How to Manage and Model Unstructured Business Processes: A Proposed List of Representational Requirements

Zaharah Allah Bukhsh[1](✉), Marten van Sinderen[2], Klaas Sikkel[2], and Dick Quartel[3]

[1] Department of Construction Management and Engineering, University of Twente, Drienerlolaan 5, 7522 NB Enschede, The Netherlands
z.allahbukhsh@utwente.nl

[2] Department of Computer Science, University of Twente, Drienerlolaan 5, 7522 NB Enschede, The Netherlands
{m.j.vansinderen,k.sikkel}@utwente.nl

[3] BiZZdesign BV, Capitool 15, 7521 PL Enschede, The Netherlands
d.quartel@bizzdesign.com

Abstract. Recent advancements in technology have enabled businesses to automate their structured business processes, thus requiring minimum intervention from end-users. This has shifted attention towards less structured processes, which are ad-hoc, often undocumented and demand frequent human decision-making. These processes are referred to as Unstructured Business Processes (UBP). Currently available tools and technologies are mainly focused on structured processes and therefore not optimally suited for management of UBP. With a representative example, we performed an experiment to compare and assess the ability of existing process support paradigms, i.e. Business Process Management and Case Management, to manage UBP. Moreover, we also investigated the limitations of Business Process Model and Notation (BPMN) and Case Management Model and Notation (CMMN) for modeling UBP. Based on our findings, a set of requirements are derived that are needed for optimally managing and modeling UBP. These requirements allow to express end-to-end business processes while providing flexibility for runtime changes. The requirements are also demonstrated with a possible extension of BPMN.

Keywords: Business Process Management · Case Management
Business Process Model and Notation
Case Management Model and Notation · BPMN · CMMN
Unstructured Business Process · Flexibility

1 Introduction

Business Process Management (BPM) has popularized the concept of business process automation, optimization and monitoring. The purpose of BPM is to innovate, maintain and optimize the business process by defining, modeling and

M. S. Obaidat and E. Cabello (Eds.): ICETE 2017, CCIS 990, pp. 81–103, 2019.
https://doi.org/10.1007/978-3-030-11039-0_5

automating it at design time. The design-time defined process is executed any number of times with various process instances. In BPM, it is assumed that each process instance has the same characteristics and will follow the process exactly in a manner that is defined at design time. However, not all the business processes can be planned and executed as defined at design time. According to a report by AIIM [19], for 51% of the companies polled, more than half of their business processes are unstructured and unpredictable in nature. Various studies [2,15,18] have defined the classification of business processes based on their level of structuredness. A business process having an ordered set of planned activities which are defined at design time, is said to be a *Structured Business Process (SBP)*. While a business process which depends on real-time events, available data and knowledge of knowledge workers is referred as an *Unstructured Business Process (UBP)*.

Companies adopt various methodologies (e.g., in-house collaborative systems, process management suites, etc.) to deal with the shift in focus from structured to unstructured business processes. Traditionally, UBP are dealt with a structured way [10]. For example, a business process is modeled at design time using Business Process Model and Notation (BPMN) while Business Process Management Suite (BPMS) implements the designed business process. Such process automation provides efficiency, however, it limits the process engineer to predefined activities and conditional flows.

Considering these limitations, some new and/or modified process management paradigms and modeling languages have been suggested that are specifically targeted to provide the flexibility for management of UBP. van der Aalst et al. [3] proposed case handling/management as a new paradigm to deal with UBP. To support the dynamic nature of business processes, a number of new modeling constructs were added in the BPMN v2.0 release [21]. Moreover, OMG proposed a new modeling language called Case Management Model and Notation (CMMN) for modeling processes where the process activities depend on real-time evolving circumstances [22]. The availability of a number of process modeling paradigms, with their advertised vendor solutions, pushes companies to rethink their tools that are used for process management. On one hand, BPMN is usually preferred since it is widely adopted and understood as an industry standard. On the other hand, the new proposed modeling language (i.e. CMMN) is attractive since it promises an increased level of expressibility for modeling of evolving business processes. The current scientific literature on process modeling languages lacks a comparison and capability assessment of BPMN and CMMN for UBP. However, a number of online discussions[1,2] and a recently published study by Hinkelmann [14] suggests the integration of BPMN and CMMN for improved process modeling benefits.

This study intends to fill the gap in the scientific literature by assessing the modeling capabilities of BPMN and CMMN with respect to UBP. Similarly, a comparison of the existing process support paradigms, i.e. BPM and Case

[1] https://www.linkedin.com/groups/1175137/1175137-5868060474150502404.

[2] http://brsilver.com/bpmn-cmmn-compared/.

Management (CM) is also performed in order to assess their support for UBP. The result of this study can assist companies, and specifically their process engineers and process consultants, in making a careful selection of the most suitable modeling and management paradigm for the process at hand by considering its requirements. Therefore, a number of representational and management requirements has been derived from literature. We believe, a process modeling language that is able to fulfill these representational requirements can model the SBP and UBP, while keeping their run-time flexibility. The work presented in this chapter is an extension of a previous study [5].

The rest of this chapter is structured as follows: characteristics of UBP are provided in Sect. 2. The details of an experiment that assess the capabilities of process support paradigms for UBP is provided in Sect. 3. To assess the capabilities of modeling languages proposed by OMG, a sample business process is modeled with BPMN and CMMN in Sect. 4. Based on the results of the capability assessment, a number of representational requirements for UBP are derived in Sect. 5. The representational requirements are demonstrated by means of an application scenario in Sect. 6. The validation of representational requirements with three business process modeling experts is presented in Sect. 7. Finally, Sect. 8 provides our conclusion.

2 Properties of UBP

Many literature studies have discussed the characteristic of UBP under the title of case management [9,16,20,31]. Following are some of the aspects of UBP which make them different from SBP.

Data Dependent: In UBP, process and data are strictly integrated which makes them data dependent [8]. The modification, addition or deletion of process data defines the future activities of the process. However, the unavailability of particular data may halt the processing of the whole process.

Goal Oriented: UBP are goal oriented, which means a process evolves through a series of sub-goals and milestones [9]. The achievement of each goal depends on a number of factors, e.g. availability of required data, execution of activities, decisions of knowledge workers, and responses from customers. Every sub-goal of a process is well-integrated with one final goal. An achieved subgoal can be modified or proven wrong as more data and knowledge emerges as the process progresses [20].

