UNIVERSITEIT GENT

FACULTEIT ECONOMIE EN BEDRIJFSKUNDE

GHENT UNIVERSITY

FACULTY OF ECONOMICS AND BUSINESS ADMINISTRATION

ACADEMIC YEAR 2014 – 2015

Case Management Theory, Modelling, Limitations and Tools: An Overview

A master dissertation submitted in partial fulfilment of the requirements for the degree of
Master of Science in Commercial Sciences

Cédric Mestdagh

under the supervision of

Prof. Dr. Manu De Backer

UNIVERSITEIT GENT

FACULTEIT ECONOMIE EN BEDRIJFSKUNDE

UNIVERSITEIT GENT

FACULTEIT ECONOMIE EN BEDRIJFSKUNDE

ACADEMIEJAAR 2014 – 2015

Case Management Theory, Modelling, Limitations and Tools: An Overview

Masterproef voorgedragen tot het bekomen van de graad van

Master of Science in de Handelswetenschappen

Cédric Mestdagh

onder leiding van

Prof. Dr. Manu De Backer

Abstract (in Dutch)

In de loop der jaren is het managen en modelleren van bedrijfsprocessen een actueel onderwerp geworden in bedrijfsmanagement. Business Process Management (BPM) systemen maakten het mogelijk om vooraf vastgelegde bedrijfsprocessen te automatiseren, wat leidde tot een algemene verbetering van de processen en hun opeenvolging, een daling van de operationele kosten, minder doorlooptijd en de opsporing en identificatie van patstellingen en knelpunten. Ondanks dat BPM-systemen erin slagen veel voordelen te genereren voor organisaties ontberen ze één belangrijke eigenschap: het vermogen om in te kunnen spelen op een flexibele en dynamische bedrijfsomgeving. Bedrijfsomgevingen vandaag de dag worden gekenmerkt door kennisintensieve bedrijfsprocessen die gemanaged worden door zogeheten 'knowledge workers'. Dit zijn werkkrachten die hun ervaring, kennis, kunde en intelligentie inzetten om ad-hoc-beslissingen te nemen en taken uit te voeren wanneer ze dat nodig achten. Toekomstige inspanningen zullen zeer sterk gericht zijn op het beheren, controleren en monitoren van deze soort processen. Case management (CM) systemen bieden bedrijven de kans om tegemoet te komen aan deze flexibele en dynamische omgevingen. Het kleine beetje onderzoek dat al over case management is gevoerd, is vaak sterk verspreid en niet ondubbelzinnig. In deze thesis probeer ik een algemeen beeld te scheppen over case management. Ik probeer te identificeren wat case management eigenlijk inhoudt aan de hand van verschillende definities die doorheen de jaren aan case management zijn gegeven. Daarnaast verklaar ik ook wat bedrijven zou kunnen aanzetten om te investeren in case management door het aanhalen van de verschillende voordelen. Echter, ik tracht professionals ook bewust te maken van de uitdagingen en de beperkingen die ze op het pad van implementatie kunnen tegenkomen. Aan de hand van deze uitdagingen en beperkingen stel ik bedrijven in staat deze te erkennen en herkennen, zodat ze in staat zijn op basis daarvan de juiste beslissingen te nemen. Verder in deze thesis bespreek ik de verschillende benaderingen die in het verleden, alsook in het heden, worden gebruikt om tegemoet te komen aan de kenmerken van de dynamische bedrijfsomgeving. Voor de academische gemeenschap kan deze thesis gebruikt worden als springplank om dieper in te gaan op de verschillende benaderingen en concepten die werden en worden gebruikt voor het modelleren van case management. Voor professionals kan deze thesis worden gebruikt als een uitgebreide en algemene inleiding op het concept case management. Na het lezen van deze thesis zouden deze in staat moeten zijn om een kwalitatieve beoordeling te maken over case management, gebaseerd op haar karakteristieken, voordelen en uitdagingen.

Preface

In 2010, I enrolled in the Commercial Sciences programme at the Hogeschool Gent, now Ghent University. This was an obvious choice since I studied economics in high school. Through the years, my interest in economics, accountancy and other financial affairs shrank. In the last year of bachelor's programme they introduced the master's programme Management and Information Technology. Immediately, I knew this was a programme that would be like a breath of fresh air to me. Soon after, I found out that there were a few subjects that drew my attention: project management, business process management, business cases and system design and analysis. These subjects gave me new insights into businesses and their challenges, very important insights that traditional approaches like accountancy, marketing, strategic management and finance failed to address.

When I was allocated the subject of case management for my master dissertation, I had no idea what it implied. My supervisor, Prof. Dr. Manu De Backer assured me it was a fascinating topic and so I began my quest. From February to early August 2015, I did research on the case management basics, different approaches, benefits and challenges. It did not cost me any blood or tears, but what I lacked in shedding blood and tears, I have made up for in sweat. When I now look back at it, it really has been an incredible learning experience. I hope this master dissertation succeeds in exploring the case management concept and deliver some new insights for professionals and academics alike. If my efforts have helped you to use this knowledge in your research or professional occupation, I have achieved my highest goal.

I would like to thank Prof. Manu De Backer for his constructive feedback and for introducing me into the wondrous world of case management. Furthermore, I would like to thank my friends and mother for their moral support, Anneliese Nimmegeers and Geneviève Westelinck for re-reading this thesis and giving me feedback, Willy De Corte for showing an interest in my master dissertation and Jelle Grammens, with whom I have shared the ups and downs of writing a thesis.

In this spirit I hope you find my master thesis informative and enjoyable to read.

Mestdagh Cédric

Table of Contents

1 List of Abbreviations

BAM	Business Activity Monitoring
BoM	Bill of Materials
BPEL	Business Process Execution Language
BPM	Business Process Management
BPMN	Business Process Model and Notation
BPMS	Business Process Management System
BRM	Business Rules Management
BRMS	Business Rules Management System
CBR	Case-Based Reasoning
CM	Case Management
CMMN	Case Management Model and Notation
CMS	Content Management System
CMSA	Case Management Society of America
CRM	Customer Relationship Management
DeCo	Declarative Configurable Process
DEMO	Dynamic Essential Modelling of Organization
ECA	Event-Condition-Action
ECM	Enterprise Content Management
EDRMS	Electronic Document and Record Management System
ERMS	Electronic Record Management System
ERP	Enterprise Resource Management
GoM	Guidelines of Modelling
KM	Knowledge Management
LAP	Language-Action Perspective
MES	Manufacturing Execution System
PDA	Product Data Model
RAD	Role Activity Diagram
REACT	Research, Evaluate, Analyse, Constraint, Task
UI	User Interface
WASA	Workflow-based Architecture to support Scientific Applications
XMI	XML Metadata Interchange
XML	Extensible Markup Language
XMLS	XML-Schema
XPDL	XML Process Definition Language

2 List of Figures

3 Abstract

In recent years, managing and modelling business processes has been a hot topic in business management. Business Process Management Systems made it possible to automate predefined business process, leading to i.a. the overall improvement of processes and sequences, less operational costs, less lead time and the identification of inconsistencies. Although they offer lots of benefits to organisations, they lack one very important feature: the ability to react to a flexible and dynamic environment. Today’s business environment gets characterised by knowledge-intensive business processes managed by knowledge workers who rely on their knowledge and skills to be able to take ad hoc decisions and perform tasks as they see fit. Future efforts will be focussed strongly on managing these types of processes. Case management systems offer businesses the possibility to cope with these flexible and dynamic environments. The little research on case management that has been conducted is scattered and often not unambiguous. In this master dissertation I try to give an overall picture of case management. I try to identify what case management actually is using different definitions throughout various standpoints and explain why organisations may make an effort to invest and implement case management by listing the benefits. On the other hand, I call attention to the challenges organisations may come up against when implementing a case management system, making it possible for them to identify these challenges and act accordingly. Further on I discuss the different approaches that have been used in the past and now to cope with the specifics of the business environment, the same specifics case management tries to find a solution for. For the academic community, this master thesis may be used as a diving board to look into different approaches and concepts to model case management. From a practitioner’s point of view this master thesis can be used as an overall introduction to case management. After reading it, practitioners should be able to make a qualitative assessment based on the case management characteristics, benefits and challenges.

4 Introduction

Over the course of time, it became clear that current modelling approaches had a major flaw: human-interaction. Over time, these modelling languages and systems introduced new versions or extensions to expand their capabilities. The Business Process Execution Language (BPEL) was used as a tool to exchange information generated by business processes through web services. With the internet becoming more and more popular for doing business and the growth of web services, interest in BPEL grew. However, the increasing use and popularity exposed a flaw in BPEL’s core operations. Focus was always put on the interaction between business processes and web services. After a while, they realised the interactions were much broader than just business processes and web services. Human contribution to the execution of business processes was undeniably strong. To enable the modelling of human interaction in processes, BPEL4People was introduced. It made use of a set of specific elements that expanded the standard BPEL aspects, so that BPEL as a modelling language could cope with the complexity of human interaction (Clément, et al., 2010). Another very popular modelling language and notation encountered problems regarding interaction between processes, systems and its human participants: Business Process Model and Notation (BPMN). In 2011, OMG introduced the concepts of choreography and conversation diagrams into the Business Process Model and Notation standard (OMG, 2011). Choreography diagrams were introduced because they could model between-process interactions and message flows (Polancic, 2014), something that was becoming more and more important in an environment where internet-based commerce increased (Earls, 2011). A conversation diagram was also introduced to provide an overview of all the people involved in a certain process or activity and to model their interactions. Human interaction has become unmistakably important in modern processes. As a result, new modelling languages and notations try to implement human interactions more than ever. Together with human interaction, businesses became more and more dependent on knowledge, hence the growing amount of knowledge-intensive processes in businesses. Depending on the industry, knowledge-intensive business processes can make up 50% of a business' total processes (Franke, 2011).

These types of processes, together with their strong relationship and dependency on human-induced interaction, created a breeding ground for case management. The basis of the case management concept comes forth from research conducted by Davenport (1994, 2005) and van der Aalst (2005). They conclude that there is an array of activities that cannot be predefined and repeated as is possible with activities in a sequenced workflow. Instead those activities depend on evolving circumstances and ad hoc decisions by knowledge workers regarding a particular situation, called a case (OMG, 2014). Davenport (1994), is the first to specify the importance of human contribution on the level of information systems, thus acknowledging the value of human interactions in information systems. An interesting excerpt from this work (Davenport T. H., 1994, p. 122) proves this: "Managers prefer to get information from people rather than computers; people add value to raw information by interpreting it and adding context." Further on Davenport (1994) points out that any design of models should incorporate human-workers since they act as an important source and processor of information. Through his work Davenport (1994) introduces the line of though in which people become a predominant factor in managing businesses and the processes those businesses engulf. It is clear that people are important in the information age since they can handle the dynamics and unpredictability that comes with it and they will get increasingly more important in the years to come. These unpredictable events that are constantly interacting with the organisational environments will become known as *cases*. This thesis will delve deeper into cases when addressing their definition. De Man (2009b) finishes his article on the Cordys Approach with stating that it is to be expected that case management automation will become the cornerstone for balancing, scheduling and accounting for knowledge worker activities. This statement supports the importance of human interaction in case management.

In ‘Automated Decision Making Comes of Age’ by Davenport and Harris (2005) automated decision-making in businesses is introduced. This evolution implies the shift from manual, worker-induced decision-making to a computer supported data analysis but with a lack of application integration to finally come to the point where we are now: the point where decision-making is embedded in the normal flow of work without having to rely on any kind of human interference. Although the paper puts the emphasis on the benefits businesses can enjoy from automation in decision-making processes, Davenport and Harris (2005) still recognize the role of people in organisations. This excerpt supports that statement: “This is not to suggest that there is no role for people. Managers still need to be involved in reviewing and confirming decisions and, in exceptional cases, in making the actual decisions. Also, even the most automated systems rely on experts and managers to create and maintain rules and monitor the results.” (Davenport and Harris, 2005, p. 86). The fact that machines and computers can execute some tasks more efficient than human workers is an argument that from the management perspective, speaks in favour of automation. Although automation offers possibilities in terms of efficiency, the art of automated decision-making is to delegate tasks in accordance with their characteristics (like flexibility, routineness…) (Trkman, 2010). It is clear automation offers new perspectives, but it is important not to forget that case management is not something that will solve all current problems regarding decision-making. Managers will still need to think about what tasks and decision can be automated and what tasks and decision will need to stay under human supervision. Davenport and Harris (2005) give an example of a situation where automation should better be left into human hands and decision-making that will be more effective when handled by automation through computers. A bank credit decision is an example for the latter because decision-making is repetitive and can be based upon an analysis of a vast amount of data. In contrast, hiring a CEO is more subjective, more prune to the personality of the interviewer and thus not very predictable. In this regard, worker-induced decision-making is more used to cope with exceptions.

The need to model case management processes lead to a Request For Information (RFI) on ‘dynamic business activity modelling’ by the Object Management Group in 2008. In 2010 this Request For Information was transformed in a Request For Proposal (Object Management Group, 2010) as an instrument to gather proposals from practitioners and researchers (members of the OMG) in order to support the modelling of case management as an extension to BPMN 2.0. As of May, 2014, the beta version of the Case Management Model and Notation is released (OMG, 2014) by the Object Management Group. The scope of this specification is to define “a common meta-model and notation for modelling and graphically expressing a case, as well as an interchange format for exchanging Case models among different tools.” (OMG, 2014, p. 1). This notation is the first step towards a standardized implementation of case management throughout businesses. As this is just a first version, it is prone to errors and inaccuracies and businesses are therefore advised to approach this document critically and with care.

However, when delving into the subject of case management (CM) it becomes clear that this is a subject still in its infancy. There is little information on the successful implementation of case management in businesses and if it is already implemented, it is often intensely manual, paper-driven, plagued by delay and poor visibility, with isolated parts of the process automated by legacy systems or spreadsheets. This does not mean however, that businesses do not acknowledge the need for standardization in case management. The need to deal with unstructured and dynamic processes will grow proportionally with that of the increasingly complex business environment. There are two main reasons why case management is so poorly supported. Firstly, it is inherently more difficult to automate than other processes because of the extent to which cases processes must support human knowledge, judgment and discretion to determine their outcome. It is harder to manage the complexity and unpredictability of a case than say, automating payroll processing or credit card transaction processing. Secondly, the available technology simply hasn’t been able to support the requirements for dynamic-user driven changes to cases as they progress (White, 2009a).

The next section introduces the case management concept by giving a definition for it, introducing its most important building blocks, characteristics, benefits and challenges.

5 What is case management?

5.1 Definition

In order to get a grasp of what case management is, it would be informative to take a look at some definitions of case management and its building blocks. Case management is not a new concept. Its application is already widely used in many areas like claim processing in insurance, patient care and medical diagnosis in healthcare, mortgage processing in banking, call handling in call centres, sales and operations planning, invoice discrepancy handling, handling business requirements in R&D, engineering to order, etc. (de Man, 2009a).

Because it is adapted in a lot of different areas as you can see above, there is no uniform definition of case management:

- “Case Management is a way to govern and control these unstructured processes, processes that ordinary Business Process Management solutions were not designed to cope with.” (Capgemini, 2012, p. 1).

- "Case management is a collaborative process of assessment, planning, facilitation, care coordination, evaluation, and advocacy for options and services to meet an individual's and family's comprehensive health needs through communication and available resources to promote quality cost effective outcomes." (Case Management Society of America, 2010, p. 6).

- "Dynamic case management (DCM) is the handling of case-related work through the use of technologies that automate and streamline aspects of each case. In this context, a case is a collection of information about a particular instance of something, such as a person, company, incident or problem." (TechTarget, n.d.).

- "A highly structured, but also collaborative, dynamic, and information-intensive process that is driven by outside events and requires incremental and progressive responses from the business domain handling the case. Examples of case folders include a patient record, a lawsuit, an insurance claim, or a contract, and the case folder would include all the documents, data, collaboration artefacts, policies, rules, analytics, and other information needed to process and manage the case." (Le Clair & Miers, 2014).

The second definition originated in the healthcare industry and has become the de facto definition for case management (when not taking into account the medical terminology) (de Man, 2009a). Using these definitions it is possible to extract some of the building blocks and characteristics of case management.

5.2 Case management building blocks

5.2.1 Case and case subject

Case management makes use of *cases* that form the basis of a case management process. "A case can be defined as a collection of tasks, actions, processes and content in support of a specific business objective." (Capgemini, 2012, p. 2). Alternatively, a case can be seen as a particular situation for which case management will try to find the desired outcome linked to that case. De Man (2009a) compares cases to a series of milestones so that event management and milestone control are considered core features in case management. Event management allows marketers, event coordinators, recruiters… to plan, register, execute and analyse a range of virtual and in-person events (Oracle, 2014). In the Case Management Model and Notation 1.0 standard, published by the Object Management Group (2014), milestones are indeed one of the aspects that case management needs to capture and formalise.

It is not hard to draw the line between event and case management. We can simply regard an event to be a case instance. The phases of event management can be visualized and modelled as states of the case, making it able to see them as milestones the case passes through during its life-cycle. The use of state-machines in case management will be discussed further on.

Rooze et al. (2007) make a distinction between different kinds of cases leading to a different approach to case management: mass cases, regular cases and special cases:

- *Mass cases* are cases that are characterised by their bulkiness. These cases are omnipresent, simple and can be managed using workflow management tools like Business Process Management Notation (BPMN).
- *Regular cases* are cases that are common. They make up the better part of all cases and are managed using "planning of milestones" or procedure management. In 'Case Management: Cordys Approach' (2009), Henk de Man (2009b) specifies the regular case. According to him, these are the cases in which the human worker is in control of how the process evolves. Although the human worker enjoys a certain level of freedom, this freedom can be constrained by a number of elements like business rules and case states (status from the case used to delegate the work over the case life-cycle). Rooze et al. (2007) made the distinction between milestones and states. A state, like a case status can become a milestone when planned dates, deadlines, etc. are set for it. By transforming states into milestones, a planning instrument is created as is suggested to manage regular cases.
- *Special cases* are cases that are least present and are characterised by a high level of complexity. Rooze et al. (2007) does not give any suggestions how to manage this last category of cases, implying that there is no clear vision on how to manage these cases.

Cases are identified by a certain *subject*. The subject of a case can be anything. For example a person, a customer complaint, an insurance claim, an invoice, a purchasing order, feedback from stakeholders etc. Actions taken are those involving the subject and will lead to achieving the desired result linked to the case. The choice of actions to take in each case involves the exercise of human judgment and decision-making since activities don't occur in a predefined sequence (Object Management Group, 2010). This links back to the beginning of this overview, with Davenport (1994, 2005) and van der Aalst (2005) stressing the importance of so called knowledge workers to support automated decision-making.

5.2.2 Case file

Probably the most important element in a case management process is the *case file*. Case files act as the backbone of a case. Efforts made for standardization will often involve case files (de Man, 2009b). Later on in this thesis, when looking into the CMMN 1.0 standard, it will become obvious that case files are indeed part of the standardisation efforts. In order to be able to make decisions and make a representation of the situation in which these decisions have to be taken, it is necessary to collect and aggregate data in connection to the *case subject*. This aggregation/collection of data is called a *case file* (Object Management Group, 2010). A case file is defined as follows by the Object Management Group (2011): “A collection of documents (a file) relating to a specific action, transaction, event, person, place, project, investigation, or other subject.” (Object Management Group, 2011, p. 3). Notice that when an activity or event is triggered, the data that is linked to that specific event or activity needs to be gathered and stored in the case file. From that moment on, this data is inherent to the case file and will define its being. It is important to understand that the accumulation of data in the case file originates from all the inputs the case is involved in. Manual case management will often be driven by classic case folders, notes and documentation. This can make things hard to trace back, monitor or share files between case workers. In a modern case management system, all this should be electronic and linked to a central database that case workers can access easily.

5.2.3 Case workers

Case workers are the equestrians of case management. They steer the case through business processes based on their experience, skill, intuition and case knowledge. Based on Thomas Davenport’s (2005) definition, it becomes possible to easily draw the line between case workers and knowledge workers: “Knowledge workers have high degrees of expertise, education, or experience, and the primary purpose of their jobs involves the creation, distribution, or application of knowledge. […]They solve problems, they understand and meet the needs of customers, they make decisions, and they collaborate and communicate with other people in the course of doing their own work.” (Davenport T. H., 2005, pp. 10-11). Case workers use their knowledge to make decisions and solve problems, they collaborate with fellow case workers, communicate and share knowledge. Consequently they have a critical role in so called knowledge-intensive business processes.

Previous studies and reports often indicated that between 25% and 40% of the total workforce consists of knowledge workers (Adams & Oleksak, 2010). The introduction of computers has turned almost everybody in some sort of knowledge worker. This has led to not asking whether or not an employee is a knowledge worker. Instead, we look at the degree in which an employee is considered a knowledge worker. In their report on the Knowledge Economy, Brinkley et al. (2009) found that there is, in general, a 30-30-40 division in the workforce between jobs with high knowledge content, jobs with some knowledge content, and jobs with less knowledge content. The importance of knowledge/case workers in case management will become more obvious throughout this thesis.

5.2.4 Knowledge-intensive business processes

Case workers are inherently involved in knowledge-intensive business processes. Knowledge-intensive business processes are immensely important in case management. Remember that knowledge-intensive business processes can make up 50% of a business’ total processes (Franke, 2011). Between 2011 and now, this number will only have gone up and will keep going up in the years to come. The shift from an industrial to a knowledge economy (Slembek, 2003) has driven the conversion towards process performance management and business process optimization (Sidorova & Işık, 2010). Papavassiliou & Mentzas (2003) formulated knowledge-intensive business processes as: "The processes considered are often complex in general, with many, but conceptually simple, (usually) document-centred activities; at the heart of these processes are few central decision steps which require personal judgment based on experience, a comprehensive knowledge about the given as well as about older, similar cases, access to much specifc information in files and forms, manifold legal regulations and standard operating procedures, etc." (Papavassiliou & Mentzas, 2003, p. 20). Işık, Van de Bergh, & Mertens (2012) defined knowledge-intensive processes as "[..] processes that require very specific process knowledge, typically expert involvement, that are hard to predict and vary in almost every instance of the process." (Işık, Van de Bergh, & Mertens, 2012, p. 3818). Knowledge-intensive business processes comprise of characteristics that are also very representative for case management: the processes are either semi- or unstructured, they are very dependent on the decision-maker (case worker) in the process (Kulkarni & Ipe, 2007), the high amount of uncertainty (Kulkarni & Ipe, 2007; Marjanovic & Seethamraju, 2008), the diversity of possible decisions (Kulkarni & Ipe, 2007), the degree of predictability (Richter-von Hagen, Ratz, & Povalej, 2005; Panian, 2011), creativity (Marjanovic & Seethamraju, 2008; Richter-von Hagen, Ratz, & Povalej, 2005; Harmon P. , 2007; Sarnikar & Deokar, 2010), repeatability (Slembek, 2003; Marjanovic & Seethamraju, 2008), automation (Panian, 2011), complexity (Eppler, Seifried, & Röpnack, 1999; Harmon P. , 2007; Marjanovic & Seethamraju, 2008; Davenport T. H., 2010; Marjanovic & Freeze, 2011; Panian, 2011) and structure (Richter-von Hagen, Ratz, & Povalej, 2005). The research conducted by Işık, Van de Bergh, & Mertens (2012) showed that information in knowledge-intensive business processes mainly comes from external sources, the scope (aim or purpose (Chang, Chang, & Paper, 2003)) of the information can be either wide or narrow, the level of aggregation can differ (detailed or aggregated on a high-level), the time horizon is typically long range and focused on the future, the information is not necessarily current ("[…] refers to the urgency of the information provided for the process." (Işık, Van de Bergh, & Mertens, 2012, p. 3819)), accuracy is often low (case workers can give their own interpretation to external information) and the frequency of information-use is low ("infrequent processes only require information at the time of their execution" (Işık, Van de Bergh, & Mertens, 2012, p. 3819)).

An interesting finding in their study showed that the studied knowledge-intensive business processes were mostly repeatable. This is counter-intuitive, since these kind of processes are regarded as being ad hoc and dynamic. Işık, Van de Bergh, & Mertens (2012) give two possible explanations for this peculiar find. First off, the repeatability can be present, but on a much lower level than is the case with non-knowledge-intensive processes. With knowledge-intensive processes we talk about months or even years, resulting in a much lower number of instances in comparisons with non-knowledge-intensive process. Secondly, the reason for this find may be based on confusion or the subtle difference between repeatable and reproducible processes (Reynolds, 2011). According to Reynolds (2011) the repeatability says nothing conclusive about the complexity of the processes. In case management, repeating certain processes isn't that interesting because processes are case specific, so a process may be repeated several times for a certain case, but outside that case, the process hasn't got any repeating value. Reproducible processes are more worthwhile here because reproducible means you can reuse the process, or parts of it in other cases too.

Although, knowledge-intensive processes are clearly very important in case management, this thesis will not dig deeper into the subject. This would lead to focussing too much on a specific aspect of case management while losing the overall picture.

5.3 Characteristics

A lot of scenarios or cases in case management will share a common set of characteristics. Although the structure or architecture of processes and activities may differ from case to case, there are elements inherent to every single case. Some are more visible than others: the involvement of case workers whose job it is to manage this complex framework from start to finish. Transitions, decisions and events are all captured and stored in a central document to be used as backbone throughout the case life-cycle. Other elements and features are less visible. What follows will be an overview of the most distinct characteristics of case management mainly based on characteristics provided by Michael White (2009), Panian (2011) and Capgemini (2012) supported by insights found in other academic research.

5.3.1 Knowledge-intensive

We talked briefly about the importance of knowledge workers in case management before. Knowledge workers acquire their knowledge, skill and expertise through their experience of working with other people. Often, they will also perform similar cases multiple times and enhance their experience in a trial-and-error fashion, learning how to cope with explicit constraints, rules and the specificity of certain tasks and processes. The problem with conventional workflow approaches is that they fail to deal with the ad hoc nature of knowledge-intensive processes, frequent exceptions and habitual changes in knowledge-intensive activities (Macintosh, 1999; Allen, 2000). A challenge addressed by Michael White (2009) is how to manage knowledge and distribute it to new members, in order to keep it within the organization. This is addressed in the 'Challenges' section.

