MASTER

Designing an architectural blueprint of a smart BPM system

Minchev, M.

Award date:
2014

Department of Mathematics and Computer Science
Department of Industrial Engineering and Innovation Sciences

Master's Thesis

Designing an Architectural Blueprint of a Smart BPM System

Martin Minchev

Supervisors: dr.ir. Rik Eshuis (TU/e)
dr.ir. Claudia-Melania Chituc (TU/e)
dr.ir. Ronny Mans (TU/e)

Tutor: ir. Erik van der Starre (Capgemini)

Eindhoven, 17 December 2013

Management summary

In this study, we link together the fields of BPM technology and smart objects. On the one hand, BPM technology already exists for many years and it is widely used in larger organizations today. On the other hand, the field of smart objects is relatively new and it is developing very rapidly in last several years.

Due to the immaturity of the field of smart objects, there is still a lot of controversy about what exactly a smart objects is. In this study, we look at a smart object as a physical object which has a representation in the digital world. This representation can be stored on a device or on a remote information system. The link between the physical object and its digital representation is maintained via ICT devices such as tags, sensors and actuators. Information collected with the help of sensors and tags can be used to update the digital representation of a smart object while actuators can be used to modify the state of the physical object. The functionality related to smart objects is exposed to the external world via services. These services can be used by information systems in order to work with smart objects.

The problem addressed in this study is related to the fact that, nowadays, people are the main resource on which BPM systems rely when they have to interact with their physical environment. When BPM systems need to capture information about the state of the physical environment, they rely on people who observe it and provide input based on their perceptions. Similarly, when BPM systems need to modify the state of the physical environment, they rely on people who perform the required activities in it. The problem with people being the link between BPM systems and the physical environment is that they are quite limited when it comes to the accuracy, speed and cost of their work. These limitations of people influence the capability of BPM systems to capture information about and modify the physical environment in an accurate, timely and cost efficient manner. In turn, this limits the capability of BPM systems to provide organizations with real-world visibility, control and responsiveness. One possibility for improving the current situation would be to reduce the dependence of BPM systems on people. This can be done with the help of devices, since they are able to work more accurately, quickly and cheaply compared to people. Organizations can use devices in order to transform objects from their physical environment into smart objects. Then, their BPM systems can be adapted to work directly with smart objects in order to collect information about or modify the physical environment in more accurate, timely and complete way. Although this seems as a viable idea, based on the literature review we have conducted, we claim that there is still no complete solution how BPM systems can utilize smart objects in order to improve the real-world visibility, control and responsiveness of organizations. In order to verify the existence of this problem in industry, we conducted a number of meetings with experts (including two vice presidents) from Capgemini Nederland. The result of the meetings was positive, since most of the experts expressed an interest in the topic.

In order to address this problem, in this study, we develop a generic, industry-independent architectural blueprint of a BPM system which can utilize smart objects in order to improve the real-world visibility, control and responsiveness of organizations. This architectural blueprint defines a new class of BPM system which we call Smart BPM (SBPM) systems. The architectural blueprint includes analysis of the organizational context and functional requirements of the system and, based on them, the development of a reference model for the system. The methodology we use in this study is based on COMET which is a use case-driven, model-focused methodology for specifying and implementing software products [1]. As suggested by COMET, we capture the organizational context of the SBPM system in a business model. The main focus is on capturing the different stakeholders of the system and their goals related to the system. As suggested by COMET, we capture the functional requirements of the system in a requirements model. We do that with the help of use cases. Finally, we design a reference model which presents the functional view of the SBPM system. The goal of the model is to facilitate the future design of concrete SBPM systems.

In order to validate the quality of the models which we have developed in this study, we used two different approaches. The first approach involved the description of the SBPM system in the context of a concrete industry scenario related to outage restoration. On the one hand, this evaluation gave us evidence for the completeness of our generic business and requirements models. On the other hand, it illustrated how the SBPM system can be applied successfully for the improvement of real-world visibility and responsiveness of organizations in the energy sector. As far as the second approach is concerned, it involved 4 interviews with experts from Capgemini Nederland. The focus of the interviews was on evaluating the quality of only the most important generic models in our study. Based on the positive results from the interviews, we concluded that, in general, the models we have developed have a good level of understandability, completeness and soundness.