Business Rule Driven: Conformance to business rules and standards is one the most convincing arguments to automate a business process. However, due to the uncertain and emergent nature of UBP, knowledge workers are required to maintain the business rules and standards during process execution. All the process activities are influenced by particular rules and policies of business [9].

Coordination and Collaboration: Execution of UBP highly relies on the coordination and collaboration among the knowledge workers [20]. Usually, a single process involves many knowledge workers [9]. As the process progresses, new knowledge workers may get involved or existing knowledge workers may leave their roles.

To sum up, it is argued that in SBP the predefined routing rules drive the process while in UBP the characteristic of the particular process instance drive the process [1]. UBP requires tacit knowledge, collaboration and decision making skills from knowledge workers. The knowledge work of an organization cannot be straight-jacketed into an automated process and electronic forms due to its unstructured and evolving nature [3]. Eshuis et al. [11], suggested an approach to convert the UBP to SBP to be able to model them with imperative modeling languages.

3 Process Support Paradigm for UBP

For effective resource utilization, business organizations employ various methods, techniques and methodologies to optimize their business operations. Due to various types of process, process support and process improvement paradigms, an organization has to make a decision on which process support paradigm is more useful in dealing with a particular type of business processes. In this section, two process support paradigms are compared to better understand their strengths and weaknesses in managing UBP. For comparison purposes, an experiment is conducted. The experiment will provide us insight about which process support paradigm is more efficient and easy to use for implementing and maintaining an UBP.

3.1 Experiment Setup

Business Process Management (BPM) and Case Management (CM) are process support paradigms with difference in focus. An experiment is planned to assess and compare the capabilities of BPM and CM to manage UBP. The approach of the experiment is adopted from [30]. This section details the experiment setup, factors and factor level and threats to validity.

1. Subject: The purpose of this experiment is to understand the BPM and CM methodological differences while dealing with an UBP. Considering the purpose of this experiment, the subject of this experiment is only one researcher who is also responsible for experiment setup as well as for experiment execution.

2. Object of Study: The object of study is the admission of a student to a university, which is, to some extent, a UBP. The detailed process description is provided in Sect. 4.1.

3. Factors and Factor Levels: The factor or independent variable of this experiment is process support paradigm with two factor levels which are BPM and CM. Bizagi modeller and Bizagi Studio [6] are selected as BPMS to analyse the BPM capabilities. According to the Gartner magic quadrant [26], Bizagi is one of the visionary vendors of BPMS. As a CM, the Cognoscenti software tool is used. Cognoscenti [27] is one of the few CM tools that is available free of cost for research purposes [29].

4. Response Variable: The response or dependent variable of this experiment is ‘effort of implementing an unstructured process’. The usage effort will be measured by implementing an admission process with Bizagi (BPM) and with Cognoscenti (CM).

5. Analysis Procedure: The analysis procedure provides the design of experiment. It is difficult to quantitatively gauge the effort of implementing UBP as well as to assess the differences between the BPM and the CM process management methodology. Hence, certain aspects of process management are proposed for analysis purposes. These aspects include (a) process/case modeling, (b) data management (c) business rules specification (d) user roles specification (e) process/case progress view (f) process/case control (g) activities execution and (h) process/case setup effort.

6. Threats to Validity: We identified the following threats to validity for this experiment:

1. It is planned to assess the difference in methodology of BPM and CM based on their software suites. However, the platform-specific features provided by these suites can influence the results of this experiment.
2. The experiment has very limited subjects, i.e. only one researcher.
3. The object of this study, i.e. the admission process, can have possible biases for one process support paradigm over the other.

To mitigate the effect of these threats, some additional steps are performed. For example, Bizagi provides the functionality of process modeling while Cognoscenti lack this feature. To bring balance of functionality between both platforms, a Microsoft Visio stencil was created and used for modeling the case using CMMN. The second threat of validity is mitigated by presenting the results of the experiment to two experienced BPM practitioners as well as to three experienced researchers. The third and final threat of validity is difficult to mitigate. As depending on the particular process instance and the choices made by designer/modeler, the admission process can be more structured or unstructured.

3.2 Experiment Execution

The implementation procedure of the admission process can be broadly divided into design-time planning and run-time execution. In design-time planning of the process, the process is defined and modeled. While, in run-time a process instance is created as a result of process initiation. In the following paragraphs, the implementation of the admission process is discussed with the software suites of Bizagi (BPM) and Cognoscenti (CM).

Admission Process with Bizagi (BPM). For the implementation of the admission process in Bizagi, process modeling is the first and most pivotal part of the process management life cycle. Based on the process model, the data model, business rules, user roles and electronic forms are designed. After deployment of the designed process, the process can be accessed on the work portal of Bizagi. On the work portal, a process instance can be initiated. The process of implementing the admission process with Bizagi is depicted in Fig. 1.

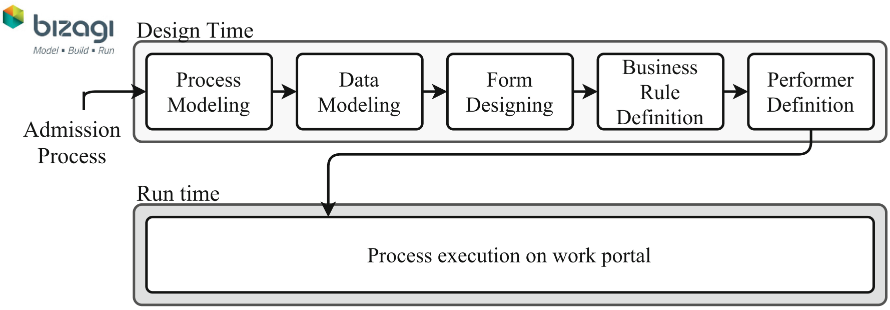


Fig. 1. Implementation step of admission process in Bizagi [4].

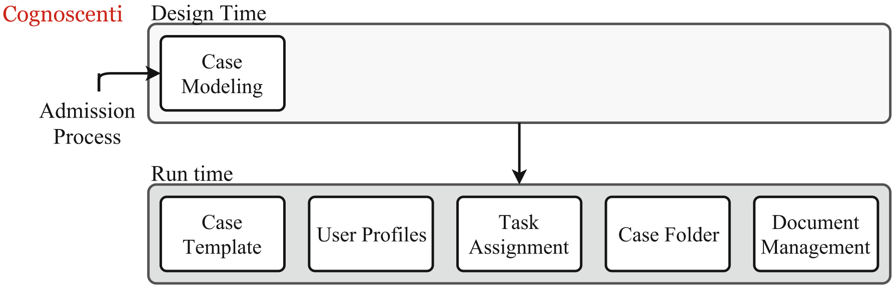


Fig. 2. Implementation step of admission process in Cognoscenti [4].