5.3.2 Information Complexity

A case is a complex composition of data, used both as driver for decisions and as a means to keep track of decisions, states and activity output (case history). It usually contains multiple supporting documents attached to it at various points in the life-cycle, making case management not only knowledge-intensive, but also content intensive. All data related to the case must be easily accessible to the case workers with the right authorisations. This is often hard to manage properly and the risk of losing important notes, documents or records is omnipresent. Present-day organisations rely on case worker experience and an adequate physical filing system to retrieve correct information (White, 2009; Panian, 2011).

5.3.3 Dynamic

Requirements change based on individual circumstances (each case is unique). It is not possible to predefine a sequence of tasks, activities or processes for a case instance to pass through (van der Aalst, Weske, & Grünbauer, 2005). Circumstances can and will change depending on the case environment, so cases need to be flexible enough to be steered in the right direction in run-time (Figure 1). However, this does not mean that every element of the case needs to be dynamic and flexible. Some elements may be fixed (e.g. lead time, budget constraints…), leading to a restriction in the variability of the execution of cases (White, 2009). Inflexible requirements are often driven by legislation and service level agreements. Cases are not only dynamic based on the tasks that have to be executed and the decisions that need to be taken but also based on how they are managed. Franke (2011) sees the overall management of cases not as a cycle but as a parallel flow. Even though cases will go from phase to phase, flowing though their individual life-cycle (just like projects or products do), managing cases happens simultaneously. A manager will need to design, execute, monitor, analyse and govern the different tasks, activities and processes within a case (Franke, 2011). Since case management is dynamic and so is its management, all those different aspects of managing (design, execute, monitor, analyse and govern) can occur at the same time. Sometimes specific aspects will occur multiple times while others may be left out.

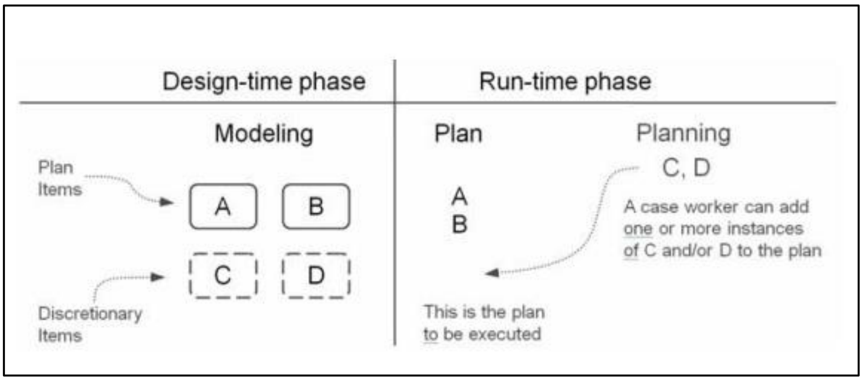

Design-time phase
Run-time phase
Modeling
Plan
Planning
Plan Items
A
B
C
D
Discretionary Items
A
B
C, D
A case worker can add one or more instances of C and/or D to the plan
This is the plan to be executed

Figure 1: Design-time modelling vs. run-time planning (Oracle, 2014)

5.3.4 External events can affect cases

It is important to understand that the dynamic nature of case management often is illustrated when external events alter the direction in which the case workers need to steer the case. Alternative paths can be chosen throughout the case life-cycle in run-time based on those events, in order to be able to address this event.

5.3.5 Long running

Cases typically run over long periods of time. This differs from customer relationship management (CRM), where case life-cycles generally run much shorter. Because of this long running life-cycles, case management is susceptible to a dynamic environment (see the 'Dynamic' feature): "Because a case is long running, it changes hands over time, different people work on different aspects and no single individual has an accurate view of the case as a whole." (White, 2009, pp. 3-4).

5.3.6 Collaborative

Case management requires the involvement of multiple users and multiple departments/groups in an organization. Often this collaboration will result in a team-based approach, consisting of a panel of experts and specialist working on different aspects of a case (Levinsen, 2004). As an extension to the knowledge-intensive characteristic, collaboration is also used to exchange knowledge between colleagues. "Collaboration is particularly important in knowledge-based case management because workers rely on each other's advice and experience when making decisions on a case." (White, 2009, p. 4).

5.3.7 Multiple participants, multiple roles

A wide range of roles, skills and responsibilities can be involved in a case. Who is involved in what stage of the case needs to be defined in the case data (White, 2009; Panian, 2011).

5.3.8 Interrelation

Cases can be interrelated. The outcome of one case can be influenced by the outcome (can also be intermediate) of another case. "Business cases can be explicitly linked or they may be linked by inference and conducted with this inferred link in mind." (Panian, 2011, p. 469).

5.3.9 Semi-structured or unstructured

Semi-structured refers to a framework where activities are somehow related to each other using a connector without the connector being a pathway constraint. It can also refer to structuring the model (e.g. in a state-based approach). Some case management models may contain ad hoc tasks which are added at various points. These activities are unstructured because they are not modelled in a relation to another activity. Nonetheless, they can be structured somewhat by using them in a state-based case management approach (see 'State-based case management' and 'Case Study: Cordys Approach to Case Management').

5.3.10 Hierarchical

There can be multiple levels of tasks and actions to be managed (Capgemini, 2012). A good example of how hierarchy can be built into case management is with the help of state-machines. This thesis will address a state-based approach to case management further in this thesis as it is an interesting approach that deserves a more detailed elaboration.

5.3.11 Security

When adding a means to track access levels and roles of case workers, case management can be used as a control tool to track authorisations over activities. In most environments, efforts in access and data manipulation restrictions will be necessary (e.g. in finance, healthcare…) making a security a must-have feature in case management.

5.3.12 Other

Michael White (2009) also takes time to clarify two other features or characteristics of case management. It was a conscious choose not to add them to the list above because they apply to the use of case management as performed by a majority of organisations today. Consequently, they illustrate features of a premature, non-modelled case management approach. Nonetheless, the two features are interesting because they give an insight into elements that are needed to be improved in a mature case management model:

- Difficulty in gaining visibility of case progress: In case management as is applied today (paper-based), the follow-up on progress and the monitoring of decisions, events, internal triggers, data-context, activity outcomes… is difficult. Especially for high-level managers, it is difficult to oversee progress on different cases, their costs, time-frame and the work distribution. Problems regarding visibility also occur with cases that are halted because they depend on an external trigger in a specific stage of the case life-cycle or when certain stages in the case life-cycle repeatedly lead to bottlenecks. The visibility of case management models will be addressed further as a benefit in '5.4 Benefits'.

- Isolated pockets of automation: Physical folders, paper documentation, spreadsheets, notes, e-mails… are the main pillars of today's case management. Legacy systems only automate portions of the entire case. As a result you get a system that consists of islands of automated processes without anything to connect them, thus losing a continuous start-to-end pathway in the case instance.

In the next section there will be an overview of the most common benefits case mangement has to offer. After having gone over the benefits, this thesis will take a look at the challenges case mangement has to deal with from a theoretical viewpoint and wrap-up with a more in-the-field approach.

5.4 Benefits

As addressed already in the introduction, research on case management is scarce and consequently benefits of case management are barely addressed in case management literature. This part will sum up the most found benefits listed by experts and organisations specialized in this matter.

Case management is widespread in the healthcare sector. One of the reasons case management originated in healthcare, apart from being heavily case-based, is because of the benefits it brings for all parties involved. There are numerous academic papers, reviews and studies on case management and its benefits in treating illness. One study showed that case management is capable to improve overall quality of care in cases where people were diagnosed with a serious illness that requires complex and/or drastic treatment (Oregon Evidence-based Practice Center, 2013). The study also demonstrated a positive correlation between case management results, outcomes and previous levels of healthcare use. This may not come as a surprise as this is in line with an elaborate trial-and-error approach, something that was already mentioned in the knowledge-intensity characteristic of case management. This results in gathering an extensive amount of data regarding the case which here also implies the patient, just as a customer would in a business environment. Case management is used as a tool to achieve client wellness and autonomy through advocacy, communication, education, identification of service resources and service facilitation (Case Management Society of America, 2010). The philosophy of the CMSA regarding case management is as follows: “The underlying premise of case management is based on the fact that, when an individual reaches the optimum level of wellness and functional capability, everyone benefits: the individuals being served, their support systems, the health care delivery systems and the various reimbursement sources. Case management services are best offered in a climate that allows direct communication between case manager, client and appropriate personnel, in order to optimize the outcome for all concerned.” (Case Management Society of America, 2009, p. 9).

Although there are a lot of comparisons to be made between a healthcare-based management approach and a business/organization-based management approach, the two are quite different. Hospitals focus on generating the most beneficiary circumstances for a patient’s wellbeing, while businesses and organisations lean on added value for stakeholders, focussing more on generating profits than on wellbeing for everybody involved. As good the intentions of a company may be, making a profit will always come first. A second point to be made is the fact that businesses generally employ a more diversified case team than hospitals do. Hospitals do have a wide range of specialists and experts available to tackle various cases but patients will almost only end up with one of them. Case workers will often work in teams, combining and sharing their know-how and skills to tackle cases. This leads to a third point: the case itself. Most cases will be repetitive in a medical environment. Rare conditions will often be identified and transferred to hospitals that are specialized in that specific illness. Cases in a business environment are far more dynamic and flexible. They are also highly mouldable depending on internal and external events.

Not only in healthcare is case management used extensively. Legal case management systems are used to support the practice of law and aid them in the transition from a paper-based to an electronic environment. For lawyers, case management can help to organise diaries and even entire firms, manage deadlines, execute mundane tasks automatically, keep track of people, coordinate communications, do data control (avoid double-entries), avoid malpractice claims and/or profits… (Bilinsky, 2002). The list keeps going and going.

However, a business environment can't be compared to a hospital or a solicitor's office. So what are the business-specific benefits a company can reap from implementing case management?

5.4.1 Improved performance and productivity

Previously unable-to-automate processes, like those who are very knowledge-intensive and dynamic in nature, can be structured or semi-structured and cast into a model. Repetitive tasks and activities will be automatically performed by the system and more variable activities will be manually managed by case workers. The result of this symbiosis is reduced completion times, customer response time, overall efforts and costs (White, 2009; Le Clair & Moore, 2009; Le Clair & Miers, 2014; IBM, 2015; Capgemini, 2012).

5.4.2 Reduction in paper-intensive tasks

where cases traditionally consisted of physical documents, notes, spreadsheets, invoices, folders, surveys… case management substitutes most with electronic files, stored in electronic files, which make up the better part of the case instance. Paper records and forms are reduced to a minimum, reducing paper-use (resulting in lower costs) and increasing effective document management (White, 2009; Miers, 2007; Le Clair & Moore, 2009).

5.4.3 Greater visibility

As already been discussed in the part on 'Characteristics', visibility was a drawback in the early case management implementations. Good case management succeeds in giving a clear overview of all the cases being processed, which case worker or team is involved, what state the case currently is in, actual of predicted costs, timeframe… With the help of user interfaces, case workers can keep up-to-date and react to bottlenecks, incidents, calamities… in real-time (White, 2009).

5.4.4 Greater consistency

Case management is characterised by knowledge-intensive business processes. A survey by Işık, Van de Bergh, & Mertens (2012) found that this kind of business processes are primarily considered repeatable by the interviewees. Based on research by Reynolds (2011), Işık et al. (2012) conclude this may be explained by a subtle difference between repeatable and reproducible knowledge-intensive business processes. In short, in case management, repeatable processes are processes that can be executed multiple times a case. Reproducible processes on the other hand are processes that can be reproduced over different cases. Repeatable parts of the case can be grouped together in benchmark processes, that allow organization to handle a larger number of cases of varying nature. This allows for far more consistency throughout the case model and a reduction in errors and failures due to lower usage of paper-based manual data-input (White, 2009; Le Clair & Moore, 2009).

5.4.5 Scalability

Scalability is a natural result of improved performance, productivity, consistency and visibility. By scalability, Michael White (2009) intends the possibility of managers to predict how many cases they can administer given a set of resources and use this prediction to be able to expand the number of cases that can be managed in an effective and efficient way.

5.4.6 Capturing and sharing process knowledge

Later on, this master dissertation will be addressing the artefact-based approach to case management where a case file will act as the central data-context to support the whole case and make monitoring efficient. Such a central data-context will avoid case workers from losing valuable information and since it is electronic it is available and accessible on multiple devices. Based on the data generated throughout the case, case workers can learn from the past, adjust processes, activities and actions accordingly while predictions and decisions can be made in real-time. Outcomes will be gathered and stored in the case file so that they, on their turn can be used as input for future decisions (White, 2009; Capgemini, 2012; IBM, 2015).

5.4.7 Manage complexity

A survey by Forrester (2009) found that "Dynamic Case Management products usually support a number of features designed to handle the complexities and multi-faceted attributes of varying types of information." (Peacock, 2011). Case management is designed to create order where there is chaos, while at the same time not neglecting its flexibility and ability to adjust to its volatile environment (Le Clair & Miers, 2014).

5.4.8 Flexibility and agility

In case management, processes have the flexibility to change at run-time. Agile case workers can then adapt accordingly, changing their approach as is required based on the newly available data. Not having this flexibility, case management would quickly fall prey to deadlocks (White, 2009; Franke, 2011; Capgemini, 2012; Le Clair & Miers, 2014; IBM, 2015). Marisa Peacock (2011) sums it up very plain: "Because the world of corporate governance is in flux, having a system that can effectively monitor changing rules and regulations, from state to state, country to country, is attractive within the enterprise." (Peacock, 2011).

5.4.9 Better coordination of existing systems

"Good case management joins up and coordinates the various supporting systems, reducing the need for repeated data entry and manual hand-offs." (White, 2009, p. 11). In this regard, case management can be seen as a sort of Enterprise Resource Management (ERP) for modelling.

5.4.10 Business case control

"Strong design time case management emphasizes control over highly repeatable processes. Strong runtime case management supports use cases where the ultimate paths are highly variable." (Le Clair & Miers, 2014, p. 1). Control is obviously also a consequence of the other benefits. Flexible processes and agile case workers have more grip on external and internal circumstances, increasing their overall control on the case. Coordination between systems allows for more accurate and detailed inputs, making sure that system and the case workers get a better understanding of what is happening. So does capturing and sharing knowledge. Also, according to the Forrester report on Dynamic Case Management (Le Clair & Moore, 2009) visibility can theoretically increase control.

5.4.11 More effective collaboration

Collaboration is a feature and benefit of case management often found in research and academic literature. The reason is very simple: collaboration is a distinct and very important characteristic of case management. It is highly interconnected with sharing of knowledge and information throughout the case (Le Clair & Moore, 2009; White, 2009; Capgemini, 2012; Le Clair & Miers, 2014).

It is important to note that this list containing benefits of case management is not exhaustive. It offers a clear insight into the advantages of case management and subsequently it helps to understand why companies are interested and willing to invest in case management. After addressing the benefits, this thesis will go over the challenges businesses may encounter when incorporating case management.

5.5 Challenges

Investing in new software will always be a dilemma for a lot of companies and organisations because there are significant challenges and risks connected to their implementation. With a worldwide market share of $25.4 billion in 2012, ERP software is strongly imbedded into today's business management (Pang, Dharmasthira, Eschinger, Motoyoshi, & Brant, 2013). ERP doesn't need much of an introduction, the software is vastly known among companies and its benefits are unquestionable. Yet, there are still a lot of companies out there that are suspicious to incorporate ERP into their own business. There is research dedicated to the key risks of ERP-implementation and its critical success factors (Sumner, 2000; Khvalev, 2004; Hakim & Hakim, 2010). Although ERP and case management are very different in the way they function and add value to the company, the reasons for companies for not being eager to implement them are fundamentally the same. It will always be a difficulty for suppliers of business software to persuade companies into taking the plunge.

Michael White (2009) and Zeljko Panian (2011) give a good overview of the challenges regarding case management. The thesis try to augment their findings with other research and recent conclusions regarding the topic, provided by experts in the field:

5.5.1 Striking balance between practice and procedures

“There are many aspects of processes that must be coordinated to achieve excellence: people, systems, rules, policies, sequences (flows), documents, decisions and others.” (Hill, 2007, p. 4). It is already known from previous sections that case management is characterised by highly knowledge-intensive business processes, where aforementioned aspects are mostly unpredictable and depend on the judgement performed by knowledge workers. According to Meirs (2006) and Harmon (2006), jobs can be scaled on a continuum ranging from ‘defined procedures’ to ‘discretionary practices’. This thesis is not going to go deeper into the continuum as this is not relevant. People interested should check out Meirs’ (2006) and Harmon (2006)’s research and Micheal White’s (2009) visualization of the continuum on page 9 of ‘Case Management: Combining Knowledge with Process’. What this continuum can shows is that managers, modellers and designers have to be very careful not to over-proceduralise processes. Proceduralising knowledge work to much without balancing it out with more practice-based work in case management will result in possible deadlocks in situations where processes don’t match the predefined procedures. These deadlocks may lead to workers trying to work around the procedures in order to avoid them. Balance is of utmost importance.

5.5.2 Sharing implicit rules and tacit knowledge

Knowledge workers will often apply knowledge or expertise that is not easily transferred between people (Liebowitz & Wilcox, 1997). This kind of knowledge is called tacit or automated knowledge and is not well supported by formal training (Sternberg, et al., 1999). “Tacit knowledge is made up of best practices, experience, wisdom and unrecordable intellectual property that lives within individuals and teams. Since tacit knowledge exists within minds, it cannot be reduced to the digital domain as a material asset, or be manipulated directly. However, it expresses in the social realm as the response ability of individuals (productivity, innovation and initiative), and teamwork (communication, coordination and collaboration).” (CDS, 2003). Knowledge workers will apply tacit knowledge to manage and govern processes with support of explicit procedures and policies. Tacit knowledge is very valuable for businesses, but according to Kingston (2012), it is often futile trying to capture it. Instead, organisations should make an effort trying to share tacit knowledge between employees using approaches that don't require capturing knowledge (Kingston, 2012). An example of a non-capture approach to sharing tacit knowledge is training (Milton, 2010). Implicit rules are the unspoken rules that are common within the organisation. How decision-making is done in different stages, how employees cope with certain circumstances and exceptions or how knwoledge workers deal with unstructured knowledge are good examples of implicit rules (White, 2009; Panian, 2011). The big challenge with implicit rules and knowledge will be to capture and structure what is possible to capture and structure and try to automate it accordingly, while focussing on sharing the remainder of implicit knowledge and rules.

5.5.3 Formalizing experience – supporting learning

When case workers apply their knowledge to guide cases through a number of processes, they fine-tune certain skills and build experience. These skills and experiences in their turn will help the case worker to execute future processes more efficiently or respond more effectively to exceptions or external events in the environment. A good case management system should support this kind of learning by formalizing previous experiences into the definition of new processes and procedures or by offering better online help (Panian, 2004; White, 2009). Case-based Reasoning (CBR) is closely related to this approach. In CBR, previously attained experiences are used to understand and solve new problems. Bassically, experiences build when solving old cases are used to asses new cases and solve them based on the knowledge of the previous case(s), similar to how an attorney would use precedented cases for constructing and justifying arguments in new cases (Kolodner, 1992). CBR is based on two principles, elaborated by David Leake (1996): "similar problems have similar solutions" and "the types of problems an agent encounters tend to recur" (Leake, 1996, p. 1). The first principle supports the idea that a combination of solutions for similar problems that occurred previously, are a good basis for virgin problem-solving. The second principle simply says that similar problems will keep occur in future cases. So if a problem occurs, you can guarantee it will happen again somewhere in the future.The CBR- cycle on the right (Figure 2) depicts the four task-driven phases found in CBR (Aamodt & Plaza, 1994). Each phase shows a task the reasoner has to achieve in order to attain full case understanding and apply a suiting

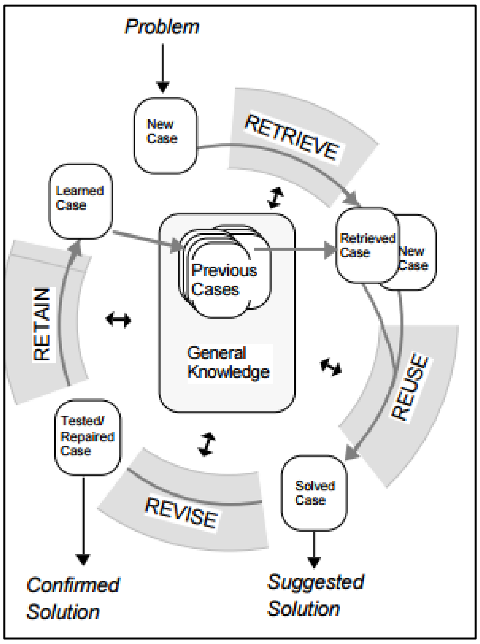


Figure 2: The CBR-cycle (Aamodt & Plaza, 1994, p. 8)

solution. The case-based reasoning approach is an interesting and intuitive concept for businesses to use in a case management system and to improve overall learning of cases.

5.5.4 Supporting ad hoc change

I already spoke about the dynamic character of case management. It is possible to define processes and activities at design-time but this does not make it possible to analyse or define cases at run-time. Every case management system should take into account the unpredictability of cases and their processes and possible changes in the case's environment. This often sounds easier than it is in reality. It requires knowledge (both explicit and tacit) and certain skills, as already has been discussed in previous points. The dynamic nature and the ability and skill of case workers to cope with ad hoc change really define the strength of case management. Being able to react to unpredictability and change is therefore both one of case management's most distinct characteristics and at the same time one of its most pressing challenges.

5.5.5 Involving participants in the design of knowledge processes

Being able to respond to events quickly requires the intervention of case workers, making changes to processes where they see fit (White, 2009; Panian, 2011). Although not everybody should have the power to alter processes in a case, case workers that are authorised to do so should have full support of management and the management should have full confidence in their ability to react to events appropriately.

5.5.6 Supporting collaboration

In order for case workers to be able to perform their job efficiently they have to be able to access the right information regarding the case at the right time. Since case workers will often cooperate when working on cases, there has to be an instrument to enhance communication between them so they can share case history, discussions, correspondence and previous decisions (Panian, 2011). But it is not just a matter of enabling instant messaging or document sharing. Case management should be an intelligent system, knowing what info to provide when (White, 2009). At the same time, all irrelevant information should be withheld from the case worker to avoid confusion and inefficient decision-making. Confidential information should only be accessible to those people that are authorised. Supporting collaboration between case workers can possibly make them benefit from the experience of others. In this aspect, supporting collaboration should be considered connected to supporting learning. As Margret Wallace (2012) states, participants can rely on the benefits from the knowledge of more experienced co-workers. In projects where less experienced workers were put together with a more experienced worker, the less experienced workers learned how to better approach certain tasks (Wallace, 2012).

5.5.7 Support decisions

When talking about 'Involving participants in the design of knowledge processes' (5.5.5) previously, this thesis already mentioned the importance of confidence and support by the management. This is paramount in letting case workers make right decisions quickly. Next to giving support and confidence, rules can be used to support decision-making too (Panian, 2011). Yet, people tend to go for full-out automation and as a result they often fall prey to systems that lapse into rule-based complexity. In case management rules are there to support decision-making. Control should always stay in the hands of the case workers (Panian, 2011; White, 2009).

5.5.8 Effectively coordinating participants

Just like the right information needs to be provided to case workers at the right time, the appropriate tasks need to be directed to the right case workers. This point is strongly related to 'Supporting collaboration' (5.5.6), because redirecting tasks to other case workers implies informing them of the case specifics using the right information. Case management will need to support coordination through using sophisticated workflows, synchronising process flows while making sure milestones are monitored, controlled and met. Coordination of participants will also require supporting and identifying possible delays and/or disruptions in workflows (White, 2009; Panian, 2011).

5.5.9 Managing complexity

Since case management requires so much input of internal and external information, things can become very complicated very quickly. Companies implementing case management should really focus on structuring and organising data efficiently so case workers don't become overwhelmed or confused of all the information coming at them (White, 2009; Panian, 2011). A recent comment by Peter Whibley (2015) on a blog post about the challenges today in regards to case management identified the complexity, not of the information, but of the design and integration. According to Peter Whibley (2015), the complexity will only increase as the number of business applications in organisations keeps rising. It will be a huge challenge for organisations to integrate the complex web of applications, legacy systems and new systems and create an environment where all those different applications work together as a whole to add value for the organisation (White, 2009; Panian, 2011; Capgemini, 2012).

5.5.10 Managing artefacts

Related to managing the complexity of information, managing artefacts focusses on storing data in a structured and secure ways. When talking about artefacts, the central case file is seen as the artefact that forms the backbone of our case. It contains all the information case workers need to take the right decisions. Artefacts are addressed in more detail in Chapter 6 'Different approaches to modelling case management' under 'Artefact-based process control' (6.3), an approach to modelling case management. Since case artefacts are used in case management to store and retrieve information, they will often be supported by Content Management Systems (CMS) or Electronic Record Management Systems (ERMS). In this regard, managing artefacts will also relate to the complexity of integrating different systems (Panian, 2011; White, 2009).

5.5.11 Other

A remarkable problem today, as learned from consulting the blog concerning today's biggest challenges for case management (Schooff, 2015), is very different in nature. Not the case management systems seem to be the problem but the way in which they are advertised and offered by vendors. All businesses have both structured and unstructured processes, but want only one system to manage them both. It is up to the vendors to mould this complexity into their applications, without selling enhanced workflow management systems in the lower end of the market as mature case management systems. Other problems mentioned in the blog (Schooff, 2015) are the lack of a uniform definition for case management and consequently, the confusion that has been created in the market, a polished standard notation (fine-tuning of the Case Management Modelling Notation 1.0 beta version), making the standard notation easy to communicate towards customers and the flexibility and creativity in designing case plans and files (Schooff, 2015).

