MASTER

SPExFlex

a method for balancing standardization and flexibility in business process execution

van de Coevering, D.

Award date:
2017

Eindhoven, September 2017

SPExFlex: A method for balancing standardization and flexibility in business process execution

Dennis van de Coevering

Student ID: 0939075

In partial fulfillment of the requirements for the degree of

Master of Science

in Business Information Systems

Supervisors

dr. ir. I.T.P. Vanderfeesten (Irene)	**TU/e**
dr. D. Fahland (Dirk)	**TU/e**
dr. P.M.E. van Gorp (Pieter)	**TU/e**
Ms. S. Mooi (Suzanne)	**MUMC+**
Ms. C.M.G.H Winckers (Cindy)	**MUMC+**

Abstract

At the Maastricht University Medical Centre (MUMC) the outpatient departments are structured according to the different branches of medicine, such as rheumatology or cardiology. From an organizational point of view that is an effective structure, because resources often needed in certain situations are located together. However, in many cases it is not the optimal solution for the patient, because patients that have conditions that require consultations of multiple branches of medicine are faced with a lot of different locations, appointments, paperwork and duplicate actions. In an effort to improve the experience of the patient, the hospital is in the process of standardizing and centralizing her outpatient departments.

Combining the processes of all the different departments in a single standardized process is a difficult task. First of all, acquiring all the knowledge to design the centralized process is hard. Furthermore, supporting the process with an information system that does make sure the standard is followed by employees, but does not restrict their behavior too much so they cannot do a proper job anymore is hard. In other words, an implementation is required that balances both process standardization and process flexibility. Previous research in both the fields of process flexibility and process standardization has been combined in the SPExFlex approach to provide a solution to the problem. By applying the approach to the case at the MUMC it is shown that application of the approach in practice results in an implementation that considers both flexibility and standardization in a successful way.

Preface

Despite the fact that I enjoyed the last two years of studying Business Information Systems at the TU/e very much and that I definitely learned a lot, I still did not consider myself to be anything close to a scientist before starting this project. Therefore, I aimed for a graduation project with a practical focus to be able to apply the theory I learned in practice. The project at the MUMC+ was exactly what I had hoped for, by providing an inspiring environment with an interesting problem to be solved. However, the longer the project lasted the more I found myself becoming interested in the scientific foundation that lies underneath the practical work that I was doing. Finally, that resulted in this graduation thesis of which I can say that I am proud on both the practical as well as the scientific contributions that it achieved. This result would not have been as it is without the help of a number of people that I would like to thank.

As should be clear by now, it took a while before I got a clear vision on what my scientific value would be. Therefore, I admire the patience and consistent support showed by Irene while supervising me throughout the entire period. Irene, I can imagine that sometimes it must have been hard to understand where I was going with my project, whereas even I sometimes had no clue. Thank you for investing the time in me to find a common ground together and always remain positive throughout the project. Dirk, I would like to thank you for your feedback as it provided the final pieces of the puzzle to complete my work.

Cindy, thank you for involving me in the POM project and making me feel at home at the MUMC. By introducing me to a lot of people, you made sure I had the ability to acquire the information I needed and if not you were always available to provide me more. Suzanne, thank you for being a valuable sparring partner that I could always go to for both a good discussion and a nice chat. Henk, thank you for being the male companion that was sometimes needed in a female dominated team. Especially, your input in defining the different tasks was valuable to me. To all the other people I met at the MUMC, thank you for your help and for making my MUMC period a very pleasant one.

Finally I would like to thank family and friends for their support during my entire period of university studies. Mom, your ability to motivate me without me realizing it at the moment still amazes me. Thank you for your ongoing support and for providing a stable and loving home. Sis, your visits over the weekends always bring me the renewed energy needed to start the new week and our combined Sunday afternoon studying sessions were always a treat. To all my friends I would like to say thank you for providing a nice distraction, a listening ear and a helping hand during all these years, hopefully I could return the favour.

Dad, knowing me graduating from university was one of your biggest dreams for me was the biggest motivation of all. Your values and moral standards about commitment and collaboration have brought me where I am today and I want you to know that I still feel like I am on your shoulders.

Index

1. Introduction

At a hospital, the outpatient department (OPD) is the department where a patient enters the hospital (if not an emergency) and is at the center of the healthcare process. It is the main point of contact for the patient and treatment decisions are made by the physicians at these departments. The Maastricht University Medical Center (MUMC) is no exception with 34 different OPD departments all specialized in a specific area of healthcare. Currently the outpatient departments at the MUMC are organized as separate departments with their own business processes. Every department does not only own its own registration desk and waiting area, but it also operates fully autonomously in treating their patients.

That is not only visible in the daily routines, but it also propagates to the underlying fundamentals, such as the information system infrastructure. Although every department is working with the same ERP system, each has its own environment and information storage. As a result, substantial amount of patient information is not easily accessible by employees from different departments. Obviously, that means communication and collaboration between departments is limited. Another key issue is that resources are not shared between departments. A study by the MUMC on the utilization rates of its resources, such as examination rooms, showed that even at peak moments the utilization was still far from 100%. It was found that the main aspect that causes that, is the fact that resources aren't shared between departments, so that one department could have a resource shortage while another department experiences a surplus. These two issues lead to issues for patients that undermine patient satisfaction.

First of all, it is difficult for patients to find the right location for their appointment in a maze of registration desks for the different departments. Furthermore, when they attend multiple appointments at different departments they experience a lot of repetition in the procedures which could be avoided with proper communication. Finally, although not experienced directly, it might be that the quality of care could be better when a better insight in patient data could be provided. Therefore, the service for patients would greatly benefit from a unification of the outpatient departments into a single department with a single face towards the patient. Behind the scenes that would mean a standardized process with standardized tasks that ensure high quality and easy accessibility of data. Furthermore, a new infrastructure would allow for sharing of resources and closer collaboration between departments to improve process efficiency. However, to get from a situation which is very flexible and allows for many variations between the departments to a standardized environment is a difficult task.

To get to the standardized process the current processes would have to be merged into one general process for all departments. Although the resulting process would provide a clear representation of the behavior present at the departments, supporting execution of the process by information systems would be hard. Differences between the departments would be too substantial to be captured in a single standardized model, because it would too much restrict the behavior of physicians for them to properly do their job. In other words, it would be hard to find a good balance between standardizing the process and allowing flexibility in the execution of the process. The struggle to provide a process tailored to specific customers, but at the same time assure efficiency and quality of the process by standardizing the process can be found in other domains as well. Think for example about the insurance domain in which every customer has its own specific insurance and specific questions although they in general belong to a small number of different insurance packages. Therefore, an effective approach for tackling this problem is desirable.

Existing techniques have been developed in previous research that focus on both process flexibility and process standardization. However, an effective approach for combining these techniques could not be found. Therefore, this research aims at designing an approach based on those existing techniques, that, when executed, accomplishes a good balance in process execution standardization and flexibility.

1.1 Motivation

Techniques that focus on improving and monitoring business processes are clustered under the umbrella term 'Business Process Management', as the following definition shows:

"*Business Process Management (BPM) is the discipline that combines knowledge from information technology and knowledge from management sciences and applies this to operational business processes*" (van der Aalst, 2013).

Business process standardization is one of the highly valued aspects of BPM, because it increases performance, quality of products and services, collaboration, cost savings and it improves decision making (Púchovská and Závadský, 2016). Furthermore, in the healthcare domain standardization of processes can support in adhering to the many obligatory regulations and guidelines.

Another important aspect, however, is process flexibility, which allows for the execution of processes with a high level of uncertainty (Weber and Reichert, 2012a). It is estimated that, in the healthcare domain, in only 50% of the cases adherence to the imposed guidelines can be achieved (Zuiderent-Jerak, 2007), which shows that capturing actual healthcare processes in standardized models is a difficult tasks and flexibility during runtime is desirable.

It is clear that both aspects are important, but it is also clear that in their current use they contradict each other. Flexible systems are fully focused on supplying the user with all the variability he requires without taking standardization into account. While at the other hand standardization techniques lack the support to be able to deviate from the standard in a structured way. Therefore, a clear approach on how to apply the best of both practices and combine them in a single implementation is desirable.

1.2 Research goal

As the previous section shows flexibility and standardization are two contradicting aspects, and balancing them in an effective system for process support has proven to be a difficult task. Execution support for a process is often established through a workflow management system. Typically, such a system is established by tailoring it to a specific process to adhere to its specific needs. However, analyzing a process and effectively modelling and implementing it is a time consuming and error prone task (Aguilar-Savén, 2004)

Process flexibility and approaches to apply have been defined (Schonenberg *et al.*, 2008) that allow for the deferral of choices to the moment at which the decisions are actually made, instead of deciding on them beforehand. In this context, that would mean physicians can decide on what task to perform for a patient, instead of the choices having been decided upfront. Unfortunately, the mechanisms designed for allowing flexibility in workflow management undermine standardization aspects, because they often require underspecification of models.

On the other hand, techniques exist that allow for configuration of standardized models to fit specific contexts while still adhering to the standard (La Rosa, Wil M.P. van der Aalst, *et al.*, 2015). In this context that would mean there would be a standard model for all departments, but some slight changes to the model by departments are allowed. However, that would still mean all the choices for execution are made beforehand and the physician has no influence on the process regardless of the patient. Furthermore, the technique is restricted to the design phase, by lacking propagation to the execution level. That means, the resulting configured models are still process models that are not directly executable and would still require additional analysis of the specific processes to implement the process in a supporting system.

These insights combined lead to the conclusion that the techniques for a good balance between flexibility and standardization in the healthcare domain exist, but a good approach that defines how to combine these techniques effectively to get to a useful implementation in practice is missing. That leads to the goal of this research, which is to design such an approach that allows for a balanced implementation of a workflow management system in the healthcare domain.

Different types of processes exist in an organization, such as the MUMC, which are defined in a typology (Mooney, Gurbaxani and Kraemer, 1995). As the typology shows a main distinction can be made between operational and management processes. Operational processes are the key processes of an organization that make up the value chain of the organization. On the other hand, management processes that act as a support to the operational processes exist. When applying the topology on the healthcare domain, the operational processes are the healthcare processes that aim to cure a patient and the management processes are the, often more administrative, processes that support the people who participate in the healthcare processes. A workflow management system's main purpose is to manage task execution and resources, which are typically found in management processes. Therefore, the focus of the research is on management processes of the OPDs.

The previously outlined aspects can be combined a single research question:

> How can flexible workflow management and process standardization techniques be combined to balance standardization and flexibility in business process execution of management processes at outpatient departments?

The main goal of the research is to answer the aforementioned question by designing an approach and applying it in practice to evaluate its usability in an actual situation. The next section further elaborates on the steps taken through a concrete research design.

1.3 Research design

Peffers et al. (Peffers *et al.*, 2006) have defined a framework for conducting information systems (IS) research. The framework was created using a combination of techniques proven successful in previous publications on IS research. The resulting framework is shown in figure 1 below and used as a reference when designing this research.

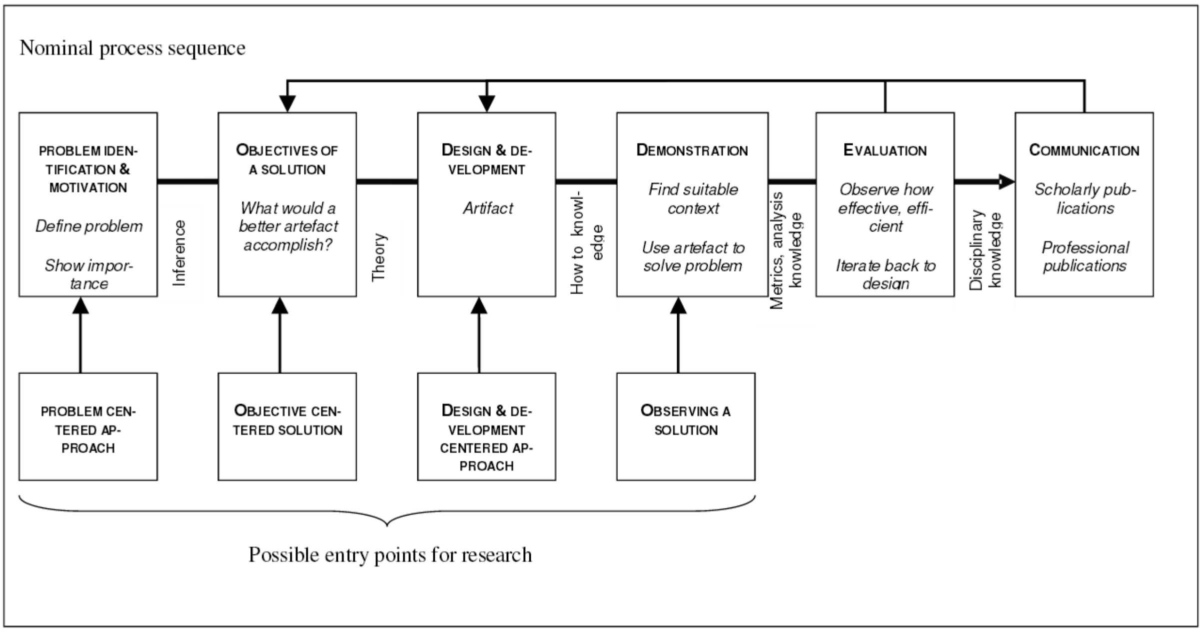


Figure 1: Peffers' framework for IS research

As the figure shows, an IS research consists of six main phases: problem identification, defining objectives, designing and developing a solution, demonstrating the solution, evaluating the solution and communicating the solution. Furthermore, the model shows several entry points for the process from which a research can commence. As mentioned before, the initial motivation for conducting the research emerged from a practical problem. Therefore, this research had a problem centered approach meaning the process commenced from the first phase onwards.

Apart from the main path, the model shows two recursive cycles that can be performed. The outer cycle allows to design new objectives based on the communicated results. After the final findings are presented they can input for future research, but that is outside the scope of this project. The second (inner) cycle loops over design, demonstration and evaluation to allow for an iterative development approach.

Based on the phases in the research as defined by the framework, the main research question was divided in five different sub-questions as shown below.

Q1. What is the main problem and why should it be solved?
Q2. What should the research accomplish?
Q3. What are the existing techniques that enable process flexibility and how have they been applied so far?
Q4. What are the existing techniques that enable process standardization and how have they been applied so far?
Q5. How can the techniques that were found be combined to a single approach?
Q6. How well does the approach function when applied in practice?
Q7. Can changes to the model be applied based on the acquired experience in practice?

Each question relates to a specific phase of the IS research framework as shown in figure 2. The previous sections have provided answers to both Q1 and Q2 by staging the context and problem of the research and defining the research goals based on that. The remainder of this chapter provides an overview of the steps that were performed to answer the remaining questions.

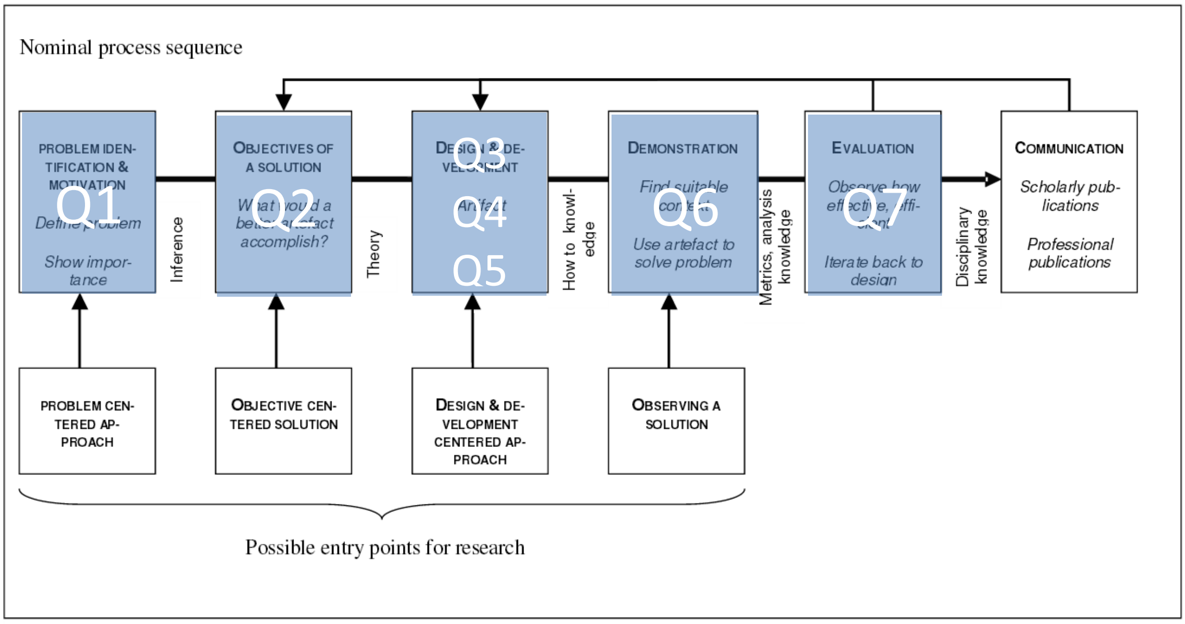


Figure 2: Relations between research questions and the research design framework

1.3.1 Design and demonstration

The design phase started off with an extensive literature review on the existing techniques for flexibility in workflow management and process standardization techniques. The aim of the literature study was to gain a thorough understanding of existing techniques in these fields to identify the strengths of both and utilize them in the further design of the approach. Finishing this phase resulted in the answers to Q3 and Q4 that are presented in chapter 2.

The next step was to design the main artifact of the research. The artifact for this research is the approach as it was defined when defining the research goal. During this phase the acquired knowledge from both literature and practice were combined to create the artifact. Creating the artefact provides an answer to Q5.

In the demonstration phase, the approach created during the previous phase was put into practice during a case study at the MUMC+ to find the answer to Q6. The goal of the case study was to put the approach into practice by performing the steps as proposed by the approach to implement a standardized process for the outpatient departments. More detail on the context and scope of the case study will be provided in chapter 4.

Executing the approach resulted in a prototype of a workflow management system that supports execution of the process. The prototype was used to evaluate the results of using the developed approach by testing whether the methodology resulted in adequate process support that is perceived as valuable. The results of the case study are presented in chapter 5.

1.3.2 Evaluation and communication

After demonstration, the full process was evaluated. The proposed steps by the approach were compared to the actual steps taken during demonstration and changes were proposed accordingly. An explanation of the proposed changes is presented in chapter 6. Furthermore, it provides an answer to Q7, thereby answering the final sub-question. Based on the found answers to the sub-questions, an overall answer to the research question could be formulated.

Finally, Peffers mentions the communication phase, which is important to allow easy access to the acquired knowledge for future reference. This report is the main part for communication of this research. Besides the report, two presentations were organized. One at the hospital for people interested in the progress and the results and a second at the university for people interested in the academic achievements.

2. Theoretical Background

The goal of this chapter is to provide an overview of the existing literature. On the one hand about workflow management and related flexibility concepts and on the other hand standardized process modeling techniques and related configuration concepts were analyzed. For both fields, first a general definition is provided, before highlighting typical applications in practice.

2.1 Workflow management

Workflow management refers to the automatic execution of processes by focusing on the people and software involved in the process. Typically, a process model is designed that defines the ordering of process steps and the relations between them. The process model can be designed and implemented in a workflow management system (WfMS) which executes the process by assigning tasks to specific users and providing an interface for task execution (OuYang *et al.*, 2010a).

