Flexibility Schemes for Workflow Management Systems

- regular paper -

R.S. Mans[1], W.M.P. van der Aalst[1], N.C. Russell[1], P.J.M. Bakker[2]

[1] Department of Information Systems, Eindhoven University of Technology, P.O. Box 513, NL-5600 MB, Eindhoven, The Netherlands.
`{r.s.mans,w.m.p.v.d.aalst,n.c.russell}@tue.nl`

[2] Academic Medical Center, University of Amsterdam, Department of Innovation and Process Management, Amsterdam, The Netherlands.
`p.j.bakker@amc.uva.nl`

Abstract. Currently, many hospitals are investigating the use of workflow management systems in order to provide support for healthcare processes. However, contemporary workflow management systems fall short in supporting care processes which require flexible execution options. In this paper, we investigate the flexibility requirements that need to be satisfied in order to support various kinds of healthcare processes Our evaluation shows that different systems need to be used in conjunction with each other in order to fully support the various types of care processes.

Key words: workflow management, flexibility, healthcare

1 Introduction

In a competitive health-care market, hospitals need to focus on ways of streamlining their processes in order to deliver high quality and safe care while at the same time reducing costs [7]. Consequently, there is a need for technological support in controlling and monitoring healthcare processes for patients [12] and workflow technology is potentially a means for achieving this end. Workflow Management Systems (WfMSs) support processes by managing the flow of work such that individual work items are done at the right time by the proper person [1]. The benefits being that processes can be executed more rapidly and can be monitored.

A number of difficulties commonly arise when hospitals attempt to automate healthcare processes as a consequence of the fact that these processes are *diverse*, require *flexibility* and that *several medical departments* can be involved in the diagnostic and treatment process. For a group of patients with the same diagnosis, the number of different examinations and treatments required can be high and the order in which they are conducted can vary greatly.

Therefore, an interesting and challenging question arises: *What are the considerations with regard to process flexibility when applying workflow technology* *in hospitals?*. When we look at how to achieve process flexibility, four different approaches can be identified [14] which differ in the timing and manner that they are applied. More details can be found in [14].

Flexibility by design: the ability to incorporate alternative execution paths within a process model at design time allowing the selection of the most appropriate execution path to be made at runtime for each process instance.

Flexibility by deviation: the ability for a process instance to deviate at runtime from the execution path prescribed by the original process without altering its process model. The deviation can only encompass changes to the execution sequence of tasks in the process for a specific process instance, it does not allow for changes in the process model or the tasks that it comprises.

Flexibility by underspecification: the ability to execute an incomplete process model at run-time, i.e., one which does not contain sufficient information to allow it to be executed to completion. The model needs to be completed by providing a concrete realization for the undefined parts.

Flexibility by change: the ability to modify a process model at run-time such that one or all of the currently executing process instances are migrated to a new process model.

To answer the previous question, we implemented a *representative* healthcare process in four workflow systems. Based on the above four flexibility types, we will discuss what kind of flexibility is actually needed in order to support the representative healthcare process and healthcare processes in general.

As the representative care process, we have taken the diagnostic process of patients visiting the gynecological oncology outpatient clinic in the AMC hospital, a large academic hospital in the Netherlands. The healthcare process under consideration is a large process consisting of around 325 activities. We choose to implement the care process in workflow systems which demonstrate various kinds of flexibility. For this purpose we selected YAWL [2, 6], FLOWer [5], ADEPT1 [17], and Declare [16]. YAWL was chosen because it is a powerful open-source system supporting most of the workflow patterns [13]. FLOWer is considered to be the most successful commercial system providing flexibility support. ADEPT1 and Declare are two academic systems providing new and powerful ways of supporting "extreme" flexibility. Moreover, the selected systems cover distinct areas of the Process Aware Information Systems (PAIS) technology spectrum, such as adaptive workflow (ADEPT1), case handling (FLOWer), and declarative workflow (Declare). In Table 1, it is shown which flexibility types are supported (+) and not supported (–) by each workflow system. A detailed evaluation can be found in [14]. Together with the identified flexibility requirements this allows for examination of the conditions under which a workflow system can be applied in the healthcare domain. Note that we only focus on the control-flow perspective of a process. Other factors which might be relevant are not considered.