In 'Managing Chaos: unstructured processes and dynamic BPM' Capgemini (2012) lists other possible challenges. Lack of feedback, not only within the company between case workers and managers, but also towards customers and other external stakeholders. Connecting with customers is something that will need attention in future case management. Case management will need to offer a consistent way of monitoring and managing customer complaints, else complaints will build up and as a result the number of discontented customers and stakeholders will increase, leading to a possible loss of business. Using an electronic system to gather and store data (often of customers) means that organisations will need to meet the accepted regulations on data storage and use. With the emergence of mobile applications, these regulations may proof to be difficult for organisations using case management systems (Capgemini, 2012).

5.6 Case management vs. Business Process Management

Case management cannot be seen as a standalone specification, but as a complement on existing concepts like Business Process Management (BPM) and Enterprise Content Management (ECM). In recent years BPM has really grown out as a worldwide tool for design, enactment, management and analysis of predetermined (predictable) business processes (see van der Aalst, 2013). MarketsandMarkets (2014) predicts an exponential BPM market growth of $4.71 billion in 2014 to $10.73 billion by 2019. Gartner states that "by 2016, 60% of BPM initiatives will fail unless they can demonstrate direct positive impact on organisational strategy." (Gartner Inc., 2013). Since in today's business environment, knowledge and unpredictability become more and more important, the focus on case management will grow. As already mentioned above, businesses operate in volatile environments,and thus can not only function through sequential workflows. In 'Case Management: A Review of Modelling Approaches' (de Man, 2009a), a few elements are highlighted to illustrate why certain processes (case management processes) can't be modeled the same way it has been done with sequential workflows (like BPMN). These elements basically depict the basic differences between case management process modelling and business process modelling (de Man, 2009a):

- A case file is the central artefact, it creates context throughout all activities in the case. BPMN lacks a similar artefact.
- Case management is event- and data-driven. It is difficult to agree with Henk de Man's (2009a) point of view when he considers case mangement merely as event-based. Being event-driven means that events can trigger processes, activities and tasks in whatever sequence. In case mangement the data, that is gathered and stored in the case artefact, helps case workers in their decision-taking abilities and new processes, tasks and activities will be triggered based on the case workers evaluation of that data. BPMN 2.0 allows event-based elements (gateways, subprocess) with the difference that a process still is excecuted in a predefined sequence and overall. BPMN 2.0 also allows data-based exclusive gateways to insert the possibility to check the data based on an expressed condition (Tscheschner, 2008). Where case workers take decisions based on their knowledge, skill and intuition, data-based exclusive gateways will automate this decision-making based on predefined conditions.
- Case management is also very collaborative (because the case is collaborative). In a way, BPMN is collaborative too because different elements (illustrated by swimlanes) can be part of the same process but that collaboration can be seen as mechanic, since it is all part of a predefined sequence whereas the collaboration in case management is ad hoc, situation dependent and thus far more unpredictable and flexible.
- Case management is based on the imporatance of human workers, also called knowledge workers. They are not supposed to just execute certain tasks in a sequence as they are in BPMN. They benefit the freedom and expertise to be creative in the way they handle certain situations (they are "not restricted to what should be done, but should rather be guided by what can be done" (de Man, 2009a, p. 3)). As a consequence, these knowledge workers can have major influence on a process.

Joern Franke (2011) adds additional differences between BPM and case management:

- In BPM, business assets only played a minor role in regards to the execution and modelling of processes. In case management, those business assets are getting much more attention. Cohn and Hull (2009) talk about the data manipulated by processes focussed on acitvity-flows (like BPMN). In these processes, data, like the business assets, is seen as a second-citizen throughout the process and is not really taken into account. Example of the importance of business assets in cases: “An insurance case governs all insurance contracts, incidents, processes, rules, regulations, data, customer, people related to it. Thus, an insurance case contains important business assets of any insurance company.” (Franke, 2011).

- Case management is characterised by a strong integration with Business Rules Management (BRM). A BRM system (BRMS) is defined by IBM as “a system that enables organizational policies – and the operational decisions associated with those policies – to be defined, deployed, monitored and maintained separately from core application code.” (IBM, 2015). According to Schlosser et al. (2014) one of the prominent drivers for BRM is the need for organizational agility in order to allow fast and ad hoc decision-making, hence the importance in case management.

Business processes modelled in BPM and case mangement are fundamentally different. BPM tries to automate a stable, predefined sequence of activities and tasks, case management focusses on unpredictable ad hoc processes. Although they are very different, they both try to achieve the same thing: increase the efficiency of the process flow. Because they try to achieve the same goal, and both systems focus on a different aspect of the process spectrum, companies are experimenting to combine them into a system that is capable of managing both structured and unstructured processes: the BPM-based case management approach was born. The next section will look into the different approaches that have been used to cope with the unpredictibility of processes. I’ll finish the overview of modelling apporaches with the BPM-based approach to case management.

6 Different approaches to modelling case management

We now have identified some of the most common benefits and challenges case management can throw at us. Still, this gives no support whatsoever on how to implement case management into the operational activities of an organisation. This chapter is meant to give an overview of the different approaches to case management in the literature. These approaches will introduce different concepts which have been used to address the difficulties of modelling unpredictability. The next section will start off with giving an overview of how case management was approached in the past.

6.1 Case management in the past

In the past there have been multiple attempts to model case management. The main approaches to support case management have been custom-built applications, Customer Relationship Management (CRM) systems and Electronic Document, Record Management Systems (EDRMS) (de Man, 2009a) and Enterprise Content Management (ECM). With the help of Michael White’s (2009) analysis, I’ll address the reason why these approaches are not appropriate to be used as support for case management.

Custom-built applications assume that it is doable to define all the possible paths in the system, the sequence of execution, beforehand. Hence, this kind of applications makes it difficult to support ad hoc processes and the dynamic characteristic of case management. Another reason why custom-built applications are not very effective as case management systems has to do with the business logic. The business logic will be buried deep into the software source code rendering it inaccessible for case workers. Case workers should have access to this code so they can not only maintain the system but also alter its code when needed to (White, 2009).

The reason why Customer Relationship Management systems aren't suited for case management support is partly because of the characteristics of the interactions and partly because the underlying process logic is, just like with custom-built applications, buried within the technology so case workers cannot access, manipulate and change them. CRM-cases typically have a much shorter life-cycle (days/weeks vs. months/years with case management cases). They also come in much higher volumes, often focussing more on the individual instead of focussing on multiple external parties at once. The most important reason why CRM-systems are not fit to be used as a valuable case management system is their inability to cope with ad hoc processes and change that occurs throughout the case life-cycle (Miers, 2007). Although CRM-systems obviously fail to deliver the functions to be used as a case management system and fail to meet the totality of case management requirements, it is clear that it has a certain supporting value (e.g. sorting customer data). Electronic Document and record Management Systems (EDRMS) can play a crucial part in case management. Just like Enterprise Content Management systems they are used to control critical business data, streamline information access and link structured and un-structured content to be able to perform efficient decision-making (ECM², 2015). However, they have to deal with the same problem as CRM-systems. They are too limited to be used as a standalone system, focussing only on the gathering, storage and management of information related to the case. They do not possess the functionalities to manage dynamic processes and the complexity of human-interaction (White, 2009).

While demonstrating systems that have been used in the past (and some will still use them today) as a means to manage cases, there are some concepts that are more theoretical and have been used throughout the years by academics and researchers to try and build a case management model. The thesis will give an overview of those different approaches in the next part of this master thesis.

6.2 Process Control

When trying to model a certain case, efforts should be focussed on the *regular* and *special cases*. *Mass cases* can be sufficiently modelled using a workflow model like BPMN. De Man (2009a) presents a structured overview of different approaches of process control. Process control is defined by PAControl (2006) as follows: "Process control refers to the methods that are used to control process variables when manufacturing a product." When linking process control to case management it may be better to look at process control as a way to actively change a process based on the results of process monitoring (Natrella, et al., 2010). In the next three sections there will be a brief summary of these approaches and stress the most important elements with attention to case management.

In general, three paradigms can be identified regarding process control based on research conducted by Berry (1998), Manolescu (2001) and Manolescu (2002):

- *Activity-based paradigm*: This paradigm focuses on a sequence form activity to activity. Examples of activity-based process-control are BPMN, BPEL and XPDL.
- *Communication-based paradigm*: In this paradigm, goals are achieved by the interaction between process participants. There is no explicit modelling of tasks performed by individuals.
- *Artefact-based paradigm*: Activities are defined in the context of a business artefact. Data-events act as triggers to execute those activities. Execution of activities can be restricted by business rules linked to the business artefact. Bhattacharya et al. (2005) and Bhattacharya et al. (2007) distinguish four aspects of artefact-based process-control:
    - Records used to store information dependent on the business context
    - Distinct life-cycle from creation to completion
    - Unique identifier that allows identification of an artefact across the enterprise (for tracking purposes)
    - Records meaningful to the business user

  The main purpose of artefact-based process-control is to focus on the core entities within a business that are paramount to business users and to track current organizational goals.

6.3 Artefact-based process control

After going over the specifications of the different paradigms it is clear that case management leans most towards artefact-based process-control. When going over the differences between case management and BPM this thesis already mentioned the case file as being the artefact in the system. There are several approaches to case management that use artefacts as a shared context that is passed through between processes.

6.3.1 State-based case management

This concept focuses on state transitions and business rules that lead to the execution of the business artefact. A sequence of activities is dependent on the artefact and is consequently very flexible (Bhattacharya et al., 2007). In line with Bhattacharya et al. (2007) lies research conducted by Cohn and Hull (2009): “Artefacts are business-relevant objects that are created, evolved, and (typically) archived as they pass through a business.” (Cohn & Hull, 2009, p. 2) They expand the concept of artefact-based process control by including two models typically found in artefact types: an information model to keep track of data of business objects and a life cycle model to describe how and when certain tasks can be executed regarding the business objects (Cohn & Hull, 2009). Ferguson and Stockton (2005) worked out an artefact-based process control with use of state machines, similar to the state transitions used by Bhattacharya (2005 and 2007) (see Figure 3 and Figure 4 below).

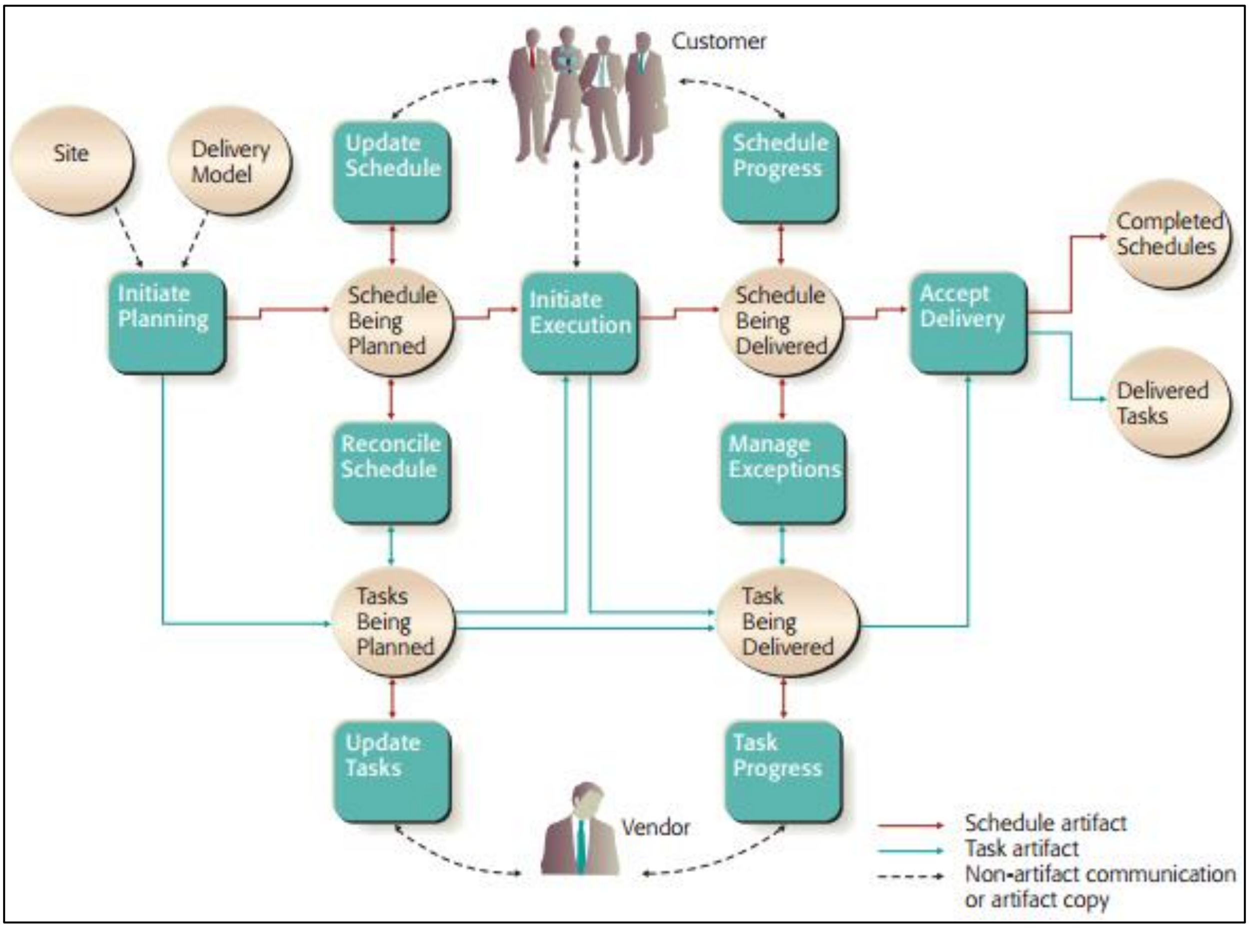


Figure 3: Operational model for schedule and task artefacts (Bhattacharya, Caswell, Kumaran, Nigam, & Wu, 2007, p. 716)

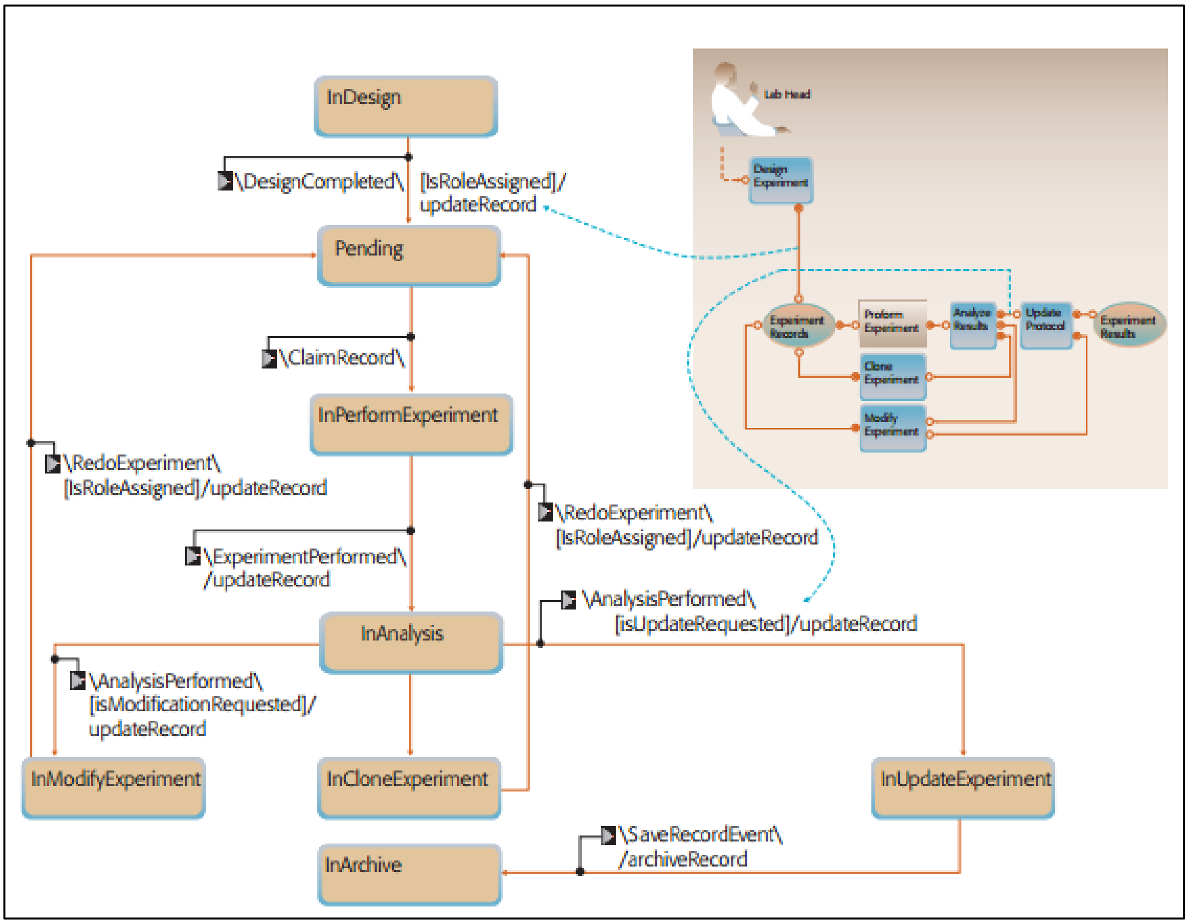

InDesign
\DesignCompleted\ [IsRoleAssigned]/ updateRecord
Pending
\ClaimRecord\
InPerformExperiment
\RedoExperiment\ [IsRoleAssigned]/updateRecord
\ExperimentPerformed\ /updateRecord
\RedoExperiment\ [IsRoleAssigned]/updateRecord
\AnalysisPerformed\ [isUpdateRequested]/updateRecord
InAnalysis
\AnalysisPerformed\ [isModificationRequested]/ updateRecord
InModifyExperiment
InCloneExperiment
InUpdateExperiment
\SaveRecordEvent\ /archiveRecord
InArchive
Lab Head
Design Experiment
Experiment Records
Proform Experiment
Analyze Results
Update Protocol
Experiment Results
Clone Experiment
Modify Experiment

Figure 4 : Mapping of the experiment record life cycle to the finite state machine (Bhattacharya, et al., 2005, p. 155)

State-machines are to be situated in object-oriented systems, where states are identified for a certain class. A state can be defined as “a set of values that describes an object at a specific point in time, and it represents a point in an object’s life in which is satisfies some condition, performs some action or waits on something to happen.” (Tegarden, Dennis, & Wixom, 2013, p. 255). In this case, the object that is being described is the case itself and its position in the life-cycle, defined by its unique values and features stored in the case file. Strahonja (2005), also makes use of state-machines to model case management. Strahonja (2006) utilizes UML (Unified Modelling Language) as a modelling tool (Figure 5). Research suggests that the use of BPMN and UML does not differ regarding end-user usability/readability (Peixoto et al., 2008).

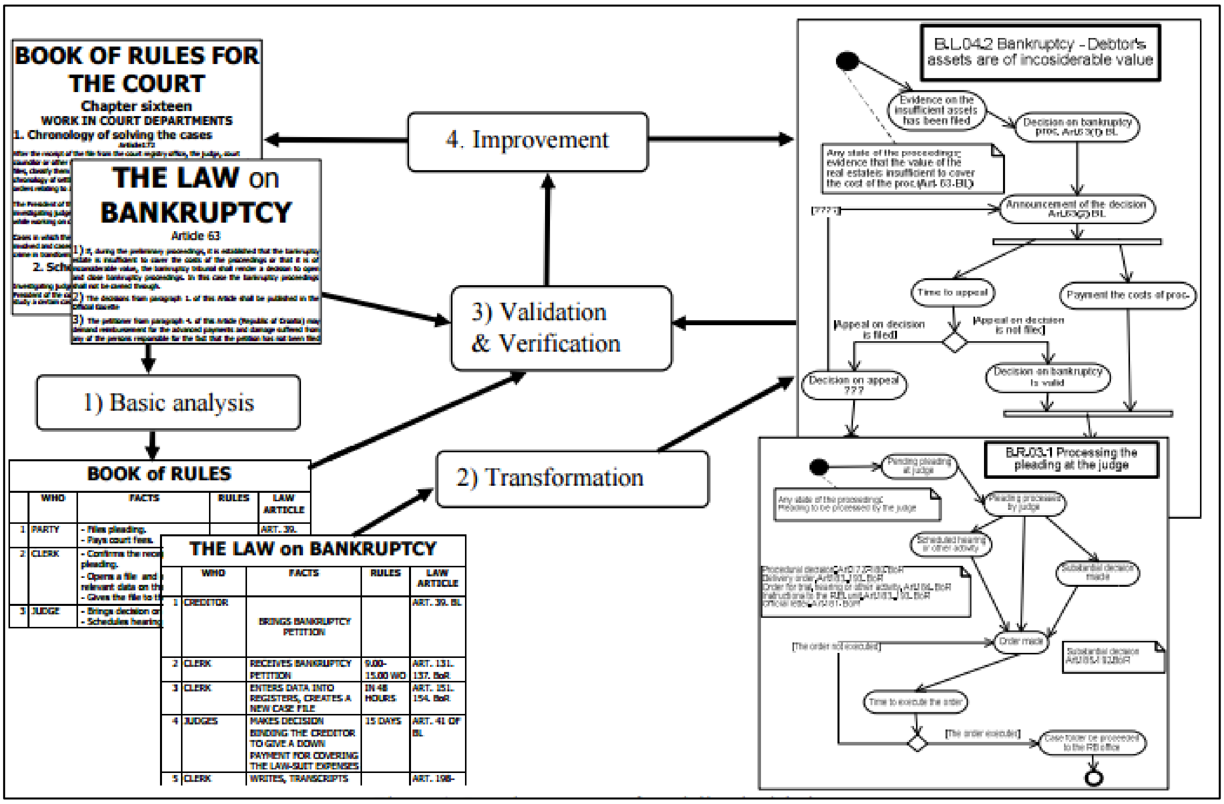

BOOK OF RULES FOR THE COURT
Chapter sixteen
WORK IN COURT DEPARTMENTS
1. Chronology of solving the cases
THE LAW on BANKRUPTCY
Article 63
4. Improvement
3) Validation & Verification
1) Basic analysis
2) Transformation
BOOK of RULES
THE LAW on BANKRUPTCY
WHO
FACTS
RULES
LAW ARTICLE
B.L.04.2 Bankruptcy - Debtor's assets are of incosiderable value
B.R.03.1 Processing the pleading at the judge

Figure 5: Iterative process of modelling legislation (Strahonja, 2006, p. 2)

6.3.2 Case Handling

Van der Aalst et al. (2005) came up with a hybrid approach to case management, called Case Handling, where a logical flow still exists, similar to Bhattacharya et al. (2007), but the individuals in the process, the case workers, have the ability to skip activities and return to them allowing the level of flexibility that comes with case management.

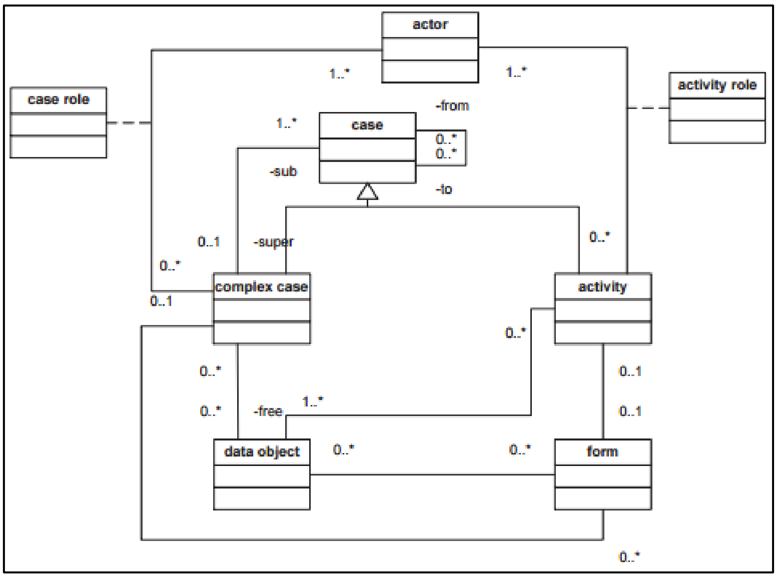
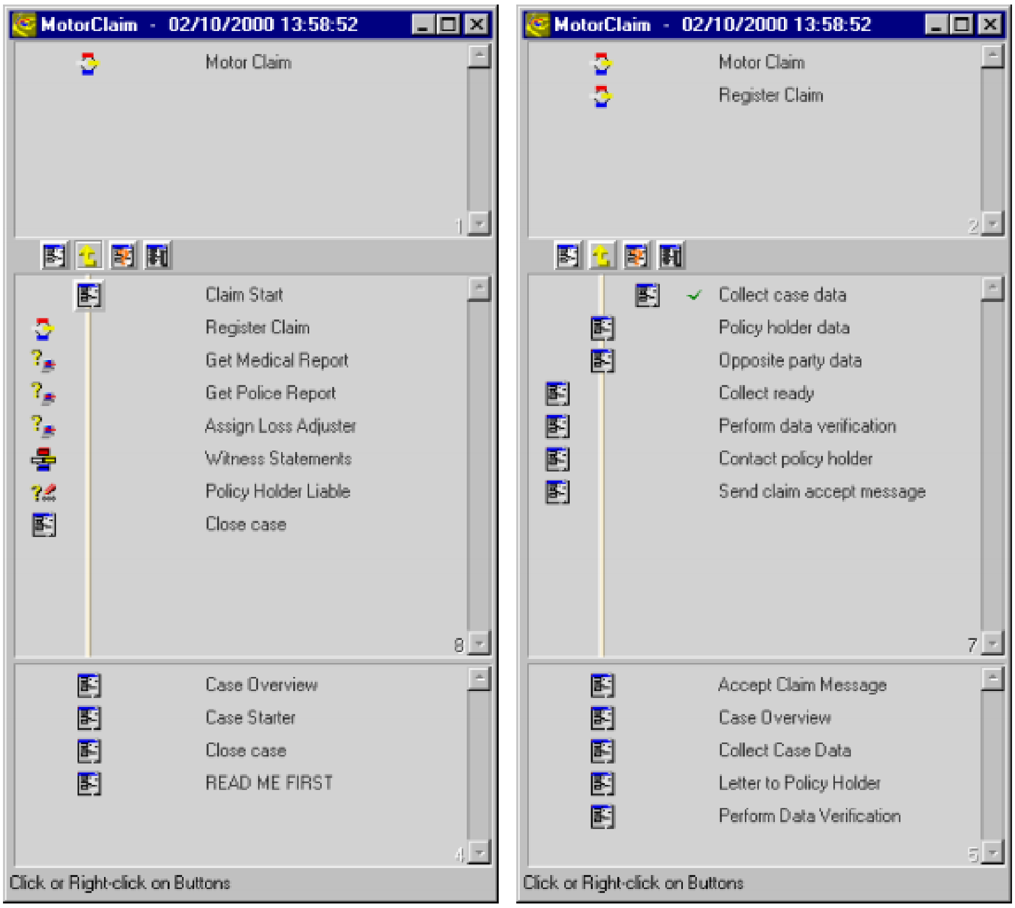


Figure 6: Case handling meta model, instance level (van der Aalst, Weske, & Grünbauer, 2005, p. 140)

Figure 6 shows a meta-model of a Case Handling instance. Case handling's strength lies in its ability to offer flexibility to processes by putting the emphasis more on data and not so much on the control-flow aspects of a process (Billington, Jensen, & Koutny, 2009). Pallas Athena made an attempt to model case-handling and that resulted into a commercial workflow management system called FLOWer. It succeeds in offering organizations mechanisms to react to change in the environment. They call this 'operational flexibility'. Before FLOWer, standard workflow systems didn't allow for a flexible approach of business processes. Exceptions needed to be modelled explicitly and the process control had to be done by a different employee for every other step in the process. Case handling offers the employees insight into the whole context of the case so that employees can execute all the tasks that are necessary.