Motivators for using workflow management can not only be found in the efficiency and the quality of the process, but also process analysis and legislative advantages were identified. Applying workflow management is an effective approach for assuring compliance with guidelines/regulations and it also limits differences in the execution of specific cases making it easier to trace back errors and improve the process (Reijers, Rigter and Van Der Aalst, 2003; OuYang *et al.*, 2010b).

2.1.1 Typical applications

The workflow management coalition defined a reference model for workflow management systems and their usage in practice (Hollingsworth, 1995). A high-level approach for applying workflow management systems, as defined in that reference model, is depicted in figure 3 shown on the right.

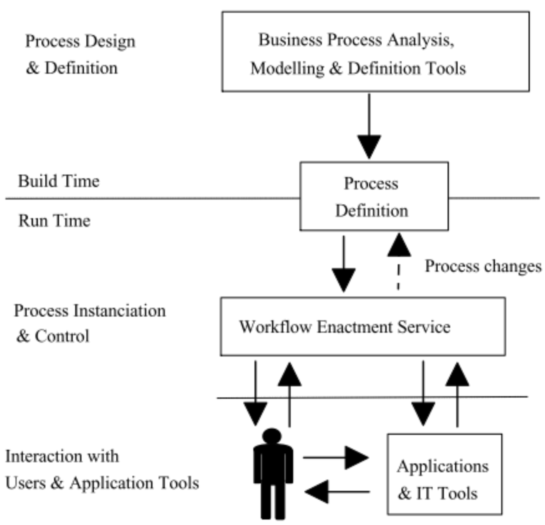


Figure 3: Workflow management in practise

As the figure shows, two main components make up the workflow management system: the process modeling and definition tool and the workflow enactment service. They are the results of two main phases in workflow management system implementation. The modeling/definition tool is used to model the behavior of the process during 'build time' and the enactment service supports execution of the modeled behavior during 'run time'.

In workflow management systems, the modeling tool is typically a visual modeler that allows business analysts to create a computerized definition of a business process. Usually, the modeling language allows for tasks that can be ordered. Furthermore, actors that perform the tasks and rules that determine how the tasks should be executed are included as well as the information aspect that shows the input and output of data that is required for each task to be executed properly. The executable process model including all its task, resources and data is called the process definition.

The enactment service assigns the modelled tasks to the correct roles at the right moment in time and ensures the tasks are executed correctly. Furthermore, it manages cases to make sure no data is corrupted in case of errors during the execution.

2.1.2 Flexibility in workflow management systems

Traditional workflow management was aimed at well structured, repetitive processes that can be easily captured in a workflow specification. However, often processes require a level of flexibility to deal with exceptions and/or to cope with uncertainty. A taxonomy for process flexibility has been designed that identifies four main categorizations of flexibility required in process support: variability, looseness, adaptation and evolution (Weber and Reichert, 2012a).

Variability allows execution of different case variants based on pre-defined variables, for example the assembly of different product types. Looseness ensures adaptations to the process based on parameter values that are not known beforehand or that change during execution. Adaptation is needed to anticipate to occurrences of exceptional events. Finally, evolutionary flexibility allows the implemented process to evolve according to the corresponding business process.

Furthermore, four main approaches to ensure these types of flexibility were identified (Schonenberg *et al.*, 2008): flexibility by design (variability), flexibility by deviation (adaptation), flexibility by underspecification (looseness) and flexibility by change (evolution). Extensive research resulted in several techniques implementing one (or multiple) of the approaches as mentioned before.

The principle of flexibility by design is to incorporate the possible variants in the process model during the design phase. The Workflow Management Coalition (WfMC) has defined a series of 'workflow patterns' (van der Aalst, ter Hofstede and Barros, 2003), which are standardized combinations of routing elements that cover all possibilities to be flexible when designing the process. Most workflow management systems support these patterns both in design as well as during execution.

Techniques that cover flexibility by adaptation are referred to as 'exception handling' techniques (Hagen and Alonso, 2000; Russell, Van Der Aalst and Ter Hofstede, 2006). A common approach is to define a series of protocols that can be executed when an exception occurs depending on the type of exception.

Looseness is typically achieved by applying a declarative modelling approach (Pesic and van der Aalst, 2006; Van Der Aalst, Pesic and Schonenberg, 2009). Declarative modeling is opposite to classic process modeling techniques since it focusses on modeling what should not happen instead of modeling the complete behavior. By defining constraints, choices can be deferred to runtime at which choices can be made based on the specific case context.

Evolutionary flexibility focusses on changes that are made during a new design phase and how to propagate these changes down to the execution level. Important is how to apply these changes to running cases (Weber and Reichert, 2012b)

Extensive research in how to apply these flexibility approaches in workflow management systems has resulted in the notions of 'adaptive workflow management' (Han, Sheth and Bussler, 1998), 'adaptive case management' (Motahari-Nezhad and Swenson, 2013) and 'case handling' (Reijers, Rigter and Van Der Aalst, 2003; Van Der Aalst, Weske and Grünbauer, 2005; Chiao, Künzle and Reichert, 2013). These techniques show a transition from a focus on the control flow towards a more data- and process driven approach. Such an approach enables a focus on individual cases and their specific contexts during process execution. When relating that to the workflow reference model, it can be observed that by applying these techniques the strict boundary between the build and runtime is softened. By incorporating flexibility concepts during the build time, it is ensured that not all decisions are made at build time as it was done in classic workflow management system implementations.

2.1.3 Applications in healthcare

The potential of applying workflow management in the healthcare domain was already identified seventeen years ago (Dwivedi, Bali and James, 2001). With the large amounts of patient data and the high amount of repetitive tasks the healthcare domain seemed suitable for workflow management application. Six years later the potential was still not exploited, because process aware information systems in hospitals were still rare (Lenz and Reichert, 2007) and workflow support by an IT system was still identified as a premise for the future.

Again three years later, in 2010, it was identified that usage of workflow management was still limited in the healthcare domain (Reijers *et al.*, 2010) and a method was proposed to adjust classical workflow management to be suited for the healthcare domain. In short, this method included an extensive analysis of all the possible exceptions and applying the flexibility techniques as described before to adjust the existing process so it could cope with the identified exceptions. The method was proven useful in a case study, but multiple limitations were identified that have proven to be too big to allow application on a big scale.

When doing a research in recent literature, systems that provide adaptive case management for use in the healthcare domain can be found, such as a system to support treatment of chronic patients (Cano *et al.*, 2015). That system is mainly based on flexibility through looseness by having a human actor actively assign tasks and goals to other users. Although this is a good effort, evidence of a widespread acceptance of operational data-driven adaptive case management systems that support physicians (and other users) in executing their daily tasks could not be found.

Therefore, the conclusion can be drawn that workflow management in the healthcare domain is proven to be harder than expected at first. That observation is backed by the problems observed at the MUMC as it was one of the motivators for performing this research.

2.2 Process Standardization

Standardizing its processes provides a business with multiple benefits (Hammer and Stanton, 1999). First of all, overhead costs are reduced, because only a limited amount of processes needs to be managed. That means, for example, that only limited training material is required and a smaller amount of information systems is required to capture all required functionality. A second benefit is that a single face is shown to customers and suppliers. Finally, it provides more flexibility from a resource perspective, because resources can more easily be relocated to other parts of the organization, since they are already aware of the procedures.

However, standardization is not only applied within organizations, but also to enable knowledge sharing in the domain. Modeling a process is a time consuming and error prone task (Aguilar-Savén, 2004), and therefore, standardized models for certain non-specific processes is desirable. More and more of these so-called 'reference models' have been created in recent history (Fettke, Loos and Zwicker, 2005).

Multiple process design techniques have been developed that can be used to design a process reference model. A top down approached can be used in which the desired behavior is analyzed from scratch by thinking of the behavior the process should allow and modeling that behavior in a model. An example of such an approach which is based on first determining the goals and deduct the model from that is described by Kueng and Kawalek (Kueng and Kawalek, 1997). Another approach is to observe the current process, create an as-is model from the current situation and then alter that model to get to the to-be situation. Conventional knowledge acquisition techniques (Cosgrove, 1991) can be used to get the information needed for creating the model, but recently more often a data-driven approach is chosen. A common data-driven approach is called process mining (Rojas *et al.*, 2016) which aims at extracting process models from log data from systems currently in place. However, using those techniques is only desirable if current information systems already provide proper logging of activities and enough data is available to mine a model which is a good representation of practice.

Common processes that occur in many organizations, such as support processes, can relatively easy be captured in a reference model that applies to many organizations. However, it was found that when applying reference models in processes still a lot of changes had to be made to the original reference model, which undermined its original purpose. As a solution to that problem, configurable process models emerged which allow the configuration of reference models to a specific context (Hallerbach, Bauer and Reichert, 2008; Gottschalk, 2009; La Rosa *et al.*, 2009; Gröner *et al.*, 2013). In that way, reference models can both easily be applied as well as being fit to the specific organizational context. The use of configurable models in practice is illustrated by figure 4 which shows how the reference model functions as a starting point for process model implementation in multiple organizations.

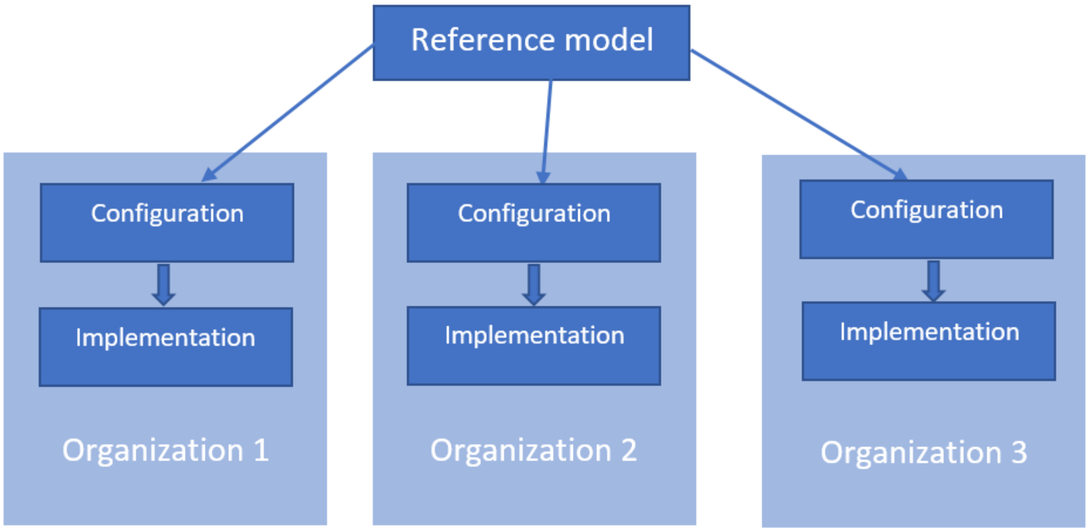


figure 4: Application of configurable reference models in practice

Several techniques for configuring process models have been defined (La Rosa, Wil M. P. van der Aalst, *et al.*, 2015). The common aspect shared by the techniques is that 'variation points' are identified in a reference model at which modifications to the model can be made that typically include addition, deletion or skipping of activities. Consequently, a concept was introduced that enabled easy access and control over process families, which is the complete set of configured models corresponding to a certain reference model. Process families are typically managed through a process repository (Yan, Dijkman and Grefen, 2012). A process repository allows for easy management and access of process models, comparable to a database for data. The next section provides a more elaborate overview of process model configuration.

2.2.1 Model configuration

An extensive survey was written on different process configuration techniques and their comparison (La Rosa, Wil M. P. van der Aalst, *et al.*, 2015). From that survey, several commonalities among the different techniques can be observed. In general, the approaches consist of three main phases as shown in figure 5.

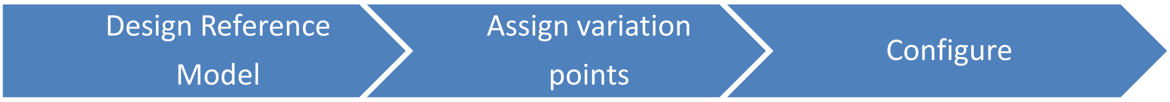


Figure 5: Main approach for configurative process modelling

The first phase consists of defining a reference model for the process. During this phase, a general model that captures all the behavior that is allowed in the execution of the process is designed. Most model configuration techniques assume a reference model for the process already exists. If not, standard process modelling techniques are used to create one.

During the second phase variable elements in the process model are defined. These are the elements or groups of elements that can be configured during the final configuration phase. The methods for configuration have in common that they involve selection of tasks or groups of tasks that can be deleted or hidden from the reference model. A small example of a configurable process model is provided in figure 6. As the example shows, configuration options have been highlighted on the model for each task. In this case, a task can be allowed (green arrow), blocked (red sign) or skipped (orange arrow).

Examples of applications for modeling configuration options are model projection ((Becker *et al.*, 2004), in which views of a process model can be created by selecting parts of the reference model to be deleted, configurable EPC's ((La Rosa *et al.*, 2009) in which OR elements are introduced that allow for configuration choices in the model and configurable workflows that allow for direct configuration of workflow models (shown in figure 6)(Gottschalk *et al.*, 2008). Furthermore, techniques exist that use model transformations (Puhlmann *et al.*, 2005; Hallerbach, Bauer and Reichert, 2010) to configure the reference model. Interesting about these techniques is that insertion of elements is also included as opposed to the other techniques. Finally, more abstract methods exist that, for example, define configuration points by creating constraints on a meta-level (Czarnecki and Antkiewicz, 2005) or apply cardinality attributes for configuration (Reinhartz-berger and Sturm, 2004).

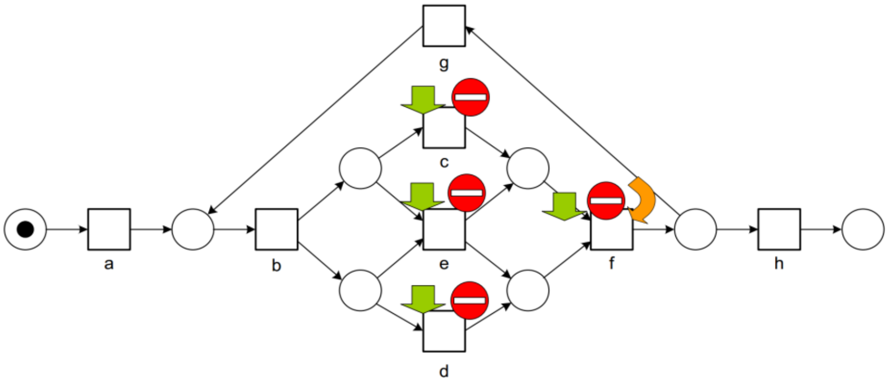


Figure 6: A simple example of a configurable workflow model

Similar to the workflow management field, a trend towards a more data-centric approach can be observed. As an example, a technique was designed as an extension to configurable workflow models (Gottschalk *et al.*, 2008) that allows for data based configuration (Fahland and Hagen, 2016). By introducing dynamic skipping and blocking of activities, configuration of the model can be (partly) done based on data provided during run time. Therefore, a more case based configuration can be achieved.

In the end, the configuration phase consists of making pre-defined choices in such a way that the model is suited to a specific context. Whereas the goal is to configure reference models in specific domains, techniques have been developed to abstract from specific modeling languages such that domain experts without modeling knowledge can perform configurations. Two main techniques were identified for decision support in model configuration (La Rosa, Wil M.P. van der Aalst, *et al.*, 2015).

The first includes Feature models (Schobbens *et al.*, 2006), which are a visual representation of the key features of a process. By creating a feature model, domain experts can decide on dependencies between process features. The feature model can then be translated to a model configuration by identifying the feature dependencies and translating them to task dependencies. Another type of decision support focusses on questionnaires (La Rosa *et al.*, 2006). By translating configuration points into straight forward questions, a model can be configured by answering a series of questions.

In most approaches, the end result is a configurable reference model that can be used to create a family of process models. That means the created models are still not executable models that can directly be used in practice. That means it is a strong method for improving standardized models, but it does not provide support for effective execution of these models.

2.2.2 Applications in healthcare

Care pathways are defined as: “structured multidisciplinary care plans which detail essential steps in the care of patients with a specific clinical problem” (Campbell *et al.*, 1998). According to that definition and the definition for reference models as described before it can be concluded that care pathways function as reference models for medical treatment processes.

Care pathways are a broadly accepted practice and are used in most hospitals as a reference to structure their processes. Furthermore, configuration has been introduced for care pathways as well (Milla-Millán *et al.*, 2013), however it focusses on the planning aspects of the process instead of direct execution support.

That means standardization is introduced for the medical processes, but not so much for the management processes. Furthermore, execution support is only introduced for simple planning tasks and it is not used to support physicians in their daily work. Therefore, there is still a lot to gain.

2.3 Conclusion of findings

The relevant concepts that were identified while performing the literature study are summarized in figure 7. The figure shows the two main concepts, workflow management and standardization, and the techniques that have been described that were introduced to bring them closer together. The gap that was identified is shown in the figure by the red circle, and the goal of the artifact will be to narrow this gap and allow for a combination of both aspects together in practice.

.

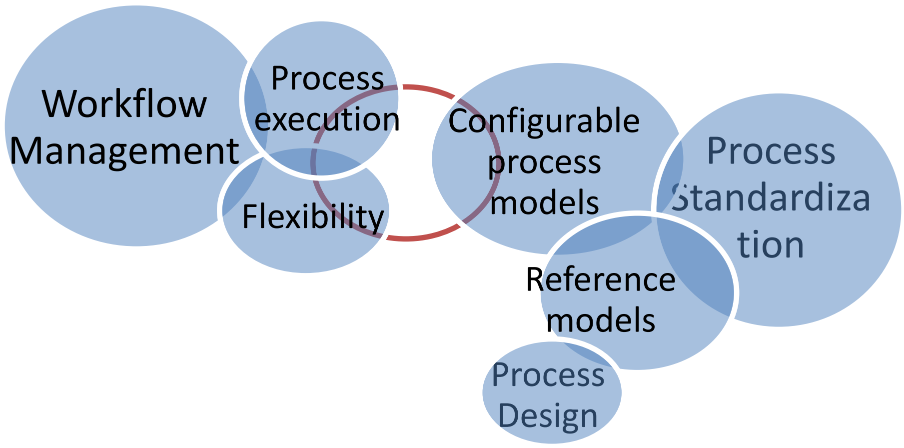


Figure 7: Theoretical concepts and their relations

3. The SPExFlex approach

The goal of this chapter is to provide an answer to Q5: How can the techniques that were found be combined to a single approach? The chapter starts of with an analysis of the techniques presented in the previous chapter before combining them in the Standardized Process Execution Flexibility (SPExFlex) approach. To gain a full understanding of the approach, each step of the approach is then explained in more detail.

3.1 Combining techniques

When the workflow reference model of figure 3 and the application of configurable reference models as shown in figure 4 are directly combined the resulting situation would be as depicted in figure 8. As the figure shows, the reference model is an input for the design phase and functions as a template that is used during the design phase. Therefore, configuration of the model would occur during the design phase. When the design phase is finished, a configured model is created that captures the behavior of the process for that specific organization.

Based on that model the other steps are performed as defined by the workflow reference model. First a process definition is created by adding information about data and roles to the model. The resulting process definition is an executable model that is then implemented in a workflow management system. If implementation is successful, the workflow management system will then be able to support execution of the process.