Admission Process with Cognoscenti (CM). Cognoscenti, an open source software, is available for research purposes. According to Swenson [28], Cognoscenti is not a complete product, but a testbed to show the capabilities of case management. Due to this reason, Cognoscenti has limited features.

As compared to Bizagi, the process implementation steps in Cognoscenti are directed. Cognoscenti does not automate the business process but facilitates in its management. Figure 2 provides the steps that assist in process management in a flexible manner. It is important to notice that the execution steps of the admission process are performed at the runtime with no directed links.

3.3 Results of the Experiment

As mentioned earlier, the procedure of implementing the business process is dependent on the functionalities provided by software suites. Each software suite has different capabilities, which can either make the process management more or less difficult. Execution of this experiment is assessed based on the process management aspects defined in the above-mentioned analysis procedure. Table 1 provides a summary of the experiment execution process.

Considering the result of this experiment, it can be said that BPM and CM are significantly different in managing an UBP. However, both paradigms have its strengths and weaknesses. From the UBP perspective, BPM straitjackets the

Table 1. Difference between BPM and CM process paradigms [4].

Business Process Management (Bizagi)	Case Management (Cognoscenti)
Process/Case modeling	
The process model is the first and most important activity in the process specification as all other process activities (e.g. data modeling, electronic forms design) are dependent on it. The process model is used as a road map which is followed by each process instance on run-time	Case modeling is a useful activity in which activities and goals of a case are defined. The case model provides guidance for case processing. It is not necessarily followed by case instances at run-time
Data management	
In BPM, visibility of data is very limited to users. End-users can provide and obtain data only through the electronic forms	The focus of CM is on data. A case folder maintains all the relevant records of the particular case. Based on available data, the further goals of the case are defined and assigned to knowledge workers
Business rules specification	
Business rules can be defined at design time. These rules specify the flow and execution of activities of the process	Cognoscenti, as a tool, does not provide the functionality to define the business rules. However, in CM, the specification of business rules is an important task
Users role specification	
In the BPM process model, the lanes are used to define the user roles. The activities assigned to users are defined on the process model. User roles and assigned activities cannot be modified at run-time	User roles are assigned as soon as the case is created. New users can be added at run-time as well as existing users can be deleted. Each uploaded document can be assigned with different access level permissions
Process/Case progress view	
A limited view of the overall process progress is visible to end-user. The process view is only accessible by electronic forms, where process model is not visible during process execution	With the milestone concept of CMMN, the process view is visible to process engineers on the process/case model
Process/Case control	
A BPM system is able to conform to the business level standards. Sequential and conditional flow of the process, predefined input of data and inherited business rules provide control to the business. But on the downside, it limits the process on the predefined flow of activities which might not depict real-world situations	The CM system is an open-ended platform where conformance to standards is a difficult task. It does not provide the conditional flows or predefined forms for data input. But on the positive side, it can represent the real-world situation by performing the needed activities
Activities execution	
The focus is on activities and control. Activities defined in the process model are executed with certain sequential flow and control. All the activities that are defined in the process model need to be performed except if omitted by if-else conditions	New activities can be initiated during run-time, while, the defined activities can be skipped, executed or deleted. Activities are defined and executed at run-time based on the data
Setup effort	
A BPM process requires intensive setup steps at design time. The setup includes process modeling, data modeling, forms design, business rules specification, users' profiles specification and finally deployment of the process. The setup efforts also require some prior knowledge of data modeling and form designing	A case does not require intensive setup steps before the execution. The setup step includes only case modeling and the creation of the particular case project. The case project can be considered as a case folder which contains all the information about the case and its processing

process into predefined activities and flow, but at the same time conformance to business rules and business standards are assured. On the other hand, CM provides an open-ended platform that provides run-time activity specification, task assignment and collaboration among knowledge workers, thus providing the required flexibility to unstructured business processes. However, without the predefined users and activities, the case/process can take longer in its processing than expected. For example, the run-time task assignment to users can cause resource dependency or even deadlocks in certain situations.

By analysing the strengths and weaknesses of both paradigms, it can be concluded that a software suite that contains the features of BPMS and CMS will facilitate management of an UBP in the most effective way. For example, most of the processes of a business are combination of structured and unstructured activities.

4 Evaluating BPMN and CMMN for Modeling UBP

Two process modeling languages i.e. BPMN and CMMN have been introduced and evaluated, respectively, to assess their ability to model UBP. We use an application scenario of an admission process to investigate and compare the capabilities of these notations to model an UBP.

4.1 Application Scenario

The admission process is a knowledge intensive unstructured process which demands collaboration and communication among number of departments to perform the smooth intake of students. Following is the detailed description of the admission process.

> With the announcement of admission, the students can send their documents to the university through an online form. Students are required to submit their personal information with their academic certificates, motivation letter and language certificate. Once the admission application is *submitted* by the student, the **admission office** is *notified.* Based on documents received, each admission file might go through at number of assessments before the final decision can be made. Initially, the **admission administrator** *checks* the application for its correctness and completeness. The admission file is then forwarded to the corresponding department of university for *assessment.* The **admission coordinator** will *review* the admission file to check the attached academic certificates. The final *decision* can be made by the admission coordinator only or it can require the *discussion* and *decision* from the **admission panel**. During the decision process, the provided details can be *verified* and new documents can be requested from the student. At the end, a student can be admitted, rejected or conditionally admitted. The involved knowledge workers and the decision highly depend on the particular admission file. Finally, the student is *informed* about the decision based on his admission file.

In this scenario description, verbs in italic letters show the activities of the admission process while nouns in bold letters represent the involved knowledge workers.

4.2 Modeling UBP with BPMN

BPMN is one of the widely adopted process modeling notations due to its ease of use and expressibility. A BPMN process model provides a layout of the business process by modeling the set of ordered activities, events, and process flow logic [10]. BPMN is often regarded as the modeling notation of choice for SBP [24]. Figure 3 shows the admission process modeled using BPMN modeling constructs.