The FLOWer Case Guide (Figure 7) is an illustration of a whole case. The top shows the context of the case, in our example this is a 'Motor claim'. Underneath, the different sub-processes and decisions are shown that are part of the 'Motor claim'. Processes and decisions don't need to be executed in a predefined order; workers can choose what and what not execute. At the bottom, all the forms are shown. Forms can be opened when initiating a certain activity in the box above. In our example, the activity 'Claim Start' can be triggered to open the first form. When this activity is completed, the activity moves from the status line to the right as illustrated in Figure 7(b). The next activity on the left would be to register the claim. The icon on the left of 'Register Claim' means this isn't actually an activity but a sub-process. Figure 7(b) depicts the expansion of this sub-process (van der Aalst & Berens, 2001). For more information on the specifics of Pallas Athena's FLOWer system, please consult van der Aalst & Berens (2001) who give a good overview of its functionalities and possibilities.



MotorClaim - 02/10/2000 13:58:52

Motor Claim

1

Claim Start
Register Claim
Get Medical Report
Get Police Report
Assign Loss Adjuster
Witness Statements
Policy Holder Liable
Close case

8

Case Overview
Case Starter
Close case
READ ME FIRST

4

Click or Right-click on Buttons

MotorClaim - 02/10/2000 13:58:52

Motor Claim
Register Claim

2

✓ Collect case data
Policy holder data
Opposite party data
Collect ready
Perform data verification
Contact policy holder
Send claim accept message

7

Accept Claim Message
Case Overview
Collect Case Data
Letter to Policy Holder
Perform Data Verification

5

Click or Right-click on Buttons

Figure 7: Example of the FLOWer case guide (van der Aalst & Berens, 2001, p. 6)

6.3.3 Product-based flow design

Product-based flow design is another take on case management approaches. Here, Vanderfeesten et al. (2006) see a case as a product that is manufactured in a process.

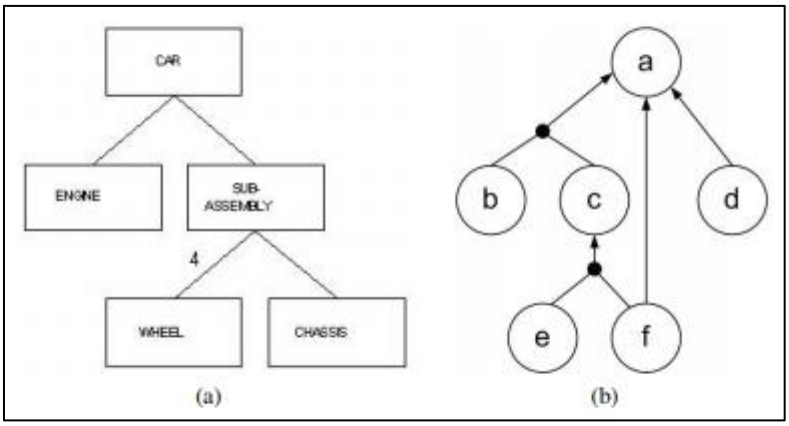


Figure 8: (a) The Bill of Material (BoM) of a car. (b) The product data model which represents the decision on the suitability to become a helicopter pilot. The meaning of the elements is as follows: (a) decision for suitability to become a helicopter pilot, (b) psychological fitness, (c) physical fitness, (d) latest result of suitability test in the previous two years, (e) quality of reflexes, (f) quality of eye-sight. (Vanderfeesten, Reijers, & van der Aalst, 2006, p. 190)

The data structure of the case is comparable with that of a so called Bill-Of-Material, a list of the parts or components that are required to build a product (TechTarget, n.d.), so that data in a case can be seen as parts used in manufacturing. Tasks, on their turn, can be seen as assembly operations through which these data elements are mounted into the case. In a Product Data Model (PDA) (seen simplified in Figure 8(b)) there are no physical parts that have to be assembled. Instead, they are data elements that have to be processes to achieve new data (Vanderfeesten, Reijers, & van der Aalst, 2006). The similarity with manufacturing assembly is striking, and can be illustrated by Manufacturing Execution Systems (MES) that were introduced to tackle the constraints of ERP (Enterprise Resource Planning) systems and to offer human-workers increased support (de Man, 2009a). Knowledge workers need an even higher degree of support in human-driven business processes than shop-floor workers do. In order to transform a product-based design model (Figure 9) into a process model, data elements or operations should get mapped and specified to the systems concept properly so that the product-based design model can be translated into the system. Those data elements and operations should also be grouped and be defined as activities. It is very important to define the correct sequence of activities because the model depends on precedence relationships (Vanderfeesten, Reijers, & van der Aalst, 2006). Before the model can be translated into a system, it should also define information on resources, conditions or activity duration (Vanderfeesten, Reijers, & van der Aalst, 2006). For a detailed study on product-based workflow design, please consult ‘Design and Control of Workflow Processes’ by Reijers (2002).

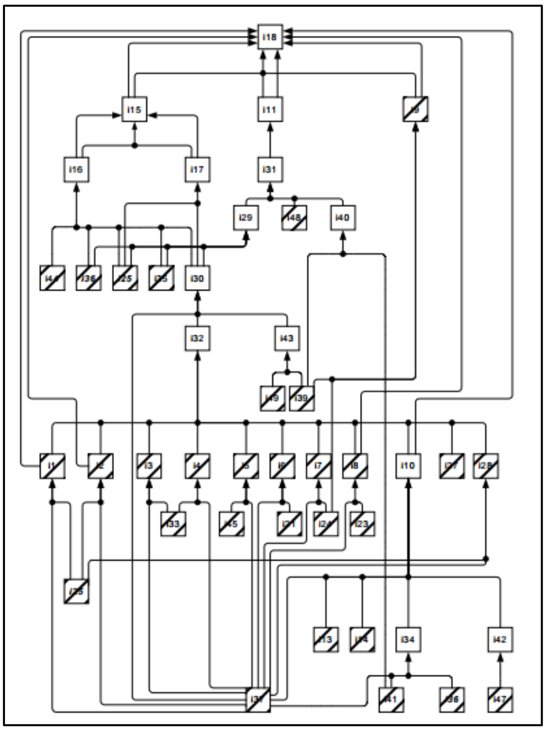
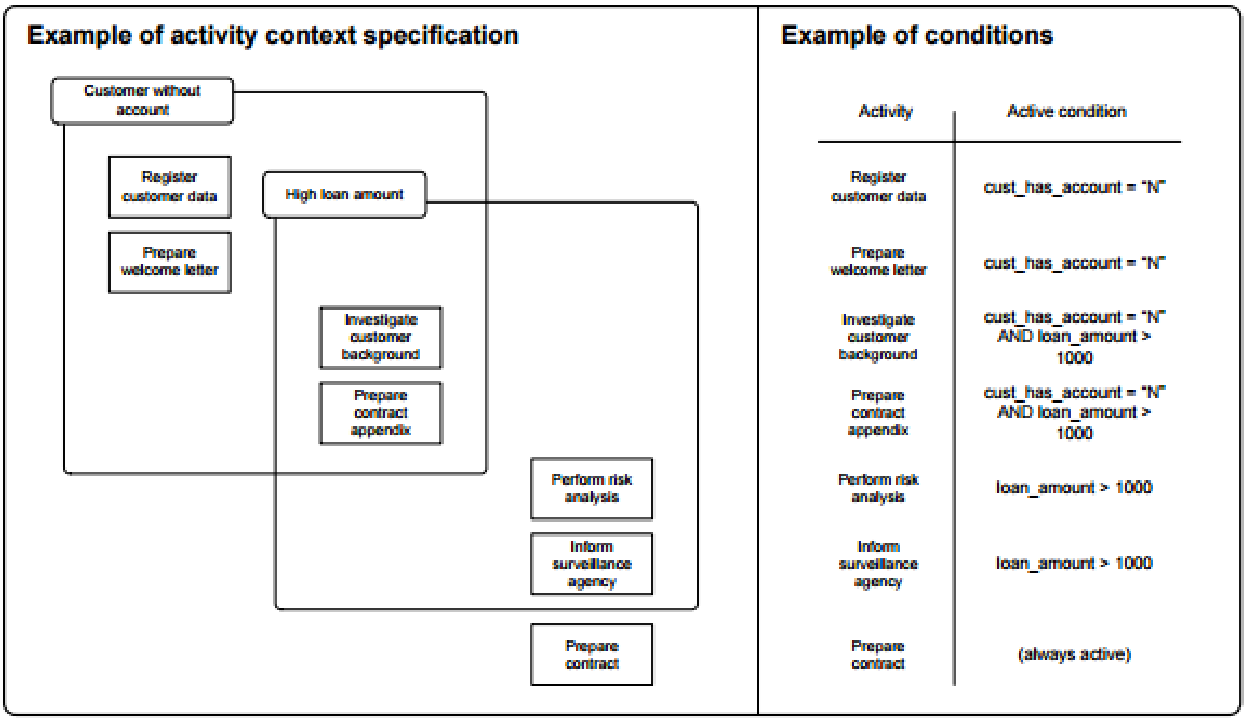


Figure 9: Product-based design model based on the GAK case (Reijers, 2002, p. 212)

It is safe to say that because of this, the proposed concept by Vanderfeesten et al. (2006) may need more artefact-based and less activity-based processes in case management. In the Cordys Approach, Henk de Man (2009b) revisits the resemblance between case worker support and shop-floor worker support. Furthermore, this concept has already been implemented in FLOWer.

6.3.4 Venn-diagram approach

Kaan (2005) uses venn-diagrams (Figure 10) in order to group activities based on certain conditions that depict their dependency on rules. This way of working probably is desirable when working with a control flow that isn't dependent on many rules, because that could complicate the diagram dramatically, thus rendering it unusable. Figure 11 gives a graphical summary of how a venn-diagram model came to be according to Kaan (2005).



Activity	Active condition
Register customer data	cust_has_account = "N"
Prepare welcome letter	cust_has_account = "N"
Investigate customer background	cust_has_account = "N" AND loan_amount > 1000
Prepare contract appendix	cust_has_account = "N" AND loan_amount > 1000
Perform risk analysis	loan_amount > 1000
Inform surveillance agency	loan_amount > 1000
Prepare contract	(always active)

Figure 10: Example of using contexts to specify the conditions of activities (Kaan, 2005, p. 28)

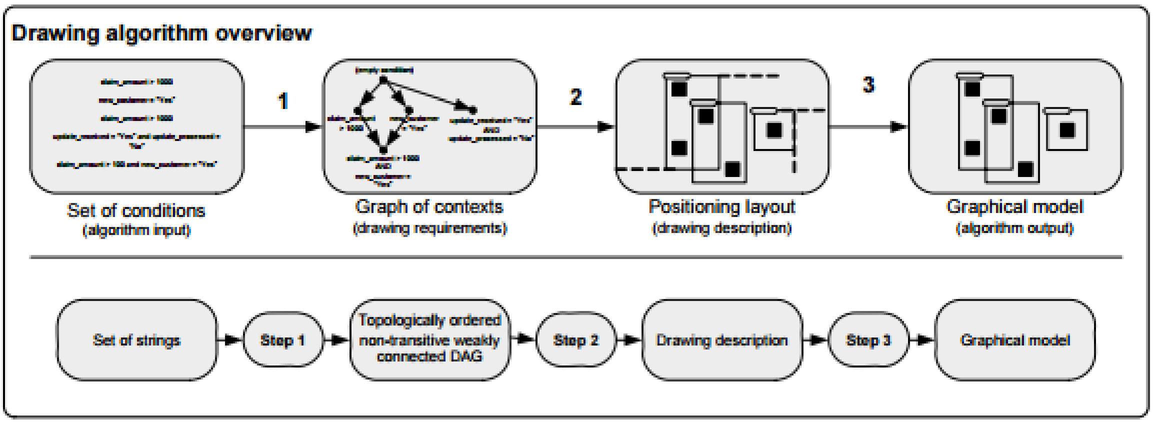


Figure 11: Four steps in drawing a graphical model from a set of conditions (Kaan, 2005, p. 33)

6.3.5 Workflow-based Architecture to support Scientific Applications (WASA)

Medeiros et al. (1995) described a concept whereby activities can be individually selected and performed by knowledge-workers, depending on certain constraint present in the environment. This concept is especially interesting for scientific experiments where knowledge-workers have to evaluate restrictions and possibilities in the environment and tune their actions accordingly. Basically, this concept allows a system to make several suggestions based on data and the users (knowledge-workers) can interpret these suggestions freely, applying or denying them as they see fit. Although this concept offers possibilities in certain areas of expertise, a proposal to cast this into a process model has not been made. Figure 12 gives you a look into the architecture of WASA.

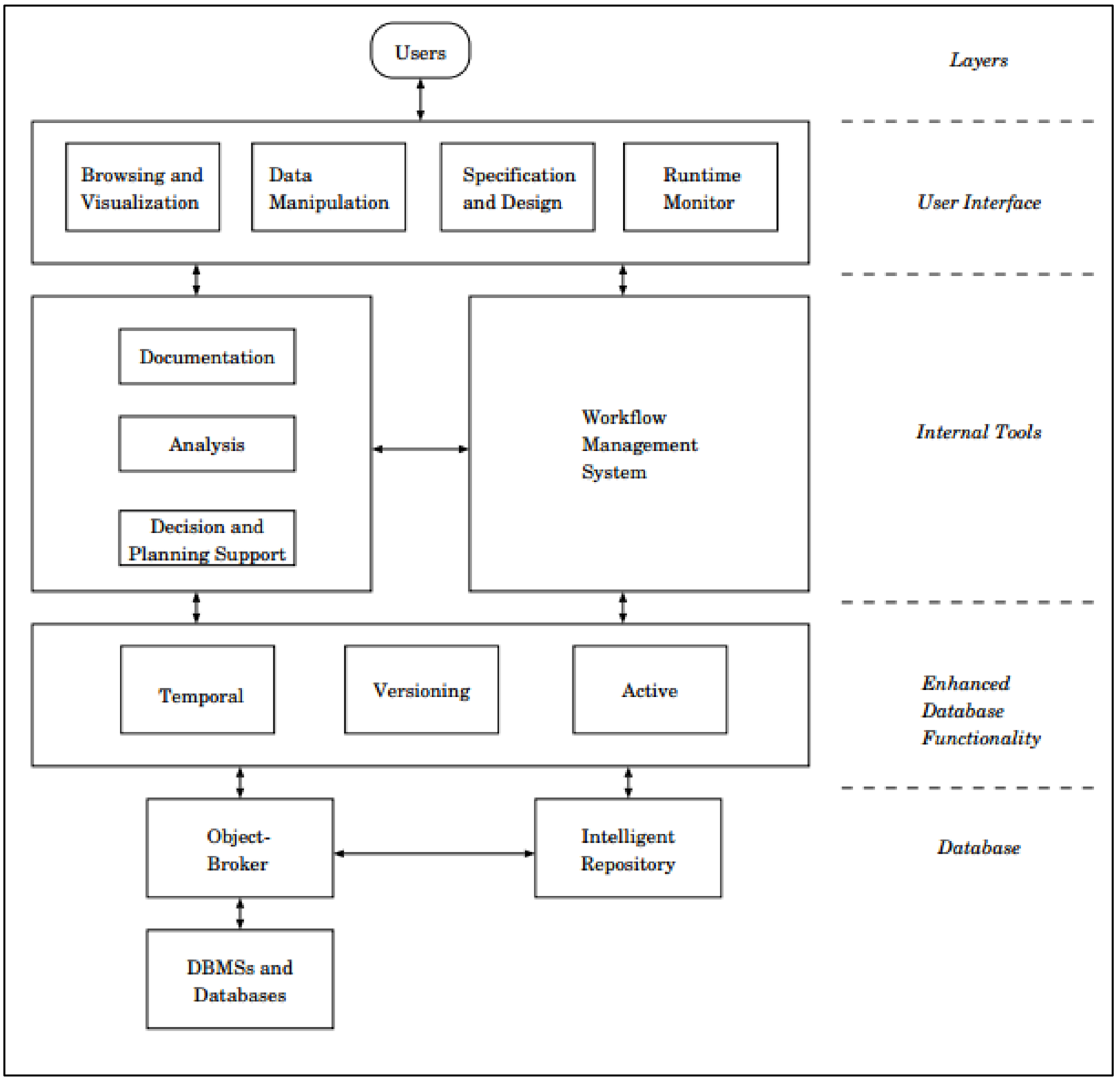


Figure 12: WASA architecture (Medeiros, Vossen, & Weske, 1995, p. 8)

WASA offers some important benefits for scientific applications that have to deal with lots of data. The architecture clearly separates different layers of functionalities which allows for separate experiment specification, control and execution of experiment control and documentation. WASA is built to be able to cope with different types of databases ensuring interoperability. It supports the dynamic execution of tasks, experiment re-usability and reproducibility, ensures function grouping so that specific functions can be accessed by specific modules and the distribution of procedures. More info on WASA can be found in ‘WASA: A Workflow-Based Architecture to Support Scientic Database Applications’ by Medeiros et al. (1995).

6.3.6 Vortex

Vortex is another concept (Figure 13) regarding workflows, proposed by Manolescu (2001) and Hull & Su (1999). Vortex can be seen as a concept in order to support workflow that does not follow a predefined number of steps. It works using Event-Condition-Action (ECA) flows, where a certain action is triggered by an event taking place while meeting the artefact- based conditions prioritized in the model. There exists a graphical notation for Vortex but is it rather cryptic so it does not share the readability that UML and BPMN have. More details on Vortex are provided by Hull et al. (1999), Dong et al. (2000) and Manolescu (2001).

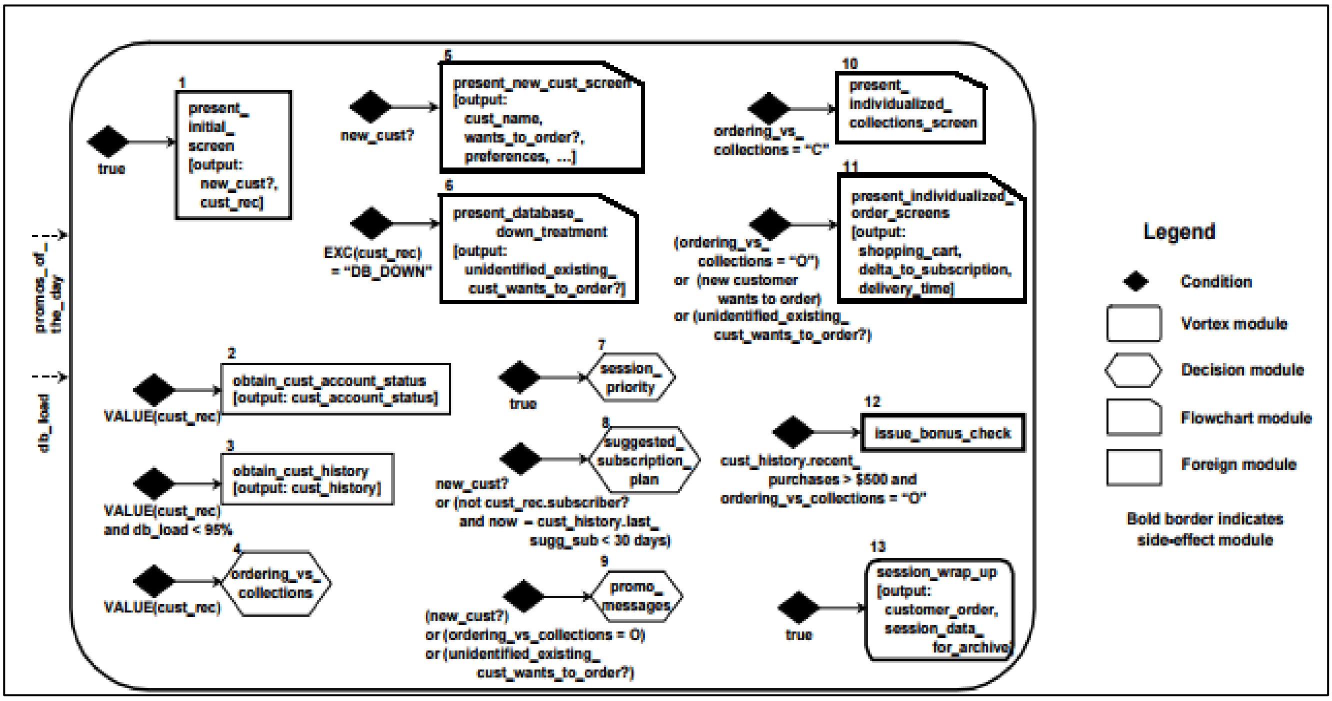

promos_of_the_day
db_load
1
present_initial_screen [output: new_cust?, cust_rec]
true
5
present_new_cust_screen [output: cust_name, wants_to_order?, preferences, ...]
new_cust?
6
present_database_down_treatment [output: unidentified_existing_cust_wants_to_order?]
EXC(cust_rec) = "DB_DOWN"
10
present_individualized_collections_screen
ordering_vs_collections = "C"
11
present_individualized_order_screens [output: shopping_cart, delta_to_subscription, delivery_time]
(ordering_vs_collections = "O") or (new customer wants to order) or (unidentified_existing_cust_wants_to_order?)
2
obtain_cust_account_status [output: cust_account_status]
VALUE(cust_rec)
3
obtain_cust_history [output: cust_history]
VALUE(cust_rec) and db_load < 95%
4
ordering_vs_collections
VALUE(cust_rec)
7
session_priority
true
8
suggested_subscription_plan
new_cust? or (not cust_rec.subscriber? and now – cust_history.last_sugg_sub < 30 days)
9
promo_messages
(new_cust?) or (ordering_vs_collections = O) or (unidentified_existing_cust_wants_to_order?)
12
issue_bonus_check
cust_history.recent_purchases > $500 and ordering_vs_collections = "O"
13
session_wrap_up [output: customer_order, session_data_for_archive]
true
Legend
Condition
Vortex module
Decision module
Flowchart module
Foreign module
Bold border indicates side-effect module

Figure 13: Example of a Vortex workflow for web-storefront of grocery store (Hull, et al., 1999, p. 70)

6.3.7 Document-driven workflow

A similar concept is provided by Wang & Kumar (2005), called ‘document-driven workflow’ (Figure 14). The execution of the process, similar to the action with Vortex, is purely driven by documents. The documents here function as events, triggering the whole process. A diagram mixes both control-flow (‘hard constraints’ or business policies) and data-flow (‘soft constraints’). The model has a drawback though. Data-based constraints are also visualized by using connectors. As the model gets more and more complex or a business problem gets increasingly difficult, the model will take the form of a ‘bowl of spaghetti’, with connectors flooding all overview, which makes it not suitable for high-complexity models (Ould, 2004). Like with previous concepts, the value by the authors is derived from the flexibility it offers, as opposed to control-flow based processes.