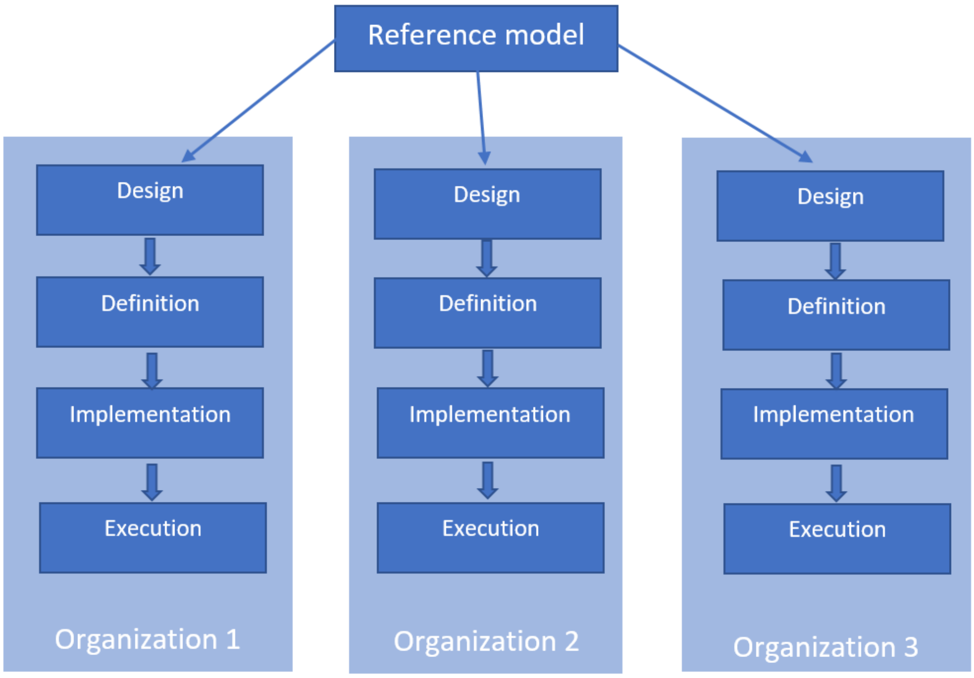


figure 8: Schematic overview of combining configurable models and workflow management techniques

The next step is to alter this direct combination in such a way that it solves the problems as highlighted before. As was highlighted in chapter 1, the goal is to standardize the different processes of the different OPDs at the MUMC in to one standardized process. When looking at figure 8, several aspects can be identified that undermine this goal. The focus of this research is to apply configurable process modeling within a single organization instead of between multiple organizations as shown in figure 8. Because the OPDs operate according to a process that is similar at a high level and that only differences occur due to differences in the medical specialism, it is believed that the configurable process modeling techniques can be applied in this context as well.

In the original application, the configuration and the implementation of the models is strictly separated. Every configured model has its own implementation of that single model and no awareness of the fact that the model is a configured model or not is necessary. However, when applying model configuration in a single organization that changes. Within a single organization, it is not desirable to have a different workflow management system for each department, because it would be a great effort to implement and maintain all these different systems. Therefore, a more desirable solution would be to have a single implementation of a workflow management system that is capable of handling different model configurations as shown in figure 9.

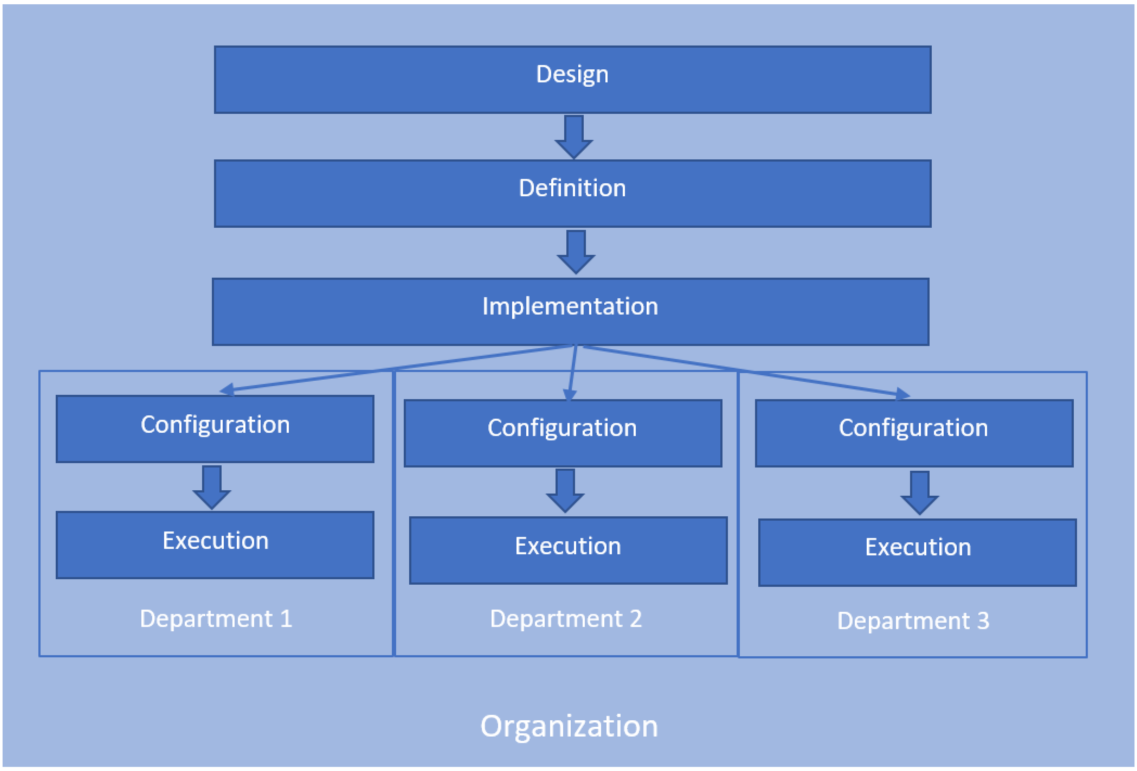


figure 9: Required solution

As opposed to figure 8, an extra 'configuration' step has been added. In the original situation, the configuration is included in the design phase, because the reference model is used as a template for the design phase. In this case, the design phase focusses on creating the reference model and implementing it. Configuration of the model can then be done at the different departments before executing the process. The next section provides an overview of how that would work exactly.

3.2 Detailed overview of the approach

Based on figure 9 a general high level overview of the approach consisting of five stage was created as shown in figure 10. The first four phases each have a specific goal and create an artefact that is utilized in the next stage. Finally, in the fifth stage this all comes together through process execution. In this section each of the five stages are explained in more detail before presenting the final full detail approach.

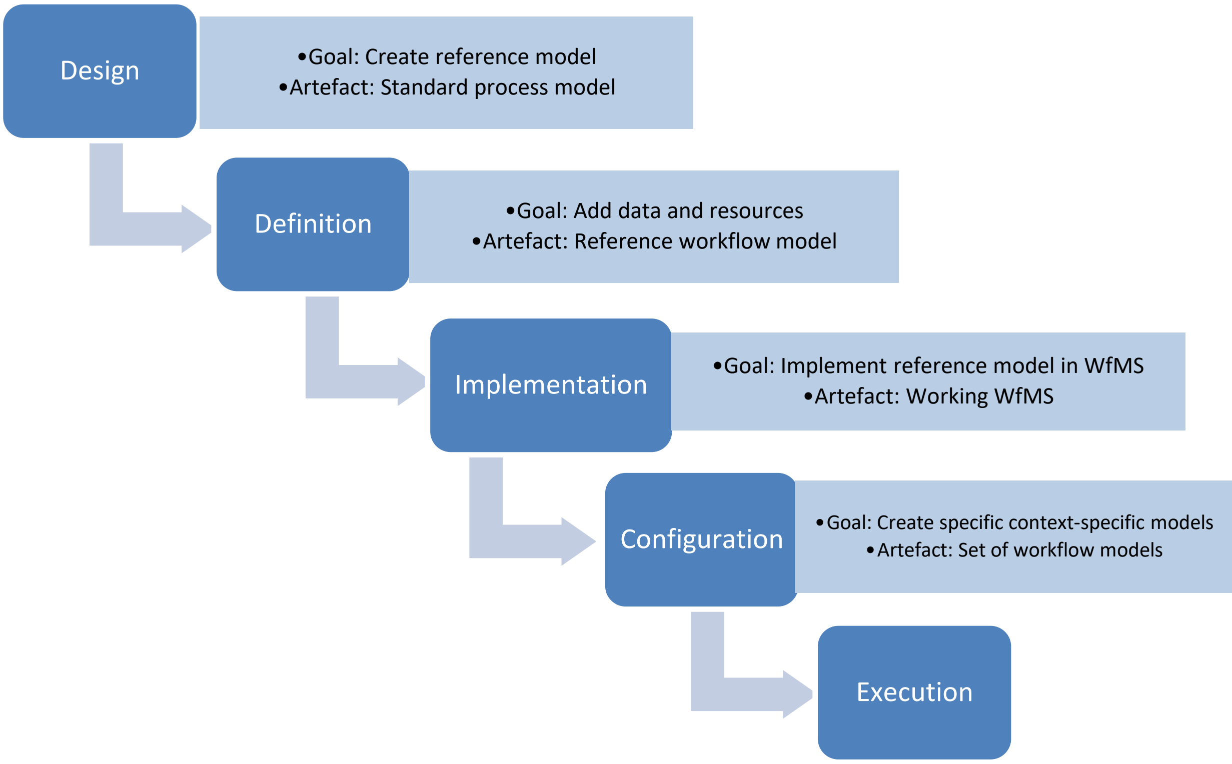


Figure 10: High-level overview of the five phases of the approach

3.2.1 The design phase

The first phase is the design phase and has as goal to capture the behavior of the model in a reference process model. The reference model should possess all behavior that can be found at the different departments. A previous study on the design of a process in the healthcare domain (Reijers *et al.*, 2010) resulted in a validated methodology for the design phase that is used as reference for this design phase. Figure 11 below shows that methodology consisting of four stages.

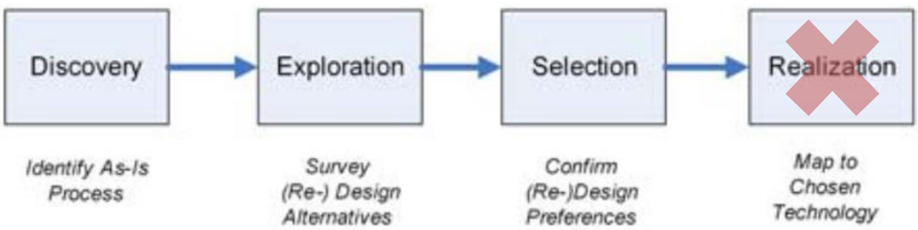


Figure 11: The four steps of IS design

During the discovery step the as-is situation is analyzed and modelled. For the analysis, common process design techniques as they were explained in section 2.2 can be applied. During the exploration phase, all possible extensions and/or changes to the as-is situation are discovered. Usually, this occurs during brainstorm sessions with users and other stakeholders. The selection phase focusses on selecting the aspects that will be incorporated in the to-be model from all the aspects that were collected in the exploration phase. The selected changes are than processed on the as-is model to get to the desired model of the to-be situation. In the methodology used by Reijers et al. (2010), a realization step is also included, that consists of a system selection based on the required functionality. This step is part of the 'implementation' phase of our approach, and therefore not incorporated in the design phase.

To capture all required behavior in the model, flexibility by design has to be applied. That means that different alternatives for certain parts of the model have to be modelled to capture the fact that different departments might perform similar tasks in a different way. In the next phase, the allowed behavior is restricted by defining the variation points.

3.2.2 The definition stage

Finishing the design phase results in a reference process model that functions as input for the process definition stage. The goal for the definition stage is to extend the reference model to become a process definition. To recall from figure 3, the process definition is the executable workflow model that is the input for the workflow enactment service. According to the workflow reference model (as introduced in section 2.1.1) *"a process definition normally comprises a number of discrete activity steps, with associated computer and/or human operations and rules governing the progression of the process through the various activity steps".* To achieve that, three main steps were identified that together make up de process definition stage: Task definition, Role definition and Rule Definition.

Figure 12: Steps for the process definition phase

Typically, after a process design the tasks are defined by informal descriptions or by a single label. For the task definition step, the purpose is to make the different tasks more concrete. To do that, a data model for each task has to be created that shows what information is required as input, what information is produced by each task and what needs to be done to get from the input to the output. Furthermore, it is important to know where the input is retrieved from and where the output should be stored.

Apart from defining the tasks that were included in the reference model, extra tasks might have to be added that are required by the system to execute the model. The so-called 'system tasks' are tasks that will not be visible to users as actual tasks, but that are required to perform certain operations in the background. Typically, these are automated tasks that are performed based on the information gained during process execution. An example of such a task would be that the system automatically sends an email with the summary of a specific task after it has finished. The system tasks also need to be specified and the code that needs to be executed when the task is executed needs to be defined.

When concrete tasks have been defined it is important to specify who should perform the tasks. The role definition step focusses on defining the different roles involved in the process and assigning a role to each task. For the execution of the process it is only important that the system knows who to assign a task to. Therefore, for the workflow management system the only thing that is required for a role definition is a unique identifier. However, within the organization a clear understanding of each role is required to be able to determine who to give what role and what roles to assign to a task.

Finally, the rule definition step should ensure that all extra execution rules are defined that are not captured in the original reference model. Examples of such rules are that a task might only be performed by someone who also performed another task or that a delay is required before a task is offered. Also included in this step is the modeling of the variation points to extend the reference model to be a configurable model. Using one of the configuration approaches as described in section 3.1, the configurative aspects of the reference model can be identified. The goal of this model is to highlight all parts of the reference model that are configurable. Furthermore, the constraints on the configurations need to be specified. To apply standardization to the highest degree possible the goal is to create a complete model in the definition phase, and hence, to not allow adding of model element by configuration. Therefore, only configuration techniques should be used that apply elimination/hiding of model elements disallowing flexibility through looseness.

Besides highlighting the variation points in the reference model it is also important that the conditions are stated for application of the configuration. A set of variables should be defined that determine in what specific situation that model should be applied. An example of such a set of variables could be: department, diagnosis, type of appointment and the physician that is consulted.

3.2.3 The implementation phase

At this point all the behavior and the possible variations have been modelled in a configurable process definition. The goal of the implementation phase is to implement the configurable process definition in a workflow system in such a way that is can still handle configuration. Two main steps can be identified in the implementation phase: system selection and system integration. The system selection step ensures the functionalities required are analyzed and the required system(s) are selected. During integration it is analyzed how the selected systems can be connected to the existing systems in order to successfully implement the system(s) in the current environment.

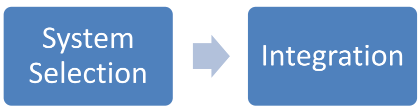


Figure 13: Steps for the implementation phase

In a typical workflow management system implementation, the fully specified model would be deployed in the workflow management system and would be fully defined to be ready for execution. However, in this case the model still requires configuration. Two options can be identified when trying to resolve that issue. The first option would be to configure the model in the workflow management system modeler, for example by applying Configurable YAWL (Gottschalk *et al.*, 2008) or the WebFlow application as provided by SAP (Gottschalk, Van Der Aalst and Jansen-Vullers, 2007). An advantage of that option is that only one system is needed for the modeling, implementation and execution.

Disadvantages are that you are bound to a specific workflow modeling language and that abstraction from the modeling language for configuration is more difficult, because the configuration has to happen within the environment of the workflow management system.

A second option would be to set up the workflow management in such a way that configurations can be considered by the workflow enactment service at runtime. The process can then be configured outside of the workflow management system which allows for more freedom in the variety of modeling languages and model configuration techniques. A disadvantage lies in the fact that a more complex infrastructure is required to import the configurations in the workflow management system. As was identified in chapter 2, one of the strong aspects of applying model configuration is that it can be done by domain experts who do not have explicit process modelling knowledge. To allow for that, model abstraction from the original modeling language is required. Therefore, this option was considered the better one, and applied in this approach.

For the system selection step, the first aim is to select a workflow management system. Many systems have been designed and based on the required functionalities a system can be selected. Furthermore a system has to be introduced for model configuration. The functionalities of that system include that it can translate a process model to an abstracted version and that it can allow users to configure the model based on the abstraction. An example of such an abstraction is to translate variation points in a process model to questions and combine them in a questionnaire that needs to be completed by the user. The answers of the questions can then be used to configure the model.

Furthermore, the configured models need to be stored. Often, process repositories are used for that purpose. A business process repository is a database for process models that offers standard database functionality such as storage, retrieval, updates and deletion of process models (Ma *et al.*, 2007). Process repositories are used to manage large collections of process models and will be used to manage all the configured process models. A framework for realizing a process repository system has been defined (Yan, Dijkman and Grefen, 2012) and an application for how workflow models can be managed and how existing knowledge can be used to improve existing models has been presented (Madhusudan, Zhao and Marshall, 2004). Besides providing support for managing process models the repository can also be used by the workflow management system to retrieve the models that need to be executed. Based on the variables that were defined in the process definition a query can be created that can be used to query the process repository for the correct configuration. That configured model can then be accessed by the workflow engine and the configurations can be applied during execution.

Note that the configurable process definition implemented in the workflow engine is not a configurable model anymore. When implementing the process definition, flexibility by design should be used to incorporate all the different possible configurations. One way of doing that, would be to include data based decisions that check a certain condition, that is set when initializing the case based on the configured model.

After that, a clear overview of the functionalities that are needed is established together with the systems that will provide that functionality. To integrate the system in the existing infrastructure two important aspects were identified. First, a link should be established onto the existing data structure. The workflow management uses data from a central storage and, therefore, it is important that the variables needed for execution are stored in a single place in the database and that other systems access the same variables. Furthermore, the data model that is defined for each task in the process definition can be used to identify the links that are required to other systems. Based on those two, the links can be created to the existing systems that necessary for the workflow management system to function.

3.2.4 Configuration and execution

Configuration happens through the abstraction tool as defined before. Domain experts, such as physicians, can define their model configurations based on how they would like to see the process for different kind of patients. The different configurations are stored as explained before. The simplest example of a clear case type distinction is the distinction between new patients and returning patients. For new patients different information might have to be gathered, that is different for returning patients.

When initialization of a case is triggered, the workflow system checks the defined variables that determine the case type and queries the corresponding configuration from the repository. The case variables are set based on the configuration and the first task is assigned. When a decision has to be made that is defined by the configuration, the workflow engine can check the set case variables and, hence, make the decision.

Combining all the steps as described in the previous sections the resulted in the complete SPExFlex approach overview as it is shown in figure 14. Thereby, Q5 has been answered and the next step is to apply the approach in practice of which the context is staged in the next chapter.