Following are some problems of modeling an UBP with procedural modeling language like BPMN [25].

Task Ordering: BPMN, as a procedural modeling languages, poses the ordering and task dependency in process executions. For example, in Fig. 3, the task ordering implies that the activity *'Send certificate for authentication'* will be only performed after the task *'Review admission form'* has been completed. While, in reality, the verification of certificates and review of admission form can be performed in parallel.

Unavailable Optional Tasks: In BPMN, the execution of tasks can be skipped only by employing conditions on an exclusive gateway. However, tasks that are defined with a sequential flow on the process model without any conditions cannot be skipped. Even if the tasks are not required by the particular process instance, the tasks are needed to be executed to continue the process flow. For example, the activity *'Send certificates for authentication'*, in Fig. 3, should be regarded as an optional activity if the authentication is not needed.

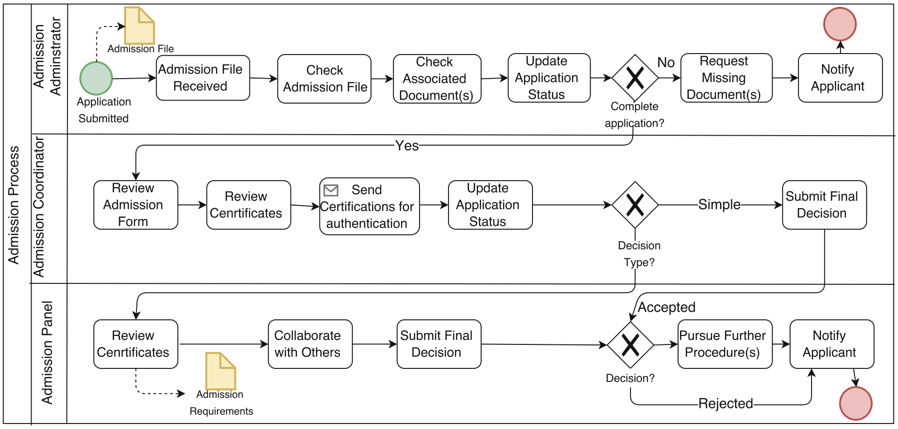


Fig. 3. Process model of admission process using BPMN [5].

Limited View on Data: BPMN provides a very limited view on data. Business processes like the admission process are data-intensive in nature; the provided data can define the flow of activities. With BPMN, the data input and output flow can be depicted, but the changing state of data can not be defined.

Some of the problems that are highlighted with BPMN can be mitigated by using the extended BPMN elements [21, p. 30]. The concept of ad-hoc subprocess has been found to be most useful for modeling an UBP. An ad-hoc sub-process does not specify the ordering among activities. The activities in an ad-hoc subprocess can be executed any number of times without any pre-defined ordering. Based on process instance requirements, the activities of ad-hoc sub-processes can be done, redone or even skipped. However, according to the BPMN version 2.0 standard specification [21], many process engines don't provide support for ad-hoc sub-process execution. Moreover, use of extended BPMN elements results in a very complex process model. The activities defined inside the ad-hoc subprocess cannot be labeled to indicate whether activities are optional, required or re-executable. The use of various events and sub-processes can negatively influence the understandability and readability of the process model.

4.3 Modeling UBP with CMMN

CMMN is for modeling the case/process where the activities are not strictly defined, but dependent on evolving circumstances and decisions of knowledge workers [22]. As compared to BPMN, CMMN is a relatively new process modeling language with unique constructs. Modeling construct of CMMN, which are exploited in Fig. 4 for admission process model, are the following:

A rectangle shape with the title of 'Admission Process' is called *case folder*, while the title depicts the name of the case/process. A *case folder* is a container that consists of all CMMN elements to model the process. A rectangular shape with angled corners shows the episodes of a process which are called *stages*. 'Check Admission File', 'Assess Admission File' and 'Decision on Admission File' are stages of the admission process. Shapes with half-rounded corners are called *milestones*; they represent the goals to be achieved in a process. 'Completed Admission File' and 'Final Decision Submitted' are *milestones* that are required to be achieved in processing of the admission file. Finally, diamond shapes in the model are called as *sentries*; they define the entry and exit criteria for tasks and stages.

Following are problems that were encountered while modeling an UBP with CMMN.

Predefine Users: CMMN doesn't have any notation to represent the assigned user roles. According to the CMMN specification [22], the user roles are defined semantically when the case/process is initiated.

Limited View on Data: CMMN is meant to model those processes that evolve with time and where the execution of a process is mainly based on data and knowledge workers' decisions. CMMN has a concept of case file along with file

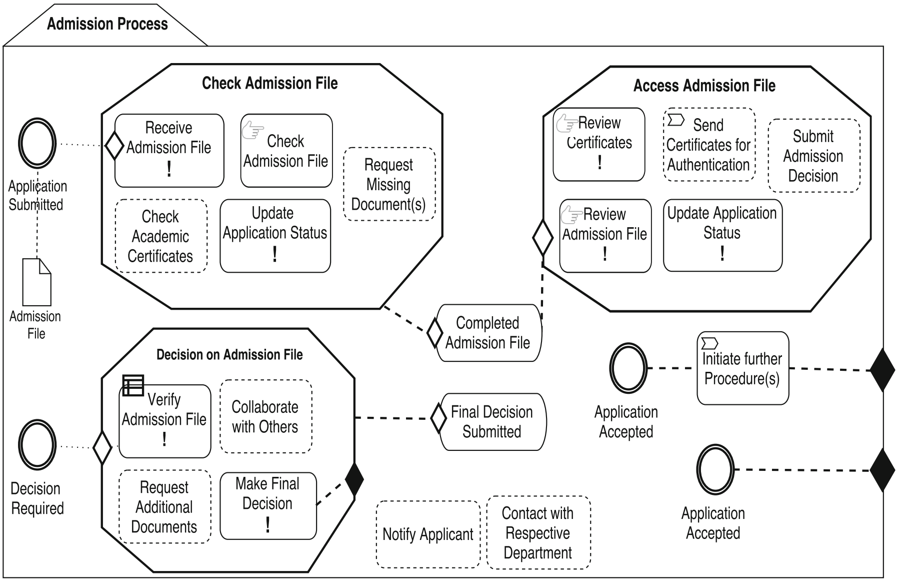


Fig. 4. Case model of admission process using CMMN [5].

versioning. However, the versioning of a case file is defined semantically. From a visualization perspective, CMMN provides a very limited view on data.