This paper is structured as follows: Section 2 introduces the gynecological oncology healthcare process in general and a subpart of it in detail. Section 3 discusses the corresponding implementation in each of the different workflow systems. Section 4 examines the flexibility needed for supporting healthcare

Flexibility by	ADEPT1	YAWL	FLOWer	Declare
design	+	+	+	+
deviation	−	−	+	+
underspecification	−	+	−	−
change	+	+	−	+

Table 1. Product evaluations.

processes. Related work is outlined in Section 5. Section 6 concludes the paper.

2 Case of gynecological oncology

In this section, we introduce the diagnostic part of the gynecological oncology healthcare process, which we studied. In Figure 1, a snippet showing the most important part of the process is given. Moreover, for the "referral patient and preparations for first visit" node a part of the corresponding subprocess is shown in Figure 2.

Figures 1 and 2 model the gynecological oncology process using so-called *Colored Workflow Nets (CWN)* [4], which are a specific class of *Colored Petri Nets (CPNs)* [10]. Furthermore, a CWN is a *workflow model* in which we restrict ourselves to concepts and entities which are common in most workflow languages.

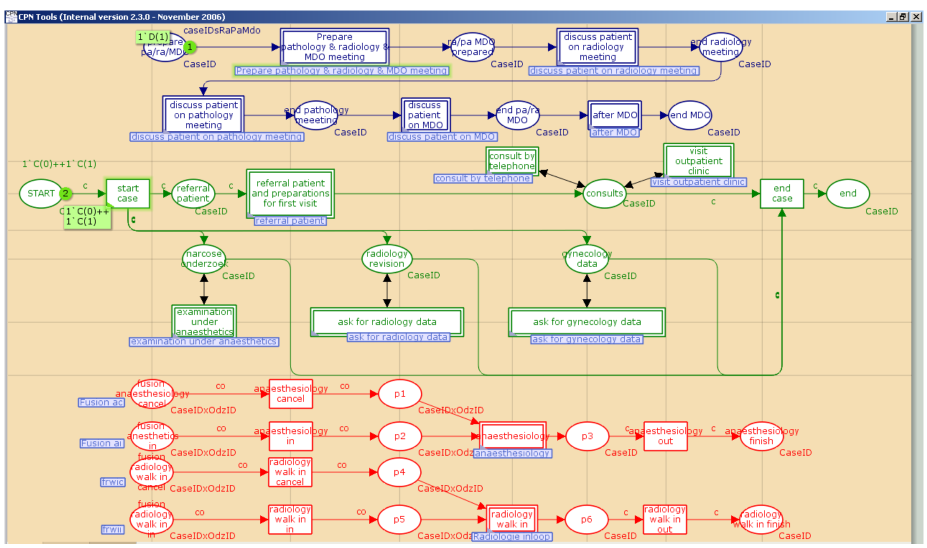

Fig. 1. General overview of the diagnostic process of the gynecological oncology healthcare process. The green and blue nodes and arcs represent respectively the first and second part of the process. The red nodes and arcs represent the interactions with different medical departments.