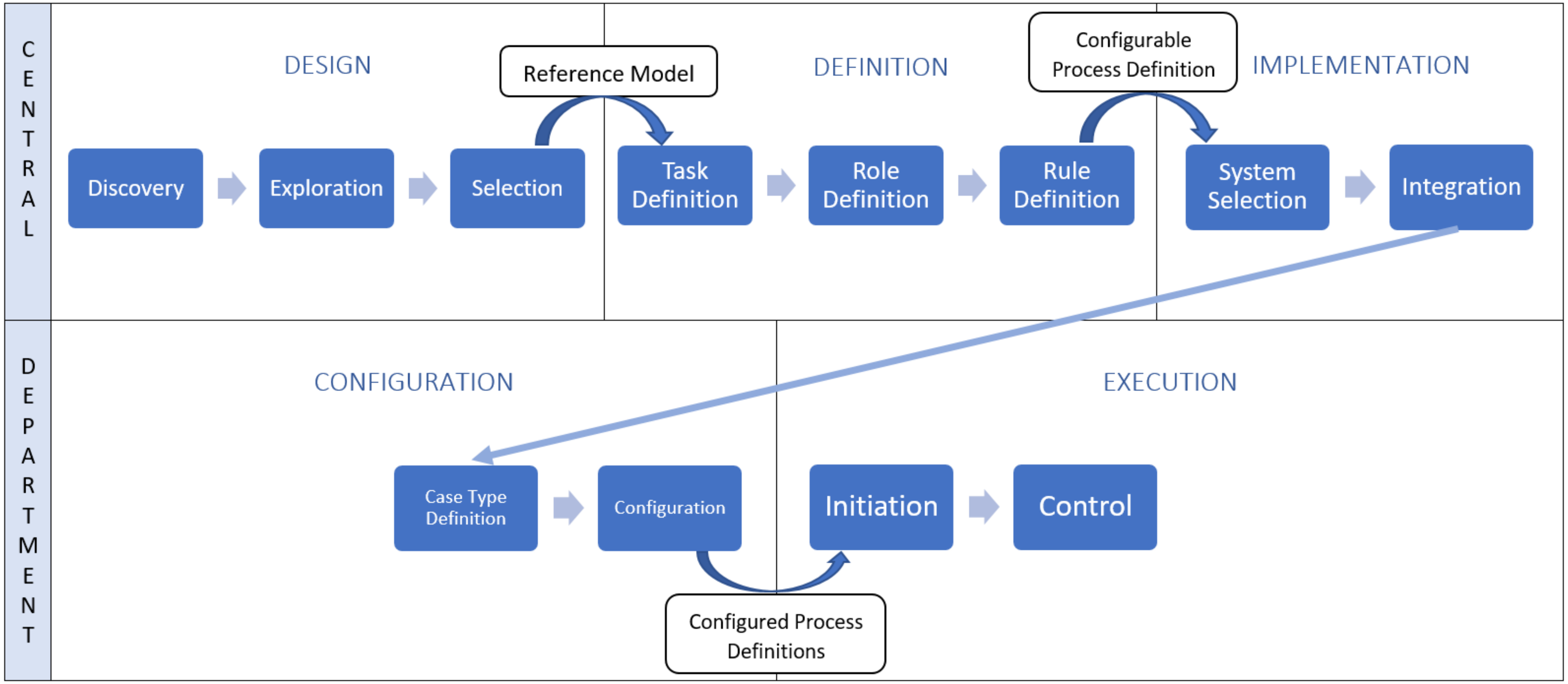


Figure 14: The SPExFlex Approach

4. Case study: Setup

The proposed approach was tested in a practical setting during a case study. The case study was performed in collaboration with the MUMC+ and the results are presented in this section. This chapter will provide the context in which the case study was performed and it will define the scope and goals of the case study. The results of the case study will be presented in chapter 5. The first sections provide additional information about the organization and the practical problem as introduced in chapter 1. Furthermore, the situation at the starting moment of the case study was identified. Finally, section 4.5 explains the goals and scope of the case study.

4.1 MUMC+

The Maastricht University Medical Center (MUMC) is a healthcare organization situated in the south of The Netherlands. The organization is part of an umbrella organization, called Maastricht UMC+ (MUMC+) in which the MUMC and the Maastricht University are combined to ensure close cooperation between the two organizations. The organizational structure is shown in appendix 1.

In total, about 7000 employees and 4000 students are part of the MUMC+. Most relevant for this research is the MUMC, which consists of eight main Results Responsible Units (RRUs) supported by service and staffing departments. Each RRU is responsible for all the facets of care relevant for her specific field. Examples of these facets are operating theaters, outpatient departments and nursing departments.

The focus of this research will be mainly the outpatient departments of the MUMC. For 2018 a large renovation of the outpatient departments section of the hospital is planned. To create a new situation that is future proof, it was decided that a redesign of the processes would be leading in designing the requirements for the new building. Therefore, a project team was founded that is analyzing the current processes and redesign the process in collaboration with the outpatient department employees and physicians. The project is a separate organizational body and, hence, cannot be situated directly at one of the organizational elements as shown in appendix 1. However, the project is authorized by the management board AzM and the results will be directly reported back to them. The result of the project will be an impact analysis consisting of the impact on capacity, personnel, IT, planning and costs. This research will be conducted as part of this project and, therefore, the next section will provide more detail about the outpatient departments.

4.2 Outpatient departments

An outpatient department (OPD) can be defined as a part of an hospital where patients are treated that do not need to be hospitalized. It allows patients to see a specialist within the context of an hospital, and with the facilities of the hospital at hand. Typically, the specialists working at the OPDs are the first point of contact for patients entering the hospital. Examples of actions being performed at an OPD are for patients to get diagnosed, to get prescriptions or to be transferred to other departments for further tests/treatment. OPDs mainly consist of simple examination rooms in which specialists can do simple examinations of patients and discuss test results and treatment plans.

The UMC houses 34 OPDs that can be visited by patients. They are organized based on the different medical areas, meaning a patient visits a specific outpatient department based on his or her medical condition. Patients can enter an OPD through a referral by a general practitioner or through the emergency room. In the current situation, although situated in the same hallway, every OPD has its own location complete with its own registration desk and administration. Registration desks are not only responsible for registering patients, but also for planning appointments and printing passes needed for other departments. Furthermore, every department has its own dedicated waiting area and its own staff. In other words, from a process point of view the departments are completely independent from each other.

The main ERP system used in the hospital is based on the SAP system and is also used at the OPD to schedule appointments, view patient information and for desk employees to communicate with specialists. The electronic patient administration is an organization wide system that is the same for each department. However, for scheduling and processing of consultations each department has its own environment within the SAP system. However, these specific environments are basically just applied filters on the entire information available (e.g. selecting a specific OPD filters out all appointments for other OPDs) combined with some configured settings. Hence, from a technology point of view the departments are not that independent and centralization should be possible.

4.3 Identified problems

As mentioned before, a redesign of the business processes will occur prior to the physical modification of the workspace. Several problems have been identified that lead to the application of this strategy.

Through measurements over time the utilization rate of all examination rooms was analyzed and it was found out that even at peak moments the maximum capacity of the rooms did not exceed 70%. Therefore, it was concluded that improvement opportunities to better use the available space exist. However, to accomplish that different departments would have to share examination rooms to be able to divide the load. Consecutive reasoning resulted in the idea to use general examination rooms that could be used by different types of specialists for different types of patients. To examine the implications of implementing that way of operating on the required number of rooms and the physical layout of the to-be build location, the renovation was postponed and the project team as described before was introduced.

More research in what the process would look like with a patient-centered approach resulted in the conclusion that, on top of generalizing examination rooms, patients would also benefit from having a central registration desk and waiting area. The results show that it would be more convenient for the patient resulting in a higher patient satisfaction which is the goal with the highest priority. However, that means a more drastic redesign of the process is necessary. Therefore, the goal of the project was to design a standard process to be able to merge parts of the different OPD processes into one process without constraining specialists too much and to increase patient satisfaction.

4.4 Initial situation

The project team had already initialized the redesign of the process before the case study was performed. Therefore, this section presents the observed situation at the start of the case study.

Through analysis and workshops, in collaboration with employees from as many different departments and levels of expertise as possible, a reference process model has been defined. It is agreed upon by this group of people that all outpatient departments could operate according to that reference process. The process is shown in figure 15, to verify the outcomes of the workshops, the execution of the tasks in practice was measured over about 100 patient consultations. Furthermore, during the brainstorm sessions the different high level aspects were discussed to make them more concrete. As a result, a more concrete collection of informal task descriptions was created.

Figure 15: High level process overview

For every aspect of the desired process a pilot was introduced that focusses on designing a solution for implementing the desired solution. Examples of these pilots are a pilot that focusses on online appointment scheduling, another that focusses on what the general examination rooms should look like and one that focusses on how patients can register at the hospital and where they are staying while waiting.

This case study was performed at the pilot called 'task substitution'. The goal of the pilot is to redesign the process of the actual consultation of the patient with a physician as highlighted in figure 15. The redesign should make sure the tasks that do not necessarily require to be performed by a physician are taken over by trained employees. The idea is that if that is accomplished physicians will have more time to focus on the tasks that are actually related to their specialism, and hence, patient care would improve. The case study had as goal to design a solution for supporting that new process with an information system.

It was decided to design the prototype for the rheumatology department only to limit the number of activities that have to be implemented within the case study. The main practical reason for choosing the rheumatology department lies in the fact that the department has multiple physicians participating in the project, and hence, the configuration will be based on the views of more than one person. Furthermore, because the physicians are willing to participate in the experiment probably means they have given the concept more thought, and hence, might have more clear views on how they would want to configure their process as compared to physician who have no relation with the project at all.

4.4.1 Results for the design phase

As highlighted before, the case study focused on 'task substition' and, hence, on the consultation/treatment phase of the high level process as presented in figure 15. Therefore, the design phase focused on modeling the process from the point the preparations for a patient appointment start until the appointment and corresponding administrative tasks have been finished. Based on the knowledge acquired through the workshops a model was created that identifies six main phases in the process. Furthermore, the tasks corresponding to each phase are visualized as shown in figure 16.

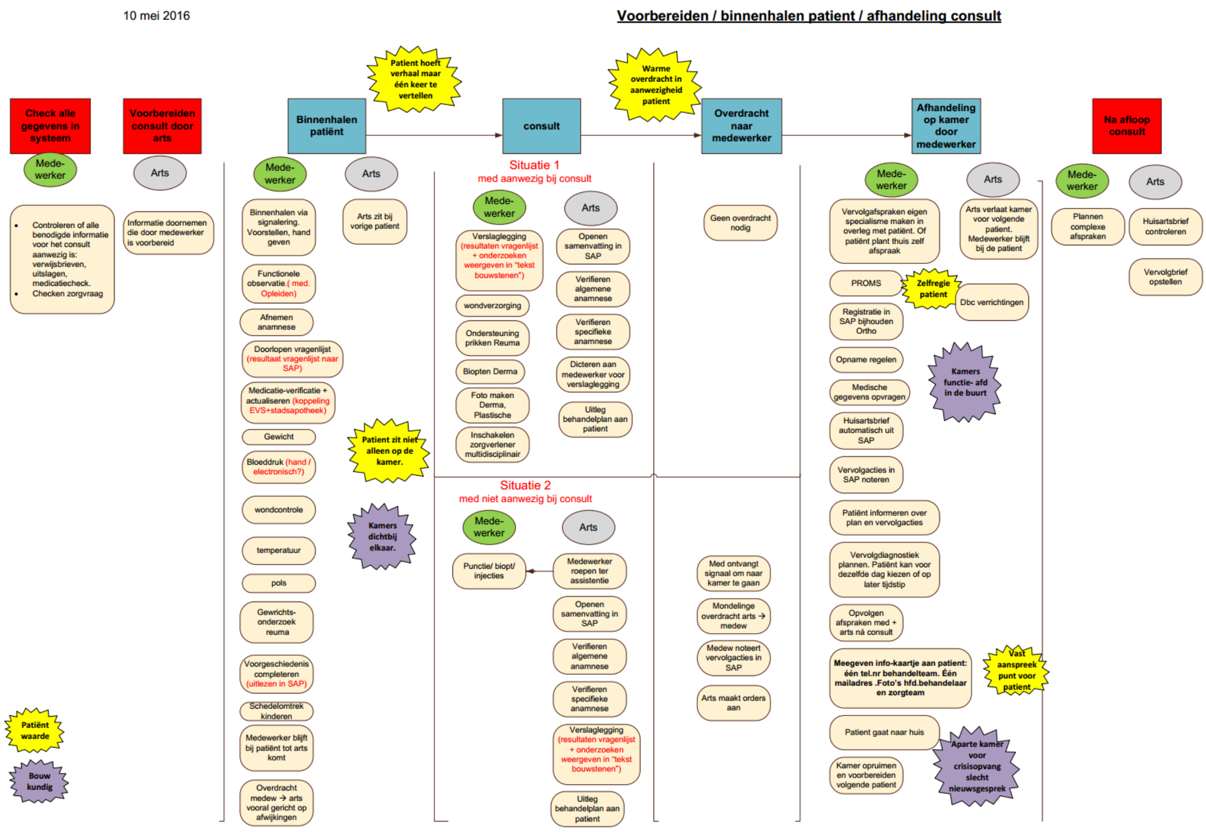


Figure 16: High level overview of current progress

The first and last column consider tasks that are related to the consultation, but are performed at an undecided before or after the consult without presence of the patient. Hence, these tasks are not included to be supported by the workflow management system. Furthermore, the tasks that were identified as irrelevant for the rheumatology department were filtered out based on the opinion of the rheumatology physicians. For example, Pulse and body temperature measurements are rarely performed at the rheumatology OPD, because they have no predicting value for rheumatology related problems. Furthermore, questionnaires do not need to be discussed, because they are integrated in a national platform that automatically processes them and provided physicians with relevant insights.

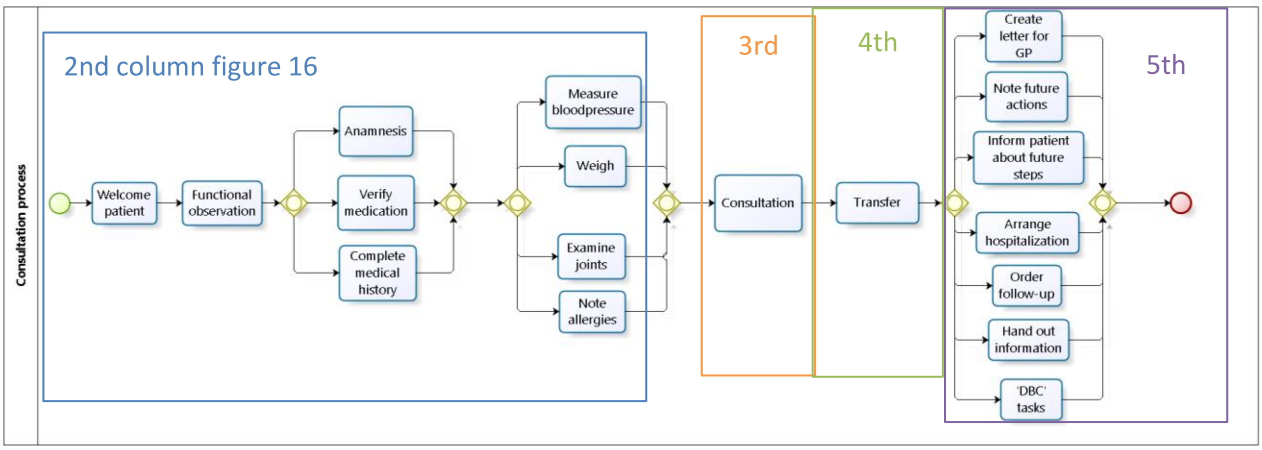


Figure 17: Reference process model

The remaining tasks were modelled in a BPMN model to function as the reference model as shown in figure 17. The BPMN language (Chinosi and Trombetta, 2012) was chosen, because it is a broadly accepted modeling language and supported by many workflow engines. But also for practical reasons, because personal experience had already been acquired with the language.

The first step when the patient arrives for his or her appointment is to welcome the patient and to move to an available examination room. The consult preparation can be roughly divided in an introductory phase, in which some basic informaion is checked and an execution phase in which some tasks are executed. During the introductory phase, the main state of the patient is acquired by a small conversation (anamnesis) with the patient. The anamnesis consists of a number of questions about the well being of the patient, and is also used to set the mood and make the patient feel comfortable. After the anamnesis, the patient information is checked and possibly altered. Especially important are to perform a medication verification, which ensures there is an up to date overview of the medication that is taken by the patient. Furthermore, it is important to have a complete overview of the medical history of a patient who arrives as a new patient in the hospital.

The second phase consists of straight forward tasks that result in valuable information the physician can use during the consultation. It includes basic checks such as a weight or a pulse rate measurement, but it also consists of more complex tasks such as a joint examination or wound care. After all these tasks have been completed, all the necessary information is required for the physician to perform his consultation.

What happens during the consultation is very patient specific and, hence, not modelled explicitly. However, in general it includes an interpretation of test results and the information acquired in the preparation. Based on the interpretation the physician determines the future treatment or decides to further examine the patient to retrieve some more information before deciding. The future steps are than explained to the patient and registered in the information system.

After the consultation is completed, a transfer takes place between the physician and the assistant. The transfer includes a small summary of consultation and some instructions for the assistant on what needs to be completed before the patient leaves. The transfer can either be in spoken form or it can be communicated digitally.

Based on the transfer information, the assistant concludes the consultation by providing more information to the patient and performing administrative tasks. An example of a form of information provision is an assistant that shows the patient how to use a digital platform where test results can be found and answer questions about the future steps. Arranging future appointments, hospitalization and DBC registrations (necessary for insurers) are examples of administrative tasks that need to be performed.

4.5 Scope

The goal of the case study was twofold. First, from a practical point of view the goal was to see if applying the SPExFlex approach would result in a system that properly supports the process and that provide added value to the employees. The second goal was to evaluate the SPExFlex approach by conducting its phases and to experience what purpose.

The scope of the case study with relation to the approach is shown in figure 18. Executing all phases of the approach completely would be a time consuming task which would not have been realistic to complete within the limited time span of the case study. Therefore, the case study was scoped. The goal was to fully evaluate the proposed approach, and therefore it was decided not to skip any phases of the approach. However, for different reasons it was necessary to perform a limited version of some of the phases as explained below.

As explained in the previous sections, the design phase was as good as completed when the case study started. Therefore, it was expected that only minor effort was required to generate a suitable reference model and, hence, the design phase was largely omitted. However, the process definition was not yet defined and it was, therefore, the starting point for this case study. The variability modeling will be performed and the goal is to perform this phase as it is prescribed by the approach. A configuration technique will be chosen and used to create the configurable process definition.

Especially the implementation phase was limited in the case study. An actual implementation as prescribed by the approach would require integration of a new system in the existing IT architecture and extensive research in to the functionality existent in the current information systems in use in the MUMC. Furthermore, implementing the system in practice would mean sensitive and important data is processed meaning errors could have serious implications. Because it would extensively exceed the scope of the research to deal with these issues at this point, it was decided to implement a prototype of the workflow management system using the SPExFlex approach and describe how it would fit in with the organization in a theoretical manner. The prototype will exist of an implementation of the configured process model and execution of the model. Information required from and provided to other systems is simulated and importing the model in to the workflow management system would be done manually, hence no integration step would be performed.

To test the configuration abilities of the model it was decided to pick a single department and configure the model according to their process. For practical reasons of not having to implement an abstraction tool, it was decided that abstraction from the model would be accomplished by configuring the model by interviewing the associated physicians which was possible due to the small scale of the configuration. Assumed is that this would yield similar results as the abstraction techniques explained in section 3.2.2.

Execution of the model would be simulated by acting out some test cases with selected employees who would work with the system in real life. Their acceptance of the implementation was evaluated according to the Technology Acceptance Model (TAM) (Moody, 2003).