We believe that the architectural blueprint we have design in this study will be useful for both industry and academia. On the one hand, when it comes to business scenarios which require the integration of smart objects and BPM systems, the architectural blueprint can help organizations decide if they need a SBPM system, what functionality they need and how to design one. On the other hand, it describes a SBPM system which is a new class of BPM system. This creates a number of opportunities for future research.

Table of contents

1 Introduction 1
1.1 Research context and motivation 1
1.2 Research problem 2
1.3 Research goal 2
1.4 Research approach 3
1.5 Report structure 7
2 Background 9
2.1 Smart objects 9
2.2 Business processes 13
2.3 Information flow processing 14
2.4 Business activity monitoring 15
2.5 Event-driven BPM reference model 16
2.6 Conclusions 17
3 Business model 18
3.1 Vision for change 18
3.2 Context statement 20
3.3 Goal model 23
4 Requirements model 28
4.1 Use case model 28
4.2 Architecture analysis 30
5 Reference model 32
5.1 General structure of the model 33
5.2 Modules 34
6 Evaluation through a scenario 39
6.1 Business model 39
6.2 Requirements model 45
6.3 Conclusions 47
7 Evaluation through interviews 48
7.1 Smart object conceptual model 48

7.2 Organizational goal model 49
7.3 Requirements model 49
7.4 External systems 50
7.5 Reference model 50
8 Conclusions, limitations and future work 51
8.1 Conclusions 51
8.2 Limitations and future work 52
9 References 54
10 Appendices 59
10.1 Interviews 59
10.2 Extended literature review 64
10.3 Smart energy 78
10.4 Use cases 81

List of figures

Figure 1: Information Systems Research Framework [12] 4

Figure 2: Relationships between COMET models 7

Figure 3: Conceptual model of a smart object [16], [17] 11

Figure 4: The BPM lifecycle [3], [27] 14

Figure 5: High-level view on an IFP system [28] 15

Figure 6: Event-driven BPM reference model [30] 17

Figure 7: People and smart objects as mediators 19

Figure 8: IT landscape of the SBPM system 22

Figure 9: Context statement 23

Figure 10: Organizational goal network 26

Figure 11: System boundary model 30

Figure 12: Grouping use cases into modules 31

Figure 13: SBPM system reference model (functional view) 33

Figure 14: High-level model of an ASC system [47] 36

Figure 15: High-level model of an IFP system [28] 37

Figure 16: Outage restoration process [49], [50] 41

Figure 17: Goal network 44

Figure 18: Gartner's 2012 Hype Cycle for Emerging Technologies 66

Figure 19: Types of business processes [3] 67

Figure 20: The devil's quadrangle [35] 69

Figure 21: Workflow reference model [41] 71

Figure 22: An integrated view of an information system [67] 74

Figure 23: Reference architecture process [66] 75

List of tables

Table 1: Smart capabilities 9
Table 2: Stakeholders 20
Table 3: Layers in the IT landscape 21
Table 4: Organizational goals 24
Table 5: User goals 27
Table 6: Use cases 28
Table 7: Mapping ASC and SBPM system elements 35
Table 8: Mapping IFP and SBPM system elements 36
Table 9: Stakeholders 41
Table 10: External systems 42
Table 11: Organizational goals 42
Table 12: User goals 44
Table 13: Use cases 45
Table 14: Use case - Define IFP rule 81
Table 15: Use case - View monitoring objects 81

1 Introduction

1.1 Research context and motivation

Nowadays, companies around the world consider business processes to be their most valuable asset and therefore they spend a considerable amount of effort on managing them [2], [3]. In doing so, they heavily rely on information technology and especially on Business Process Management Systems (BPMS). The purpose of these systems is to support the design, configuration, enactment and evaluation of business processes.

At the same time, the rapid development of information technology enables the transformation of more and more "*everyday*" objects around us into "*smart*" objects [4]–[7]. This is achieved with the help of ICT devices which are embedded in them or placed in proximity to them. Usually, these devices have some computation, communication, storage, sensing or/and actuation capabilities. One way to look at a smart object is as physical object which has a representation in the digital world [7]. This representation can be stored on a device or on a remote information system. The link between the physical object and its digital representation is maintained via tag, sensor and actuator devices. Information collected with the help of sensors and tags can be used to update the digital representation of a smart object while actuators can be used to modify the state of the physical object. One recent example of a smart object from the consumer market is a smart door lock [8]. An embedded actuator enables the locking and unlocking of the lock via Bluetooth. An embedded sensor and storage enable the smart lock to collect and store information about the people who have recently used it. This information is in fact the digital representation of the lock.