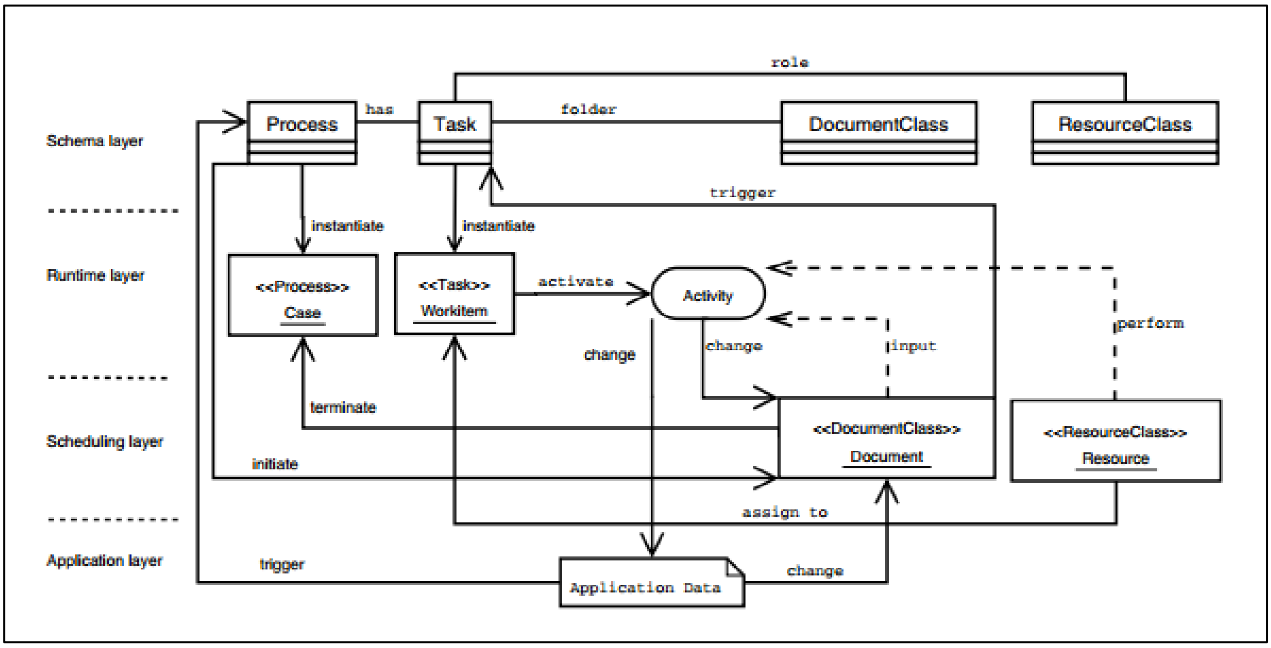


Figure 14: Document-driven workflow framework (Wang & Kumar, 2005, p. 289)

6.3.8 Process-based knowledge management

When you spend some time, digging into process-based knowledge management, you will find a lot of common ground between process-based knowledge management and case management. One could even ask him- or herself what the difference between the two is. In knowledge management (KM) it is possible to regard the knowledge itself as the artefact, since it is the central data-context in knowledge mangement and is passed through between different entities, just like the case file is in case management. The big difference between the two lies in the central artefact and their purpose. First of all, the case artefact is very broad. It contains all the information a case gathers when flowing through the business processes during its case life-cycle. As proposed by Papavassiliou & Mentzas (2003), a data object can be used as a knowledge-artefact. Data objects are used both as input and outputs for tasks, flowing from one entity to the other. The knowledge-artefact is more specific and narrow than the case artefact is. The purpose of the two conepts is also very different. While case management strives to manage a complete case through a dynamic sequence of processes, tasks and activities, knowledge management strives to support other systems by providing them the necessary information about the knowledge needed to execute processes or take decision in these systems.

The field of knowledge management was introduced for the first time roughly 25 years ago (Koenig, 2012) and has been a well-documented activity in organisations for over 15 years (Davenport & Prusak, 1997; Nonaka, 1994). Knowledge Management and displays resemblance with case mangement, but Henk de Man (2009a) fails to adress knowledge mangement as a possible apporach to case magnement in his paper.

With the growing intrerest in knowledge and the emergence of knowledge management, there have been several modelling attempts (Bukowitz & Williams, 1999; Gamble & Blackwell, 2001; Botha, Synman, & Kourie, 2008; Karadsheh, Mansour, Alhawari, Azar, & El-Bathy, 2009). The problem with knowledge mangement modeling is that it is mostly meant as a support to create, store, represent and retrieve knowledge (Alavi & Leidner, 2001) while case mangement is a full-fledged, sustainable system that incorporates all this knowledge-based approaches with common business processes. It is therefore save to summarise their connection into following sentence: case management is partly knowledge management but knowledge management is not case management. There have been approaches to support the integration of knowledge mangement in BPM-systems (e.g. KnowMore system (Abecker, Bernardi, Hinkelmann, Kuhn, & Sintek, 2000), Knowledge-in-Context (Kwan & Balasubramanian, 2002) and CommonKADS (Schreiber, et al., 1999)) but they lack a generalised set of design guides (Sarnikar & Deokar, 2010). Process-based knowledge management systems have been introduced to address the problem of linking knowledge management of the organisational processes. They are used to support knowledge-intensive processes, processes that rely hard on the knowledge and expertise of whoever is involved in executing activities (Sarnikar & Deokar, 2010). In their paper, Sarnikar & Deokar (2010) propose a guideline for process-based knowledge management based on the design science framework (meta-requirements, design process, kernel theories and testable design process hypothesis) of Hevner et al. (2004). Despite giving a good and qualitative overview of the design process and its value, there is no concrete model proposition. The paper is merely meant as an incentive for future empirical analysis and research.

Knowledge management is undoubtedly an interesting topic and will get more and more attention in the future as a result of the increasing knowledge-intensive nature of businesses and their processes. It is surprising that there is no literature that links case management with knowledge management, seen that they share so much features. Even though it shares a lot of features with case management, this is not the place to go delving deeper into the specifics of it. How knowledge management can support case management in its management of knowledge-intensive processes should be the topic of a separate study.

6.3.9 BPM-based case management

A BPM-based approach to model case management is probably the most interesting one of all approaches. It uses the maturity and popularity of BPMN to build case management models. The approach is further elaborated in 6.5 ‘BPM-based case management: an emerging approach to case management’.

6.4 Communication-based process control

Next to artefact-based process control, communication-based process control is the other go-to paradigm when going over case management. As been seen before, Davenport (1994, 2005) and van der Aalst (2005) stressed the importance of human workers in supporting automated decision-making and essential to human collaboration is communication.

6.4.1 Role Activity Diagram (RAD)

The best known communication-based process modelling technique is probably RAD (Role Activity Diagram) (Figure 15). Ould (2005) wrote an entire book, just on the appliance of RAD on case management. The problem with RAD is that for case management, the RAD notation is not sufficient enough to meet the needs in order to be able to act as a support for process execution because data inputs and data events are not specified (List & Korherr, 2005). Based on the analysis of Harrison-Broninski (2005), Henk de Man (2009a) concluded the following items with regards to RAD:

- RAD in itself cannot create a division of activities into tasks, so this has to be added through a process modelling system or as plain text (Harrison-Broninski, 2005), thus rendering it useless for case management.
- Parallel treads of conversation, a strongpoint of RAD according to Ould (2005) is by Harrison-Broninski (2005) qualified as of no use for human-driven processes since human-workers can work simultaneously on activities across multiple branches (as opposed to machines).
- Since RAD fails to provides a notion of data, events or its related states and rules it's unsuitable to define control based on case data. Instead of focusing on the business artefact or the state of the case, it focusses on the state of conversation. According to Harrison-Broninski (2005), even though there is no real case data involved with RAD, each role is assumed to have its own data sources. This assumption comes with two problems: first, this is in no way visualized in the model. Second, case management is characterised by having a central and shared case data-context, unlike a set of personal information sources seen with RAD.
- Since human workers work on activities in any sequence, a lot of detail must be left out of the RAD activity sequence in order to be able to make use of it in a more flexible, ad hoc environment. The vertical sequence (conversation read-line) needs to be removed since there is no predefined sequence. Instead, the assumption grows that each activity is accompanied by a pre- and post-condition, where a post-condition of one activity has to match the pre-condition of another in order to be able to be linked to each other in a vertical sequence.

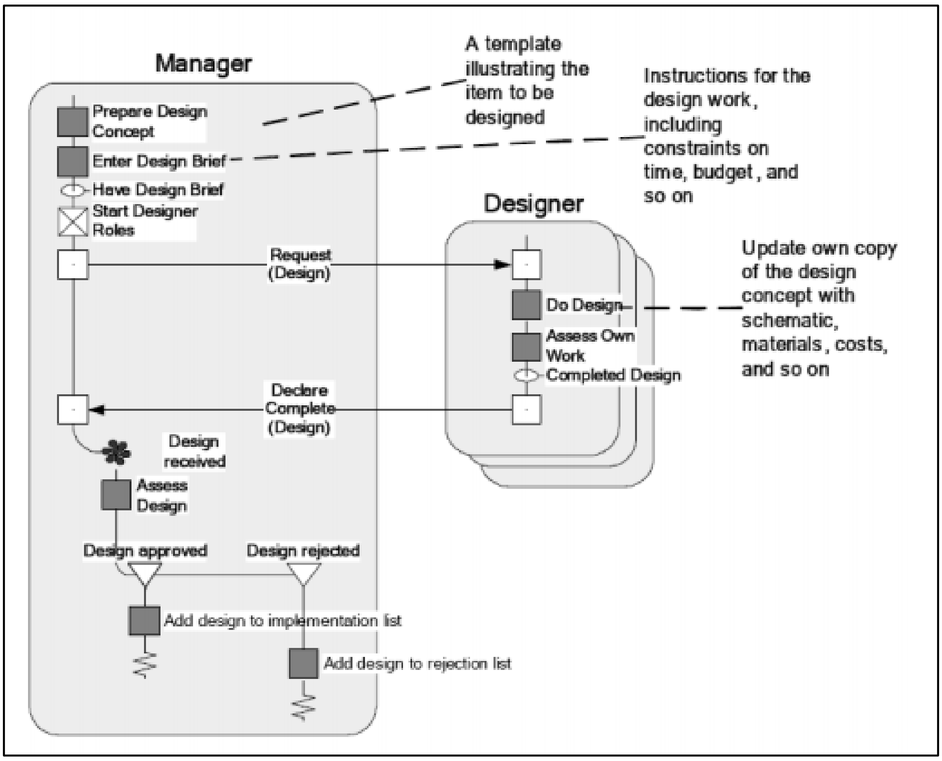

Manager
Prepare Design Concept
Enter Design Brief
Have Design Brief
Start Designer Roles
A template illustrating the item to be designed
Instructions for the design work, including constraints on time, budget, and so on
Designer
Request (Design)
Do Design
Assess Own Work
Completed Design
Update own copy of the design concept with schematic, materials, costs, and so on
Declare Complete (Design)
Design received
Assess Design
Design approved
Design rejected
Add design to implementation list
Add design to rejection list

Figure 15: RAD model (Bushell, 2005)

Although it is possible to conclude the RAD model is not the go-to model for case management, Harrison-Broninski (2005) provides interesting insights regarding communication- based process modelling. One of the most interesting insights they provide is probably that of the 'REACT' pattern. This pattern is based on the fact that "human-driven processes, just like case management processes, are different from other processes, in that work preparation and work distribution design are incorporated into the processes itself. Process workers define processes and the process of defining a process is part of the process." (de Man, 2009a, p. 11)

REACT (de Man, 2009, p. 11):

- Research: explore a new subject
- Evaluate: how that subject can be applied in their own business
- Analyze: decompose work into task for other workers to involve in the project
- Constrain: Determine controls and authorisations for these tasks
- Task: get tasks executed

6.4.2 Language-Action Perspective (LAP)

There is another concept that is present in the world of communication-based process modelling, although it doesn't get much attention because of the restrictions in use and the criticism it subsequently received. The concept is called the Language–Action Perspective (LAP). LAP is discussed in all its facets by Kethers & Schoop (2000) and Goldkuhl (2003). Following LAP, people reach goals by communicating. The basic unit for communication is called the 'speech act'. "A speech act is a minimal functional unit in human communication like 'request', 'promise', 'demonstrate performance', and 'accept'." (Jaworowska, 2001). There are two approaches to LAP used in business process management: ActionWorkflow and DEMO. LAP received a lot of criticism, mainly because it is an academic concept that fails to be relevant for applications in industry. Also, there is no reason why people would choose LAP as a means to work around certain problems because there are sufficient alternatives that manage to perform better overall. Kethers & Schoop (2000) go deeper into evaluating ActionWorkflow.

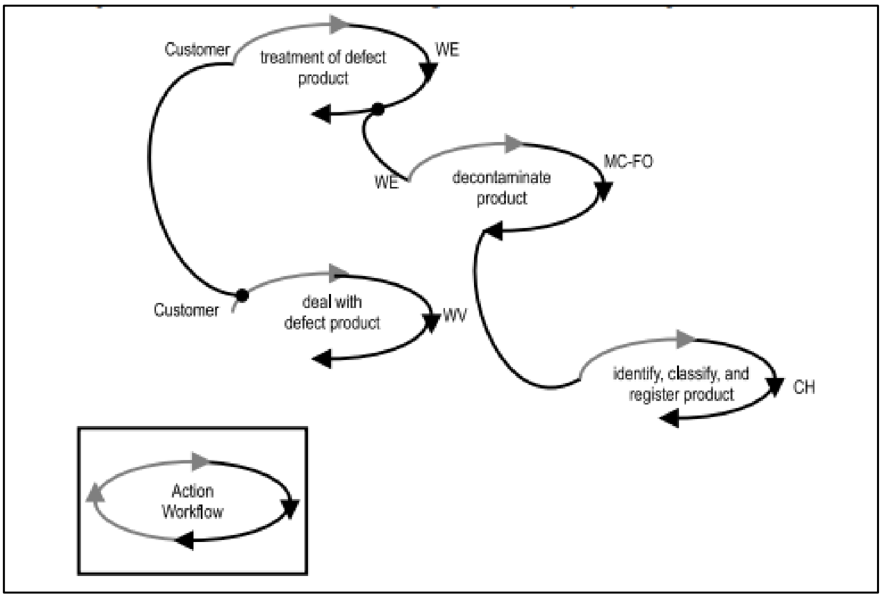


Figure 16: Example of a Service-oriented model of ActionWorkflow (Kethers & Schoop, 2000, p.159)

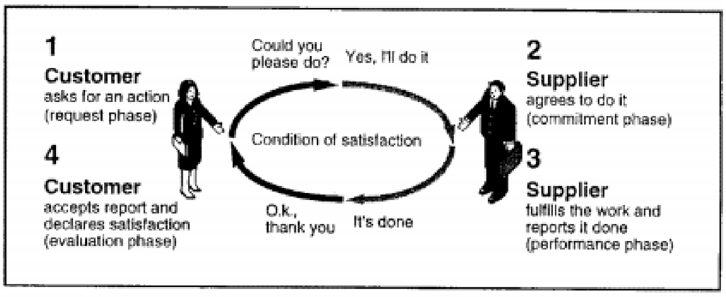


Figure 17: Action Workflow Loop (Schäl, 1998, p. 38)

As is depicted in Figure 16 and Figure 17 above, the ActionWorkflow consists of four phases: a request phase, a commitment phase, a performance phase and an evaluation phase. This loop of four phases is the core element of the ActionWorkflow methodology (Kethers & Schoop, 2000). The loop focusses solely on communication. The underlying actions are not modeled. Multiple loops can exist in relation to each other in order to be able to simulate clarifications, discussions, further actions etc. In spite of being able to provide value in terms of communication between different actors within a business environment, the ActionWorkflow overall falls short when it is evaluated against the Guidelines of Business Process Modelling (GoM) (Becker et al., 2000). There are no remarks to be made when talking about syntactical correctness. That can't be said of the semantical correctness. In order to meet semantical correctness, the ActionWorkflow has to be fitted with clear and exact semantics to prevent inconsistencies from popping up in the model. Another problem regarding semantics is the lack of existence of an object that runs through the workflow so that it is hard to determine whether or not a certain condition has been fulfilled at the end of the loop. "The fulfilment of the condition of satisfaction is decided on by the customer, during her evaluation phase, and will often depend on "soft" criteria, such as timeliness, or friendliness of service, which cannot be expressed by means of the Action Workflow approach." (Kethers & Schoop, 2000, p. 164). Other problems with correctness are the focus on the actor 'customer', the inability to backtrack to previous phases, define alternatives and situating the workflow in time, resulting in a strictly sequential flow. Because of the problems regarding semantics, the comparability of the ActionWorkflow to other models is restricted. The conclusion to make is that "modelling is highly subjective, and strongly depends on the modeller's personal understanding of the process, point of view, and modelling goals." (Kethers & Schoop, 2000, p. 165).

6.4.3 DEMO

DEMO (Figure 18), which has been inspired by the ActionWorkflow approach (Weigand, 2005) and hence inherited a lot of its criticism, is not sufficient and specific enough as a basis for process automation. It is therefore more suitable for capturing high-level business interaction (Dumay et al., 2005 and Dumay, 2005). According to Rittgen (2004), the only way DEMO can in some way support dynamic processes is by leaving the dynamic transactions out of a DEMO diagram and replace them with rules-based behavior. But since rule-based behavior cannot be defined in DEMO model, it lacks the capability to model dynamic processes, which makes it completely useless for case management processes (Rittgen, 2004). On his turn, Lyytinen (2004) stresses the possibility to use DEMO as a tool for business process analysis, not for business process execution. This conclusion comes not as a surprise as Dietz himself referred to DEMO as "a methodology that can be an effective help in various situations concerning the analysis and optimization of business processes." (Dietz, 1999, p. 14).

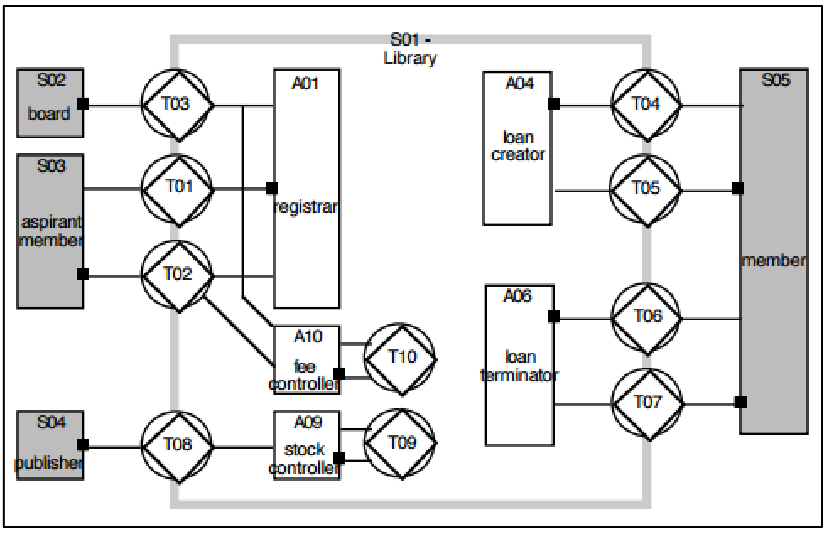


Figure 18: DEMO Construction Model of a library (Dietz & Halpin, 2004, p. 5)

Harrison-Broninski (2005) has also given thought to the idea of incorporating LAP into RAD, specifically the idea of 'speech acts' in order to create a more detailed overview of the interactions within RAD. General conclusion regarding communication-based approaches of process modelling are included in Henk de Man's 'Case Management: A Review of Modelling Approaches' (2009a). It is possible to summarize them into one sentence: Communication-based process control as it is now is not sufficient to fulfil the requirements that are put forward on modelling dynamic business processes or case management.

6.5 BPM-based case management: an emerging approach to case management

As most state-of-the-art BPMS systems adopt BPMN for process modelling, case management processes cannot yet benefit from the power of these systems yet. For this reason, in many administrative environments, case management often remains a paper-based process. E-mail is probably the most used tool for many case management processes. However, management of cases that are just administrated on paper, or that are just 'automated' by e-mail, is often not productive enough and does not sufficiently comply to today's regulations e.g. aim to protect personal information of clients, citizens, patients, students, etc. Such systems also lack the possibility to adequately standardise, balance, schedule, authorize, audit, monitor and account for case work. (de Man, 2009a).

In '5.6 Case management vs. Business Process Management' the thesis already talked about the exponential increase in BPM spending that is expected in the near future. With BPM being used in more and more organisations to automate and optimise a range of business processes, there is a growing interest to fuse BPM and case management concepts together to create the ultimate process management system. The popularity of BPM is not the sole reason to conclude they should be combined. Control flow-based processes and artefact-based processes have to be used together anyway to achieve a complete business process system. Predefined, structured processes should be able to be initiated by a case management process and vice versa (de Man, 2009a). One of the strengths of BPM is that is allows support of different technologies to create an environment where each technology supports one aspect of the business process concept. This way a BPM-based case management system can support the capabilities of systems like CRM and EDRMS while at the same time being able to profit from the control possibilities with regard to flexible, dynamic and ad hoc processes (White, 2009; Yockelson, n.d.). A case-based mangement system combines the best of two worlds: predefined, structured business processes can be joined seemlessly on dynamic ad hoc processes, allowing run-time interaction, communication and intervention by human case-workers while being supported by the capabilities of technologies like CRM, Content Management, Knowledge Mangement and EDRMS to increase overall understanding of the case (White, 2009; de Man, 2009a; Franke, 2011; Yockelson, n.d.).

Below I'll introduce the Case Management Model and Notation (CMMN) and DeCo as approaches to BPM-based case management. I'll end the modelling approaches with the BPM-based case management case study by Henk de Man (2009b).

6.5.1 Case Management Model and Notation (CMMN)

On the 10th of May 2014 the Object Management Group released its first version of the Case Management Model and Notation standard with the support of some of the largest IT-firms in the world (e.g. SAP, Oracle, IBM…). The Object Management Group is an international non-profit technology standard consortium founded in 1989. OMG is the driving force behind modelling standards like the Unified Modelling Language (UML) and Model Driven Architecture (MDA) (OMG, 2015). Their mission statement is to develop technology standards that provide real-world value for thousands of industries (OMG, 2015). Following this philosophy, OMG defined a meta-model and notation for modelling and graphically representing a case as well as "an interchange format for exchanging case models among different tools" (OMG, 2014, p. 1). The actual goal of the specification is to capture the elements that are commonly used in case management products. The case management model and notation standard wants to be to case management what BPMN is to business process management (OMG, 2014).

In order to be compliant and conformant with CMMN 1.0, software that claims to achieve compliance and conformance needs to fully match the applicable compliance points stated in the specification. This means that software that only partially matches these compliance points can never claim compliance and conformance with the specification (OMG, 2014). There are four compliance points listed in the standard: Visual Notation Conformance, Case Modelling Conformance, BPMN Compatibility Conformance and CMMN Complete Conformance (OMG, 2014). Not all compliance points are required. We'll need to look at the combination to decide whether or not a compliance point is required. The CMMN (1.0) also implements a XML model for Interchange (XMI) and a XML-Schema (XMLS) to be able to exchange the models between different case management vendors environments and tools. This meta-model specification ensures interoperability. Next to the meta-model, the document also goes over the case management elements like case files, tasks and case workers. The document dedicates a lot of pages on the meta-model and the class diagrams that form its core structure. This master disseration won't go over all the diagrams since they do not add value to the purpose of this thesis. People interested into the CMMN meta-model can always consult the document directly.

The document also lists a series of Model Elements (Core Infrastructure, Case Model Elements, Information Model Elements and Plan Model Elements). These elements illustrate all the elements that represent the initial building blocks of a case (e.g. abstract base classes, case attributes, case files, attributes…) together with all the elements that support future evolutions in run-time (Plan Model Elements). Their diagrams are all extensively discussed in the CMMN. A further elaboration of these diagrams does not contribute to what this thesis tries to achieve. This master dissertation is all about understanding the basic concepts of case management while keeping it readable and understandable for everybody. The concepts introduced with the diagrams are not new and the diagrams themselves are mainly characterised by a high amount of technicality. To avoid all unnecessary copying, people interested in these diagrams should consult the document directly.

The CMMN notation is what is most interesting. Since the notation is built up around a BPMN construct, a lot of the case notation elements will be familiar. What follows is a brief overview of the most distinct modelling elements that are introduced in this beta version of the CMMN. The first novelty is the introduction of discretionary tasks (Figure 19).

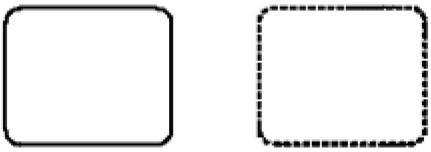


Figure 19: Normal task and discretionary task in CMMN (OMG, 2014, p. 49)

The introduction of discretionary or unstructured tasks is obvious. They allow the model to cope with ad hoc tasks. Tasks can also be accompanied so called 'Sentries' (Figure 20). A sentry is a look-out waits for a particular situation to happen that will influence the ongoing proceedings of the case. If a certain event or situation occurs, that situation or event can influence the case when it is specified by the sentry. A sentry either is an event or a condition that has to be valid in order to proceed or it can also be a combination of those two. So when a situation or event occurs that catches the sentry's attention, the sentry will do an initial check on the event or situation that occurs. If this event or situation proves to be valid and the sentry accepts it, the sentry will do a second, conditional, check. If this condition is true, only then the case can be continued through that particular flow. In this instance, the sentry will take following form: pass through on <event> if <condition>.

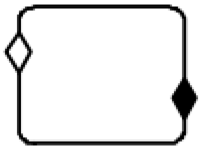


Figure 20: CMMN task with entry and exit sentry (OMG, 2014, p. 49)

Exit sentries are not new, they are also used in BPMN 2.0 in the form of ‘conditional sequence flows’, where a specified condition needs to be true in order for the sequence to continue (OMG, 2011). Entry sentries are new. In BPMN, a similar situation can be modelled using a gateway. The gateway can be expanded with a conditional intermediate event to model a process flow in cases where it would be advisable to validate a combination of events and conditions. There is also a distinction in human tasks. They can be blocking or non-blocking (Figure 21). A blocking human task means that the task will wait until the work that is associated with that task is completed. If it is not-blocking then the task completes immediately.

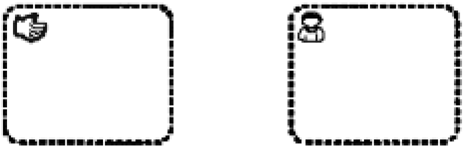


Figure 21: A non-blocking and a blocking human task (discretionary) (OMG, 2014, p. 50)

Then there are process tasks and case tasks (Figure 22). They are used respectively to call upon another process or case.

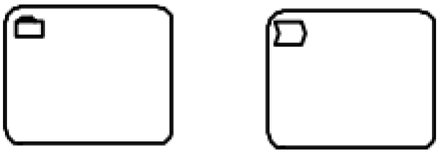


Figure 22: Case task and process task (OMG, 2014, p. 51)

Milestones are also introduced in the CMMN notation (Figure 23). They play an important role in the case management model because they allow to follow-up on the status of the case. They do not represent any activities or tasks themselves but they are mainly used as a way to model the completion of a series of tasks or the availability of key deliverables. Often milestones are accompanied by an entry condition (sentry), specifying the criteria that need to be fulfilled before reaching the milestone.