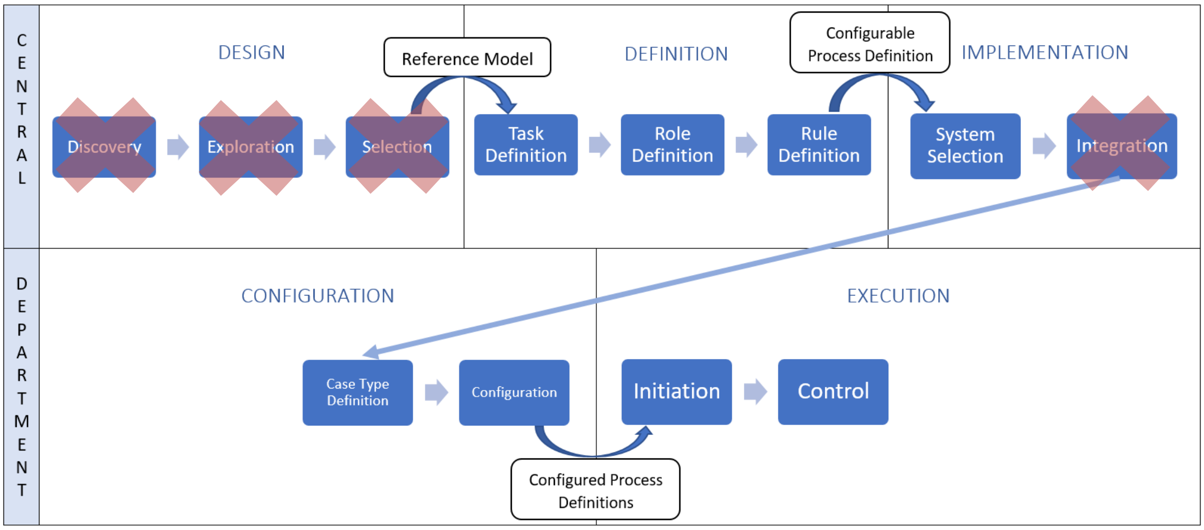


Figure 18: Scope of the case study

5. Case Study: Results

The results of the case study are presented in this chapter. For each phase of the tested approach, the steps performed are explained before showing the results for that phase. As the design phase was already completed, the first section focus on the define phase.

5.1 Define

Executing the process definition phase resulted in three main artefacts. The first is a data model that defines what information is involved with each task. Second, it is defined what actors are involved with the system and who should perform what task. Finally, the reference model is extended with system tasks and roles to end up with a complete process definition.

5.1.1 Data model

The first step in extending the reference model to a process definition was to formalize the tasks based on a data model. As a healthcare organization, the MUMC is bound to strong regulations on what information can and what information must be recorded for certain treatments or tasks. On top of that, the hospital has its own in house agreements on what information is required for certain tasks. Therefore, it is important to standardize task registration as much as possible and, hence, reduce the amount of flexibility allowed for user input. In practice, that means open text fields are undesirable and most information should be recorded by registering pre-defined choices.

The easiest tool to achieve that, is by applying a data standard. To ease communication between healthcare institutions the national institute for information technology in healthcare (NICTIZ) has defined such a data standard in the form of 'zorginformatiebouwstenen' (ZIBs) (Shackleton-dijkstra, 2017). A ZIB, which translates to 'healthcare data building block', exactly defines what data needs to be recorded for a certain medical task and in what form. As it turns out, the MUMC has the intention to apply the standard as much as possible, but in practice it is rarely used, because the systems at the

OPDs mainly rely upon free text inputs. Therefore, the data models of the tasks were made from scratch by combining the informal task descriptions and the corresponding ZIBs.

An example, of the ZIB that defines the registration of a body weight measurement (the 'weigh' task in figure 17), is shown in figure 19. As the figure shows, every data element has an associated type, id, conceptname in both Dutch and English, cardinality, definition, definitioncode and reference attribute. The type attribute refers to the data type in which the attribute is stored, such as integer, Boolean or timestamp. Furthermore, the definition elaborates on the meaning of the conceptname in case the conceptname itself is not clear enough on what is meant. The definition code refers to specific medical equipment that is referred to in the definition and that is used in a different standard. The final column houses references to other ZIBs. Sometimes, ZIBs are reused in multiple other ZIBs. For example, the ZIB that defines the recording of a pain score is applicable in multiple situations.

Type	Id	Concept	Alias	Card.	Definitie	DefinitieCode	Verwijzing
	NL-CM:12.1.1	Lichaamsgewicht	EN: BodyWeight	1	Rootconcept van de bouwsteen Lichaamsgewicht. Dit rootconcept bevat alle gegevenselementen van de bouwsteen Lichaamsgewicht.		
PQ	NL-CM:12.1.2	>GewichtWaarde	EN: WeightValue	1	Het lichaamsgewicht van de patiënt. Het gewicht wordt uitgedrukt in kilogram (kg). Bij pasgeborenen onder 3 kg wordt het gewicht veelal in grammen uitgedrukt. Dit concept kan ook gebruikt worden om een geschat lichaamsgewicht vast te leggen als het niet mogelijk is om het exacte lichaamsgewicht te meten - bijvoorbeeld, het wegen van een tegenwerkend kind, of een schatting van het gewicht van een ongeboren kind.	27113001 Body weight	
ST	NL-CM:12.1.3	>Toelichting	EN: Explanation	0..1	Opmerking over de gewichtsmeting, zoals eventuele problemen of factoren die van invloed kunnen zijn op de meting van het lichaamsgewicht; bijvoorbeeld timing in de menstruele cyclus, timing van de recente stoelgang of het noteren van een amputatie.		
TS	NL-CM:12.1.4	>GewichtDatumTijd	EN: WeightDateTime	1	Datum en (eventueel) tijd dat het gewicht gemeten of geschat werd.		
CD	NL-CM:12.1.5	>Kleding	EN: Clothing	0..1	De kleding die de patiënt aan had tijdens de meting.		KledingCodelijst

Figure 19: Example ZIB

The ZIBs were used as reference for the creation of the data model as shown in appendix 2. Each task has been given an unique name and for each task it is specified what information is required to execute the task and where the information should come from. Furthermore, for each task the information that should be registered is highlighted and how/where that information should be stored. For the tasks for which no corresponding ZIB exists a data model is defined in a similar style as the ZIBs. Figure 20 shows the resulting data model for the bodyweight example.

Task	ZIB	Data in	Data from?
Weight	BodyWeight	-	-

Data out	ZIB	Data type	Cardinality
WeightValue	BodyWeight	Physical Quantity	1
Explanation	BodyWeight	String	0..1
WeightDateTime	BodyWeight	Timestamp	1
Clothing	BodyWeight	Coded description	0..1

Figure 20: Data model for bodyweight

5.1.2 Role specification

Until now, two main types have been introduced when mentioning actor roles: the physician and the assistant. The physician is a clear notion and won't differ from the role of physician in the current system, except from the fact that the tasks performed by the physician change. However, the 'assistant' is a broader term that requires more specification. It is a role that does not exist in the as-is situation and it is unclear what qualifications are required for a person to be able to function as an 'assistant'.

The main idea is that the assistant will be responsible for acquiring patient information and that the physician's main task will be to interpret information. Therefore, in general the assistant should be capable of performing the medical tasks and register them in the system and the assistant does not require the medical knowledge to necessarily know what the information means. However, when analyzing that behavior several problems were identified that make the role more complex. First of all, in order to acquire knowledge from the patient that might contain sensitive information requires good communicative skills. On top of that, multiple situations were thought of in which the assistant would require more medical knowledge in order to properly to the job. For example, medication verification might be a difficult task when lacking knowledge. An illustrative example is that a patient might say something like 'I take that bright pink pill every day'. A physician who prescribed the medication will directly know what medication the patient is talking about, and this might be harder for an assistant making it more difficult to complete the job.

At the time of the case study there was no clear consensus on what capacities the assistant should possess and more important it was not clear whether the assistant would be one single actor role or if it might have to be split in to multiple roles. For example, an idea would be to have multiples types of assistants with an increasing level of medical knowledge and assign tasks to the assistant with the proper level. In that case, the medication verification task could, for example, be assigned to an assistant with more medical knowledge as compared to a simpler task such as measuring blood pressure.

It was outside of the scope of this research to fully investigate this topic and define the different roles for an assistant. At the same time this research was performed a professional coach with extensive knowledge on human competencies did an investigation on this aspect, so the results will later be used to decide on this issue. Therefore, for the case study some assumptions were made for the implementation of the assistant. It was decided to keep the assistant as one single role which could perform all consultation preparation as well as conclusion tasks. However, it was kept in mind that in the future that might change.

Another aspect which was identified when analyzing the differences between the current and the future roles is the role of the registration desk employee. Currently, this role is not only responsible for registering patients and making appointments, but also for keeping track of the planning and for general support. Those tasks need to be taken in the new situation, because the registration desk employee is going to disappear. Part of the tasks will be taken over by the main registration desk that will handle registration of all patients for all OPDs and part of the tasks will be taken over by the assistant and performed as part of the consultation preparation. However, it was identified that some of the tasks were still missing and that an extra role was required in order to fill that whole. The new role was named 'problem manager', because it is a person who will keep an overview of the procedures and step in if problems occur. Currently, for example, if a patient was waiting in a room for too long without anybody seeing him the registration desk would notice, because he/she is responsible for a small number of patients so it would stand out. In the new situation, however, that might not be the case, and it is important that this 'manager' would than be alerted to find out what the problem is. Another problem that might occur is that the person assigned to perform a certain task cannot perform the task for whatever reason, and the system is not capable of finding a solution and the manager might have to solve the problems manually.

To conclude, three main roles were identified: the physician, the assistant and the manager. The physician does only perform the consultation. The assistant performs all preparation and conclusion tasks, although the person performing the preparation might be a different person from the person performing the conclusion tasks. And a manager keeps an overview of the entire process and is alerted in case of problems.

5.1.3 Rule definition

Based on the analysis presented in the previous sections the reference model was extended to function as a process definition. The roles were added as well as the system tasks that are needed for process execution. For the purpose of simplicity, the preparation and conclusion tasks have been hidden in a sub process. The final step is to define the additional rules that are needed for execution. For clarity, the rules are explained according to the final process definition as shown in figure 21.

As the model shows, a case is initiated by the system. The idea is that this task is triggered when a patient is registered as being present in the hospital. Initialization consists of retrieving the specific patient variables from the ERP system and retrieving the corresponding configuration from the repository. If the system cannot retrieve the correct configuration, for example, because the patient information is not complete and key information is missing, the manager can manually solve the problem. For example, he might do this by providing the extra information needed by the system to retrieve the configuration.

The first task is to welcome the patient and the resources required for that task are an assistant, an examination room, a physician and possible extra equipment that is not included in the standard equipment of the rooms. The physician is not directly assigned to the task, but a check is performed whether the physician will be available at the time when the preparation is finished, based on expected task durations. A timer is included, because at this point the patient is waiting in the waiting room and it is not desirable to keep the patient waiting for too long. Therefore, if assignment of the resources takes too long, the manager can manually assign the resources, for example by slightly changing the requirements for the required equipment, because a certain type of equipment is not available in that moment.

During the preparation/consultation tasks performed by the assistant, it might happen that the assistant cannot finish the tasks and a different employee needs to take over. Examples of such situations can be when an employee gets sick or immediate assistance is required at another location. The same counts for a physician who might have to leave in the middle of a consultation, for example to assist at the ER. Therefore, escapes for these kinds of situations have been modelled so that these problems are solved, either by the system itself or by interference of the manager.

As can be noticed when comparing the reference model and the process definition, the process definition is significantly more complex as the reference model. The primary reason for that is that in the design phase already performed by the hospital the logistic aspects of workflow management and the tasks that need to be performed in the background were not taken into account. Therefore, many extra tasks had to be added, such as the tasks performed by the problem manager.

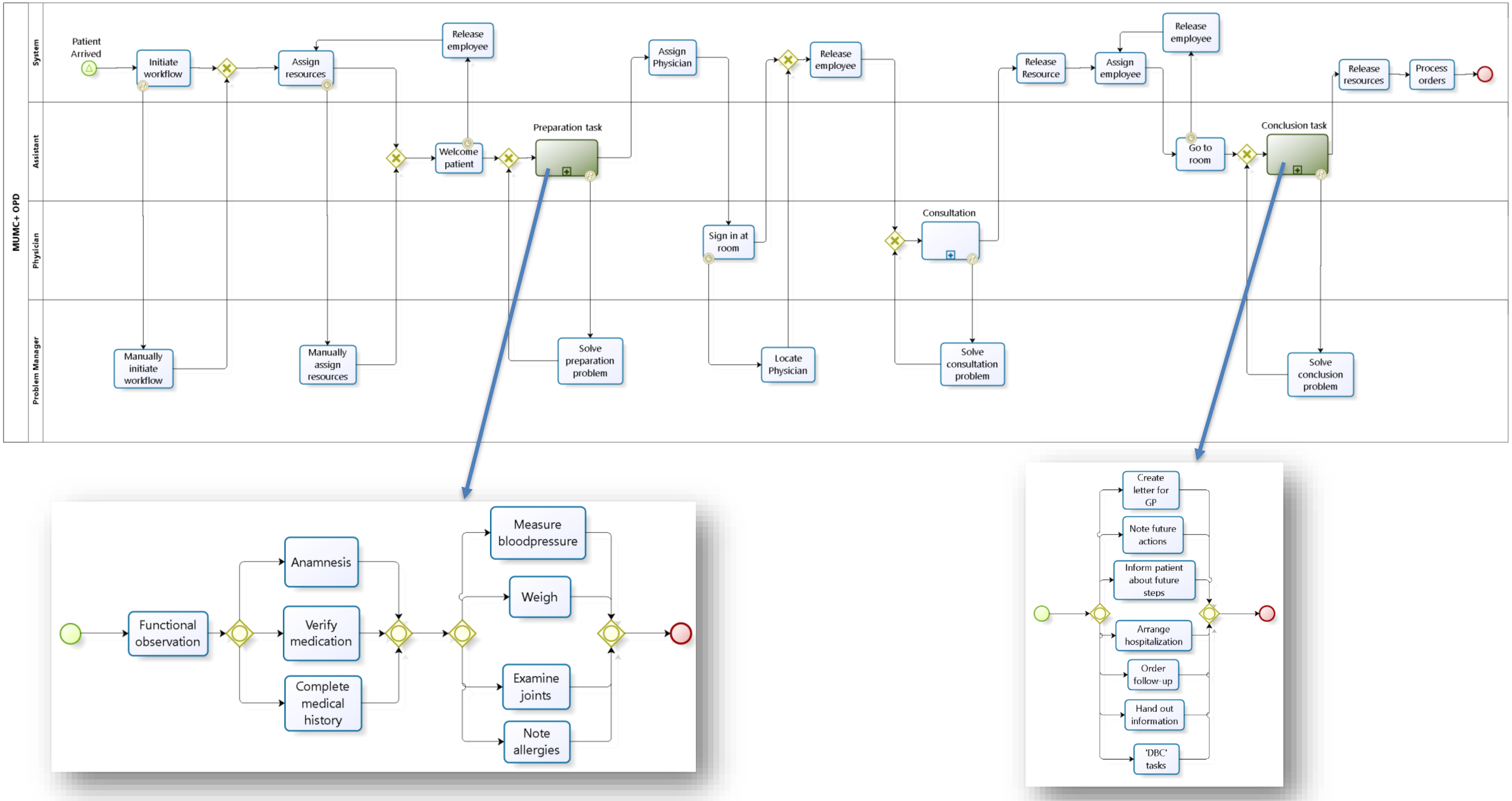


Figure 21: Process Definition

5.1.4 Variability Modeling

The final step in completing the definition stage was to model the variation points in the model to create the configurable process definition. As explained before, variability modeling refers to the process of highlighting what tasks are mandatory, what tasks can be deleted from the model, and what tasks are hidden. As also explained before, addition of tasks is not considered as an option for this approach.

First, a modeling technique was chosen to model the variabilities in the model. The configurable workflow technique (Gottschalk *et al.*, 2008) was chosen, because of its focus on configuring workflow models and the assumption that that would ease the import phase of the model in the workflow management system. Especially, since a formal method is provided for translating the configured model to a workflow model. Furthermore, it supports all three aspects of allowing, blocking and hiding tasks and it is proven to be working by its implementation in the form of the C-YAWL language.

The variation points are highlighted in the model by using symbols on the tasks. Green arrows are used to indicate tasks that remain in the model, yellow arrows depict tasks that are hidden and the red signs indicate blocked tasks. The difference between hide and block is that for hidden tasks the path can still be followed, but the task is not executed. For blocked tasks the path can also not be taken.

The configurable model is shown in figure 22. For each task, the allowed configurations are shown. As the model shows most variations can be found in the preparation and conclusion tasks subprocesses. The only tasks that are mandatory to be executed in these subprocesses, are ‘welcome patient’, ‘anamnesis’, ‘verify medication’, ‘consultation’, ‘transfer’ and ‘inform patient about future actions’.

For the ‘welcome patient’ task it is not only the case that the patient will always be called in for an appointment, but also another important security measure is assured by making this a mandatory task. An identity check is incorporated in this task to make sure the right patient is being treated and no mistakes can be made. The check is performed by comparing the person with a photo and by asking the person about his or her birth date. The anamnesis is mandatory, because is not only an effective way of gathering general information about the patient’s complaints. It also ensures hospitality by engaging in some real conversation instead of just completing a list of tasks, which might be perceived as impersonal.

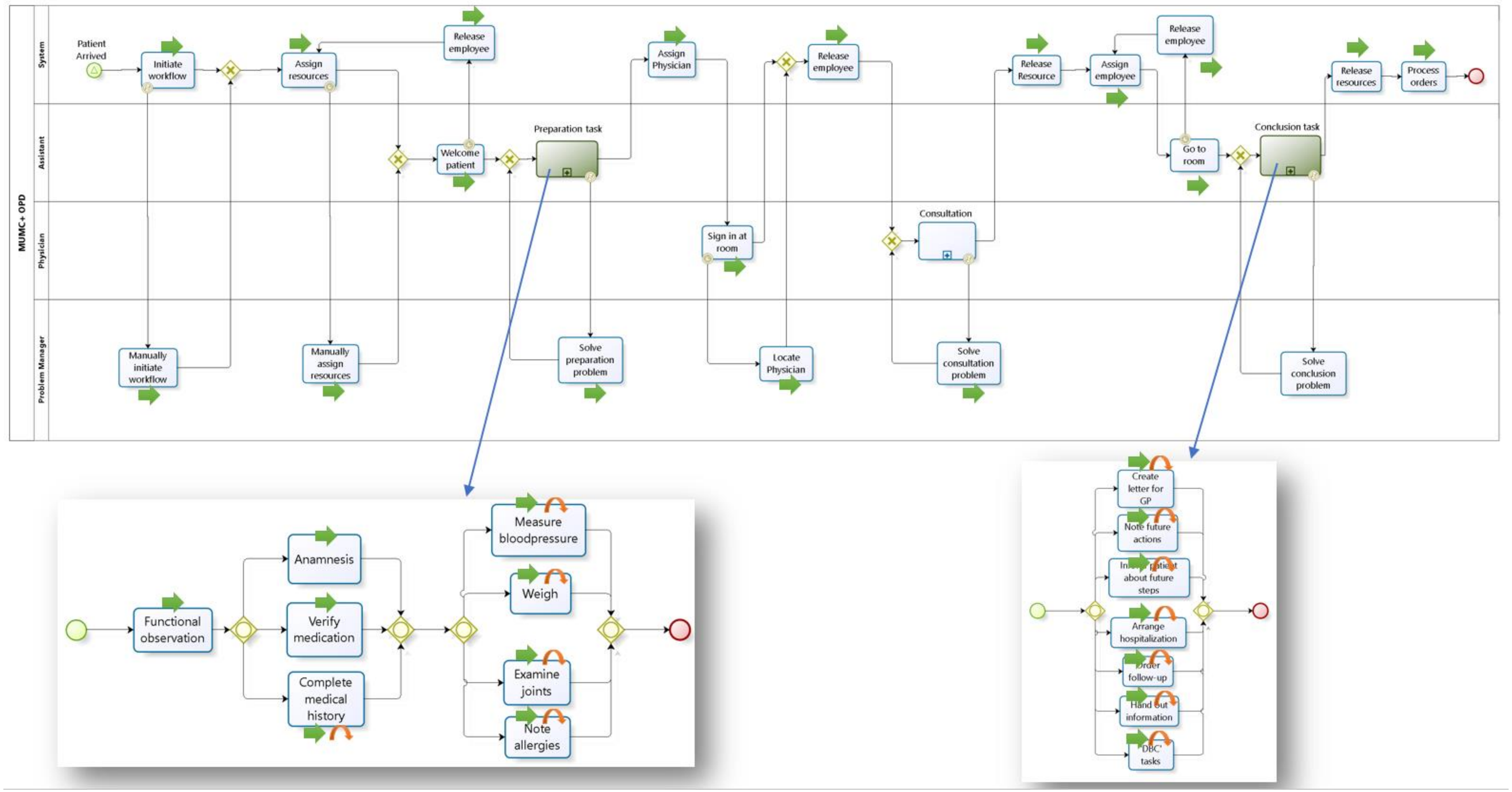


Figure 22: The configurable process definition

Another task that is mandatory due to guidelines that need to be adhered, is the medication verification. It is mandatory to check medication use of patients for each consultation. As it turns out, that is a complex and time consuming task to perform, and therefore it is worth it to investigate how and when this task can best be performed during the process. Furthermore, it is unclear how the medication verification should be incorporated in the new process and who should perform it. More information on how that is handled for the prototype will be provided in section 5.5.