Task Dependency: Connectors and sentries represent the concept of task dependency in CMMN. The tasks will be executed only if the entry/exit condition, associated with it, is fulfilled. However, as compared to BPMN, the combination of connector and sentries provides poor readability. For example, in Fig. 4, the stage of *Assess Admission File* will only be executed if the milestone *Application Check Completed* has been achieved.

Unlike BPMN, CMMN is a declarative language. It is used to specify what should be done in the process instead of how it should be done. The purpose of a CMMN model is to provide a guidance map which instructs the process engineers on what can be done for successful process execution. Instead of design-time defined conditional flows, the evolving data and knowledge of knowledge workers drive the process execution. Consequently, BPMN is more expressive in its process flows as compared to CMMN. On the other hand, the discretionary tasks and stages of CMMN provide a better understanding of which tasks can be skipped during process execution as compared to ad-hoc sub-processes of BPMN. A detailed comparison of BPMN and CMMN notations is provided in [4, Sect. 4.4].

5 Representational Requirements of UBP

In this section, representational requirements of UBP are presented. The proposed requirements are based on the lessons learnt from BPM and CM comparison experiment provided in Sect. 3 and limitations of BPMN and CMMN discussed in Sect. 4. Cardoso et al. [7] also comparatively evaluated the BPMN with another modeling language and concluded that, despite its popularity, BPMN is limited in its ability to model UBS.

Therefore, representational requirements for modeling UBS are presented in this study, which will facilitate us into modeling the UBP in a flexible manner. A few literature studies [8,9,13] have also proposed the requirements for the development of an adaptive process management system, which support flexibility in management of a knowledge-intensive process. The large set of these requirements are taken from an previous study conducted by authors [5].

To define requirements concretely, we adopted the convention from Chiao et al. [8], where each requirement is explained with the help of an application example.

5.1 Process Specification Requirements

Each process has some general requirements that need to be fulfilled to represent the real-world scenarios.

Support to Capture Real-Time Events: It should be possible for UBP to capture and respond to real-time events. These real-time events can be related to process start or end, arrival of data, modification of existing data, or they can be triggered by user activity.

> When an applicant submits his admission application, the admission office is notified. The *notify event* can be a start event to initiate the admission process.

Support to Quantify and/or Qualify the Conditions: On certain steps in processing of an UBP, the decision to execute next process activities is taken. It should be possible to represent the conditional flow on process model.

> A complete check in admission process is an example of quantifying condition.

5.2 Activities Specification Requirements

Activities define the work that is expected to be performed for successful execution of a process. The requirements of activities specification from the perspective of an UBP are discussed below.

Support for Ordered and Unordered Activities: A business process consists of structured and unstructured parts of the process. It should be possible to define and follow the control flow among the activities as well as skip the activities' execution, if needed.

Ordered Activities: The steps like *submission* of admission file by student and *notifying* it to admission office are ordered set of activities. These process activities are required to be executed one after another.

Unordered Activities: An *assessment* activity which consist of check on academic certificates, analysis of their authenticity and review of other related documents are example of unordered process activities.

Support for Required and Optional Activities: Due to nondeterministic and emergent nature of UBP, it should be possible to define the process activities as optional or required.

Required Activities: Irrespective of type of admission file, it is required to inform the student about the status of his application.

Optional Activities: During the *assessment* activity, the activity of certificates authentication can be treated as an optional activity based on admission application.

Support for Re-execution and Undo Activities: An UBP mainly relies on decisions made by knowledge workers. Such decisions may lead to undo or re-execute the previously performed activities.

Re-execution of Activities: An admission application from the recognized national university might require single review while the international admission application might go through a number of reviews.

Undo Activities: For example, a request to defer the admission for a specific time can lead to undo certain activities that had marked the student as an upcoming admitted student.

Support for Collaboration Among Activities: In addition to parallel execution of activities, it should be possible to define and depict the collaboration among the individual process activities. BPMN depicts the collaboration between the external and internal process through message passing but not within a process.

The activities of *discussion* and *decision* require collaboration and can further leads to *verification* of the admission application. Therefore, the collaboration activities should be explicit.

Support for Varying Levels of Granularity: A process model with low level of granularity provides the flexibility for knowledge workers in process execution while a process with high level of granularity limits the knowledge workers' freedom.

Assessment and *verification* are examples of those activities that be modeled with varying level of granularity.

Support for Process and Data Alignment: Unlike traditional business process, where data are limited to defining control flows, UBP have an abundance of data with changing states. With process and data alignment, it should be possible to trace back data through a process and vice versa.

Almost each activity of admission process have associated data e.g. admission documents, remarks, decisions, etc.

Support for Process/Activity Call: It should be possible to model the already available process or activity. The callable aspect will reduce the burden of re-modeling/re-doing the same activity.

In case the applicant, who had applied for admission, also submitted his application for a scholarship. With activity/process call, the results of the authentication activities can be reused from admission process.

5.3 Data Specification Requirements

UBP are fundamentally data-centric, which means that the process and data are strictly bounded [3,17]. The execution of process highly relies on available and evolving process data.

Support for Data Representation: UBP produce and consume data during execution. It should be possible to clearly define the inflow and outflow of data files for a particular process activity.

In the *assessment* activity, the admission application can be represented as an input data file while remarks as an output data file.

Support for Data Authorization: With the involvement of number of knowledge workers in UBP, it is should be possible to define the access level of data.

> The admission application should not be accessible to the admission coordinator and admission panel before it is verified by admission administrator.

Support for Version Control of Data: Due to evolving nature of data, the version control of data is important. The concept of versioning for UBP is introduced by OMG in CMMN version 1.0 [22]. Data versioning can be modelled as data states on a process model.

> The remarks and the decision on the admission file have evolving nature which can be revised, added or deleted.

5.4 Business Rules Specification Requirements

To conform to standards and business policies, business rules need to be employed during process execution. These rules provide information on how certain business processes should be performed and how the resources can be used [23]. The alignment of process with business rules will answer the questions about 'how and why certain activities were performed and specific decisions were made'.

> The admission deadline defined by an institute is one example of business rule, which is related to admission process.