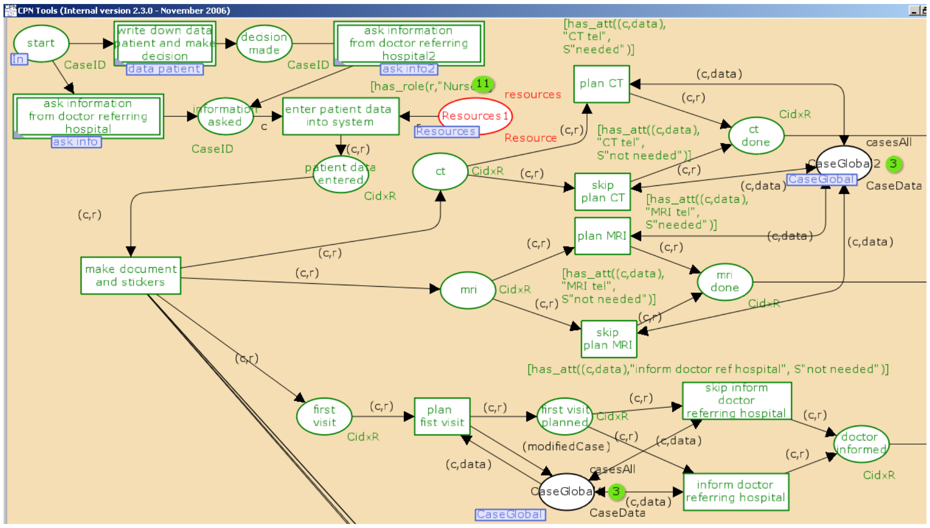

Fig. 2. CWN for the first part of the gynecological oncology healthcare process.

To this end, a CWN covers the *control-flow*, *organizational*, *data* and *operational* perspectives. More details about a CWN can be found in [4]. In Figure 1, the topmost page of the CWN model is shown which gives a general overview of the diagnostic process of the gynecological oncology healthcare process in the AMC hospital. As we are dealing with a large healthcare process it is only possible to show a small part of the *overall* model.

As can be seen in Figure 1, the gynecological oncology process consists of two different processes of which only one will be considered in detail. The process, which is modeled in the lower part of the picture and colored green, deals with the diagnostic process that is followed by a patient when referred to the AMC hospital for treatment, up to the point where the patient is diagnosed. In this process, the patient can have several consultations with a doctor, either by visiting the outpatient clinic or via telephone. During such a consultation, the status of the patient is discussed and a decision is made about whether examinations and consultations need to be requested, canceled, or rescheduled. Moreover, during the course of the process, several administrative activities such as brochure recommendation and patient registration can also occur. A doctor can request various tests, performed by different medical departments. The interactions with these medical departments and also the processes adopted by them are modeled at the bottom of Figure 1 (the red colored nodes). It is important to note, that in the future new tests can become available, even new types of medical departments. In this way, it becomes clear, that in order to cater for varying interactions with medical departments, at runtime we need to decide which interactions are needed which can be provided by flexibility by underspecification.

Having introduced the gynecological oncology process, we will focus on its initial stages (i.e. substitution transition “referral patient and preparation for first visit”), in which a doctor of a referring hospital calls a nurse or doctor of the AMC hospital resulting in an appointment being made for the first visit of the patient. At that moment it is also necessary to schedule appointments for diagnostic tests. This part of the process is shown in Figure 2. For example, we see that the first visit of the patient needs to be planned, and that it is possible to make an appointment for an “MRI”.

The process, shown in Figure 2, is considered to be a “standardized procedure” for these patients at the AMC. From the figure, it can be seen that there a number of possible courses of action that may be taken (and the figure only shows half of the process). Furthermore, as healthcare processes are unpredictable, there can also be the need to skip or to add activities. Respectively, the first is an example of flexibility by design and the latter is an example of flexibility by deviation.

3 Realization of the system in different Workflow Systems

In this section, we will discuss how several different workflow systems have been configured in order to support the healthcare processes discussed earlier. The workflow systems YAWL, FLOWer, ADEPT1 and Declare have been chosen as candidate systems. Each of them demonstrates a specific kind of flexibility, which is deemed relevant when implementing a healthcare process in a workflow context. In the remainder of this section, we will examine how the flexibility provided by each workflow system has been used or can be utilized during the execution of healthcare processes. Due to space limitations, we will do this in detail for YAWL in Section 3.1 and present the main findings for the other systems in Section 3.2.