The final two mandatory tasks are mandatory for more obvious reasons. They include the consultation itself, which is the center of the process and the transfer between physician and assistant. For the preparation and consultation tasks, high variability is allowed, because it is highly dependent on the department and the disease what tasks are important and what tasks can be skipped. The hospital trusts her medical specialists to assess necessity of tasks for their patients, and therefore clinical guidelines are not included for these tasks.

The reason no configuration options can be observed outside of the subprocesses, is because these tasks are mostly background tasks that form the main body of the process and hence cannot be removed.

5.2 Implementation

Based on the process definition the implementation phase could be started. As highlighted before the system could not be integrated within the existing infrastructure. Therefore, this section explains the system selection step in which a workflow management system is selected and the process definition is deployed on its workflow engine.

An analysis of the functionalities the workflow management system should possess to be able to function as the prototype was performed. When looking at the process definition, it can be observed that it is not highly complex and, therefore, complicated model support is not required. When looking at the execution, the most important feature is that flexible task assignment is possible. It should be able to assign tasks to pools of people and, also, for some tasks assignment of a task is based on assignment of different tasks. Furthermore, although not crucial, it would be convenient for this case if the system would support the BPMN language for its modeler, so a translation of the model to a different language is not required. Finally, the system should be an open source system, because no means were available to invest in a commercial tool.

Based on these features the Camunda system was selected for the implementation. Using the 'Camunda Modeler', the configured model was modelled in the Camunda language. The resulting model is shown in figure 23. The model will be evaluated with people who are not used to the English language, therefore the process model has been (partly) translated to Dutch.

The Camunda modeler does not use swim lanes to assign roles to tasks, hence, the model has a different visual representation from the original process definition. The way it works, is that tasks in the Camunda modeler have associated attributes that can be set to guide execution of the tasks.

A HTML file is associated to each user task that is used to define the task form the user can use to complete the task. Furthermore, java classes can be associated to add functionality to the execution engine that is not included in the standard implementation. Finally, for each task a task type is associated. These task types define how the task is executed, for example by a user or automatically.

The goal for the task forms was to create forms that have a simple, but clear lay out and that adhere to the data model as defined before. The task form for the 'weigh' task (BMI stands for Body Mass Index) is shown in figure 24 as an example. As the figure shows, most information that needs to be registered is encapsulated in small fields with a specific purpose or in pre-defined choices to ensure data quality. The only free text area still available is a general comments section, that can be used to note unusual encounters. Not noticeable from the screenshot, but also included in the task forms, is the aspect that some fields are mandatory to be filled in before a task can be completed. In that way the cardinalities as defined in the data model are implemented. The forms for the other user tasks have a similar structure.

The java files are used for simple purposes, such as to hide certain pieces of forms in certain contexts to keep forms simple, but also for more complex purposes. It is, for example, important that all preparation tasks are performed by the same assistant that performed the 'welcome patient' task. Therefore, java code is used that checks who claims the first tasks and then automatically assigns that person to the other preparation tasks. The same principle is applied to the conclusion tasks. Finally, java code is used to throw and catch errors. These are necessary to detect possible problems such as the assistant having to leave in the middle of the preparation. The way that is implemented, is by adding a checkbox for that in the task form and have the java class detect it when the box is checked and throw an error for the execution engine. In the prototype, system tasks are often not implemented to do the task they claim to do, because that would often require links with other systems. Therefore, the results of the system tasks are often simulated, but the tasks are included for clarity and to show what the complete implementation would require. An example of such a task is the assign resources task which is currently not really doing anything, because no system is in place that keeps track of room and resource availability. Therefore, for the prototype task assignment is hard coded and not performed during process execution.

The task types are shown in the model by the symbols in the top left corner of the tasks. The only two types used for this model are 'human task' and 'scripting task'. A human task is a task that includes a task form that needs to be completed by an user in order for the task to be completed, and is noted by a small symbol of a torso. The scripting task refers to an automatic task that is executed by the system and is noted with the symbol including the two small gears.

As the model shows some of the conclusion tasks were not included in the implementation, although they were part of the process definition. Unfortunately, implementation of the tasks could not happen, because it turned out to be too difficult to implement a standardized task for these tasks. The excluded tasks consisted of complex administrative tasks that are not easy to capture in a single task. A more thorough analysis of what these tasks should look like and whether they should be part of this process is required. Furthermore, the 'examine joints' task was not implemented, because in hindsight this task was not suitable for substitution and, hence, was included in the consultation. Of course this changes are not desirable and raise some questions, especially about the design phase. A more detailed discussion on these experienced issues will be provided in chapter 6.

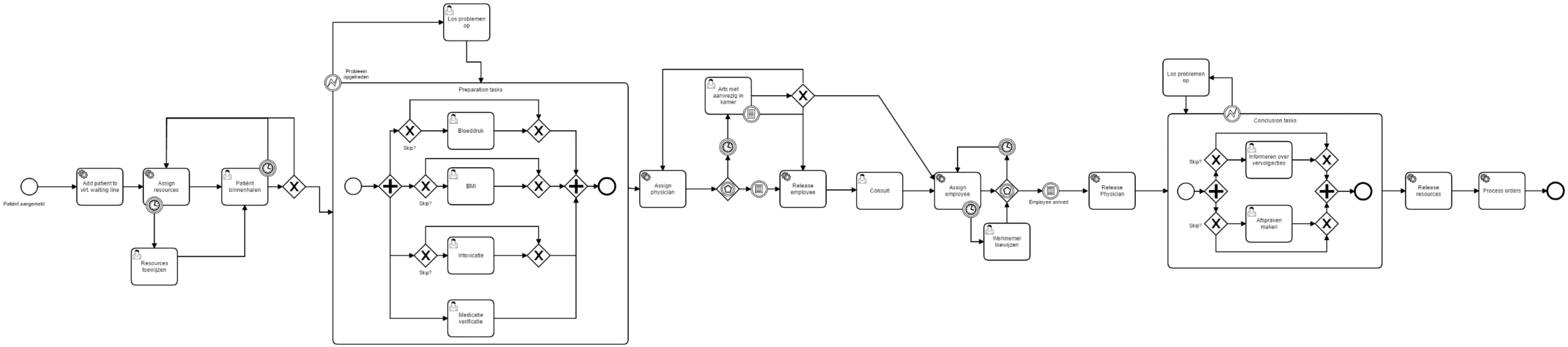


Figure 23: Implementation of process definition in the Camunda modeler

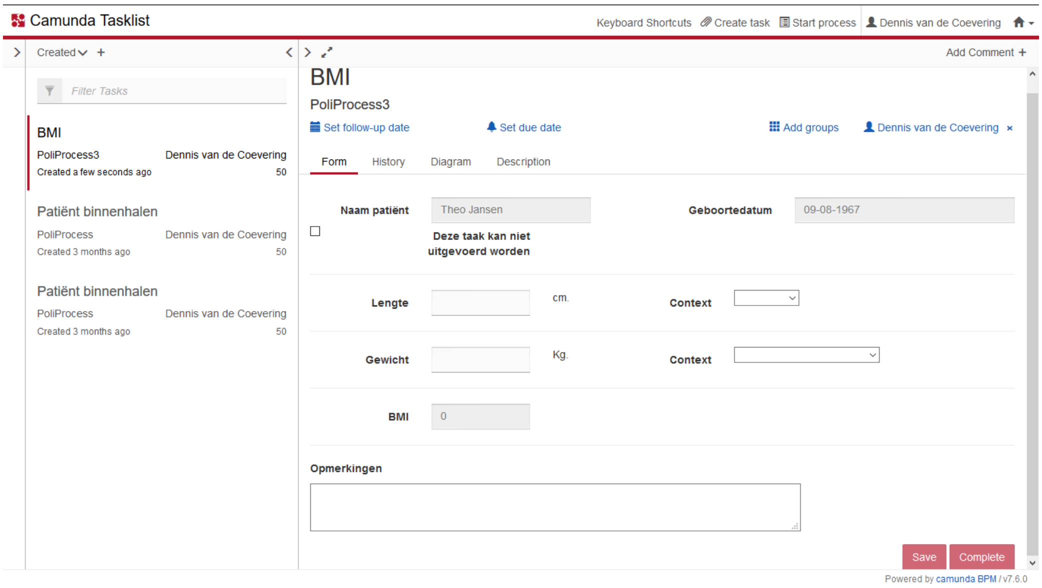


Figure 24: Example of a task implementation for the 'weigh' task

In the sub processes the flexibility by design constructs can be observed that allow for skipping of activities if the configuration dictates the activities should be skipped. The conditions on the XOR choices, that check the certain data variables that are set based on the configuration, determine whether the task is skipped or not.

5.3 Configuration for the rheumatology department

With the implementation completed, the configuration phase could be performed. To test working with the new roles in practice and to measure its performance as a comparison to the current process an experiment was set up. That experiment included physicians executing their consultations according to the desired process. In order to set up the experiment the physicians were already interviewed about their ideas for tasks that they would prefer being substituted by assistants. Therefore, it made no sense to conduct similar interviews again, and to use the results obtained.

To recall from the configurable process definition, only configuration options were found in the sub processes containing the preparation and conclusion tasks. For clarity, only those subprocesses are shown in this chapter to be able to clearly show the configurations. Figure 25 shows these parts of the model. As the figure shows, the tasks that could not be implemented are now marked as hidden automatically to resolve that issue.

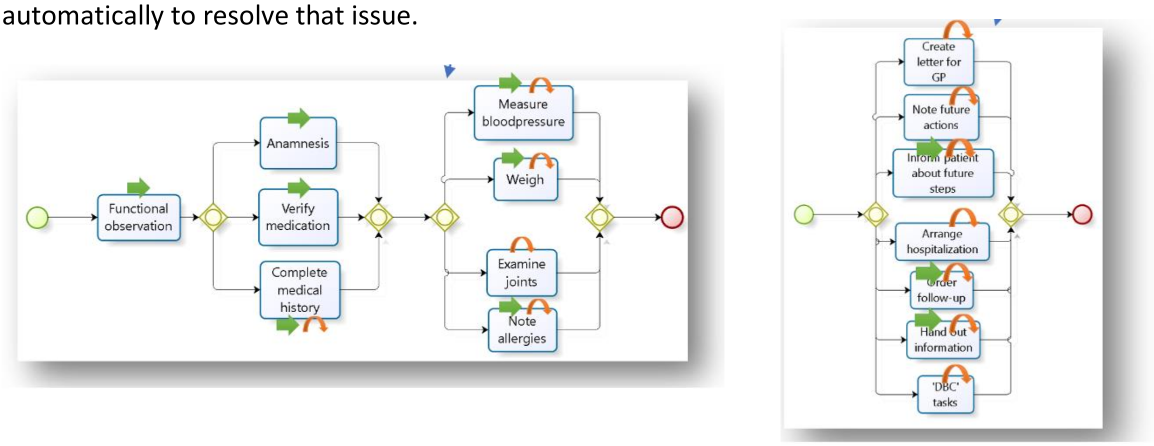


Figure 25: Parts of the process definition with configurable elements

For the case study, two case types were defined based on whether the patient is a new or a returning patient. For a return patient less information needs to be gathered, so especially changes can be made to the preparation tasks. To distinguish these two case types only a single variable can be set that shows whether the patient has had a previous appointment or not. To apply the configuration, for each variation point this variable can be checked and based on its value an activity can be skipped or not.

It turned out that for new patients, the configuration wanted every task to be completed, because the physicians want to have as much information about the patient as possible to define a proper diagnosis. For the returning patients, some of the tasks could be skipped as shown by figure 26. As the figure shows, the 'note allergies' and 'complete medical history' tasks were configured to be skipped.

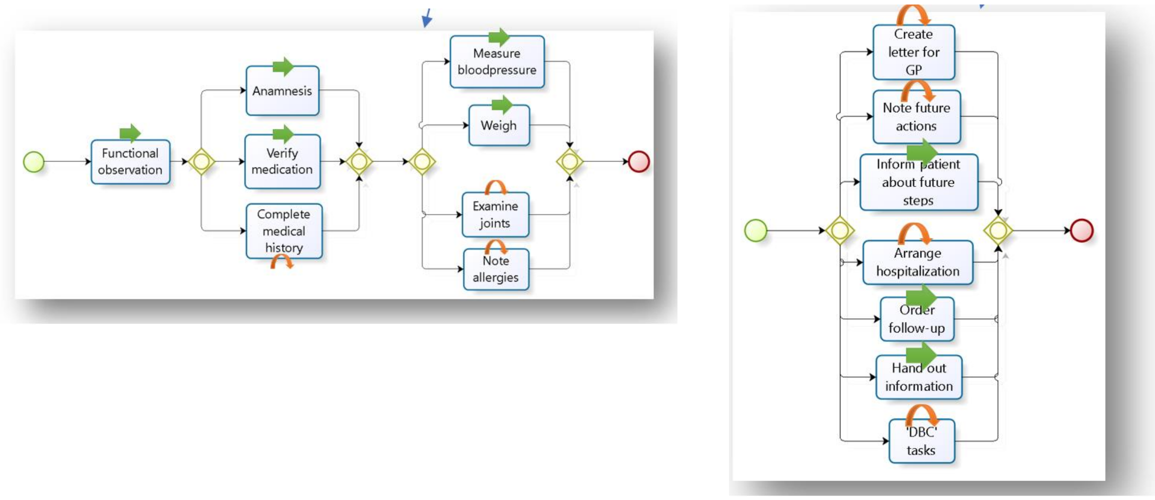


Figure 26: Process model for rheumatology department according to configurations

5.4 Execution

To get an idea of the perceived usefulness of the implementation as small scale evaluation was performed by simulating execution of the process. The involved participants will be involved in the future process in different roles. Two cases were written that depict a realistic scenario that the system might have to deal with. The first scenario was a scenario in which every task is executed smoothly, whereas in the second scenario the system had to deal with some unexpected events.

The participants were given the scenarios and were asked to register the information as it was provided in the case using the prototype. In this way it could be observed how well the system speaks for itself and whether tasks are executed smoothly. Furthermore, the perceived usefulness of the prototype was tested using the Technology Acceptance Model (TAM) (Moody, 2003). By using that model, a good insight could be gained in both the expected learning curve and the perceived usefulness of the prototype. On top of the standard question as proposed by moody (Moody, 2003), two extra question were added. The first question asked about positive and negative aspects of the prototype as experienced by the participant to gain a better understanding in the reasons behind the answers given in the survey. The second question asked about foreseen problems in the daily routines when the transition would be made from the current system to the proposed system. Both The scenarios and questionnaire used for the evaluation are added in appendix 3. Most participants were not used to working with the English language, therefore both the cases as well as the questionnaire was translated in Dutch.

The pool of participants consisted of 3 assistants, 1 project managers and 1 physician. By assistant a person is meant that is a potential candidate for performing the preparation and/or conclusion tasks of the consults. Therefore, these people are the people that will actually work with the system. The physician was added, because although not directly working with the system, it needs to rely on the information that is acquired by the system and has to delegate tasks when the system will be used. Therefore, the daily routine of the physicians is significantly influenced by the system and hence it is useful to take their opinion and ideas into account. Project managers have the ability to look at the process from a more abstract overview and will have a better understanding of the implications of using the system in practice and, hence, might have different view on the usefulness of the system. The evaluation was conducted separately for each participant, so they had no influence on one another.

The results of the closed questionnaire section are presented in table 1. To clarify the results, the answers were translated to a plus or a minus to indicate a positive or a negative response to the question. The first thing that attracts attention, is that all participants were very likeminded and mostly positive. That means that, although the evaluation was performed on a small sample, still some general conclusions can be drawn based on this evaluation.

As described by Moody (Moody, 2003), questions 1, 4, 6, 9, 11 and 14 are designated to determine the perceived ease of use and questions 2,3,5,7,8,12,13 and 15 have as purpose to test perceived usefulness. Question 10 checks whether the participant has the intention to use the prototype herself. When looking at perceived ease of use the only negative answer was to question 14, which was about whether the participants believed the prototype could be used in practice in its current state. As expected, the project manager has a broader view on the implications of using the system in practice and, hence, still sees some serious problems that still need to be overcome to apply the system in practice. However, in general we can conclude that the participants find it easy to work with the prototype and that they would picture themselves working with the prototype in its current form. A similar conclusion can be drawn when looking at perceived usefulness. Beside the fact that they would be able to work with the prototype the participants also believe the prototype would provide added value if applied in practice. Question 2 shows the most doubt among the participants. That question is about whether it would reduce the amount of work done by the participants. However, apparently even if the amount of work would stay the same or increase, it would not outweigh the other advantages given the answers given to the other questions. Finally, all participants agreed that they would use the prototype to support them in their work if it would be available for them.

	Assistant 1	Physician	Project Manager	Assistant 2	Assistant 3
Q1	+	+	+	+	+
Q2	-	Undecided	+	+	+
Q3	+	+	+	+	+
Q4	+	+	+	+	+
Q5	+	+	+	+	-
Q6	+	+	+	+	+
Q7	+	+	+	+	+
Q8	+	+	+	+	+
Q9	+	+	+	+	+
Q10	+	+	+	+	+
Q11	+	+	+	+	Undecided
Q12	+	+	Undecided	+	+
Q13	+	+	+	+	+
Q14	+	+	-	+	+
Q15	+	+	+	+	+

Table 1: Questionnaire results (closed questions)

Especially, the perceived usefulness and the intention of use aspects give an indication of the success of the configuration. If the configuration of the model would have resulted in a selection of tasks that would not provide sufficient information for the physician to execute the consultation, it can be expected that the perceived usefulness would be lower. Furthermore, because the opinion of the physician was taken into account, and the physician obviously has to take into account if she can execute the consultation based on the provided information, a more certain conclusion can be drawn.