5.5 Process Goals Specification Requirements

Goal-orientedness is one of the most distinguishing characteristics of UBP. Based on the main goal, a process evolves into a number of sub-goals and milestones as process progresses. To provide an overview of process, it should be possible to model goals and sub-goals.

> The main goal of admission process is final verdict of acceptance or rejection of admission application, while the other goals can be 'application received', 'application reviewed', and 'application verified'.

5.6 Knowledge Workers' Specification Requirements

Knowledge workers play a critical role in managing and solving UBP. Knowledge workers' primary job is to create, distribute and apply their tacit and explicit knowledge to comprehend the process, analyze related information and make decisions [12].

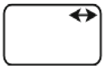
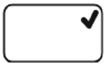
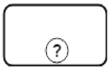
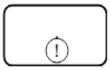
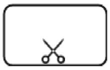
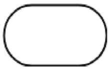
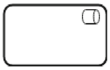
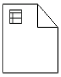
Support for Knowledge Workers' Roles Assignment: Due to involvement of many knowledge workers in process management, it should be possible to define the roles of each knowledge worker along with their assigned tasks.

> Admission administrator, admission coordinator, and admission decision panel are knowledge workers of the admission process with their assigned set of tasks.

Support to Capture Knowledge Workers' Decisions: One of the most important tasks of knowledge workers is to utilize their tacit knowledge, available data and process context to take the certain decisions. The decisions made by knowledge workers affect the process running time, its control flow, final outcome and many other process related aspects. It should be possible to capture every decision of knowledge workers.

> The admission administrator needs to make a decision about the completeness of the admission application before forwarding it to admission coordinator.

Table 2. Extended Modeling Constructs of BPMN (Demonstration) [5].

No	Name	Notations	Semantics
1	Collaborative Subprocess		Collaborative subprocess represents collaboration among different activities of the process
2	Decision Activity		Decision activity shows a decision taken during the course of process execution
3	Optional Activity		Optional activity defines an activity that can be skipped during the process execution considering the process context
4	Required Activity		Required activity defines a process activity that must be executed
5	Undo Activity		Undo activity represents an activity that can be undone considering the particular process context
6	Goal		Goal represents the purpose of the process
7	User Role		User role represents a person or a class of people who are assigned to perform the process execution
8	Business Rule		Business rule represents a related business rule on the process model

6 Illustration of Representational Requirements

To demonstrate the proposed representational requirements, a few extended modeling constructs based on BPMN are suggested in Table 2. The reason to demonstrate representational requirements with BPMN is twofold: First, as compared to CMMN, BPMN is widely known and adopted by process engineers. Second, many available modeling constructs provided by BPMN are able to fulfill a number of representational requirements.

Using BPMN and the extended modeling construct, an admission process is modeled in Fig. 5. The description of each construct is provided as added comments in Fig. 5 and with Table 1. All the activities without incoming and outgoing sequential flow are unordered, while the sequential flow defines the order between activities. Moreover, a subprocess can be attached to the conditional flow to reach to the goal. For instance, the goal *Application verified* can only be reached if the condition *verification completed* is met. Data objects with their changing states are also represented in the process model. The position of data objects in the process model shows the data access levels for the involved performers. For example, the data object applicant file with created and verifying state is only accessible by the admission administrator while the applicant file with verified state can be accessed by all the involved performers.

To improve the readability, comments have been added in Fig. 5. The whole process is placed within a single lane container, which is associated with a process name and a main goal. Each process has a number of other goals that are achieved during process execution. The admission process consists of three main sub-processes, namely *Application Intake*, *Application Assessment* and *Application Decision.* Each of these sub-processes has one associated 'goal' and one assigned 'user role'. The start of the admission application process is depicted with the timer catch event. *Receive application* is the required activity which will create an applicant's admission file. The data object file represents the name of the file as well as the state of data in file. The next step of *Verify Application* is demonstrated as sub-process which can be repeated any number of times as marked with re-executable marker. For example, *verify certificates* and *verify personal data* are marked as 'optional activities' which can be executed or skipped considering the provided data from applicant file. While the activities like *Check application completeness* and *Create Verification report* have a sequential flow that defines ordering of these activities. Moreover, an activity with the incoming sequential flow is always required. The conditional flow on the boundary of the sub-process *Application Intake* represents that the goal *Application Verified* will only be reached once the verification of the application is completed. The sub-process *Application Assessment* will start only if the 'goal' *Application Verified* has been achieved. In the review application step, the applicant file is reviewed considering the admission requirement. The admission requirement that is set by the institute is represented as 'business rule' data item. Once the application is reviewed, the activities that belong to *Application Decision* are performed. *Discussion for Decision* is a 'collaborative sub-process' which shows the interaction among user activities. All the activities inside the