3.1 YAWL / Worklets

YAWL (Yet Another Workflow Language) [2] is an open source workflow management system, which is based on the well-known workflow patterns [13] and is more expressive than any workflow language available today. YAWL supports the modeling, analysis and enactment of flexible processes through the use of *worklets* [6] which can be seen as a kind of configurable process fragment. Specific activities in a process are linked to a repertoire of possible actions. Based on the properties of the case and other context information, the desired action is chosen. The selection process is based on a set of rules. Also, during enactment it is possible to add new actions to the repertoire.

In YAWL, we used the worklet approach for modeling the interactions with all medical departments by linking a “multiple atomic task” node to the worklet service. This is represented in Figure 3(a) by the node with name “examinations” which can be executed additional times if multiple examinations are needed. In this way, for each test the right worklet can be chosen. In the case where a new test arises, it is possible to choose a corresponding process fragment, or to dynamically define a new process fragment, thereby extending the ruleset.

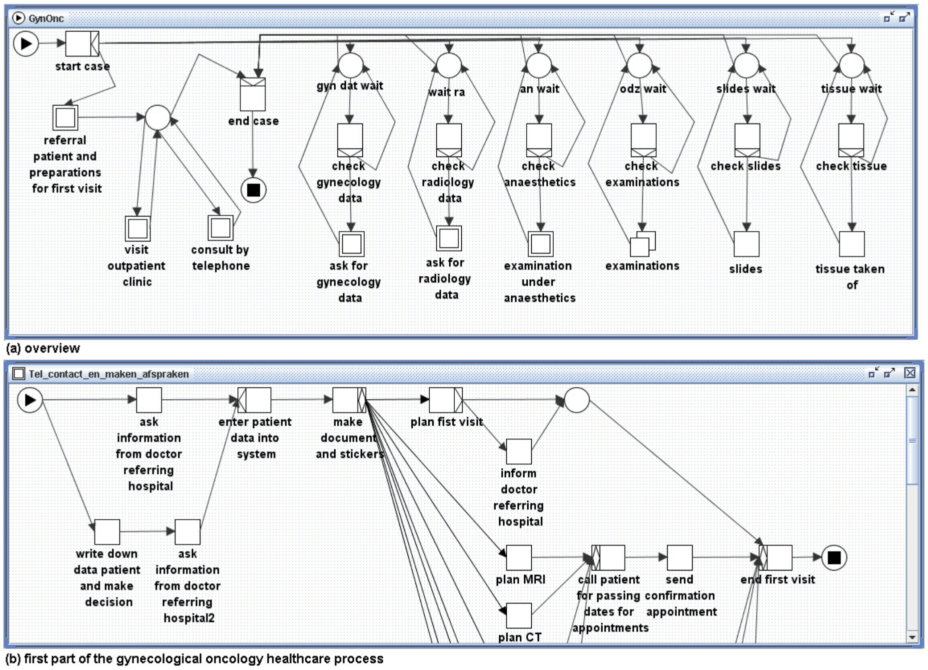

Fig. 3. Screenshots of models in the YAWL editor.

In Figure 3(b), we see the corresponding YAWL process fragment for the first part of the gynecological oncology healthcare process. Due to syntactical sugaring, less nodes were needed than are required in the CWN model of Figure 2. For example, the “make document and stickers” activity in YAWL is an OR-split, which means that one or more of the outgoing paths may be followed and others may be skipped. This OR-split is used because each of the “plan MRI” and “plan CT” activities may or may not be performed.

3.2 Realization in other workflow systems

In this section, the realization in FLOWer, ADEPT1 and Declare is discussed. Each system will be introduced shortly followed by the main findings for the system.