This study is motivated by our interest to investigate the possibilities of integrating BPM systems and smart objects. Although the field of BPM systems is already studied relatively well, there is little research focusing on the link between it and the emerging field of smart objects. Moreover, these two fields are relevant to the consulting and technology services provided by Capgemini and, therefore, the company is also interested in studying the opportunities created by their intersection.

Before describing the research problem addressed in this study, we first define the concepts of real-world visibility, control and responsiveness which are partially based on literature. Real-world visibility refers to the capability of organizations to know what is actually happening in their physical environment [9], [10]. The more accurate, timely and complete information they have about it, the better is their real-world visibility. Information is accurate if it correctly represents the state of the physical environment, timely if it is up-to-date and complete if it is sufficiently detailed and covers enough parameters of the physical environment. Real-world control refers to the capability of organizations to modify the physical environment. The more accurate, timely and complete modifications they can perform, the better is their real-world control. A modification is accurate if it changes the state of the physical environment as intended, timely if it is quick enough and complete if it is sufficiently precise and changes enough parameters of the physical environment. Real-world responsiveness refers to the capability of organizations to detect and respond to events in the physical environment [10]. The quicker they can do that, the higher is their real-world responsiveness. Note that the notions of accuracy, timelines and completeness as well as real-world visibility, control and responsiveness are dependent on the concrete situation. For example, what is considered sufficiently accurate in one situation might not be sufficiently accurate in another.

Finally, we would like to mention that this study is performed as a graduation project at Capgemini Nederland B.V., one of the largest providers of consulting, technology and outsourcing services in the Netherlands with more than 6 000 employees. Capgemini Nederland B.V. is part of the Capgemini Group, a global leader in consulting, technology and outsourcing with more than 125 000 employees in 48 countries.

1.2 Research problem

Nowadays, people are the main resource on which BPM systems rely when they have to interact with their physical environment [2]. When BPM systems need to capture information about the state of the physical environment, they rely on people who actually observe it and provide input based on their perceptions. Similarly, when BPM systems need to modify the state of the physical environment, they rely on people who actually perform the required activities in it. According to literature, one of the main reasons for human involvement are the so-called media breaks [10], [11]. They occur when people have to manually transfer data from one medium (e.g. a product cover) to another medium (e.g. a database system). The problem with people being the link between BPM systems and the physical environment is that they are quite limited when it comes to the accuracy, speed and cost of their work [10], [11]. These limitations of people influence the capability of BPM systems to capture information about and modify the physical environment in an accurate, timely and cost efficient manner. In turn, this limits the capability of BPM systems to provide organizations with real-world visibility, control and responsiveness.

One possibility for improving the current situation would be to reduce the dependence of BPM systems on people. This can be done with the help of devices, since they are able to work more accurately, quickly and cheaply compared to people. Organizations can use devices in order to transform objects from their physical environment into smart objects. Then, their BPM systems can be adapted to work directly with smart objects in order to collect information about or modify the physical environment in more accurate, timely and complete way. Although this seems as a viable idea, based on the literature review we have conducted (see section 2.6), we claim that there is still no complete solution how BPM systems can utilize smart objects in order to improve the real-world visibility, control and responsiveness of organizations. This is exactly the problem which we want to address in this study.

1.3 Research goal

This study has the following main research goal:

Main goal: *Design an architectural blueprint of a BPM system which can utilize smart objects in order to improve the real-world visibility, control and responsiveness of organizations.*

In the rest of this study, we will refer to this system as Smart BPM (SBPM) system. There are a number of constrains which we explicitly set from the very beginning. First, the architectural blueprint should be industry independent. In other words, it should be generic enough so that it can fit in the context of different industries. Second, the SBPM system do not need to support the evaluation phase of the BPM lifecycle. We leave this phase out in order to limit our scope and because we consider it the least interesting among the four phases. Third, we assume that the SBPM system will be used by organizations which are sufficiently large to manage their business processes.