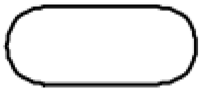


Figure 23: Milestone (OMG, 2014, p. 52)

A dependency between two tasks is not visualised by a full line like in BPMN 2.0. Full lines illustrate a predefined path between tasks or other modelling elements. In case management specific processes there are no predefined paths.

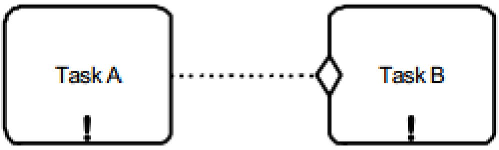
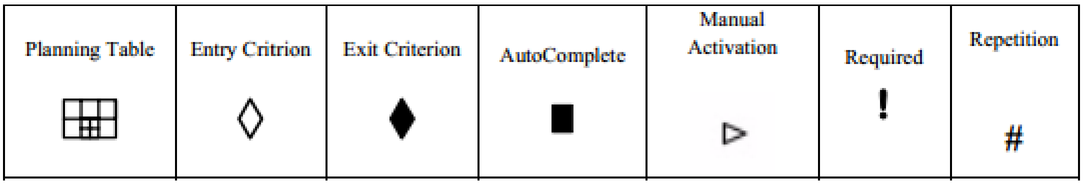


Figure 24: Dependency between two tasks (OMG, 2014, p. 53)

Dependencies between tasks are therefore shown as a dotted line (Figure 24), to illustrate the possibility to diverge from what is modelled. Also note the black exclamation mark at the bottom of the tasks. This is one of the decorators used in CMMN to make the notation as expressive as possible. Their goal is to indicate specific behaviour. An exclamation mark means that the task is characterised by required execution. Figure 25 gives an overview of the decorators used in CMMN.



Planning Table	Entry Critrion	Exit Criterion	AutoComplete	Manual Activation	Required	Repetition
⊞	◇	◆	■	▷	!	#

Figure 25: CMMN decorators (OMG, 2014, p. 62)

The AutoComplete decorator is self-evident. This decorator indicates the automatic completion of a stage or case plan. In cases where a stage or plan has to be completed manually by a user, the plan or stage is indicated by a hollow arrowhead. Tasks, stages and milestones accompanied by a '#' need to be repeated. They will create a new instance whenever a new entry criterion is satisfied, similar to a multi-instance marker in BPMN 2.0 (OMG, 2011). The Planning Table decorator indicates that the elements of the stage or task accompanied by the decorator can be used for planning purposes. This makes it possible for case workers to plan additional items into the stage or task in run-time.

One last item that needs addressing is the stage element (Figure 26). A stage in a case can be seen as a fragment of a planning, a container to visually store case elements. CMMN also describes stages as episodes of a case. However there is a big difference between stages and plan fragments in the CMMN. Stages allow additional run-time planning in a case, a plan fragment doesn't.

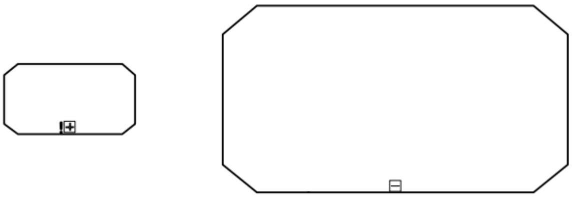


Figure 26: Collapsed and expanded stage (OMG, 2014, p. 47)

When all those elements are put together in one model it would look something like Figure 27.

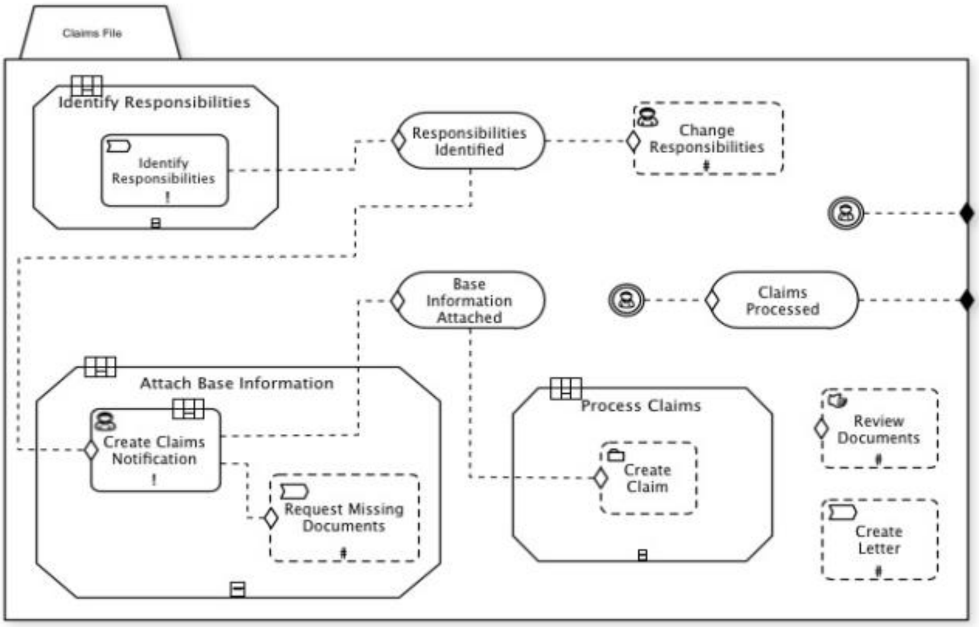

Claims File
Identify Responsibilities
Identify Responsibilities
Responsibilities Identified
Change Responsibilities
Base Information Attached
Claims Processed
Attach Base Information
Create Claims Notification
Request Missing Documents
Process Claims
Create Claim
Review Documents
Create Letter

Figure 27: Claim Management, a CMMN example (OMG, 2014, p. 64)

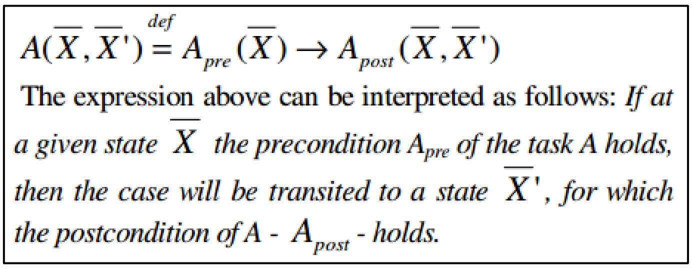
The CMMN clearly shows similarities with its business process management counterpart. For someone who has a basic understanding of BPMN 2.0, it only takes minimal effort to get into the CMMN 1.0 specifics and elements. In this way, OMG has really succeeded in constructing a comprehensive and accessible case management standard. Cases, unstructured tasks, milestones and discretionary flows are some of the elements that have been deemed important in case management throughout this thesis and the CMMN standard has succeeded in capturing these elements. On the other hand, the CMMN also introduces some other elements that may not be so intuitive: stages and decorators.

We may not forget that this CMMN standard is just a beta specification. The specification will probably witness a number of years of alterations and adjustments before organisations see its worth and are comfortable and confident enough to implement it. Standards like BPMN have had the same walk of life to be where they are now.

6.5.2 Declarative Configurable (DeCo) Process modelling notation

The Declarative Configurable Process modelling notation is an extension on the BPMN notation, enhancing it with descriptive process modelling, formal analysis and step-wise refinement (Rychkova & Nurcan, 2011). The DeCo approach succeeds in capturing some of case management's most distinct and important features: unstructured, knowledge-intensive processes. BPM however has in all those years been looked at by practitioners as a tool solely built to manage structured processes. This is one of the major reasons why, up until today, BPM fails to meet those specifications necessary to cope with dynamic environments and ad hoc processes. With DeCo, the focus doesn't only lay on the design-time specification of the model, but it adds two additional perspectives since it is necessary to have run-time specifications to be able to model knowledge-intensive processes. This approach allows BPM specialists to separate the goals from the means and guarantees them to improve the adaptability of the model through a process realization (Rychkova & Nurcan, 2011).

BPMN itself is imperative, meaning that it specifies what is allowed. Moving away from the predefined sequence is not possible, often resulting in BPMN being labelled as over-constrained (Swanson & Farris, 2015). To avoid these constraints, declarative modelling may be an answer. Declarative models focus on what the solution is, whereas an object-oriented (imperative) model describes how a solution is obtained (Hansen, Haxthausen, & Villadsen). Instead of allowing only a specific set of actions, a declarative model allows all actions at any time with the only restriction being the conditions or the sequence in which some actions need to be taken (Swanson & Farris, 2015). A simple example of a declarative modelling tool is the use of rules: if (condition) then (manipulation of case state) (Swanson & Farris, 2015). This kind of declarative modelling is also found in database queries (Hansen, Haxthausen, & Villadsen). DeCo uses a similar approach to model conditions as seen in Figure 28.



$$A(\overline{X},\overline{X}') \overset{def}{=} A_{pre}(\overline{X}) \rightarrow A_{post}(\overline{X},\overline{X}')$$

The expression above can be interpreted as follows: *If at a given state $\overline{X}$ the precondition A_{pre} of the task A holds, then the case will be transited to a state $\overline{X}'$, for which the postcondition of A - A_{post} - holds.*

Figure 28: Declarative modelling principle used in DeCo (Rychkova & Nurcan, 2011, p. 5)

This process specification used in DeCo makes it possible to declare a set of tasks without there being any restrictions or scenarios for execution. This means that tasks can be dynamically selected at run-time from a list of tasks accessible at a given case state (Rychkova & Nurcan, 2011).

I already mentioned that DeCo is an extension on the BPMN notation. Rychkova & Nurcan (2011, p. 7) summed up some features that DeCo introduces in the BPMN notation:

- Action contracts (including preconditions, post-conditions and invariants)
- Distinction between cross-boundary data objects (IN/OUT) and local data objects
- Optional vs. obligatory tasks/data/conditions (dashed or solid lines)
- Configurable vs fixed tasks/data/conditions (bold or standard lines)
- Configuration rules (if explicit)
- Task/role/data object specialization (S)
- Task/role/data object alternatives (A)

DeCo came into existence after recent discussion in the BPM community regarding solutions for descriptive, knowledge-intensive processes (Rychkova & Nurcan, 2011). The Declarative Configurable specifications allow capturing processes at three different levels of abstraction: design (which focusses on process goals), deployment (reflects process deployment environment) and execution (reflects single process enactment) (Rychkova & Nurcan, 2011). Deployment can have a serious influence on how the model is structured. Sometimes organisation may want to model every possible execution path within a single process model. Just like the document-driven workflow (Ould, 2004), this can lead to high complexity which will probably result in the model being very ineffective. One possible solution to counter this problem is to identify multiple modelling levels that have very strong relations with each other (Rychkova, Regev, & Wegmann, 2008). As a result DeCo makes a distinction between models resulting from the process design phase (called a configurable process definition) and models developed to reflect the concrete environment in which the model will operate (called a customised process design specification) (Rychkova & Nurcan, 2011). On the other hand, a lot of process details can only be identified when executing the model so they cannot be specified during customization. It is advisable for models that are execution sensitive to still allow configuration and fine-tuning. This will result in the process realisation, a reflection of the customised design specification enhanced with the dynamic process specification and decision-making. Its main purpose is to “construct the knowledge base and to contribute into further process improvement.” (Rychkova & Nurcan, 2011, p. 6).

The fact that processes in DeCo can undergo changes depending on the phase they are in implies that they mature over time. The maturity of processes in DeCo is visualised in a process life-cycle containing four distinct phases: design, customisation, execution and analysis (Figure 29). In the design phase, the process goal is set and the tasks and activities that will support this goal are defined. The design phase (Figure 30) “focusses on what can be done to accomplish the process and what is needed for it.” (Rychkova & Nurcan, 2011, p. 6). In the customisation phase (Figure 31) the process environment is selected. The process environment will be characterised by a specific set of tasks, activities, roles and rules so the initial design specification will need customisation in order to be able to cope with the environment specific information. In the customisation phase the emphasis is put on "what the organisation has to do to accomplish the process in a given environment and what will be available for it to do so." (Rychkova & Nurcan, 2011, p. 6). The execution phase (Figure 32) allows the previous customisations to be applied at run-time. As has already been mentioned above, DeCo provides a list of tasks from which tasks can be dynamically picked to be executed at run-time. The availability of tasks depends on the case state. When a task is triggered it will become part of the process realisation (Rychkova & Nurcan, 2011).

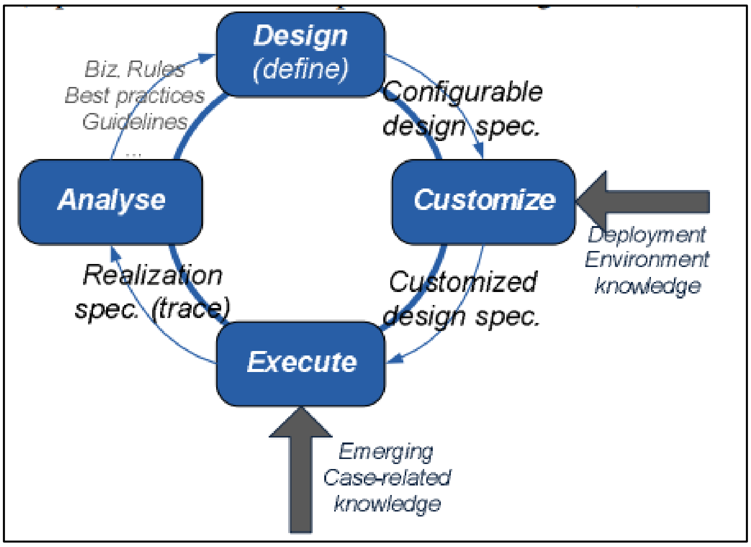

Design (define)
Biz. Rules
Best practices
Guidelines
...
Configurable design spec.
Analyse
Customize
Deployment Environment knowledge
Realization spec. (trace)
Customized design spec.
Execute
Emerging Case-related knowledge

Figure 29: DeCo process life-cycle (Rychkova & Nurcan, 2011, p. 7)

Process realisations are stored upon execution in the history record (cfr. case artefact). History records allow constructing a case history that can be analysed in order to steer business rules, define new processes, tasks and activities or to identify repetitive scenarios (Rychkova & Nurcan, 2011).

On the next page you will find an example of the DeCo process modelling notation, based on a mortgage approval process. Rychkova & Nurcan (2011) chose mortgage approval because it is a typical example of a case management process. The mortgage approval process below is the result of a description defined by several US financial institutions. The model includes guidelines, recommendations and descriptions typical for mortgage approval. The process is divided into different steps: Pre-qualification, Formal Application, Document Review, Pre-approval, Property Appraisal, Final Approval and Closing. In their research, Rychkova & Nurcan (2011) only focussed on the Formal Application.

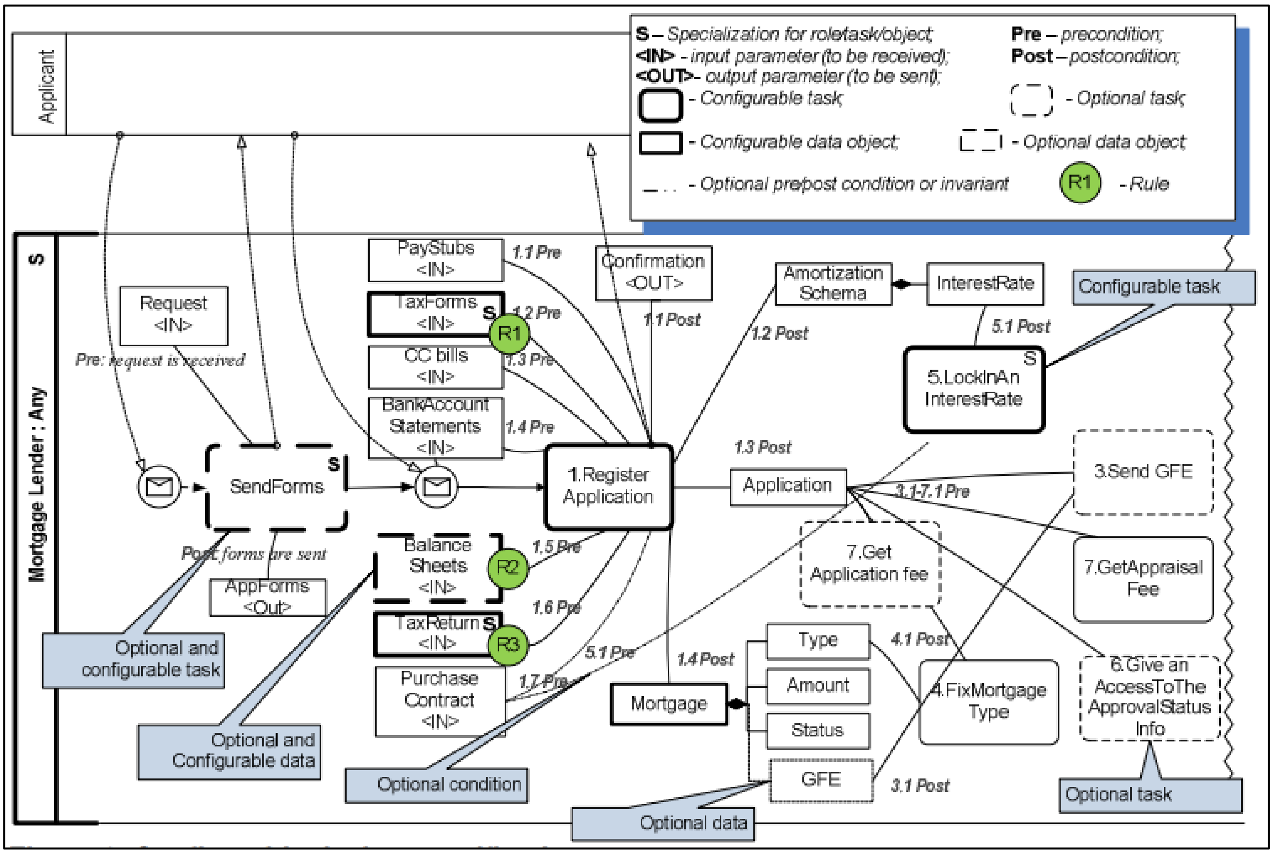


Figure 30: Configurable design specification of the Formal Application step in a mortgage approval process (Rychkova & Nurcan, 2011, p. 8)

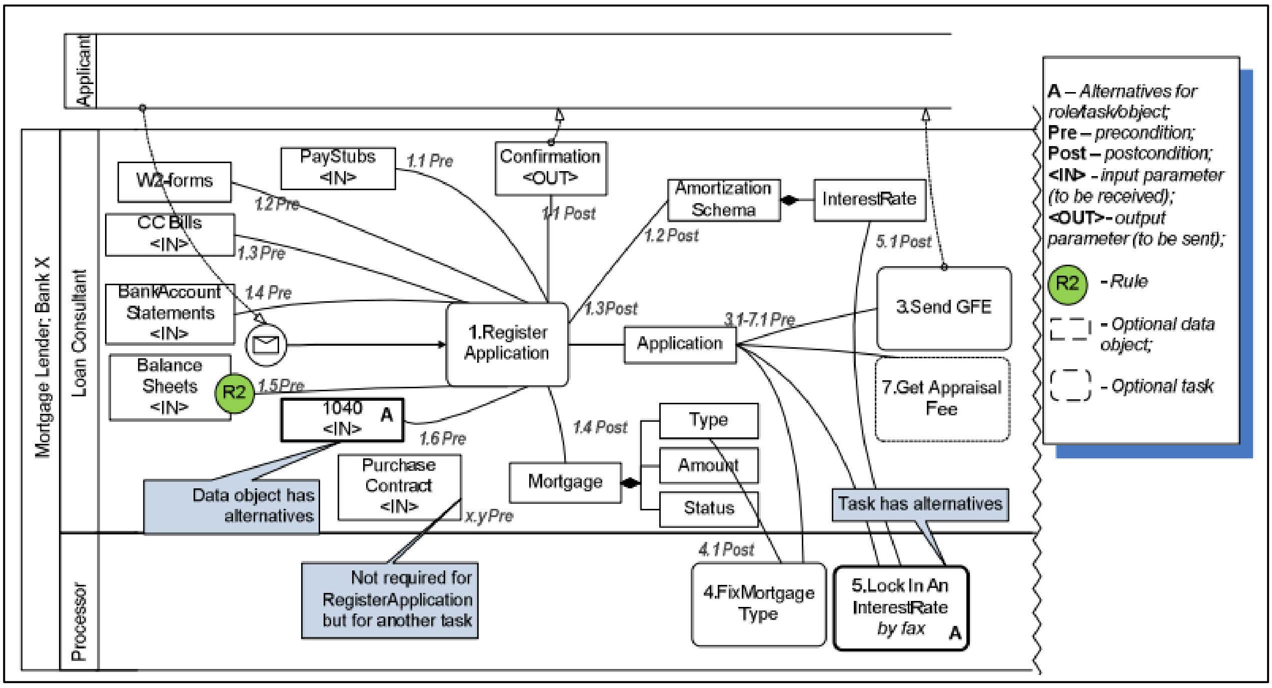


Figure 31: Customised design specification of the Formal Application step in a mortgage approval process (Rychkova & Nurcan, 2011, p. 8)

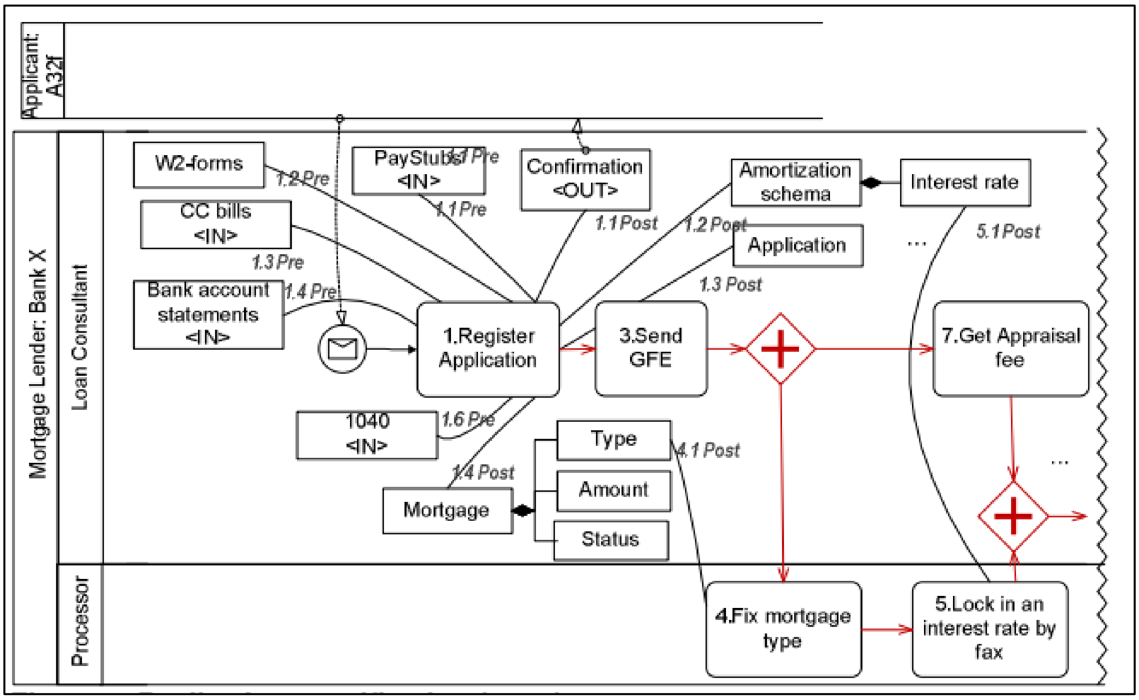


Figure 32: Realisation specification of the Formal Application step in a mortgage approval process (Rychkova & Nurcan, 2011, p. 9)

We can see in Figure 30 that tasks in the model are accompanied by pre- or post-conditions. For example, the 'RegisterApplication' task is accompanied by a series of preconditions (e.g. 'PurchaseContract', 'TaxReturn' or 'BankAccountStatements') and some post-conditions (e.g. link with the data object 'Application'). Conditions describe a link between tasks and data objects. Often this link is to establish an order of execution. 'BalanceSheets' in this model is an optional and configurable data object, making it only needed when the applicant is self-employed. 'SendForms' is an optional and configurable task. If this kind of task is triggered it is possible to specialise (e.g. send forms by mail, internet, fax…). The 'GetApplicationFee' on the other hand is only optional, not configurable. This kind of task can be decided upon in every other case but cannot be specialised. In the Customised design specification, the design elements are customised according to the process environment. In this example, the mortgage approval process is specified for Bank X, adding more detail to the execution process as defined for Bank X. Although the model is now specified for Bank X, it is still configurable. When the execution phase is reached, the model will be configured while the process is running. In this case, the model illustrates only one instance of the mortgage approval process. There are no more optional and or configurable tasks or data objects in this model because every option or configuration is already decided upon in this instance. The realisation specification will allow organisation to perform in-depth analysis and improvement of the process (Rychkova & Nurcan, 2011).

6.6 Case Study: Cordys Approach to Case Management

6.6.1 Overview

In this case study, Henk de Man (2009b) tries to explore the ideas of case management based on his previous paper, where he talks about the essence of case management and its possibilities to cope with some of today's problems regarding documenting and managing case-based processes. This thesis will try to give a comprehensive overview of the findings of Mr. de Man (2009b).

The Cordys approach to case management is explained based on a use case: calamity handling in railways (Figure 33). A use-case is a description of how a user interacts with a system to perform a certain activity. They are used to identify and communicate all the requirements for the system to the programmers (Tegarden, Dennis, & Wixom, 2013).