FLOWer *FLOWer* is a commercial workflow management system provided by Pallas Athena in the Netherlands. FLOWer is a case-handling product [5]. Case-handling aids process flexibility by focussing on the data aspect rather than on the control-flow aspect of processes. In particular the following flexibility features offered by FLOWer are used. First, work distribution is separated from authorization, which allows for additional types of actions, like skipping or redoing activities in the process. An example is the skipping of the “plan MRI” step of Figure 2. Second, workers are allowed to view and add/modify data before and after the corresponding activities have been executed. So, if the activity "make document and stickers" has already been executed, in FLOWer it is still possible to go back in the process to where the activity "enter patient data into system" was executed.

In FLOWer, we used the "dynamic subplan", which allows for concurrent execution of a subprocess, for modeling the interactions with all medical departments. However, if a new test is needed, a new version of the process needs to be introduced. Unfortunately, it is not guaranteed that already running cases can be updated to the new version of the process in a safe and secure way.

ADEPT1 ADEPT1 is an academic prototype workflow system [17], developed at the University of Ulm, Germany. ADEPT1 supports *dynamic change* which means that the process model for one individual case can be adapted. In doing so, it is possible to deviate from the pre-modeled process template (skipping of steps, going back to previous steps, inserting new steps, etc.) in a secure and safe way. That is, the system guarantees that all consistency constraints (e.g., no cycles, no missing input data when a task program will be invoked) which hold prior to the dynamic (ad hoc) modification of the process instance also hold after the modification. The intention of the next version, ADEPT2, is to provide full support for changes, including the propagation of process schema changes to already running instances[8].

So, when realizing the process in Figure 2, in ADEPT1 it is possible, for an already running case, to dynamically add the activity "order drug" after the activity "make document and stickers" and before the activity "plan first visit" which allows for ordering a drug in between the activities "make document and stickers" and "plan first visit".

Declare Declare is another academic prototype workflow system focusing on flexibility [16]. In Declare the language used for specifying processes, called *ConDec*, is a *declarative* process modeling language, which means that it specifies *what* should be done instead of specifying *how* it should be done, as is the case in imperative languages (e.g. YAWL, FLOWer). Users can execute activities in any order and as often as they want, but they are bound by certain specified rules, called constraints. For example, when implementing Figure 2, in Declare we can define that activities "enter patient data into system" and "make document and stickers" needs to be executed at least once, but it is not specified in which order they need to be executed.

Furthermore, Declare also supports *dynamic change*, so that the process associated with individual cases can be adapted. In Declare, this means that it is possible to deviate from the pre-modeled process template by adding or removing activities or constraints. Also, model correctness is guaranteed and it is checked by Declare whether the changes are allowed or not for the cases to which they are applied. As for ADEPT1, we can define that activity "order drug" needs to be done after activity "make document and stickers" and before "plan first visit" by dynamically adding a response constraint between this activity and activity "make document and stickers" and adding a precedence constraint between this activity and the "plan first visit" activity.

4 Evaluation

In Section 1, four different approaches to achieving process flexibility have been discussed. First, for the case of gynecological oncology we determined which flexibility approach is the best candidate for supporting the healthcare process under consideration. Following on from this, we distinguished different kinds of healthcare processes and offered a basis for classifying their specific flexibility requirements. Finally, we use this classification to evaluate the capabilities of the offerings discussed in Section 3 in order to determine which of them can provide the best support for various kinds of healthcare process.

For the classification we only focus on *organizational* healthcare processes. These processes consist of organizational tasks in which collaboration between people from different departments is a vital process characteristic. Moreover, the process is repetitive, but non-trivial. Unlike medical treatment processes, organizational processes do not provide any support for medical decision making[12]. Note that the focus is on presenting a classification which covers the majority of organizational healthcare processes. It is infeasible to cover everything due to the unexpected character of the processes considered. The classification itself has been made based on the insights obtained when studying the gynaecological oncology healthcare process. Moreover, the classification and the accompanying flexibility requirements are based on discussions with a medical specialist.