5.5 Recommendations

With the application in practise it has been shown that the SPExFlex approach can be applied successfully and that it can lead to an implementation with added value. However, as already explained, the implementation was a small scale prototype and there are still some challenges to overcome to get to a full scale implementation in the hospital. This section provides some recommendations, based on the experiences gained while executing the project, for how to get closer to a full implementation.

5.5.1 Information infrastructure

The most obvious aspect that has to be overcome is to integrate the system in the existing technical infrastructure. The prototype is a system which fully stands on its own and has no connections with other systems. Although, some of the links to current systems have been mapped, it was often not clear where information should come from. Furthermore, often not a single place could be pointed out where information was stored. For example, each physician documents her own consult in a free text area where it is not accessible for other people.

Therefore, one of the most important steps that should be undertaken is to introduce the information standard that was also used to define the tasks also in the data infrastructure. That means that every data element should get a dedicated location where it is stored, and that other locations where the data element is retrieved/altered all link towards a single location. The hospital is already trying to accomplish that, but more effort is required to actually establish this and create an information infrastructure that is suitable for introducing a standardized process.

5.5.2 Technological infrastructure

Another aspect can be found in the technological infrastructure. The current systems do not provide the functionalities needed to implement the proposed system. Therefore, the functionality should be provided by introducing new systems or by extending existing systems with the required functionality. The functionality will have to include a workflow management system including resource management, a process model repository and a tool for model configuration.

To see whether such systems exist and can be used in practise a similar implementation was observed in a different hospital. At the observed clinic they worked with mobile devices that were directly linked to the ERP system they used. Therefore, the physicians could note things on their mobile devices during consultations, that were directly stored in the ERP system. Limited workflow support was provided in this specific implementation, but it is a prove that the desired type of system exists and is already used in practise.

5.5.3 Organizational infrastructure

Apart from the information infrastructure, the organizational structure also requires revision. The right side of figure 27 briefly shows the steps of the approach and their relation to the artefacts produced. The left side shows the steps that are not described in the approach, but that will be important in order to sustain the standardization in the long term. The left side of the figures shows that feedback from the execution level should be able to propagate back up to the standardized models.

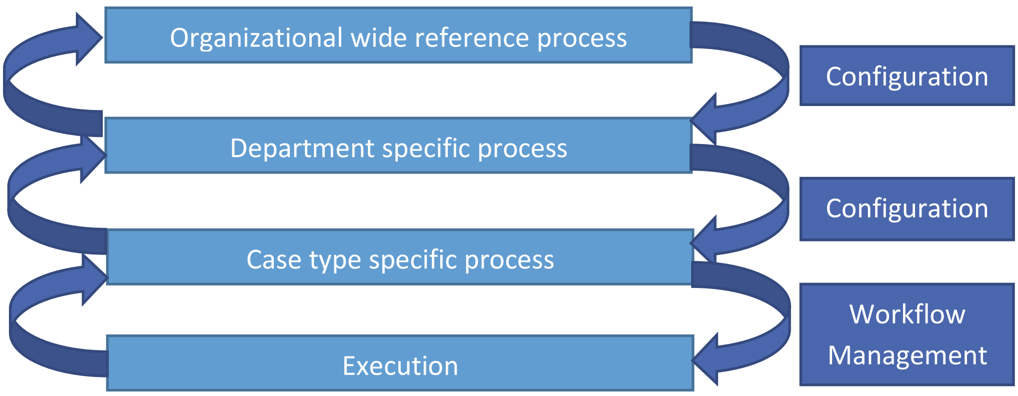


Figure 27: Schematic overview of approach

It is evident that a perfect model will not be created in a first attempt, and if it is achieved, changes will have to occur at some point. Therefore, it is important that these changes are allowed, but that they are performed in a structured matter. The main reason for that is that in order to successfully sustain a standard, strict management on changes to the standard is required. If changes are made too easily and too frequently the standardized model will contain too much specific behaviour, which might in the end lead to a model that is not a standard anymore, but a collection of a lot of specific requirements making it very hard to manage.

In the current structure neither a body for managing the standardized model nor a body for configuration of the processes exists. It is still the question what the structure will look like in the future situation, because currently every department has its own management. If that structure remains the same, the management of the department can be responsible for the first configuration phase. Furthermore, they could then be in charge of managing the set of configured models for their department. If the separated managements disappear, an option could be to form a 'committee' of current department heads and other stakeholders that together have to agree on the standardized model and that determine whether proposed changes to the model are approved. The physicians of a certain similar medical expertise can be in charge of determining the patient types for their domain and to configure the models accordingly. If their wishes require changes to the standard model they can request them at the aforementioned committee.

Besides the control structures, also the more low level organizational aspects need to be determined. Examples that have already been mentioned in previous sections are about employee roles and medication verification. While executing SPExFlex it was observed that decisions regarding these topics could not be made, because there was not enough knowledge present, or no agreement between different stakeholders. It is important to make clear decisions on these operational aspect in order to be able to properly design the process.

5.5.4 Culture

It was experienced that in the medical domain, although on paper the structure is hierarchical, in practise that does not hold. For most management decisions the hierarchy holds, but when it comes to the medical process the physicians have a big influence in the organization. Therefore, it is important for the acceptance of the system not to only convince management of its added value, but also to include physicians in the decision making process. So far, the physicians have been included in the design phase for the reference model. However, there will be more decisions that will impose significant culture changes. One important aspect is that rooms will be shared by multiple physicians as well as with assistants. That means a physician does not have his own room for which he is responsible and he will be more dependent on other people for the state of the room. Furthermore, in the current situation a physician completely manages his own time planning during a consultation session. In the new situation he will have less overview over the patient logistics, because the system will be in charge of time management. Besides handing over of tasks the physician will have to also delegate tasks to the system, which might be something that is not easily accepted by physicians.

6. Discussion & Conclusion

In this chapter a reflection on the SPExFlex approach based on the performed case study is presented. For each phase of the approach a concise explanation of some of the issues that were experienced and how they could be resolved is provided. After the reflection a general answer to the overall research question is provided resulting in an overall conclusion of the research.

6.1 Reflection on the definition phase

Because the design phase was not included in the case study, it makes no sense to reflect on it here. Therefore, the first phase discussed is the definition phase. The main issue experienced in the definition phase was to properly translate the reference model. When trying to specify specific tasks and roles it became clear that for certain tasks and roles there was still no clear image on what these exactly meant. Therefore, it took quite some effort to get to the process definition.

For the approach, that means that it might be a good idea to have a closer relation between the design phase and the definition phase. One idea for realizing that could be by defining templates for the task and role descriptions before starting the design phase that are completed during the design phase and that can be used during the definition phase. Another option might be to include a loop between the design and the definition phase to return to the design phase to be able to alter the reference model according to the experienced issues in the define phase. Future applications of the approach could focus on finding the best option for incorporating this in the approach.

6.2 Reflection on the implementation phase

The major problem that was experienced during implementation was that, although the tasks were defined in the process definition, the tasks could not be implemented in the system. The main reason for that, is that it included tasks that were largely dependent on other systems and hard to simulate within the scope of the prototype. It was identified that this was a problem purely related to the case study and not to the approach, because when the system would have been integrated within the existing infrastructure the tasks could have been implemented based on the process definition.

The integration step of the implementation phase was identified as one of the major aspects of the approach that requires further research. It is expected that integrating the system within the existing environment might cause some serious implications, for example in relation to the data. It is important to test whether defining the data and the configurations on the same model works in practice. Problems might occur if these two interfere. For example, if task a requires mandatory input data from another task b, but task b can be skipped when the model is configured, task a might not be executable anymore. Of course, these issues should be avoided while modeling the configurable process definition. However, without executing the integration step, it is unclear whether the approach sufficiently supports these and possible other related issues.

6.3 Reflection on configuration and execution

Using interviews to configure the model worked well, resulting in two configurations. Again, the simplicity of the model was a limitation, because when more decision points would be present configuration by domain experts might have become harder. Despite that, the overall experience with the setup of the configuration through abstraction was positive. It is a good way for end users to have their flexibility in the process, while not having to worry about it when performing their daily routines.

Execution of the different tasks was successful and it was shown that tasks could be assigned to the right users at the right moment, and the tasks could be performed in a correct way. However, linked to the lack of the integration as explained before, disallowed for execution within the actual setting. Therefore, the execution could not be evaluated to its full extend. Especially interesting would be to see how the cooperation with the process repository would function in practice. It would be interesting to see how often configuration could not (correctly) be retrieved and how those problems could be resolved. Furthermore, it would be interesting to see how much pressure would be on the 'problem manager', because if he would have to much work, it would be an indication that the levels of flexibility required were not met.

Overall, it can be concluded that the main challenge for future work would be to extend the scale of the implementation and apply the approach in a more complex environment. Based on that application, possible additions or changes can be made to the approach to make it suitable for application in real situations.

6.4 Conclusion

Based on the presented results and the discussion an overall conclusion can be drawn in the form of an answer to the main research question:

How can flexible workflow management and process standardization techniques be combined to balance standardization and flexibility in business process execution of management processes at outpatient departments?

By combining the strong aspects of flexible workflow management and process standardization techniques the SPExFlex approach provides the answer to the question. By introducing model configuration in the classic workflow reference model a way was found to allow for standardization while implementing a workflow management system that provides flexibility to its users.

By applying the approach in a practical case study it was shown that it is possible to get from a standard reference model for multiple departments to a department specific implementation that still adheres to the standardized model. Although, testing the approach in more complex environments is required before the approach is ready to be used in practice, a first foundation has been presented.

7. References

van der Aalst, W. M. P. P. (2013) 'Business Process Management: A Comprehensive Survey', *ISRN Software Engineering*, 2013, pp. 1–37. doi: 10.1155/2013/507984.

van der Aalst, W. M. P., ter Hofstede, A. H. M. and Barros, A. P. (2003) 'Workflow Patterns', *Distributed and Parallel Databases*, 14(1), pp. 5–51. doi: 10.1023/A:1022883727209.

Van Der Aalst, W. M. P., Pesic, M. and Schonenberg, H. (2009) 'Declarative workflows: Balancing between flexibility and support', *Computer Science - Research and Development*, 23(2), pp. 99–113. doi: 10.1007/s00450-009-0057-9.

Van Der Aalst, W. M. P., Weske, M. and Grünbauer, D. (2005) 'Case handling: A new paradigm for business process support', *Data and Knowledge Engineering*, 53(2), pp. 129–162. doi: 10.1016/j.datak.2004.07.003.

Aguilar-Savén, R. S. (2004) 'Business process modelling: Review and framework', *International Journal of Production Economics*, 90(2), pp. 129–149. doi: 10.1016/S0925-5273(03)00102-6.

Becker, J. *et al.* (2004) 'Configurative Process Modeling – Outlining an Approach to increased Configurative Process Modeling – Outlining an Approach to increased Business Process Model Usability', *Innovations Through Information Technology*, (September 2015), pp. 615–619. doi: 10.4018/978-1-59140-261-9.ch157.

Campbell, H. *et al.* (1998) 'Integrated care pathways.', *BMJ (Clinical research ed.)*, 316(7125), pp. 133–7. doi: 10.1136/bmj.316.7125.133.

Cano, I. *et al.* (2015) 'An adaptive case management system to support integrated care services: Lessons learned from the NEXES project', *Journal of Biomedical Informatics*. Elsevier Inc., 55, pp. 11–22. doi: 10.1016/j.jbi.2015.02.011.

Chiao, C. M., Künzle, V. and Reichert, M. (2013) 'Enhancing the Case Handling paradigm to support object-aware processes', *CEUR Workshop Proceedings*, 1027, pp. 89–103.

Chinosi, M. and Trombetta, A. (2012) 'BPMN: An introduction to the standard', *Computer Standards and Interfaces*. Elsevier B.V., 34(1), pp. 124–134. doi: 10.1016/j.csi.2011.06.002.

Cosgrove, S. J. (1991) 'Techniques for knowledge acquisition'.

Czarnecki, K. and Antkiewicz, M. (2005) 'Mapping features to models: A template approach based on superimposed variants', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3676 LNCS, pp. 422–437. doi: 10.1007/11561347_28.

Dwivedi, A., Bali, R. K. and James, A. E. (2001) 'WORKFLOW MANAGEMENT SYSTEMS : THE HEALTHCARE TECHNOLOGY OF THE FUTURE?', pp. 2–6.

Fahland, D. and Hagen, V. (2016) 'Dynamic Skipping and Blocking and Dead Path Elimination for Cyclic Workflows', *Business Process Management*, 9850, pp. 234–251. doi: 10.1007/978-3-319-45348-4.

Fettke, P., Loos, P. and Zwicker, J. (2005) 'Business Process Reference Models: Survey and Classification', *Business Process Management Workshops*, pp. 1–15. doi: 10.1007/11678564_45.

Gottschalk, F. *et al.* (2008) 'Configurable Workflow Models', *International Journal of Cooperative Information Systems*, 17(2), pp. 177–221. doi: 10.1142/S0218843008001798.

Gottschalk, F. (2009) *Configurable Process Models*.

Gottschalk, F., Van Der Aalst, W. M. P. and Jansen-Vullers, M. H. (2007) 'SAP WebFlow made configurable: Unifying workflow templates into a configurable model', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4714 LNCS, pp. 262–270. Available at: https://www.scopus.com/inward/record.uri?eid=2-s2.0-38049184876&partnerID=40&md5=b45a5584934d99916cbaa8cd49b1dee6.

Gröner, G. *et al.* (2013) 'Modeling and validation of business process families', *Information Systems*, 38(5), pp. 709–726. doi: 10.1016/j.is.2012.11.010.

Hagen, C. and Alonso, G. (2000) 'Exception handling in workflow management systems', *IEEE Transactions on Software Engineering*, 26(10), pp. 943–958. doi: 10.1109/32.879818.

Hallerbach, A., Bauer, T. and Reichert, M. (2008) 'Context-based configuration of process variants', *3rd International Workshop on Technologies for ContextAware Business Process Management*, 2(Activity 1), pp. 31–40. Available at: http://dbis.eprints.uni-ulm.de/199/.

Hallerbach, A., Bauer, T. and Reichert, M. (2010) 'Capturing variability in business process models: The Provop approach', *Journal of Software Maintenance and Evolution*, 22(6–7), pp. 519–546. doi: 10.1002/smr.491.

Hammer, M. and Stanton, S. (1999) 'How Process Enterprises Really Work', *Harvard Business Review*, 77(6), pp. 108–118. doi: 10.1017/CBO9781107415324.004.

Han, Y., Sheth, A. and Bussler, C. (1998) 'A taxonomy of adaptive workflow management', *Workshop of the 1998 ACM Conference on Computer Supported Cooperative Work*, pp. 1–11. Available at: http://gcc.uni-paderborn.de/www/wi/wi2/wi2_lit.nsf/0/a7fac9b815f26c87c1256c8e00669076/$FILE/CSCW98Workshop han-sheth-bussler.pdf.

Hollingsworth, D. (1995) 'The Workflow Reference Model', *Management*, am(1), pp. 1–55. doi: citeulike-article-id:1378584.

Kueng, P. and Kawalek, P. (1997) 'Goal-Based Business Process Models : Creation and Evaluation', *Business Process Management Journal*, 3(1), pp. 17–38. Available at: http://www.emeraldinsight.com/doi/abs/10.1108/14637159710161567.

LaRosa, M. *et al.* (2013) 'Business Process Model Merging: An Approach to Business Process Consolidation', *ACM Trans. Softw. Eng. Methodol.*, 22(2), p. 11.

Lenz, R. and Reichert, M. (2007) 'IT support for healthcare processes - premises, challenges, perspectives', *Data and Knowledge Engineering*, 61(1), pp. 39–58. doi: 10.1016/j.datak.2006.04.007.

Ma, Z. *et al.* (2007) 'Semantic business process repository', *CEUR Workshop Proceedings*, 251, pp. 1–11.

Madhusudan, T., Zhao, J. L. and Marshall, B. (2004) 'A case-based reasoning framework for workflow model management', *Data and Knowledge Engineering*, 50(1), pp. 87–115. doi: 10.1016/j.datak.2004.01.005.

Milla-Millán, G. *et al.* (2013) 'Knowledge-driven adaptive execution of care pathways based on continuous planning techniques', *Process Support and Knowledge Representation in Health Care*, 7738(3), pp. 42–55. doi: 10.1007/978-3-642-36438-9_3.

Moody, D. L. (2003) 'The Method Evaluation Model : A Theoretical Model for Validating Information Systems Design Methods', *ECIS 2003 proceedings*, p. 79.

Mooney, J. G., Gurbaxani, V. and Kraemer, K. L. (1995) 'Business Value of Information Technology ':, *MIS Quartely*, 33(June), pp. 137–156. doi: 10.2307/249685.

Motahari-Nezhad, H. R. and Swenson, K. D. (2013) 'Adaptive Case Management: Overview and Research Challenges', *2013 IEEE 15th Conference on Business Informatics (CBI)*, (July 2013), pp. 264–269. doi: 10.1109/CBI.2013.44.

OuYang, C. *et al.* (2010a) 'Workflow Management', in *Handbook on Business Process Management*, p. 622. doi: 10.1007/978-3-642-00416-2.

OuYang, C. *et al.* (2010b) 'Workflow Management', in *Handbook on Business Process Management*, p. 622. doi: 10.1007/978-3-642-00416-2.

Peffers, K. *et al.* (2006) 'The Design Science Research Process: A Model For Producing And Presenting Information'.

Pesic, M. and van der Aalst, W. M. P. (2006) 'A Declarative Approach for Flexible Business Processes Management', *Business Process Management Workshops*, pp. 169–180. doi: 10.1007/11837862_18.

Púchovská, J. L. and Závadský, J. (2016) 'Business Process Standardization', *Journal of Economics and Social Sciences*, 17(1), pp. 26–36.

Puhlmann, F. *et al.* (2005) 'Variability mechanisms for process models', *PESOA-Report TR*, 17, pp. 10–61. Available at: http://www.frapu.de/pdf/PESOA_TR_17-2005.pdf.

Reijers, H. A. *et al.* (2010) 'Workflow for healthcare: A methodology for realizing flexible medical treatment processes', *Lecture Notes in Business Information Processing*, 43 LNBIP, pp. 593–604. doi: 10.1007/978-3-642-12186-9_57.

Reijers, H. A., Rigter, J. H. M. and Van Der Aalst, W. M. P. (2003) 'The case handling case', *International Journal of Cooperative Information Systems*, 12(3), pp. 365–391. doi: 10.1142/S0218843003000784.

Reinhartz-berger, I. and Sturm, A. (2004) 'Behavioral Domain Analysis – The Application-Based Domain Modeling Approach', *Uml 2004*, pp. 410–424. doi: 10.1007/978-3-540-30187-5_29.

Rojas, E. *et al.* (2016) 'Process mining in healthcare: A literature review', *Journal of Biomedical Informatics*. Elsevier Inc., 61, pp. 224–236. doi: 10.1016/j.jbi.2016.04.007.

La Rosa, M. *et al.* (2006) 'Questionnaire-driven Configuration of Reference Process Models', *Microporous and Mesoporous Materials*, 93, pp. 240–247. doi: 10.1177/0885328206063905.Novel.

La Rosa, M. *et al.* (2009) *Configurable multi-perspective business process models*, *Information Systems*. doi: 10.1016/j.is.2010.07.001.