Since the idea behind SBPM systems is rather new and futuristic, by designing an architectural blueprint for such a system we want to make the first steps towards the design of concrete SBPM systems. In order to further clarify our main research goal, we decompose it into three research sub-goals. The achievement of these sub-goals will lead to the achievement of the main research goal. However, these sub-goals are not independent from each other. For example, sub-goals 1 and 2 provide a foundation for achieving sub-goal 3.

Sub-goal 1: *Analyze the generic organizational context of the SBPM system*

By analyzing the generic organizational context of the system, our aim is to motivate the design of the SBPM system. More precisely, we want to show what the main stakeholders of the system are and what goals they have related to the system.

Sub-goal 2: *Analyze the generic functional requirements for the SBPM system*

When analyzing the requirements of the system, we look at the system as a black box. Our aim is to show what functionality the SBPM system should support. We intentionally decide to skip the analysis of non-functional requirements due to time constraints.

Sub-goal 3: *Design a reference model for the SBPM system*

When designing the reference model, we look at the system as a white box. Our aim is to describe how the system will actually support its functional requirements. By "reference model" we mean a high-level architectural template for a SBPM system which can be elaborated to obtain concrete architectures. By designing this model, we aim to facilitate the design of future concrete SBPM systems.

1.4 Research approach

In this section, we briefly describe the research methodology which we use in this study. However, before that we introduce the Information Systems (IS) research framework proposed by Hevner [12] and the COMET methodology [1] which both play a central role in our research.

The IS research framework (illustrated in Figure 1) is a conceptual framework which is widely accepted in the field of IS research [12]. It positions IS research in the context of an environment and knowledge base. The environment defines the business need that has to be addressed by IS research. It includes people, organizations and technologies. The knowledge base provides the theoretical foundations and methodologies on which IS research can build on. We choose to use the IS research framework because it provides us with a well established guidelines how to execute and evaluate research in the field of information systems.

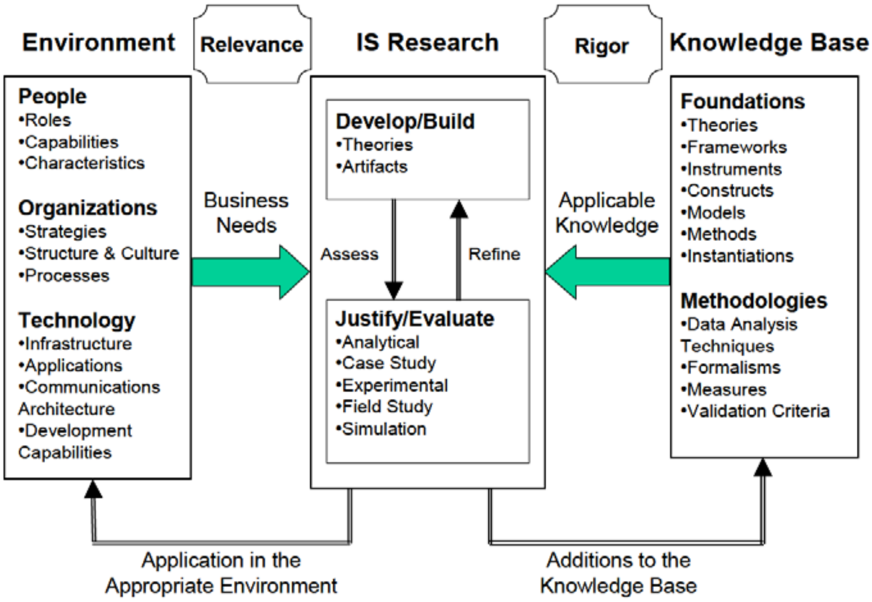

Figure 1: Information Systems Research Framework [12]

COMET (Component and Model-based development Methodology) is a use case-driven, model-focused methodology for specifying and implementing software products [1]. It specifies four main models which focus on four major aspects: *business*, *requirements*, *architecture* and *platform specific models*. Further, each of these models contains several sub-models which focus on more specific aspects of the system. COMET describes the objectives and deliverables related to each sub-model as well as the techniques and notations which can be used by architects.

