with specific needs. We showed how this approach guides users to select the necessary tasks in the right order and how tasks can be automatically inserted when required in the given context. We also gave some insights how this guidance is reflected in the underlying implementation. The approach is based on a sound formal concept which could only be sketched here due to lack of space, however. A comprehensive description can be found in [19].

An important goal was to make the implementation of clinical pathways independent from a specific BPMS. Therefore, the models used to represent the GPS graph as well as the derived clinical pathways are neutral in this sense. Mapping functions are used to transform the clinical pathways into executable process templates of the selected target BPMS. The approach has been validated by a proof of concept implementation using ADEPT2 as target BPMS.

In the context of the current project "eBusiness Platform for Healthcare"[3], we are planning to use the GPS approach to develop medical processes crossing the border of individual healthcare provider institutions. By doing so, GPS are leveraging execution of integrated workflows based on collective knowledge and in spite of heterogeneous system environments.

References

1. Reichert, M., Dadam, P.: $ADEPT_{flex}$ – Supporting Dynamic Changes of Workflows Without Losing Control. In: Journal of Intelligent Information Systems 10, pp. 93—128 (1998)
2. Dadam, P., Reichert, M., Kuhn, K.: Clinical Workflows - The Killer Application for Process-oriented Information Systems?. Proc. Int'l Conf. on Business Information Systems, BIS 2000, 4th Int'l Conf., Poznan, Poland, April 2000, Springer-Verlag, pp. 36—59 (2000)
3. Weber, B., Sadiq, S., Reichert, M.: Beyond Rigidity - Dynamic Process Lifecycle Support - A Survey on Dynamic Changes in Process-aware Information Systems. Computer Science - Research & Development, Vol. 23, No. 2, 2009, special issue on "Flexible Process-aware Information Systems", pp. 47—66 (2009)
4. Becker, J., Pfeiffer, D., Räckers, M.: Domain Specific Process Modelling in Public Administrations – The PICTURE-Approach. In: Electronic Government, Lecture Notes in Computer Science 4656, pp. 68—79 (2007)

[3] See http://www.ebpg-nrw.de/

5. Freudenstein, P., Buck, J., Nussbaumer, M., Gaedke, M.: Model-driven Construction of Workflow-based Web Applications with Domain-specific Languages. Proc. of the 3rd Int'l Workshop on Model-driven Web Engineering (MDWE'07), Como, Italy, 2007
6. Neuhaus, J., Houta, S., Reuter, C.: Ansätze bei der Umsetzung von Behandlungsplanpfaden – Flexibilisierungskonzepte am Beispiel der Behandlung von Wirbelsäulenerkrankungen. In: Hellmann, W., Eble, S. (eds): Ambulante und Sektoren übergreifende Behandlungspfade, pp. 79-97, Medizinisch Wissenschaftliche Verlagsgesellschaft Berlin (2010)
7. De Clercq, P., Kaiser, K., Hasman, A. : Computer-interpretable Guideline Formalisms. Ten Teije, A. et al. (eds.): Computer-based Medical Guidelines and Protocols: A Primer and Current Trends, IOS Press, pp. 22—43, 2008
8. Kant, K., Cohen, S., Hess, J., Nowak, W., Peterson, S.: Feature-oriented domain analysis (FODA) feasibility study. Technical report CMU/SEI-90-TR-21, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, USA, 1990
9. Puhlmann, F., Schnieders, A., Weiland, J., Weske, M.: Variability Mechniasms for Process Models. PESOA-Report TR 17/2005, Process Family Engineering in Service-Oriented Applications (PESOA), 2005
10. Beccuti, M., Bottrighi, A., Franceschinis, G., Montani, S., Terenziani, P.: Modeling Clinical Guidelines through Petri Nets. Combi, C., Shahar, Y., Abu-Hanna, A. (eds.): Proc. of the 12th Conf. on Artificial Intelligence in Medicine (AIME'09), pp. 61—70, Springer-Verlag Berlin, Heidelberg, 2009
11. Russell, N., ter Hofstede, A.H.M., van der Aalst, W.M.P., Mulyar, N.: Workflow Control-Flow Patterns: A Revised View. BPM Center Report BPM-06-29, BPMcenter.org, 2006
12. Sadiq, S., Sadiq, W., Orlowska, M.: Pockets of Flexibility in Workflow Specification. In: Proc. of the 20th Int'l Conf. on Conceptional Modeling, LNCS, vol. 2224, pp. 513—526, Springer, Berlin Heidelberg (2001)
13. Hallerbach, A., Bauer, T., Reichert, M.: Configuration and Management of Process Variants. In: vom Brocke, J., Rosemann, M. (eds.): Handbook of Business Process Management (2009)
14. Dadam, P., Reichert, M.: The ADEPT Project: A Decade of Research and Development for Robust and Flexible Process Support. Computer Science - Research & Development, Vol. 23, No. 2, special issue on "Flexible Process-aware Information Systems", pp. 81-98 (2009)
15. Reichert, M., Dadam, P.: Enabling Adaptive Processes with ADEPT2. In: Cardoso, J.; van der Aalst, W.M.P. (Eds.): Handbook of Research in Business Process Modeling. Information Science Reference (an imprint of IGI Global), pp. 173-203 (2009)
16. Weber, B., Reichert, M., Rinderle-Ma, S.: Change patterns and change support features – enhancing flexibility in process-aware information systems. Data and Knowledge Engineering 66(3): pp. 438—466 (2008)
17. Pesic, M., Schonenberg, M.H., Sidorova, N., van der Aalst, W.M.P.: Constraint-Based Workflow Models: Change Made Easy. In: Curbera, F., Leymann, F., Weske, M. (eds.): Proceedings of the OTM Conf. on Cooperative Information Systems, LNCS, vol. 4803, pp. 77—94, Springer, Berlin Heidelberg (2007)
18. Reuter, C.: Composition of Semantic Process Fragments to Domain-related Process Families. In: van Bommel, P., Hoppenbrouwers, S., Overbeek, S., Proper, E., Barjis, J. (eds.): The Practice of Enterprise Modeling (PoEM 2010), LNBIP 68, pp. 61—75, Springer Berlin-Heidelberg (2010)
19. Reuter, C.: Modellierung und dynamische Adaption klinischer Pfade auf Basis Semantischer Prozessfragmente (SPF). PhD thesis, Technical University of Dortmund, Germany (2011, to appear)