Gynecological oncology healthcare process The gynecological process, shown in Figure 1, is performed in an academic hospital (AMC, Amsterdam), and is an organizational process. In general, the art and the number of diagnostic tests to be performed is known. However, the total number of examinations is determined by patient characteristics and previously performed diagnostic tests. Clearly, complex care needs to be delivered in which many different departments can be involved. To this end, *flexibility by underspecification* is an interesting candidate in order to provide support for the process, as it allows for the definition of an incomplete model for which the ultimate realization of tasks can be deferred until runtime.

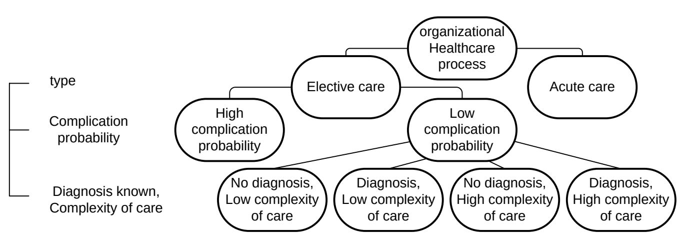

Fig. 4. Classification of healthcare processes.

Healthcare processes In addition to the healthcare process discussed earlier, there exist many other (organizational) healthcare processes with totally different characteristics for which other requirements with regard to flexibility will exist. Figure 4 shows the different types of healthcare processes that can be distinguished. In general, organizational healthcare processes can be divided into:

– *acute care* which deals with critically ill patients in which patient conditions change rapidly; and
– *elective care* for which it is still medically sound to postpone treatment for some days or weeks. Consequently, this kind of care can be planned in advance.

It is clear that acute care cannot be planned and needs to be done in an ad-hoc fashion. To this end, flexibility by change is the best candidate for supporting such an ad-hoc process as the model is not fixed and can be changed into another completely specified model.

Elective care can be planned in advance and several distinct classes of care can be identified. First of all, we propose to make a distinction between processes for which the probability that *complications* arise is *high* or *low*.

Typically, when such a complication occurs it has a high impact on the process as it requires the process to be changed dramatically in some parts. After these changes, the process needs to be made complete again so that it can be executed. Consequently, healthcare processes for which the probability on complications is high can best be supported by flexibility by change.

In contrast, when the complication probability is low, no dramatic changes are to be expected in the process execution. Nevertheless, different classes can be identified which have their own requirements with regard to process flexibility. We propose the following dimensions: *complexity of care* and *diagnosed*. Complexity of care indicates the extent of care which is delivered to a patient, which can be either high or low. Diagnosed indicates whether a diagnosis is known for a patient or not.

In situations where the complexity of care to be delivered is low, more or less, a standard procedure can be followed in which only a few departments are involved. To this end, both the diagnostic and treatment processes can be incorporated in a complete model. Nevertheless, in some cases, occasional unforeseen behavior should be anticipated, where the actual execution at runtime varies from the strict sequence implied by the process model. This can be provided by flexibility by deviation.

However, the complexity of the care to be delivered can also be high. Diagnosing a patient can be very challenging as for some patients it can not be anticipated which diagnostic tests need to be performed. Also, when a patient is finally diagnosed a careful choice needs to be made about the next steps to be done. So, the course of the process is heavily determined by patient characteristics in which collaboration between medical departments is of vital importance. Clearly, for this kind of process, the ultimate realization of some parts of the model needs to be deferred until runtime. This can be provided by flexibility by underspecification.

Flexibility by			design	deviation	under specification	change
Acute care						X
Elective care	High complication probability					X
	Low complication probability	low complexity,no diagnosis	X	X		
		low complexity,diagnosis	X	X		
		high complexity,no diagnosis			X	
		high complexity,diagnosis			X	

Table 2. Flexibility needed for each kind of healthcare process.

Table 2 summarizes which flexibility approach is considered important for which kind of healthcare process. This does not imply that if a flexibility type has not been indicated for a specific type of healthcare process that it is not relevant, rather that it is considered to be of less importance.