La Rosa, M., van der Aalst, W. M. P., *et al.* (2015) 'Business Process Variability: A Survey', *IJCAI International Joint Conference on Artificial Intelligence*, 2015–Janua(212), pp. 4207–4211. doi: 10.1145/0000000.0000000.

La Rosa, M., van der Aalst, W. M. P., *et al.* (2015) 'Business Process Variability Modeling: A Survey', *IJCAI International Joint Conference on Artificial Intelligence*, 2015–Janua(212), pp. 4207–4211. doi: 10.1145/0000000.0000000.

Russell, N., Van Der Aalst, W. and Ter Hofstede, A. (2006) 'Workflow exception patterns', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4001 LNCS, pp. 288–302. doi: 10.1007/11767138_20.

Schobbens, P. Y. *et al.* (2006) 'Feature Diagrams: A survey and a formal semantics', *Proceedings of the IEEE International Conference on Requirements Engineering*, pp. 136–145. doi: 10.1109/RE.2006.23.

Schonenberg, H. *et al.* (2008) 'Process flexibility: A survey of contemporary approaches', *Lecture Notes in Business Information Processing*, 10 LNBIP, pp. 16–30. doi: 10.1007/978-3-540-68644-6_2.

Shackleton-dijkstra, I. E. C. V. E. S. (2017) 'Kwaliteitsregistraties : van meten naar verbeteren', pp. 73–76. doi: 10.1007/s13629-016-0147-y.

Weber, B. and Reichert, M. (2012a) 'Flexibility Issues in Process-Aware Information Systems', in *Enabling Flexibility in Process-aware Information Systems*. Berlin: Springer, pp. 43–53. doi: 10.1007/978-3-642-30409-5.

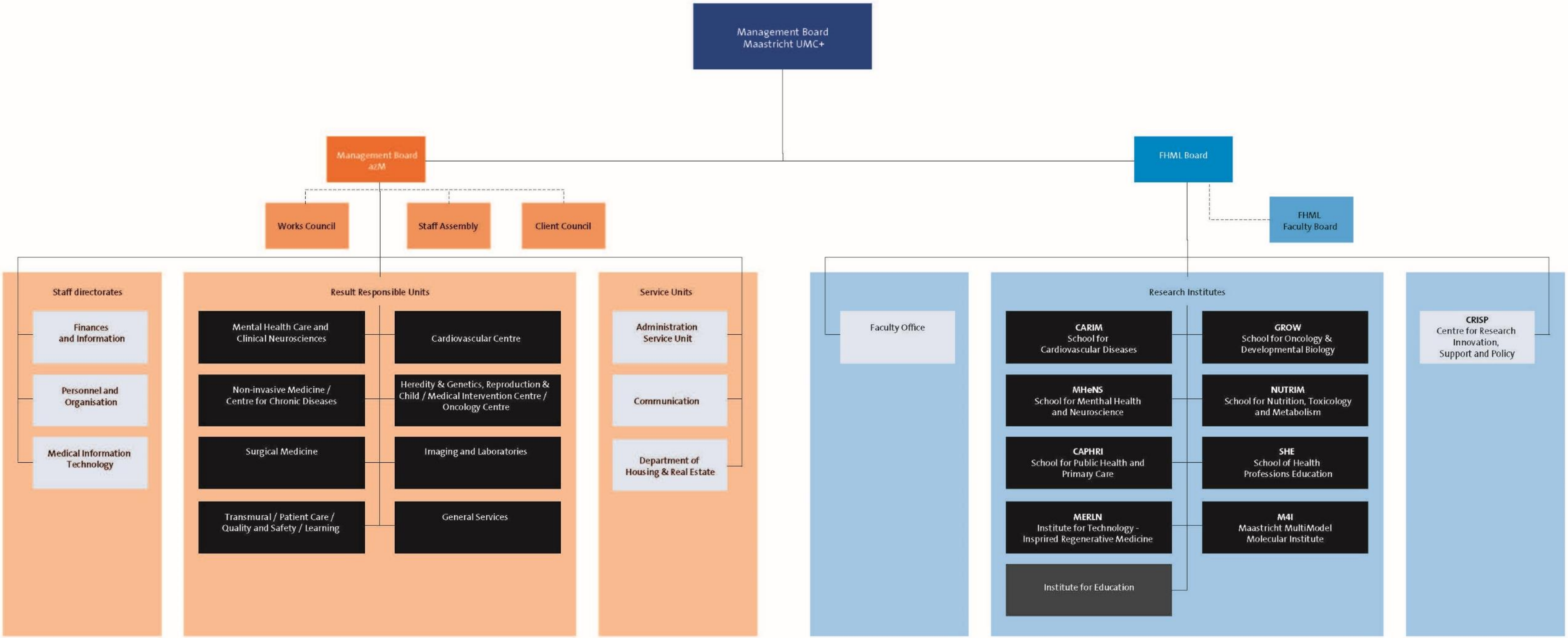
Weber, B. and Reichert, M. (2012b) 'Flexibility Issues in Processs-Aware Information Systems', in *Enabling Flexibility in Process-aware Information Systems*, pp. 43–53. doi: 10.1007/978-3-642-30409-5.

Yan, Z., Dijkman, R. and Grefen, P. (2012) 'Business process model repositories - Framework and survey', *Information and Software Technology*. Elsevier B.V., 54(4), pp. 380–395. doi: 10.1016/j.infsof.2011.11.005.

Zuiderent-Jerak, T. (2007) 'Preventing Implementation: Exploring Interventions with Standardization in Healthcare', *Science as Culture*, 16(3), pp. 311–329. doi: 10.1080/09505430701568719.

8. Appendices

Appendix 1: Organizational model MUMC+



Appendix 2: Data model

Task	ZIB	Data in	Data from?	Data out	ZIB	Data type	Cardinality
Check patient information	*Not defined*	Previous appointment summary	SAP system	-			
		Overview of test result since last appointment	SAP system				
Consultation preparation	*Not defined*	Previous appointment summary	SAP system	-			
		Overview of test result since last appointment	SAP system				
Welcome Patient		General patient data	SAP System	-			
Functional observation	FunctionalOrMentalStatus	StatusDate	System	StatusName	FunctionalOrMentalStatus	Coded description	1
				StatusValue	FunctionalOrMentalStatus	Coded description	0..1
				Explanation	FunctionalOrMentalStatus	Text	0..1
				MedicalAid	FunctionalOrMentalStatus	Text	0..*
Anamnesis	*Not defined*	Medical history	SAP System	Explanation	*Not defined*	String	1
		Previous appointment summarie(s)	SAP System				
Verifying use of medication	Use, medication	Delivery overview	Pharmacy	PatientPermission	*Not defined*	Boolean	1
		Overview of current medication use	Patient	Per product:			
		OPD medication	EVS system	Name of product	Prescription	Coded description	1
				Start date and time	Prescription	Timestamp	0..1
				End date and time	Prescription	Timestamp	0..1
				Duration of use	Prescription	Physical Quantity	0..1
				Number of Doses	Prescription	Physical Quantity	0..1
				Dose	Prescription	Physical Quantity	0..1
				Route of administration	Prescription	Coded description	1
				Rate of infusion	Prescription	Physical Quantity	0..1
				Duration	Prescription	Physical Quantity	0..1
				Explanation	Prescription	String	0..1
				Criterium	Prescription	String	0..1
				Maximum dose per time unit	Prescription	Physical Quantity	0..1
				As needed (Zo Nodig)	Prescription	Boolean	0..1
Complete medical history		Known medical history for patient	SAP System	Aangepaste voorgeschiedenis			
Discuss questionnaire		Questionnaire (paper or digital)	Brought by patient or online portal	Depends on questionnaire, processed results (typically a single score)	*Not defined*		
Weight	BodyWeight	-	-	WeightValue	BodyWeight	Physical Quantity	1
				Explanation	BodyWeight	String	0..1
				WeightDateTime	BodyWeight	Timestamp	1
				Clothing	BodyWeight	Coded description	0..1
Height	BodyHeight			Heightvalue	BodyHeight	Physical Quantity	1
				HeigthDateTime	BodyHeight	Timestamp	1
				Explanation	BodyHeight	String	0..1
				Position	BodyHeight	Coded description	0..1
Blood pressure	BloodPressure	-	-	MeasuringMethod	BloodPressure	Coded description	0..1
				CuffType	BloodPressure	Coded description	0..1
				MeasuringLocation	BloodPressure	Coded description	0..1
				DiastolicEndpoint	BloodPressure	Coded description	0..1
				SystolicBloodPressure	BloodPressure	Physical Quantity	1
				DiastolicBloodPressure	BloodPressure	Physical Quantity	1
				AverageBloodPressure	BloodPressure	Physical Quantity	0..1
				BloodPressureDateTime	BloodPressure	Timestamp	1
				Explanation	BloodPressure	String	0..1
				Position	BloodPressure	Coded description	0..1
Wound	Wound	Previous wound registrations	System	WoundTissue	Wound	Coded description	0..1
				WoundInfection	Wound	Boolean	0..1
				WoundMoisture	Wound	Coded description	0..1
				WoundEdge	Wound	Coded description	0..1
				WoundLength	Wound	Physical Quantity	0..1
				WoundWidth	Wound	Physical Quantity	0..1
				WoundDepth	Wound	Physical Quantity	0..1
				Explanation	Wound	String	0..1
				WoundType	Wound	Coded description	1
				AnatomicalLocation	Wound	Coded description	0..1
				WoundStartDate	Wound	Timestamp	0..1
				DateOfLastDressingChange	Wound	Timestamp	0..1
				WoundImage	Wound	Encoded data	0..1
				PainScoreValue	PainScore	Physical Quantity	0..1
				PainScoreDateTime	PainScore	Timestamp	0..1
				PainMeasuringMethod	PainScore	Coded description	0..1
				Explanation	PainScore	String	0..1

Temparature	BodyTemperature			TemperatureValue	BodyTemperature	Physical Quantity	1
				TemperatureDateTime	BodyTemperature	Timestamp	1
				Explanation	BodyTemperature	String	0..1
				TemperatureType	BodyTemperature	Coded description	0..1
Intoxication	*Not defined*			SmokingUseStatus	*Not Defined*	Coded description	1
				SmokingStartDate	*Not Defined*	Timestamp	0..1
				SmokingStopDate	*Not Defined*	Timestamp	0..1
				SmokingAmount	*Not Defined*	Physical Quantity	0..1
				Explanation	*Not Defined*	String	0..1
				AlcoholUseStatus	AlcoholUse	Coded description	1
				StartDate	AlcoholUse	Timestamp	0..1
				StopDate	AlcoholUse	Timestamp	0..1
				Amount	AlcoholUse	Physical Quantity	0..1
				Explanation	AlcoholUse	String	0..1
				DrugUseStatus	DrugUse	Coded description	1
				StartDate	DrugUse	Timestamp	0..1
				StopDate	DrugUse	Timestamp	0..1
				Amount	DrugUse	String	0..1
				TypeOfDrugOrMedication	DrugUse	Coded description	0..1
				RouteOfAdministration	DrugUse	Coded description	0..*
				Explanation	DrugUse	String	0..1
Allergies	AllergyIntolerance	Known Allergies	System	CausativeSubstance	AllergyIntolerance	Coded description	1
				AllergyCategory	AllergyIntolerance	Coded description	0..1
				AllergyStatus	AllergyIntolerance	Coded description	0..1
				StartDateTime	AllergyIntolerance	Timestamp	0..1
				CriticalExtent	AllergyIntolerance	Coded description	0..1
				LastReactionDateTime	AllergyIntolerance	Timestamp	0..1
				Symptom	AllergyIntolerance	Coded description	1..*
				SpecificSubstance	AllergyIntolerance	Coded description	0..1
				Probability	AllergyIntolerance	Coded description	0..1
				ReactionDescription	AllergyIntolerance	String	0..1
				Severity	AllergyIntolerance	Coded description	0..1
				MannerofExposure	AllergyIntolerance	Coded description	0..1
				ReactionTime	AllergyIntolerance	Timestamp	0..1
				ReactionDuration	AllergyIntolerance	Physical Quantity	0..1
Pulse	PulseRate			Explanation	PulseRate	String	0..1
				PulseRegularity	PulseRate	Coded description	0..1
				PulseRateDateTime	PulseRate	Timestamp	1
				PulseRateValue	PulseRate	Physical Quantity	0..1
Rheumatology joint check	*Not defined*	ESR value	Blood test results (SAP system)	SensitiveJoint (for each joint)	*Not defined*	Boolean	1
				SwollenJoint (for each joint)	*Not defined*	Boolean	1
				PainScoreValue	PainScore	Physical Quantity	0..*
				PainScoreDateTime	PainScore	Timestamp	0..*
				PainMeasuringMethod	PainScore	Coded description	0..*
				Explanation	PainScore	String	0..*
				Explanation	*Not defined*	String	0..1
Consultation		Medical history	SAP system	Decursus	*Not defined*	String	1
		Overview previous appointments	SAP system	Order for future appointment or examination (SAP/OM system)	*Not defined*	Boolean	0..*
		Test results	SAP system	Changes in medication (EVS system)	*Not defined*	-	
		Summary preparation tasks	SAP system				
Inform patient		Treatment plan	Decursus	-		-	
Arrange hospitalization		Resource availability	SAP system	Order for hospitalization	*Not defined*	Standard form	0..1
Provide information		Available information (links/folders/etc..)	Paper materials/websites	-		-	
DBC tasks		DBC code	Decursus	(changed) DBC Code	*Not defined*	Code according to DBC format	0..1
		Current DBC code	SAP system				
Draft GP letter		Automatically generated letter	SAP system	Checked/completed letter	*Not defined*	Document	0..1

Appendix 3: Evaluation scenarios and questionnaire

Casus 1

Op dit moment zit Theo Jansen in de wachtruimte, klaar voor zijn afspraak op de polikliniek. Bij deze afspraak zal jij de arts ondersteunen door voor en na het consult wat ondersteunende taken uit te voeren.

Nadat je Theo welkom hebt geheten en met hem naar de toegewezen kamer bent gegaan controleer je de persoonlijke gegevens van de patiënt. Theo geeft aan dat zijn geboortedatum 9 augustus 1967 is en dat zijn contactgegevens niet veranderd zijn. Als dat gebeurd is kunnen de overige voorbereidende taken uitgevoerd worden. Als eerste meet je de BMI van Theo, hij blijkt 1m86 te zijn en 87 kilo te wegen. Vervolgens wordt de bloeddruk gemeten die 140/90mmHg blijkt te zijn. Ook wordt Theo gevraagd naar zijn gebruik van alcohol en/of drugs. Hij blijkt gemiddeld 2 biertjes per week te drinken. Hij kan zich niet herinneren dat hij ooit drugs gebruikt heeft maar hij heeft wel van zijn 20^{e} tot zijn 25^{e} gerookt. Als laatste wordt er nog een medicatieverificatie gedaan van de medicatie die Theo. Hij vermeld daarin dan hij zelf besloten heeft de dosering van Etoricoxib te halveren, omdat hij te veel last had van bijwerkingen. Als deze taken uitgevoerd zijn wordt de arts gewaarschuwd om naar de kamer te komen en is de voorbereiding afgerond.

Je krijgt een seintje om terug naar de kamer te gaan en de afsluitende taken uit te voeren als de arts klaar is. In dit geval heeft de arts aangegeven welk beleid er uitgevoerd gaat worden en kan jij de patiënt hier meer informatie over geven. Eveneens heeft de arts aangegeven dat er een afspraak gemaakt moet worden voor een labonderzoek van de urine van de patiënt. Je maakt de afspraak voor het labonderzoek en beantwoord eventuele vragen die de patiënt verder nog heeft. Als deze taken uitgevoerd zijn begeleid je de patiënt naar de uitgang. Als de patiënt vertrokken is maak je de kamer gereed voor de volgende patiënt en is het proces voor deze patiënt afgerond.

Casus 2

De App geeft aan dat Theo Jansen zit te wachten in de wachtkamer en jij krijgt de taak om deze patiënt welkom te heten. Echter, als je naar de wachtkamer gaat om Theo op te halen, blijkt er niemand te zitten. Je geeft dit aan in de App. Tien minuten later geeft de App aan om het nog eens te proberen en toevallig ben jij dan nog steeds beschikbaar. Theo bleek even naar de WC te zijn geweest en is nu beschikbaar.

Opnieuw voer je de taken uitgevoerd zoals de App deze voorstelt. Echter, tijdens het meten van de bloeddruk voel je je opeens niet lekker worden en wel zo erg dat je besluit de kamer te verlaten. Je geeft in de App aan dat je de patiënt wil overdragen aan een collega en verlaat de kamer.

Voor deze casus ben jij ook even de collega die het werk overneemt. Je krijgt een melding en de App geeft aan welke taken er nog uitgevoerd moeten worden. Je voert deze taken uit en vervolgens wordt het consult uitgevoerd door de arts. De afrondende taken worden vervolgens zonder problemen uitgevoerd, echter bij het opruimen van de kamer ontdek je dat de bloeddrukmeter kapot is gegaan. Je geeft in de App aan dat de kamer ongeschikt is voor gebruik en verlaat de kamer.

Questionnaire

Stelling 1: Ik vind de procedure voor het toepassen van de App complex en moeilijk te volgen

Oneens **Eens** **Geen Mening**

Stelling 2: Ik geloof dat deze methode de hoeveelheid werk dat nodig is om mijn handelingen te registreren zou verminderen

Oneens **Eens** **Geen Mening**

Stelling 3: De taken die via deze manier weergeven worden zijn moeilijker te begrijpen voor gebruikers

Oneens **Eens** **Geen Mening**

Stelling 4: Over het algemeen vond ik het lastig om met de App te werken

Oneens **Eens** **Geen Mening**

Stelling 5: Door deze app te gebruiken wordt het makkelijker voor gebruikers om er zeker van te zijn dat handelingen op de juiste manier geregistreerd worden

Oneens **Eens** **Geen Mening**

Stelling 6: Ik vond het makkelijk om met de App te leren werken

Oneens **Eens** **Geen Mening**

Stelling 7: In het algemeen vond ik de App nuttig

Oneens **Eens** **Geen Mening**

Stelling 8: Door deze app te gebruiken wordt het moeilijker om patiëntgegevens te onderhouden

Oneens **Eens** **Geen Mening**

Stelling 9: Ik vond het moeilijk om de casussen uit te voeren met behulp van de App

Oneens **Eens** **Geen Mening**

Stelling 10: Ik zou deze app zeker niet gebruiken om mijn taken uit te voeren

Oneens **Eens** **Geen Mening**

Stelling 11: Ik vond het makkelijk en duidelijk wat ik wel en wat ik niet met de App kon doen

Oneens **Eens** **Geen Mening**

Stelling 12: In het algemeen denk ik dat deze App geen effectieve oplossing biedt voor de problemen

Oneens **Eens** **Geen Mening**

Stelling 13: Door deze App te gebruiken zou de communicatie tussen gebruikers verbeteren

Oneens **Eens** **Geen Mening**

Stelling 14: Ik ben er niet van overtuigd dat ik deze App nu in de praktijk zou kunnen gebruiken

Oneens **Eens** **Geen Mening**

Stelling 15: In het algemeen vind ik de App een waardevolle toevoeging aan het huidige systeem

Oneens **Eens** **Geen Mening**

Open vragen

Vraag 1: Wat is in jouw ogen het meest positieve en wat het meest negatieve aspect van de App en waarom?

Positief:

Negatief:

Vraag 2: Welke problemen voorzie je bij de overstap van de huidige situatie naar de situatie waarin met de app gewerkt wordt?