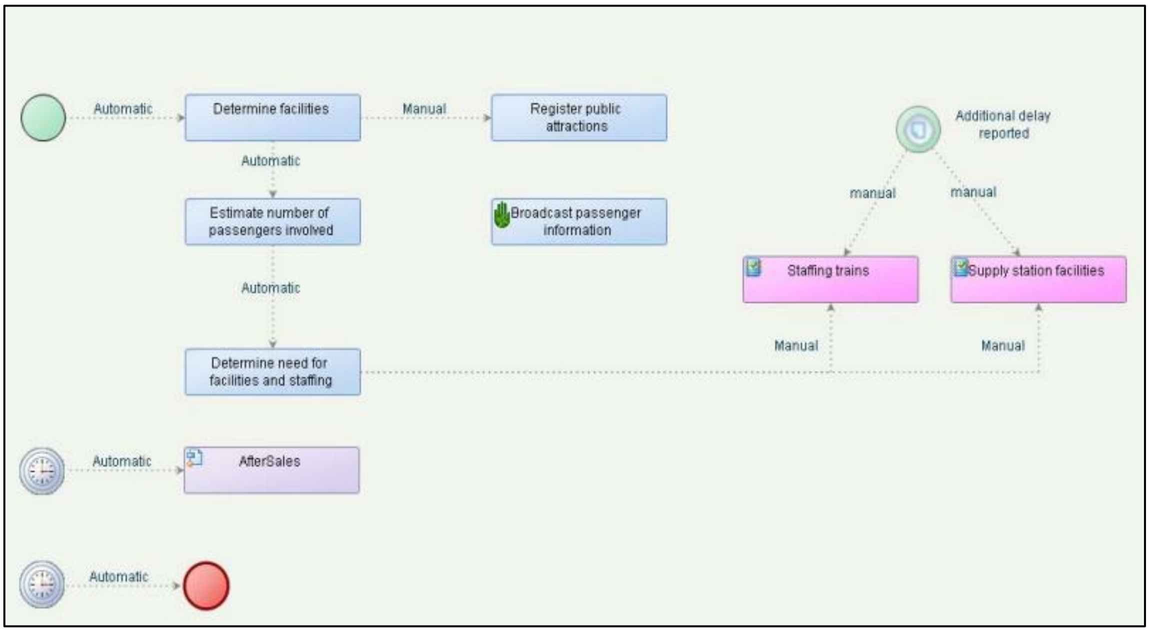

Automatic
Determine facilities
Manual
Register public attractions
Additional delay reported
Automatic
manual
manual
Estimate number of passengers involved
Broadcast passenger information
Staffing trains
Supply station facilities
Automatic
Manual
Manual
Determine need for facilities and staffing
Automatic
AfterSales
Automatic

Figure 33: Railway calamity case handling overview (de Man, 2009b, p. 2)

The situation modelled above is as follows: "A train comes to a stand-still, possibly as a result of an accident or a technical problem. There is a major delay that effects other stations and other trains as well. The case ('case file') represents the calamity itself and all related information. The focus here is put on explaining modelling constructs, not on overseeing and understanding a real-life case in all detail." (de Man, 2009b, p. 2) The model shows a high degree of comparison to BPMN-models but there is a major difference. In BPMN, events and activities are connected through a sequence flow. In this model, there are no sequence flows, only dotted connectors. Those dotted connectors denote follow-up relations. The 'follow-up' consists of all the activities associated with a certain case. In a way, the follow-up models the development of a case through its specific states. Based on certain case events, a follow-up decision can be made, either by the system (automatic) or by a human case worker (manual). (Doganov, Haralanova, & Lutfy, 2005).

6.6.2 Activities

In the model it is possible to see four types of activities, not three as depicted by de Man (2009b). The first type of activity is an activity (case task) that is executed by case workers (e.g. Determine facilities). The second type of activity visualizes a decomposable case or sub-case (e.g. Staffing trains). A third type of activity shows a sequential workflow (e.g. After Sales) (cfr. BPMN process) since a sequential workflow can be started from a case process. Integration in the other way is also supported. This type of modelling, where the modelling of a sequential workflow in a case management diagram is supported, allows a separate BPMN-style diagram. The fourth type of activity is introduced for the modelling of an ad hoc activity (e.g. Broadcast passenger information). This type of activity can be performed by a case worker, provided he has the authorisation to do so. Follow-up activities triggered by an ad hoc activity do not have to be shown in the diagram. In this instance the ad hoc activity is not connected to any other activity in the diagram. The reason behind this is to explicitly model an activity for ad hoc planning, even though this activity could be modelled or, in this case planned, through a follow-up relationship. When going over ad hoc activities in this model, it is important not to confuse them with ad hoc in the sense of ad hoc sub-processes in BPMN. The big difference of course, is the central case artefact (data-context) in case management. BPMN-processes do not include such a notion, nor does it include sharing the case data-context across the activities that are executed in context of the case.

6.6.3 Events

Apart from activities, the model also consists of several events comparable to the events in BPMN. In this model there are three different types of events to be distinguished. The first type of event is the ‘create case events’. This event basically resembles the ‘start event’ icon in BPMN and does exactly what it says it does: it creates the central case data-context which is shared across the different activities. The second type of event is the ‘periodic time event’. These events are visualized by the time event icons. Activities linked to this type of event are triggered when a certain period of time passes. In our example the ‘After Sales’ activity is executed after a time interval which processes the compensation handling for mislead passengers. Remember that ‘After Sales’ is a sequential workflow that can be started from within a case process. The third event shown in this model is the ‘case change event’. This event is represented by a conditional event icon. This type of event models a situation where an exception occurs (e.g. a calamity that is more serious than average). In this case the exception may be a reported additional delay for passengers. In this situation, the model allows to take three decisions: send a service crew to a stranded train to help passengers (cfr. ‘Staffing trains’) and/or providing additional facilities to a station that is affected by the calamity (cfr. Supply station facilities). The fourth and last event is not explicitly modelled but is nonetheless considered an event by Henk de Man (2009b). The event this paper talks about is the 'completion of case activities' which triggers a decision in the follow-up by the case worker, leading to the next activity. An event not talked about by Henk de Man (2009b) is the event that closes the case, the end event. In this model, the end event is linked by a dotted connector to a time event. The reason behind this construction could be to protect the flexibility of the case worker and his/her decision to close the case independently from the time event. If there is no interference from the case worker, the case will be closed after the modelled period of time. When using a conventional connector (cfr. BPMN) there would be no space for a case worker to decide upon the closure of the case. The case would be closed after the recorded period of time no matter what. This is where the strength of case management comes to the surface.

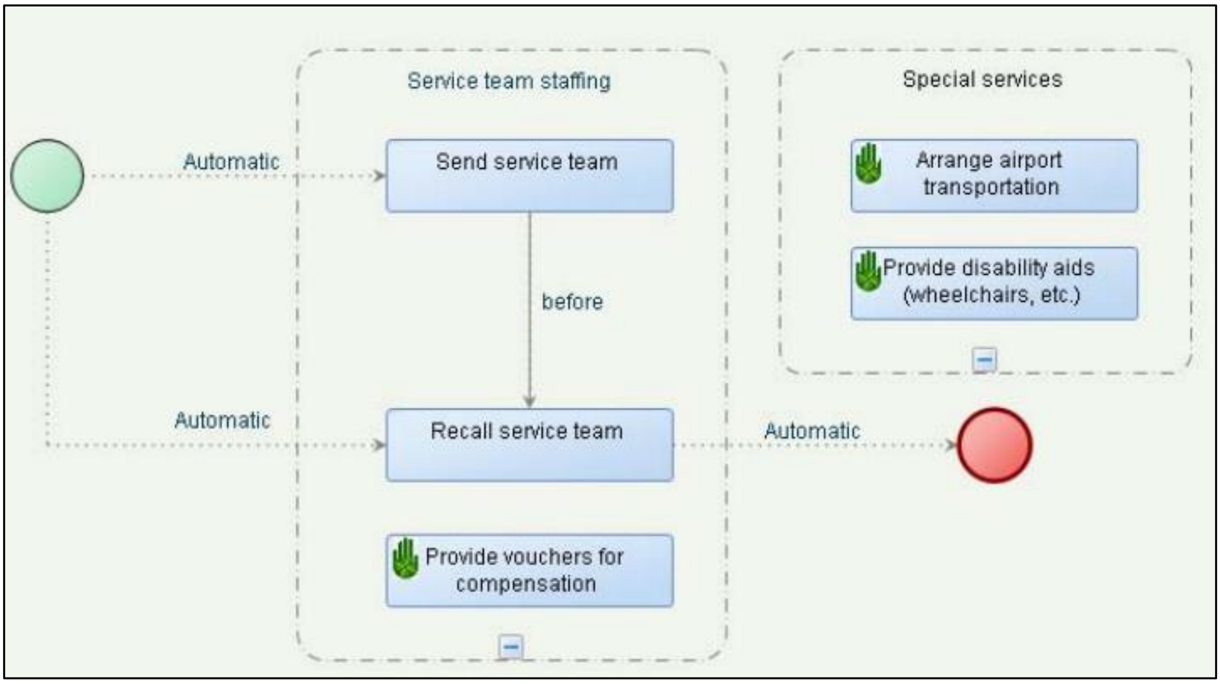

Service team staffing
Special services
Automatic
Send service team
Arrange airport transportation
Provide disability aids (wheelchairs, etc.)
before
Automatic
Recall service team
Automatic
Provide vouchers for compensation

Figure 34: Example of subcase in the Cordys Approach (Staffing trains) (de Man, 2009b, p. 4)

Figure 34 introduces the idea of an activity cluster (e.g. Service team staffing & Special services). They share great resemblance with the subcases in BPMN. The activity clusters serve several purposes. They allow functional grouping of activities, also in run-time (BPMN only allows the grouping of activities in design-time). The grouping of activities provides structure to the case worker's user interface (UI) and to process statistics in Business Activity Monitoring (BAM). "BAM provides runtime information about the business, allows real-time analysis of the processes, shows bottlenecks and unreliable tasks, measures time of each task and provides tools to visualize all that information." (Kolár, 2009, p. 21). The second purpose it serves is to cope with the efficiency in specification of the follow-up. It depends on the target whether an activity will be added to the case instance. If the entire cluster of activities is the target, all the activities that are part of the cluster will be added to the case instance (in run-time). When only one activity in the cluster is targeted, only that specific activity will be added to the case instance. In order to understand how subcases work in the 'Cordys Approach' this thesis will delve deeper into them.

The cluster (subcase) 'Service team staffing' contains two planned activities. They are automatically planned by the system. Unlike BPMN, this does not mean they have to be executed immediately. It's the responsibility of a case worker (here a calamity coordinator) to decide when and in which sequence the activities need to be executed. It's logical that a service team needs to be sent out first before they can be recalled. This implies a sequence restriction between these two activities, indicated by the connector (labelled 'before'). The sequence restriction signifies that these activities are connected and one can only be done when the other has been carried out. If a dotted line would be used here, there would be no sequence in which the case worker should perform the tasks. He could recall the service team without even sending it out, which is absurd. The restriction shows a lot of similarity with what has been seen in 6.5.2 'Declarative Configurable (DeCo) Process modelling notation'. This restriction however does not mean that the activities need to be executed immediately, one after the other. This is logical because the calamity coordinator should have the power to decide when the service team should be called back. In casu, this coincides with the use of a conditional intermediate event between activities in BPMN. Not that it can be replaced by the intermediate events in BPMN, because that would be conflicting with the philosophy of case management regarding run-time decision-making by case workers. When looking further into 'Service team staffing' it is possible to notice an ad hoc activity. These activities are typically executed in between. As soon as the service team has been called back, the subcase ends automatically, no matter what.

6.6.4 State-based approach

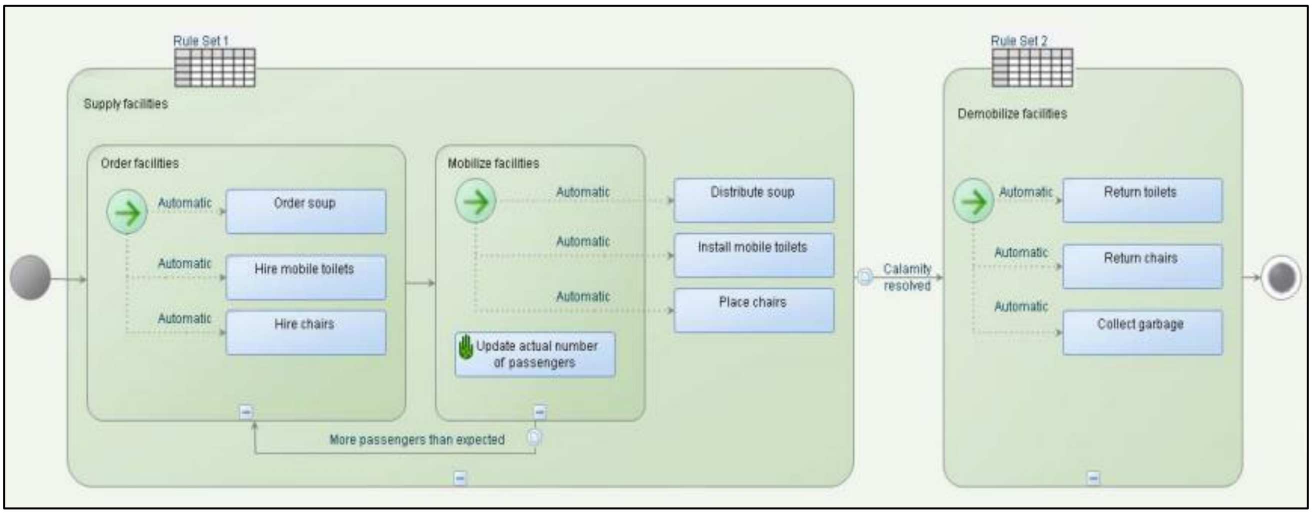


Figure 35: Example of subcase in the Cordys Approach ('Supply facilities station') (de Man, 2009b, p. 6)

Figure 35 introduces another subcase, 'Supply station facilities', with more modelling constructs. The subcase now has a life-cycle which translates into a state-machine in the background in run-time. The subcase starts in the state 'Order facilities'. In this subcase there are three activities that are automatically planned (design-time) as a follow-up on the entry in that state. The entry in a specific state is symbolised by an event icon with a green arrow. Note that these activities are prone to a set of rules (Rules Set 1). When the activities in this first state are completed, the case life-cycle moves on to the following state ('Mobilize facilities'). In this state, the activity 'Update actual number of passengers' can be performed by the calamity coordinator at any time. When this activity is executed, an event will be triggered (as seen at the bottom connector in Figure 35). As a result there will be a flow back to the 'Order facilities' state and extra facilities will be distributed accordingly. However, personnel should continue distributing the facilities that have already been ordered the first time. These activities should not be terminated while updating the needed facilities. When the calamity coordinator learns that all problems are resolved, he enters this knowledge into the case data and it triggers the 'Calamity resolved' event. At this point, any activities in the 'Supply facilities' state should be terminated.

Henk de Man (2009b) continuous by asking himself if the way of modelling activities in Figure 35 isn't characterised by a too low level of granularity. This is where he brings up the analogy again between shop-floor workers in manufacturing. He refers to Sly & Gopinath (2006) to make the connection with manufacturing process planning to make a distinction between the "smallest amount of moveable work" (tasks) (Sly & Gopinath, 2006, p. 11) and "the lowest amount of definable work" (task elements) (Sly & Gopinath, 2006, p. 11). Henk de Man (2009b) states that process control is concered with tasks, not with the elements that define the tasks. In a way this emphasises the similarity of this kind of modelling with BPMN. In BPMN tasks are modeled on an atomic level. This means tasks can not be divided further into other tasks, only work instructions. Generally, a task that can be executed by a single person is considered an atomic task. (De Backer, 2014). The activities modeled in Figure 35 can indeed be performed by a single person. "The criterion should be at which level human work should be identifiable for the purpose of standardization, balancing, scheduling, authorisation, auditing, progress monitoring and accounting." (de Man, 2009b, p. 6).

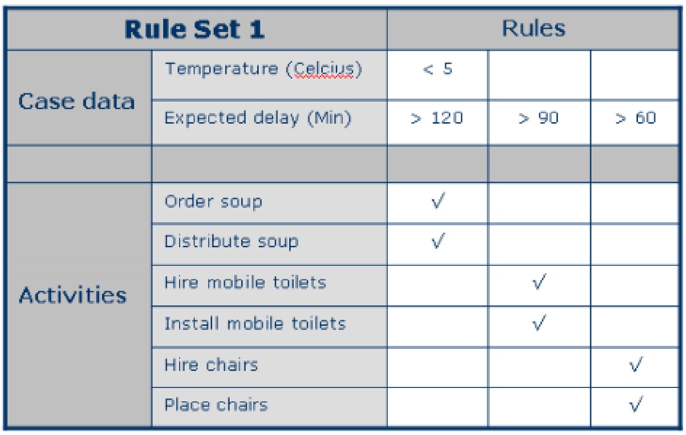
Henk de Man (2009b) explains the use of a state machine. Apparently they are not used only to show case milestones as tasks are completed but also to indicate the functional milestones a case can reach. The idea behind this is to enable planning and monitoring of cases, in accordance to Rooze et al. 's (2007) 'planning of milestones' as a coordination mechanism for regular cases. The other purpose of the state machine is to introduce restriction in the sequence of execution of the cases. Certain cases can only start after other cases have been completed and certain activities can only be executed while in a specific state. This kind of state machine semantics is applied to model case management behavior.

The last purpose of this state-machine approach is to identify the rules that have to be applied in a phase of the case life-cycle.

A state-based approach allows for using more types of rules like state invariants and state transistion guards as descibed by the Object Management Group (2015) in the 'OMG Unified Modelling Language Version 2.5'. A state invariant "specifies conditions that are always true when this state is the current state. In protocol state machines, state invariants are additional conditions to the preconditions of the outgoing transitions, and to the post-condition of the incoming transitions." (Object Management Group, 2015). In other words, a state invariant can express under which condition a certain state can be reached or maintained (de Man, 2009b). "A guard is a constraint that provides a fine-grained control over the firing of the transition. The guard is evaluated when an event occurrence is dispatched by the state machine. If the guard is true at that time, the transition may be enabled." (Object Management Group, 2015). In some cases, the use of state invariants and transition guards can be a valuable addition to the rules in a state-machine.

De Man (2009b) proposes to make use of state-based models in cases where there are no follow-up relationships. In these cases, all activities in a state are planned ad hoc and activities are filtered only based on their applicability. In my opinion, the added value from this kind of modelling would be to be able to track the urgency of a certain problem since it will transition between states depending on the ad hoc activities that need to be performed. When taking a look at the example (Figure 35), there is no reason to believe that the use of a state-machine is the wrong approach to modelling the calamity handling. State-machines and follow-up decisions can go hand in hand. Regular cases will mostly be modelled through follow-up decision because they are characterised by a certain degree of predictability. Exceptions or special events can trigger ad hoc activities that are modelled in the states next to the follow-up activities. Every type of modelling should be characterised by some sort of predictability, even in case management. To support this statement it is necessary to refer to the perfect flexibility discussed before. Without any kind of structure, in our case a follow-up structure, the model would be made up exclusively from ad hoc activities that are not connected. The result is that you will model virtually nothing because there are no constraints built in your model and that would go against the purpose of modelling.

6.6.5 Rules



Rule Set 1		Rules		
Case data	Temperature (Celcius)	< 5		
	Expected delay (Min)	> 120	> 90	> 60
Activities	Order soup	√		
	Distribute soup	√		
	Hire mobile toilets		√	
	Install mobile toilets		√	
	Hire chairs			√
	Place chairs			√

Figure 36: Rule Set in state 'Supply Facilities' (de Man, 2009b, p. 7)

Rules play an important part in the applicability of the activities. In this article de Man (2009b) talks about two kinds of rules: applicability rules and release rules. Applicability rules help to identify the activity that should be executed based on the conditions that are met. In a way they determine which activities should be considered for execution based on the case data (e.g. 'Hire mobile toilets' should only be considered for execution if the delay amounts to more than 90 minutes). Applicability rules are also incorporated similarly in the CMMN 1.0. In CMMN these rules determine whether or not an item, modelled in the case plan, is available for execution. In order to check the execution eligibility, the conditions are checked and evaluated based on information in the case file. Release rules are a bit more counterintuitive. They are used to determine whether or not conditions are met to actually execute the activity in question. If conditions are met, the activity is 'released' and executed. In a human case worker environment, the moment the activities are released, they are added into a work list of the corresponding case worker or team. In a sequential workflow, the release of an activity would lead to the automatic execution of that activity. A case worker stays in control over the execution of the activity. He/she can skip the activity when they are considered as 'not required'. He/she can also repeat an activity when the outcome was not satisfactory. They can be repeated as many times as necessary.

Figure 36 gives an overview of the rules applied on the 'Supply facilities' subcase (Rule Set 1) in the format of a decision table. A set of rules is connected to the subcase or state an activity belongs to. The rules illustrate ongoing behaviour as long as this behaviour applies to activities within that specific state. Rules from Rule Set 1 only apply to the activities in the 'Supply facilities' state. Figure 36 perfectly demonstrates that rules depend on the case data generated throughout the case life-cycle. The same goes for rules defined in the CMMN 1.0. Rules in CMMN 1.0 depend on the case file data. If there is a calamity and temperatures drop below 5 degrees Celsius and delays are expected to be longer than 120 minutes, action has to be taken in order to provide stranded passengers with soup. The thing this decision table teaches the most, is the fact that it is a perfect illustration of how case data, converted into the case file, translates into the central artefact case management is built on. Decision tables don't need to be used exclusively with state machines. They can be used together with activity clusters. When used in combination with activity clusters, the decision table is modelled through a similar graphical representation. The only difference is that the rule will not apply to a certain state of a case but to the activities contained in the cluster. Rules can also be attached to the case itself, meaning that they apply to the entire case life-cycle.

Henk de Man (2009b) tests the use of rules in this approach to the use of rules in decision gateways in BPMN. Although they seem to pursue the same goal, they are still quite different he finds. BPMN decision gateways are part of a predefined sequence flow. This is where case management distinguishes itself from BPMN again. Decisions are often made in the heat of the moment, without anybody being able to foresee where a certain decision will take place in design-time. Decision gateways on the other hand are predefined in a specific location of the activity sequence paths. The whole idea behind case management is that it is not possible to plan or model predefined paths. Case data is subject to its environment and will change accordingly. In our example, the number of stranded passengers is an environmental factor case workers have no control over. The number can go up, but it can also go down. In the case, it is necessary to just observe that environment. These observations translate into case data and with the help of rule sets, action can be taken in accordance with the needs our observations identified. The reason why decision gateways would fail here is because of the required continuous rule-evaluation on the central case data-context, in order to be able to accommodate the unpredictability of calamity situations. Once a decision gateway has been passed, you could model a sequence flow for re-evaluation after the next activity (activities) has (have) been executed. Rigidity of the model will always be a problem with similar modelling. In some cases, modelling re-evaluation with sequence flows can work perfectly. In processes where re-evaluation is limited or performed by an automated control system, sequence flows can work perfectly. In cases where evaluation has to be available in any stage of the process, sequence flows would quickly overflow the model, increasing its complexity and reducing its readability. Also in cases where knowledge workers have or need the authority to carry out evaluations when they see fit, sequence flows are not advisable since they do not offer the flexibility needed.

6.6.6 Future model advancements

Henk de Man (2009b) finishes his Cordys Approach to Case Management with further model advancements. In this part, he goes over the possibilities state-based modelling offers to case data and case UI. Case data or properties can be enabled according to the data control applied, based on the phase the case has reached in the case life-cycle. In his opinion, four aspects of case data (properties) can be controlled in a state.

Firstly, whether a property is applicable in a certain state of the model. Some properties are not relevant yet in early stages of the model but can become important later on. Secondly, whether a property is required. The philosophy behind the first aspect also applies to the second one. Requirements can differ throughout the states of the model. Thirdly, whether a property should be shown on the UI of the case and on case activity UI's. And fourthly, whether or not values of a property have to be tracked and in which states. He illustrates the importance of tracking properties in a B2B environment with the properties of an order. Delivery time, price, quantity… all are important properties of an order that need to be tracked in order to successfully complete the order. This data is important to perform business performance analysis.

When talking about case UI's, it is very intuitive to understand that every state can have its own interface. A case UI in a state-based approach should support state-transitions based on roles and authorisations. Only specific roles should have the authorisation to execute a transition into another state. What Henk de Man (2009b) did forget to mention is not only the weight of case data as a driver for state transitions, but also the link with the knowledge worker data (e.g. info about authorisations, access levels, roles…). If this is not included in the model, role-based transitions are either not applied and everybody has the authorisation to execute transitions (which can be dangerous in some cases), or they are present but not made clear so the model is prone to misinterpretation and confusion. He does address part of the problem of knowledge worker info, particularly team info when talking about work distribution. For global organisations, similar cases can be operated by teams from Europe, Asia, South or North America. Which teams have to execute which cases or case activities have to be defined in the model. Forecasting can be used to meet the volatility of workloads. "Work distribution modelling is concerned with defining scenarios of case work balancing." (de Man, 2009b, p. 11). De Man (2009b) introduces an approach whereby work balances are defined per case model. Case models that are characterised by similar case work can often be classified in a case model family. The approach implies the identification of different scenarios. For every case model family, work assignment scenarios are defined. Modelling this concept into case management is not shown by Hen de Man (2009b).

6.6.7 Summary

To wrap this case study up, there will be a brief summary of the most important elements to increase overall understanding of the characteristics, concepts and results as shown in this case study:

- Process planning is shifted from design-time to run-time in case management.