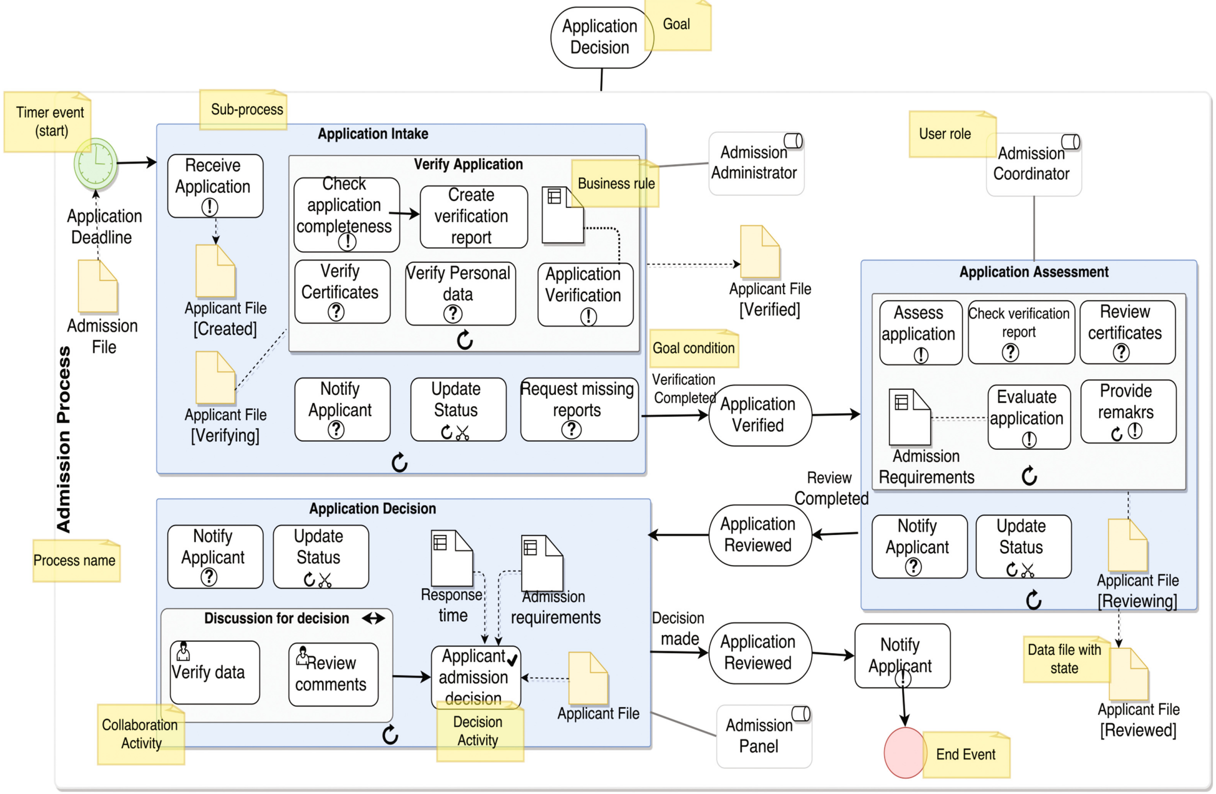


Fig. 5. Process model of admission process [5].

collaborative sub-process are dependent on each other. The final 'decision activity' takes the input from the applicant file, collaborative sub-process, business rules and finally provides the final decision. Once the decision has been made, the application decision goal and main goal of the process has been achieved and the applicant is notified.

As compared to the admission process model presented in Figs. 3 and 4, the admission model that fulfills the representational requirements offers a number of advantages.

Expressive Process Model: As compared to CMMN, the process model provided in Fig. 5 has a well-defined process start and end event. Moreover, the modeling constructs to show the required, optional, decision and collaborative tasks makes the process model easy to read and communicate.

Ability to Model (un)Structured Process: The process model shown in Fig. 5 represents the structured and unstructured process parts. Sequential flow represents the task ordering and task dependency between tasks which is a must requirement to model structured process. CMMN doesn't have the concept of sequential flow while in BPMN the use of the sequential flow inside the ad-hoc subprocess yields a semantically incorrect process model.

Ability to Model User Roles: With the user role notation, a person or group or department can be set as responsible to perform certain activities. CMMN and BPMN don't have any notation to define the user roles on the process model. However, in BPMN lanes are used for this purpose.

Ability to Model Data Access Level: The data access level is defined based on data object position on the process model. A data object that is defined inside the subprocess belongs to the assigned user only, while, the data object outside any subprocess is accessible by all the involved users of a process.

Ability to Model Related Business Rules: With BPMN and CMMN, it is feasible to represent business rule related activities either by a business rule task or planning table. However, in order to show the effect of business rules on process control flow, an extended modeling construct is used in Fig. 5.

Ability to Model Collaborative Activities: The process model provided in Fig. 5 shows the collaboration among the activities. The collaboration among activities presents that the activities are dependent on each other for their execution.

7 Validation

The validation of proposed representational requirements and their demonstration with extended BPMN constructs were performed with three experienced business process modeling practitioners. Each of these practitioners has considerable working experience with the BPMN process modeling language. Semi-structured qualitative interviews were conducted with each participant in separate sessions that lasted from 60 to 90 min.

The suggested representational requirements and their demonstration with BPMN extended constructs were mainly validated for their usefulness, ease of understanding and correctness. The result of validations is provided as follows:

Usefulness: The concepts of required, optional, collaborative sub-process, goal and decision activity are regarded as very useful for modeling unstructured business processes. However, the concept business rule is termed as unnecessary because business rules are often extensive and are difficult to be included in process model. Apart from business rules, the respondents found the concepts of data specification very powerful. According to one of the respondents, the demonstration of data specification in Fig. 5 is very intuitive as compared to technical specification of BPMN.

Ease of Understanding: The suggested representational requirements are easy to understand and yield flexibility for modeling UBS. However, the demonstration of representational requirements in Fig. 5 is indicated as difficult to read when compared to BPMN process model (see Fig. 3) and easy to read when compared to CMMN process model (Fig. 4).

Correctness: Some of the comments regarding similarities of BPMN with suggested concepts as requirements were highlighted. According to one of the respondents, the concept of optional task can be achieved by employing the BPMN gateway. But he also acknowledges the involved complexity of modeling an optional task with gateway (requiring three constructs) as compared to using a simple optional task. Moreover, the concept of undo and compensation event of BPMN is found to be similar. Another respondent suggested to keep one concept for required and optional as if some task is not required then it would be optional. However, other respondents find the separate concepts of required and optional very useful as it will bring clarity to the process model.

Overall it is found that representational requirements and a set of extended BPMN constructs are able to model USB without incorporating unnecessary details and complexity while representing the needed run-time flexibility.

8 Conclusion

Unlike structured business process, UBP are goal-oriented, data dependent, emergent, and demand run-time flexibility. Business Process Management (BPM) and Case Management (CM) as a process support paradigm are compared in order to assess their support for UBP. The results of the comparison suggest that BPM provide no flexibility for UBPm however, it assures the conformance to business rules and standards. CM, in contrast, enables the activities specification and task collaboration at run-time, however, it does not explicitly define the users. From the modeling perspective, taking the unique nature of UBP into consideration, a number of modeling limitations of BPMN and CMMN are identified. For instance, BPMN introduces task dependency in process execution whereas CMMN is unable to model user roles/task assignments in process modeling. Although BPMN provides a number of useful constructs (e.g. ad-hoc sub-processes, re-execute task) for modeling UBP. But use of various modeling constructs results into a very complex process model, which is difficult to communicate to business people along with its semantic content. On the other hand, the expressibility of CMMN modeling constructs is found to be insufficient for process modeling.

The main contribution of this paper is to derive explicit requirements for management and modeling that must be supported by a process support paradigm and a modeling language to optimally manage UBP. The modeling requirements of UBP are demonstrated by defining an extension to BPMN. We do not claim that this extension is the only or the best notations possible, but it does show that more adequate modeling notations for UBP are feasible.