System support In Table 1 it can be seen which kind of flexibility is provided by each system. Combining these results with Table 1, we can derive which system(s) can provide the best support for each kind of healthcare process.

The table shows that each flexibility type is relevant for supporting healthcare processes. For both acute care processes and elective care processes with a high complication probability, flexibility by change is needed which can both be provided by ADEPT1 and Declare. For a low complex elective care process, with a low complication probability, a choice needs to be made between flexibility by design and flexibility by deviation. As FLOWer supports both types, this system would be the best candidate. In contrast, for high complex care processes, flexibility by underspecification is needed. To this end, YAWL would be the best candidate.

5 Related Work

Careflow sytems, systems for supporting care processes in hospitals, have special demands with regard to workflow technology. One of these requirements is that flexibility needs to be provided by the workflow system [19]. Unfortunately, current WfMS significantly fall short with regard to providing flexibility, which is seen as a problem in literature [3, 11]. Also, once a workflow-based application has been configured on the basis of an explicit process model, the execution of related process instances tends to be rather inflexible [18]. The workflow systems that we chose in this paper allow for more flexibility than classical workflow systems.

Another requirement when applying workflow technology in the healthcare domain is that real time patient monitoring, detection of adverse events, and adaptive responses to breakdown in normal processes is needed [9]. As adaptive workflow systems are rarely implemented, this makes current workflow implementations inappropriate for healthcare [20]. Furthermore, in a real clinical setting, it is a critical challenge for any workflow management system that it is able to respond effectively when exceptions occur [15]. Another significant gap that can be identified is that no support is provided for the multidisciplinary nature of healthcare processes. Consequently, there exists the need to support cross-departmental healthcare processes as is stressed in [12].

6 Conclusions

In this paper, we have investigated the flexibility requirements that need to be satisfied by workflow management systems in order to support organizational healthcare processes. As a running example, we used the AMC's gynecological oncology healthcare process which has been implemented in four different workflow systems. For this process, we identified that flexibility by underspecification is a key process requirement, a feature which can best be provided by YAWL.

Furthermore, we identified that different types of healthcare processes each have their own requirements with regard to flexibility. Our results, demonstrate that all flexibility types are useful for supporting specific types of care processes. Individual systems tend to exhibit a degree of specialization in their approach to process flexibility, which has the consequence that different systems need to be used in conjunction with each other in order to fully support all types of care processes that might be encountered. In order to promote the use of workflow management in hospitals, the focus needs to be on enhancing existing tools and/or the development of new ones which provide a greater support for flexibility.

A limitation of our approach is that only one healthcare process has been considered. Future research should focus on implementing healthcare processes with a variety of characteristics in several workflow systems so that deeper insights can be gained into the requirements for process flexibility. In this paper, we only focussed on the control flow perspective of care processes. A further line of research would be to investigate what the flexibility requirements are for other perspectives, such as the data, resource and application perspectives.

References

1. W.M.P. van der Aalst and K.M. van Hee. *Workflow Management: Models, Methods, and Systems*. MIT press, Cambridge, MA, 2002.
2. W.M.P. van der Aalst and A.H.M. ter Hofstede. YAWL: Yet Another Workflow Language. *Information Systems*, 30(4):245–275, 2005.
3. W.M.P. van der Aalst and S. Jablonski. Dealing with Workflow Change: Identification of Issues and Solutions. *International Journal of Computer Systems, Science, and Engineering*, 15(5):267–276, 2000.
4. W.M.P. van der Aalst, J.B. Jørgensen, and K.B. Lassen. Let's Go All the Way: From Requirements via Colored Workflow Nets to a BPEL Implementation of a New Bank System. In R. Meersman and Z. Tari et al., editors,