- Use of 'follow-up' instead of sequential flows like in BPMN. A follow-up construction does not imply a sequence of execution (it is possible though).
- Case workers perform process planning as part of their regular job. They can plan next activities based on a manual selection of follow-up activities in the case (activity) UI.
- Since there is no fixed sequence of activities it is possible for a case worker to plan a follow-up on a running activity (before the activity is completed). This is also called an intermediate follow-up. Such a new follow-up decision is added to the case instance.
- Follow-up activities triggered by an ad hoc activity do not have to be shown in the diagram (this happens in the background). In other words, ad hoc activities do not need to be modelled in a follow-up relationship.
- A follow-up is not an explicit relationship. The follow-up is possible to any activity modelled in the diagram. New follow-up just need to be added in the case instance as mentioned before.
- Ad hoc activities can't be compared to ad hoc subprocesses in BPMN. The reason behind this is explained extensively in the overview.
- The focus is not on modelling perfect flexibility. That would result in a series of unconnected activities where basically nothing would have been modelled. Instead the focus lies in constrained flexibility. Flexibility should at least be prone to some guidelines and limitations.
- Follow-up can be filtered by applicability rules. Execution of activities can be restricted by release rules.
- Sequential workflows are compatible with this case management approach. They can be triggered from within the case process.
- What activities in an activity cluster (subcase) are triggered and added to the case instance depends on the target activity.
- Connectors in this model imply restrictions but that does not mean the activity needs to be executed immediately after the previous one (like the connectors in BPMN).
- The execution of certain activities is always decided upon by the case workers. The case worker has the final decision.
- All results and outputs from executing activities are gathered in the case data. Based on this case data, following activities are started. If an event triggers the transition to another state, the activities in the previous state should be terminated.
- The use of a state-machine to model subcases is intuitive and very natural.
- Modelling constructs can be combined and used in any diagram. Their combination and complexity depend on the problem the designer wants to model.
- Use a state-based model without follow-up relations in situations where all activities within a state are planned ad hoc and activities are only filtered based on applicability. This is probably

done to classify calamities (I use the example used by Henk de Man (2009b)) based on the ad hoc activities into different states of urgency.

- Case workers should operate in a team, overseeing the entire case.

7 Conclusion

The goal of this master thesis was to introduce the concept of case management, both understandable for practitioners and valuable for researchers. Case management will be one of the pillars of process management in the next decade. Vouching for this statement are the efforts made by corporations like Oracle, IBM, Kana Software, Isis Papyrus, Appian and OpenText (Le Clair & Miers, 2014) to incorporate case management concepts into their products. These vendors will play an important role in the future development of the case management market as has been become clear from the 'Challenges' section. Making businesses more aware of what case management actually is and what it can do to improve operations will be a first step to counter this challenge. On the long run, the growing maturity and evolutions in the CMMN standard will balance this problem out. Companies will still need to be on the lookout for the traditional limitations and challenges and be aware of the ones that have yet to arise. Throughout this thesis, the artefact-based approach has proven the most adequate to model case management. The communication-based approaches give some valuable insights and underlying concepts but they have not been executed properly, resulting in models that fail to model case management in a proper manner. The artefact-based approach on the other hand, succeeds in combining the core elements in case management as well as its most distinct characteristics. The most explored artefact-based case management approach will most likely be the one leaning on BPMN. With DeCo and CMMN 1.0, there have been taken serious steps towards a bpm-based case management system. The 'Cordys Approach' showed that even though there was no standard introduced, BPM has been intuitional to model case management.

But, there is still a long way to go before CMMN 1.0 can profit from the same popularity and maturity as BPMN 2.0. Knowledge Management displays resemblance with case mangement, but Henk de Man (2009a) fails to adress knowledge mangement as a possible apporach to case magnement in his paper. In the future, case management systems will also need to pay attention to the overall visibility a model has to offer. With a state-based approach to case management, it becomes possible to structure unstructured processes better. User interfaces that structure different states and their accompanying rules, tasks, activities and decisions will improve the visibility and readability of the whole case. This can only be guaranteed on the condition that the case architecture is properly modelled. Finally, there has to be some way to model work distribution and knowledge worker specific data to support it to cope with global business environments.

Hopefully, this thesis will have provided an informative overview of the case management concept, its core elements, benefits, challenges and approaches. It will be interesting what the future will bring for the case management concept. Until then, businesses that rely on knowledge-intensive business processes will need to help themselves with what products are available.

8 References

Aamodt, A., & Plaza, E. (1994). Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *AI Communications, 7*(1), 39-59.

Abecker, A., Bernardi, A., Hinkelmann, K., Kuhn, O., & Sintek, M. (2000). Context-aware, proactive delivery of task-specific knowledge: The KnowMore project. *International Journal on Information Systems Frontiers (ISF), Special Issue on Knowledge Management and Organizational Memory, 2*, 139-162.

Adams, M., & Oleksak, M. (2010). *Intangible Capital: Putting Knowledge to Work in the 21st-Century Organization.* Praeger.

Alavi, M., & Leidner, D. E. (2001). Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. *MIS Quarterly, 25*, 107-136.

Allen, R. (2000). Workflow: an introduction. In L. Fischer, *The Workflow Handbook 2001* (pp. 15-38). Lighthouse Point, FL.

Becker, J., Rosemann, M., & Von Uthmann, C. (2000). Guidelines of Business Process Modeling. (W. van der Aalst, J. Desel, & A. Oberweis, Eds.) *Business Process Management: Models, Techniques and Emperical studies, LNCS 1806*, 30-49.

Berry, P. (1998). *Intelligent workflow - State of the Art in Workflow.*

Bhattacharya, K., Guttman, R., Lyman, K., Heath III, F. F., Kumaran, S., Nandi, P., . . . Staudt, A. (2005). A model-driven approach to industrializing discovery processes in pharmaceutical research. *IBM Systems Journal, 44*(1), 145-162.

Bhattacharya, K., Caswell, N. S., Kumaran, S., Nigam, A., & Wu, F. Y. (2007). Artifact-centered operation modeling: Lessons from customer engagements. *IBM Systems Journal, 46*(4), 703-721.

Bilinsky, D. (2002, April). *25 Benefits of Case Management in 40 Minutes*. Retrieved July 9, 2015, from The Law Society of British Columbia: https://www.lawsociety.bc.ca/page.cfm?cid=1503&t=25-Benefits-of-Case-Management-in-40-Minutes

Billington, J., Jensen, K., & Koutny, M. (2009). *Transactions on Petri Nets and Other Models of Concurrency III.* Springer.

Botha, A., Synman, R., & Kourie, D. (2008). *Coping With Continuous Change in the Business Environment: Knowledge Management and Knowledge Management Technology.*

Brinkley, I., Fauth, R., Mahdon, M., & Theodoropoulou, S. (2009). *A Knowledge Economy Programme Report.* London: The Work Foundation.

Bukowitz, W. R., & Williams, R. L. (1999). Knowledge management process framework. In *The Knowledge Management Field Book* (pp. 9-12).

Bushell, S. (2005). *A Review of Two Books Describing a New Foundation for Business Process Management.*

Capgemini. (2012). Managing chaos: unstructured processes and dynamic BPM. *Capgemini BPM Trends*.

Case Management Society of America. (2009). *Standards of Practice for Case Management.* Little Rock: CMSA.

Case Management Society of America. (2010). *Standards of Practice for Case Management.* Little Rock: CMSA.

CDS. (2003). *Explicit and Tacit Knowledge, Cognitive Design Solutions*. Retrieved July 9, 2015, from http://www.cognitivedesignsolutions.com/KM/ExplicitTacit.htm

Chang, R., Chang, Y., & Paper, D. (2003). The Effect of Task Uncertainty, Decentralization and AIS Characteristics on the Performance of AIS: An Empirical Case in Taiwan. *Information & Management, 40*, 671-703.

Clément, L., König, D., Metha, V., Mueller, R., Rangaswamy, R., Rowley, M., & Trickovic, I. (2010). *WS-BPEL Extension for People (BPEL4People) Specification Version 1.1: Committee Specification.* OASIS BPEL4People TC.

Cohn, D., & Hull, R. (2009). *Business Artifacts: A Data-centric Approach to Modeling Business Operations and Processes.* IBM T.J. Watson Research Laboratory. Hawthorne: IBM.

Davenport, T. H. (1994). Saving IT's Soul: Human-Centered Information Management. *Harvard Business Review*, 119-131.

Davenport, T. H. (2005). *Thinking for a living: how to get better performance and results from knowledge workers.* Harvard Business Press.

Davenport, T. H. (2010). Process Management for Knowledge Work. In J. vom Brocke, & M. Rosemann, *Handbook on Business Process Management 2.* Heidelberg: Springer.

Davenport, T. H., & Prusak, L. (1997). *Information ecology : mastering the information and knowledge environment.* New York: Oxford University Press.

Davenport, T., & Harris, J. (2005). Automated Decision Making Comes of Age. *MIT Sloan Management Review*, 83-89.

De Backer, M. (2014). Business Process Modeling & BPMN 2.0 [Powerpoint Presentation].

de Man, H. (2009a). Case Management: A Review of Management Approaches. *BPTrends*.

de Man, H. (2009b, February). Case Management: Cordys Approach. *BPTrends*.

Dietz, J. L. (1999). *Understanding and Modeling Business Processes with DEMO.* Delft.

Dietz, J. L., & Halpin, T. (2004). Using DEMO and ORM in Concert: A Case Study. *Advanced Topics in Database Research, 3*, 218-236.

Doganov, K., Haralanova, C., & Lutfy, M. (2005). *Legal Case Management for not-for-profit organisations.*

Dong, G., Hull, R., Kumar, B., Su, J., & Zhou, G. (2000). A Framework for Optimizing Distributed Workflow Executions. In *Research Issues in Structured and Semistructured Database Programming* (pp. 152-167). Heidelbert: Springer Berlin Heidelberg.

Dumay, M. (2005). Demo or Practice: Critical Analysis of the Language/Action Perspective. Delft, The Netherlands.

Dumay, M., Dietz, J. and Mulder, H. (2005). Evaluation of DEMO and the Language/Action Perspective after 10 years of experience.

Earls, A. (2011, February 18). *BPMN 2.0: The emerging star of business process modeling*. Retrieved July 12, 2015, from ebiz: http://www.ebizq.net/topics/bpm_process_modeling/features/13202.html

ECM². (2015). *Case Management*. Retrieved July 15, 2015, from https://www.emc.com/services/consulting/application/expertise/case-management.htm

Eppler, M. J., Seifried, P. M., & Röpnack, A. (1999). Improving Knowledge Intensive Processes through an Enterprise Knowledge Medium. *In Proceedings of SIGCPR*, 222-230.

Ferguson, D., & Stockton, M. (2005, June 14). *SOA programming model for implementing Web services, Part 1: Introduction to the IBM SOA programming model*. Retrieved February 18, 2015, from http://www.ibm.com/developerworks/library/ws-soa-progmodel/

Franke, J. (2011, July 27). *Case Management Process Modeling (CMPM) – An Introduction*. Retrieved February 2015, from http://scn.sap.com/people/joern.franke/blog/2011/07/27/case-management-process-modeling-cmpm-an-introduction

Gamble, P. R., & Blackwell, J. (2001). *Knowledge Management: A State of the Art Guide.* Kogan Page Publishers.

Gartner Inc. (2013). *Gartner's 2013 BPM Summit.* Maryland, US: Gartner Inc.

Goldkuhl, G. (2003). Conversational Analysis as a Theoretical Foundation for Language Action Approaches. *Proceedings of the 8th International Working Conference on the Language-Action Perspective on Communication Modelling*.

Hakim, A., & Hakim, H. (2010). A practical model on controlling the ERP implementation risks. *Information Systems, 35*(2), 204-214.

Hansen, M. R., Haxthausen, A., & Villadsen, J. (n.d.). Declarative Modelling. Lyngby, Denmark.

Harmon, P. (2006). *Review of "Working Minds": A practitioner's guide to cognitive task analysis by Beth Crandall, Gary Klein and Robert R. Hoffman.* BPTrends.

Harmon, P. (2007). *Business Process Change: a Guide for Business Managers and BPM and Six Sigma Professionals.* Morgan Kauffman.

Harrison-Broninski, K. (2005). *Human Interactions: The Heart And Soul Of Business Process Management: How People Reallly Work And How They Can Be Helped To Work Better.* Meghan Kiffer Pr.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly, 28*, 75-105.

Hill, J. B. (2007). *BPM: A Change From "Business as Usual".* São Paulo, Brazil: Gartner.

Hull, R., & Su, J. (1999). The Vortex Approach to Integration and Coordination of Workflows.

Hull, R., Llirbat, F., Simon, E., Su, J., Dong, G., Kumar, B., & Zhou, G. (1999). Declarative Workflows that Support Easy Modification and Dynamic Browsing. *WACC '99 Proceedings of the international joint conference on Work activities coordination and collaboration* (pp. 69-78). New York, NY: ACM.

IBM. (2015). *What is Business Rules Management?* Retrieved February 15, 2015, from http://www-01.ibm.com/software/websphere/products/business-rule-management/whatis/

Işık, Ö., Van de Bergh, J., & Mertens, W. (2012). Knowledge Intensive Business Processes: An Exploratory Study. *2012 45th Hawaii International Conference on System Sciences* (pp. 3817-3826). Maui, HI: IEEE.

Jaworowska, J. (2001). Speech Act Theory.

Kaan, K. (2005). *Modeling support for the Case Management paradigm and Expectations Management in process innovation.* Eindhoven: Technische Universiteit Eindhoven.

Karadsheh, L., Mansour, E., Alhawari, S., Azar, G., & El-Bathy, N. (2009). A Theoretical Framework for Knowledge Management Process: Towards Improving Knowledge Performance. *Communications of the IBIMA, 7,* 67-79.

Kethers, S., & Schoop, M. (2000). Reassessment of the Action Workflow Approach: Empirical Results. *Proceedings of the Fifth International Workshop on the Language-Action Perspective on Communication Modelling LAP*, 151-169.

Khvalev, E. A. (2004). *Key Risks in ERP Implementation: Identification and Analysis by Project Phases.* Nizhny, Novgorod, Russia: Institute of Radio Engineering and Information Technologies.

Kingston, J. K. (2012). Tacit Knowledge: Capture, Sharing, And Unwritten Assumptions. *Journal of Knowledge Management Practice, 13*(3).

Koenig, M. E. (2012, May 4). *What is KM? Knowledge Management Explained*. Retrieved July 14, 2015, from KM World: http://www.kmworld.com/Articles/Editorial/What-Is-.../What-is-KM-Knowledge-Management-Explained-82405.aspx

Kolár, J. (2009). *Business Activity Monitoring.* Brno: Masaryk University.

Kolodner, J. L. (1992). An Introduction to Case-Based Reasoning. *Artificial Intelligence Review, 6*, 3-34.

Kulkarni, U., & Ipe, M. (2007). Decision Support for Knowledge Intensive Business Processes. *Proceedings of the ADPSI Conference.*

Kwan, M. M., & Balasubramanian, P. (2002). KnowledgeScope: Managing knowledge in context. *Decision Support Systems, 35*, 467-486.

Le Clair, C., & Miers, D. (2014). *The Forrester Wave™: Dynamic Case Management, Q1 2014*. Forrester.

Le Clair, C., & Moore, C. (2009). *Dynamic Case Management - An Old Idea Catches New Fire.* Forrester.

Le Clair, C., & Moore, C. (2009). *Dynamic Case Management - An Old Idea Catches New Fire.* Forrester.

Leake, D. (1996). *Case Based Reasoning in Context: the Present and the Future.* AAI Press/MIT Press.

Levinsen, K. T. (2004). Collaborative On-Line Teaching: The ineveitable Path to Deep Learning and Knowledge Sharing? *The Electronic Journal of e-Learning, 2*(1), 41-48.

Liebowitz, J., & Wilcox, L. C. (1997). *Knowledge Management and its Integrative Elements.* US: CRC Press LLC.

List, B., & Korherr, B. (2005). An Evaluation of Conceptual Business Process Modeling. *SAC '06 Proceedings of the 2006 ACM symposium on Applied computing*, 1532-1539.

Lyytinen, K. (2004). The Struggle with the Language in the IT – Why is LAP not in the Mainstream?. *Proceedings of the 9th International Working Conference on the Language-Action Perspective on Communication Modelling (LAP 2004).*

Macintosh, A. (1999). *Adaptive work • ow to support knowledge intensive tasks.* Edinburgh: Artifcial Intelligence Applications Institute (AIAI).

Makhfi, P. (2011). *Introduction to Knowledge Modeling*. Retrieved July 14, 2015, from World of Knowledge Modeling: http://www.makhfi.com/KCM_intro.htm

Manolescu, D. (2001). *Micro-Workflow: A Workflow Architecture Supporting Compositional Object-Oriented Software Development.*

Manolescu, D. (2002). *Workflow Enactment with Continuation and Future Objects.*

Marjanovic, M., & Seethamraju, R. (2008). Understanding Knowledge-Intensive, Practice-Oriented Business Processes. *Proceedings of the 41st Hawaii International Conference on System Sciences.* Hawaii.

Marjanovic, O., & Freeze, R. (2011). Knowledge Intensive Business Processes: Theoretical Foundations and Research Challenges. *Proceedings of the 44th Hawaii International Conference on System Sciences.* Hawaii.

MarketsandMarkets. (2014). *Business Process Management Market by Solutions (Process Modeling, Automation, Integration, Content & Document Management, and Monitoring & Optimization*

Management), by End User (SMBs, Enterprises & Large Enterprises) - Global Forecast to 2019. marketsandmarkets.com. Retrieved June 25, 2015, from http://www.marketsandmarkets.com/PressReleases/business-process-management.asp

Medeiros, C. B., Vossen, G., & Weske, M. (1995). *WASA: A Workflow-Based Architecture to Support Scientic Database Applications.*

Miers, D. (2006). *Process Innovation and Corporate Agility: Balancing Efficiency and Adaptability in a Knowledge-Centric World.* BPMFocus.

Miers, D. (2007). *Business Processes and Customers – Difficult domains to integrate.* BPMFocus.

Milton, N. (2010). *The Lessons Learned Handbook: Practical approaches to learning from experience.* Chandos Publishing.

Natrella, M., Carroll, C., Tobias, P., Filliben, J. J., Hembree, B., Guthrie, W., . . . Prins, J. (2010). *NIST/SEMATECH e-Handbook of Statistical Methods.*

Nonaka, I. (1994). A Dynamic Theory of Organizational Knowledge Creation. *Organization Science, 5*(1), 14-37.

Object Management Group. (2010, February 9). Case Management Process Modeling (CMPM) - Request For Proposal. Needham, Massachusetts , USA.

Object Management Group. (2011). *Records Management Services (RMS).*

Object Management Group. (2015, March). OMG Unified Modeling Language (Version 2.5).

OMG. (2011, January 3). Business Process Model And Notation (BPMN) - Version 2.0.

OMG. (2014, May 10). Case Management Model and Notation. USA.

OMG. (2015, May 27). *About OMG®*. Retrieved July 20, 2015, from Object Management Group: http://www.omg.org/gettingstarted/gettingstartedindex.htm

Oracle. (2014). *Event Management Business Process*. Retrieved June 27, 2015, from http://docs.oracle.com/cd/E58407_01/crm92pbr5/eng/crm/cevm/concept_EventManagementBusinessProcess-867fda.html#topofpage

Oregon Evidence-based Practice Center. (2013). *Outpatient Case Management for Adults With Medical Illness and Complex Care Needs.* Rockville, Maryland: Agency for Healthcare Research and Quality.

Ould, M. A. (2004). Getting your head round spaghetti. *BPTrends*.

Ould, M. A. (2005). *Business Process Management: A Rigorous Approach.* BCS.

PAControl. (2006). Instrumentation & Control.

Pallas Athena. (2002). *FLOWer scoort uitstekend op workflow patterns.* Pallas Athena.

Pang, C., Dharmasthira, Y., Eschinger, C., Motoyoshi, K., & Brant, K. F. (2013). *Market Share Analysis: ERP Software, Worldwide, 2012.* Gartner Inc.

Panian, Z. (2004). *Knowledge Management: A Learning Organization Model.* Zagreb, Croatia: University of Zagreb.

Panian, Z. (2011). A Promising Approach to Supporting Knowledge-Intensive Business Processes: Business Case Management. *World Academy of Science, Engineering and Technology*, 468-474.

Papavassiliou, G., & Mentzas, G. (2003). Knowledge Modelling in Weakly-Structured Business Processes. *Journal of Knowledge Management*, *7*(2), 18-33.

Peacock, M. (2011, February 15). *What is Driving Dynamic Case Management?* Retrieved July 8, 2015, from CMS Wire: http://www.cmswire.com/cms/information-management/what-is-driving-dynamic-case-management-010196.php

Peixoto, C., Batista, V., Atayde, A., Borges, E., Resende, R., & Pádua, C. (2008). *A Comparison of BPMN and UML 2.0 Activity Diagrams.* Belo Horizonte.

Polancic, G. (2014, December, 8). Conversation vs. Collaboration vs. Choreography. [Blog post]. Retrieved 12 July, 2015 from http://blog.goodelearning.com/bpmn/conversation-vs-collaboration-vs-choreography/

Reijers, H. A. (2002). *Design and Control of Workflow Processes: Business Process Management for the Service Industry*. Eindhoven: Technische Universiteit Eindhoven.

Reynolds, M. (2011, February 21). *Repeatable or reproducible? The right process to achieve your goals*. Retrieved July 8, 2015, from Sales Operations Blog: http://salesoperationsblog.com/2011/02/21/repeatableprocesses/

Richter-von Hagen, C., Ratz, D., & Povalej, R. (2005). Towards Self-Organizing Knowledge Intensive Processes. *Journal of Universal Knowledge Management, 0*(2), 148-169.

Rittgen, P. (2004). Supporting Planned and Ad-Hoc Changes of Business Processes

Rooze, E. J., Paapst, M., & Sombekke, J. (2007). *eCase Management, An international study in judicial organizations.*

Rychkova, I., & Nurcan, S. (2011). Towards Adaptability and Control for Knowledge-Intensive Business Processes: Declarative Configurable Process Specifications. *Proceedings of the 44th Hawaii International Conference on System Sciences* (pp. 1-10). Hawaii: IEEE.

Rychkova, I., Regev, G., & Wegmann, A. (2008). Using Declarative Specifications In Business Process Design. *International Journal of Computer Science and Applications, 5*(3b), 45-68.

Sarnikar, S., & Deokar, A. (2010). Knowledge Management Systems for Knowledge-Intensive Processes: Design Approach and an Illustrative Example. *Proceedings of the 43rd Hawaii International Conference on System Sciences.* Hawaii.

Schäl, T. (1998). *Workflow management systems for process organisations.* Heidelberg: Springer-Verlag Berlin.

Schlenoff, C., Knutilla, A., & Ray, S. (1996). *Unified Process Scpecification Language: Requirements for Process Modeling.* NIST.

Schlosser, S., Baghi, E., Otto, B., & Oesterle, H. (2014). *Toward a Functional Reference Model for Business Rules Management.* System Sciences (HICSS).

Schooff, P. (2015, April, 28). What Are the Biggest Challenges Facing Case Management Today? [Blog post]. Retrieved 10 July, 2015 from http://bpm.com/bpm-today/in-the-forum/what-are-the-biggest-challenges-facing-case-management-today

Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., van de Velde, W., & Wielinga, B. (1999). *Knowledge Engineering and Management: The CommonKADS Methodology*. Cambridge, MA: The MIT Press.

Sidorova, A., & Işık, Ö. (2010). Business process research: a cross-disciplinary review. *Business Process Management Journal, 16*(4), 566-597.

Slembek, I. (2003). An evaluation framework for improving knowledge-intensive business processes. *Proceedings from IEEE 14th International Workshop on Database and Expert Systems Applications*, 813-817.

Sly, D., & Gopinath, P. (2006). *Practical Approach to solving Multi-objective Line Balancing Problem.* Proplanner.

Sternberg, R. J., Forsythe, G. B., Horvath, J. A., Snook, S., Wagner, R. W., Hedlund, J., . . . Grigorenko, E. L. (1999). *Tacit Knowledge in the Workplace.* Alexandria, Virginia: U.S. Army Research Institute for the Behavioral and Social Sciences.

Strahonja, V. (2006). Modeling Legislation by Using UML State Machine Diagrams.

Sumner, M. (2000). Risk factors in enterprise-wide/ERP projects. *Journal of Information Technology, 15*(4), 317-327.

Swanson, K. D., & Farris, J. (2015, September 17). *When Thinking Matters in the Workplace.* Retrieved July 17, 2015, from social-biz.org: http://social-biz.org/2012/09/17/imperative-vs-declarative-processes-models/

TechTarget. (n.d.). *Bill of Materials (BoM).* Retrieved February 27, 2015, from http://searchmanufacturingerp.techtarget.com/definition/bill-of-materials-BoM

TechTarget. (n.d.). *Dynamic Case Management*. Retrieved February 15, 2015, from http://whatis.techtarget.com/definition/dynamic-case-management-DCM

Tegarden, D., Dennis, A., & Wixom, B. H. (2013). *System Analysis and Design with UML: An Object-Oriented Approach.* Singapore: John Wiley & Sons, Inc.

Trkman, P. (2010). The critical success factors of business process management. *International Journal of Information Management*, 125–134.

Tscheschner, W. (2008). *Transformation from EPC to BPMN.* Potsdam, Germany: Hasso-Plattner-Institute.

van der Aalst, W. (2013). Business Process Management: A Comprehensive Survey. *ISRN Software Engineering*.

van der Aalst, W. M., & Berens, P. S. (2001). Beyond Workflow Management: Product-Driven Case Handling. *GROUP '01 Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work* (pp. 42-51). New York, NY: ACM.

van der Aalst, W., Weske, M., & Grünbauer, D. (2005). Case handling: a new paradigm for business process support. *Data & Knowledge Engineering*, 129-162.

Vanderfeesten, I., Reijers, H. A., & van der Aalst, W. M. (2006). Case Handling Systems as Product Based Workflow Design Support.

Wallace, M. (2012). *Evaluation of the Supported Decision Making Project.*

Wang, J., & Kumar, A. (2005). A Framework for Document-Driven Workflow Systems.

Weigand, H. (2005). LAP: 10 years in retrospect.

Whibley, P. (2015, April, 28). [Comment to blog post]. Schooff, P. (2015). What Are the Biggest Challenges Facing Case Management Today?. *BPM Today*. Retrieved 10 July, 2015, from http://bpm.com/bpm-today/in-the-forum/what-are-the-biggest-challenges-facing-case-management-today

White, M. (2009). *Case Management: Combining Knowledge with Process.* BPTrends.

Yockelson, D. (n.d.). *Advanced case management and BPM: Better together*. Retrieved July 8, 2015, from IBM: http://www-01.ibm.com/software/solutions/soa/newsletter/june10/advanced_case_management.html