Since a structured business process often consists of unstructured activities and vice versa, the future work of this study aims to seek or develop a comprehensive modeling language along with a support paradigm that is able to fulfill the requirements of structured and unstructured business processes without introducing unnecessary complexity and hindering the process run-time flexibility.

References

1. van der Aalst, W.M., Berens, P.: Beyond workflow management: product-driven case handling. In: Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work, pp. 42–51. ACM (2001)
2. van der Aalst, W., Van Hee, K.M.: Workflow Management: Models, Methods, and Systems. The MIT Press, Cambridge (2004)
3. van der Aalst, W., Weske, M., Dolf, G.: Case handling: a new paradigm for business process support. Data Knowl. Eng. **53**(2), 129–162 (2005)
4. Allah Bukhsh, Z.: BPMN Plus: a modelling language for unstructured business processes, master thesis, University of Twente (2015). http://essay.utwente.nl/67945/
5. Allah Bukhsh, Z., van Sinderen, M., Sikkel, K., Quartel, D.: Understanding modeling requirements of unstructured business processes. In: Proceedings of the 14th International Joint Conference on e-Business and Telecommunications - Volume 4: ICE-B (ICETE 2017), pp. 17–27. INSTICC, SciTePress (2017)
6. Bizagi: Bizagi studio (2015). Software. http://www.bizagi.com/en/bpm-suite/bpm-products/studio
7. Cardoso, E., Labunets, K., Dalpiaz, F., Mylopoulos, J., Giorgini, P.: Modeling structured and unstructured processes: an empirical evaluation. In: Comyn-Wattiau, I., Tanaka, K., Song, I.-Y., Yamamoto, S., Saeki, M. (eds.) ER 2016. LNCS, vol. 9974, pp. 347–361. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46397-1_27
8. Chiao, C.M., Künzle, V., Reichert, M.: Enhancing the case handling paradigm to support object-aware processes. In: Proceedings of the 3rd International Symposium on Data-Driven Process Discovery and Analysis (SIMPDA 2013). CEUR Workshop Proceedings, CEUR-WS.org (2013)
9. Di Ciccio, C., Marrella, A., Russo, A.: Knowledge-intensive processes: characteristics, requirements, and analysis of contemporary approaches. J. Data Semant. **4**(1), 29–57 (2015)

10. Dumas, M., García-Bañuelos, L., Polyvyanyy, A.: Unraveling unstructured process models. In: Mendling, J., Weidlich, M., Weske, M. (eds.) BPMN 2010. LNBIP, vol. 67, pp. 1–7. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-16298-5_1
11. Eshuis, R., Kumar, A.: Converting unstructured into semi-structured process models. Data Knowl. Eng. **101**, 43–61 (2016)
12. Grudzińska-Kuna, A.: Supporting knowledge workers: case manangement model and notation (CMMN). Inf. Syst. Manag. **2**(1), 3–11 (2013)
13. Hauder, M., Munch, D., Michel, F., Utz, A., Matthes, F.: Examining adaptive case management to support processes for enterprise architecture management. In: IEEE 18th International Enterprise Distributed Object Computing Conference Workshops and Demonstrations (EDOCW), pp. 23–32. IEEE (2014)
14. Hinkelmann, K.: Business process flexibility and decision-aware modeling—the knowledge work designer. In: Karagiannis, D., Mayr, H., Mylopoulos, J. (eds.) Domain-Specific Conceptual Modeling, pp. 397–414. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39417-6_18
15. Kemsley, S.: The changing nature of work: from structured to unstructured, from controlled to social. In: Rinderle-Ma, S., Toumani, F., Wolf, K. (eds.) BPM 2011. LNCS, vol. 6896, p. 2. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23059-2_2
16. Kitson, N., Ravisanskar, R., Soudamini, R.N.: Case management - managing chaos: unstructured processes and dynamic BPM (2012). Capgemini Whitepaper. www.capgemini.com/bpm-trends
17. Marin, M., Hull, R., Vaculín, R.: Data centric BPM and the emerging case management standard: a short survey. In: La Rosa, M., Soffer, P. (eds.) BPM 2012. LNBIP, vol. 132, pp. 24–30. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36285-9_4
18. Mccready, S.: There is more Than One Kind of Workflow Software. Computerworld - COWO (1992)
19. Miles, D.: Case management and smart process applications. Technical report, Association for Information and Image Management (AIIM) (2014)
20. Mundbrod, N., Kolb, J., Reichert, M.: Towards a system support of collaborative knowledge work. In: La Rosa, M., Soffer, P. (eds.) BPM 2012. LNBIP, vol. 132, pp. 31–42. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36285-9_5
21. OMG^{TM}: Business process and model notation (2011). http://www.omg.org/spec/BPMN/2.0
22. OMG^{TM}: Case management model and notation (CMMN) (2014). www.omg.org/spec/CMMN/
23. Penker, M., Eriksson, H.E.: Business Modeling with UML: Business Patterns at Work. John Wiluv & Sum 220 M. Godowski and D. Czyrnek/Requirement Management in Practice (2000)
24. Rosenfeld, A.: BPM: structured vs. unstructured. BPTrends, September 2011. http://bit.ly/1I6Ev3W
25. Rychkova, I., Nurcan, S.: The old therapy for the new problem: declarative configurable process specifications for the adaptive case management support. In: zur Muehlen, M., Su, J. (eds.) BPM 2010. LNBIP, vol. 66, pp. 420–432. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-20511-8_39
26. Sinur, J., Hill, J.B.: Magic quadrant for business process management suites. Gartner RAS Core research note, pp. 1–24 (2010)
27. Swenson, K.D.: Cognoscenti (2014). Open Source Software. https://code.google.com/p/cognoscenti/

28. Swenson, K.D.: Cognoscenti open source software for experimentation on adaptive case management approaches. In: IEEE 18th International Enterprise Distributed Object Computing Conference Workshops and Demonstrations (EDOCW), pp. 402–405. IEEE (2014)
29. Swenson, K.D., Palmer, N., Silver, B.: Taming the Unpredictable: Real World Adaptive Case Management: Case Studies and Practical Guidance. Future Strategies Inc., New York (2011)
30. Weber, B., Mutschler, B., Reichert, M.: Investigating the effort of using business process management technology: results from a controlled experiment. Sci. Comput. Program. **75**(5), 292–310 (2010)
31. White, M.: Case management: Combining knowledge with process. BPTrends, July 2009