CoopIS/DOA/ODBASE 2005, volume 3760 of *Lecture Notes in Computer Science*, pages 22–39. Springer-Verlag, Berlin, 2005.
5. W.M.P. van der Aalst, M. Weske, and D. Grünbauer. Case Handling: A New Paradigm for Business Process Support. *Data and Knowledge Engineering*, 53(2):129–162, 2005.
6. M. Adams, A.H.M. ter Hofstede, D. Edmond, and W.M.P. van der Aalst. Dynamic. extensible and context-aware exception handling for workflows. In F. Leymann F. Curbera and M. Weske, editors, *Proceedings of the OTM Conference on Cooperative Information Systems (CoopIS 2007)*, volume 4803 of *Lecture Notes in Computer Science*, pages 95–112. Springer-Verlag, Berlin, 2007.
7. K. Anyanwu, A. Sheth, J. Cardoso, J. Miller, and K. Kochut. Healthcare Enterprise Process Development and Integration. *Journal of Research and Practice in Information Technology*, 35(2):83–98, May 2003.
8. P. Dadam, M. Reichert, S. Rinderle, M. Jurisch, H. Acker, K. Göser, U. Kreher, and M. Lauer. Towards truly flexible and adaptive process-aware information systems. In R. Kaschek, C. Kop, C. Steinberger, and G. Fliedl, editors, *UNISCON*, volume 5 of *Lecture Notes in Business Information Processing*, pages 72–83. Springer, 2008.
9. Y. Han, A. Sheth, and C. Bussler. A taxonomy of adaptive workflow management. In *Proceedings on CSCW-98 Workshop Towards Adaptive Workflow System*, 1998.
10. K. Jensen, L.M. Kristensen, and L. Wells. Coloured Petri Nets and CPN Tools for Modelling and Validation of Concurrent Systems. *International Journal on Software Tools for Technology Transfer*, 9(3-4):213–254, 2007.
11. M. Klein, C. Dellarocas, and A. Bernstein, editors. *Adaptive Workflow Systems*, Special Issue of Computer Supported Cooperative Work, 2000.
12. R. Lenz and M. Reichert. IT Support for Healthcare Processes - Premises, Challenges, Perspectives. *Data and Knowledge Engineering*, 61:49–58, 2007.
13. N. Russell, A.H.M. ter Hofstede, W.M.P. van der Aalst, and N. Mulyar. Workflow Control-Flow Patterns: A Revised View. BPM Center Report BPM-06-29, BPMcenter.org, 2006.
14. N.A. Mulyar, M.H. Schonenberg, R.S. Mans, N.C. Russell and W.M.P. van der Aalst. Towards a Taxonomy of Process Flexibility (Extended Version). BPM Center Report BPM-07-11, BPMcenter.org, 2007.
15. S. Panzarasa and M. Stefanelli. Workflow management systems for guideline implementation. *Neurological Sciences*, 27:245–249, June 2006.
16. M. Pesic, M.H. Schonenberg, N. Sidorova, and W.M.P. van der Aalst. Constraint-based workflow models: Change made easy. In Robert Meersman and Zahir Tari, editors, *OTM Conferences*, volume 4803 of *Lecture Notes in Computer Science*, pages 77–94. Springer, 2007.
17. M. Reichert, S. Rinderle, and P. Dadam. ADEPT Workflow Management System. In W.M.P. van der Aalst, A.H.M. ter Hofstede, and M. Weske, editors, *BPM 2003, Proceedings*, volume 2678 of *Lecture Notes in Computer Science*, pages 370–379. Springer, 2003.
18. S. Sadiq, O. Marjanovic, and M.E. Orlowska. Managing Change and Time in Dynamic Workflow Processes. *International Journal of Cooperative Information Systems*, 9(1-2):93–116, 2000.
19. M. Stefanelli. Knowledge and Process Management in Health Care Organizations. *Methods Inf Med*, 43:525–535, 2004.
20. J. Sutherland and W.-J. van den Heuvel. Towards an Intelligent Hospital Environment: Adaptive Workflow in the OR of the Future. In *Proceedings of the 39th Hawaii International Conference on System Sciences*, 2006.