In this study, we choose to use COMET because of a few reasons. First, COMET guides us in the architectural process by describing the deliverables (models) we have to produce and the techniques and notations we can use to do that. Second, models in COMET link logically with each other very naturally which makes it easier to create sound architectures. Third, we have prior experience with COMET which gives us confidence to use it for a larger architectural initiative such as this one. We have to note that initially, instead of COMET, we were planning to use ArchiMate [13]. The reason to finally choose COMET is that it provide us with more architectural guidance compared to ArchiMate. ArchiMate is an architectural framework which only describes the models which has to be produced and the notation which should be used but does not provide any concrete guidelines or techniques for doing it [13]. According to the IS research framework, COMET can be classified as a methodology from the knowledge base. Another thing to note is that, in this study, we use Microsoft Visio 2013 in order to create all UML diagrams. Compared to previous versions of Visio, this one supports UML 2.4 and provides a more modern appearance of UML diagrams.

The first major step in our research was to identify the problem which we want to solve. In order to do this, we used two main sources of information. On the one hand, we surveyed the literature on the topics of smart objects, BPM and their intersection. As it turned out, there is no public document/architecture which describes a BPM system which can utilize smart objects in order to improve the real-world visibility, control and responsiveness of organizations. In the context of the IS research framework, we queried the knowledge base and discovered a missing piece of knowledge. One the other hand, we conducted a number of interviews with experts from Capgemini Nederland in order to verify if there is a need for such a system in the industry. These interviews had unstructured character and they involved two vice presidents, utilities expert, consultant, BPM expert and enterprise architect. In the context of IS research framework, these interviews correspond to the identification of a business need from the environment. As it turned out from the interviews, the experts were interested in the design of such BPM system. This step is useful for achieving all of the three sub-goals of this study.

Our second major step was to conduct a more thorough review of the available literature. We reviewed topics such as information flow processing, complex event processing, business activity monitoring and event-driven business process management. This step is useful for achieving all of the three sub-goals of this study.

Our third major step was to design a business model for the SBPM, as proposed in COMET. The goal of the business model is to describe the system in the context of the organizations which will use it. The main questions which this model should answer are what the system should do and why. Originally, COMET relies on input form stakeholders in order to build the business model. However, since we want to build an innovative and industry-independent reference architecture, in our case, there is no concrete organization for which we design the SBPM system. Therefore, we analyzed existing reference architectures and ideas from literature in order to develop the business model. In this study, we do not develop a complete business model but just a part of it. Our focus is on the vision for change, context statement and goal model. Due to time limitations, we leave the development of a risk analysis and community model out of our scope. During the building of the business model, we iterated through a number of static evaluations and respective improvements, as recommended in the IS research framework. This entire step is related to sub-goal 1 of this study.

Our fourth major step was to design a requirements model, as proposed in COMET. The main goal of this model is to answer what exactly the system should do. We identified requirements by analyzing existing reference architectures and technologies from literature. As suggested by COMET, we captured functional requirements via the definition of use cases. Similarly to the business model, during the design of the requirements model, we integrated through a number of static evaluations and respective improvements. As suggested by COMET, in order to identify the components of the SBPM system we clustered the identified use cases according to a few criteria. In this study, we skip the development of a prototype as well as the capturing of non-functional requirements since they are out of our scope. This entire step is related to sub-goal 2 of this study.

Our fifth major step was to design a reference component structure model for the SBPM system based on the identified components from the requirements model and our own design decisions. For the design of this model, we used our own notation which is less technical and restrictive compared the UML-based notation suggested by COMET. Although COMET suggests also the development of component interaction, interface and information models, we skip them in this study because we consider them out of our scope. This entire step is related to sub-goal 3 of this study.

Our sixth major step was to evaluate the SBPM system in the context of a more concrete scenario from the smart energy industry. This is a descriptive evaluation approach which has been proposed in the IS research framework. The decision to focus on exactly this industry was taken based on a few interviews with smart energy experts from Capgemini Nederland. Together we concluded that smart energy is a sector which increasingly relies on smart objects (e.g. smart meters [14]) and, at the same time, it relies on complex business processes which require BPM support. Therefore, we believe that the SBPM system will be especially relevant to this industry. We designed a business and requirements models for a SBPM system which uses smart meters in order to improve the visibility and responsiveness of electricity companies in the context of power outage restoration. We explicitly relate this models to the generic business, requirements and reference models.

Our last major step was to evaluate the completeness and soundness of the most important models in this study by interviewing 4 different experts from Capgemini Nederland. These experts included a system analyst, data architecture and two enterprise architects.

Finally, in Figure 2, we illustrate the relationships between the different COMET models that we have developed in this study.

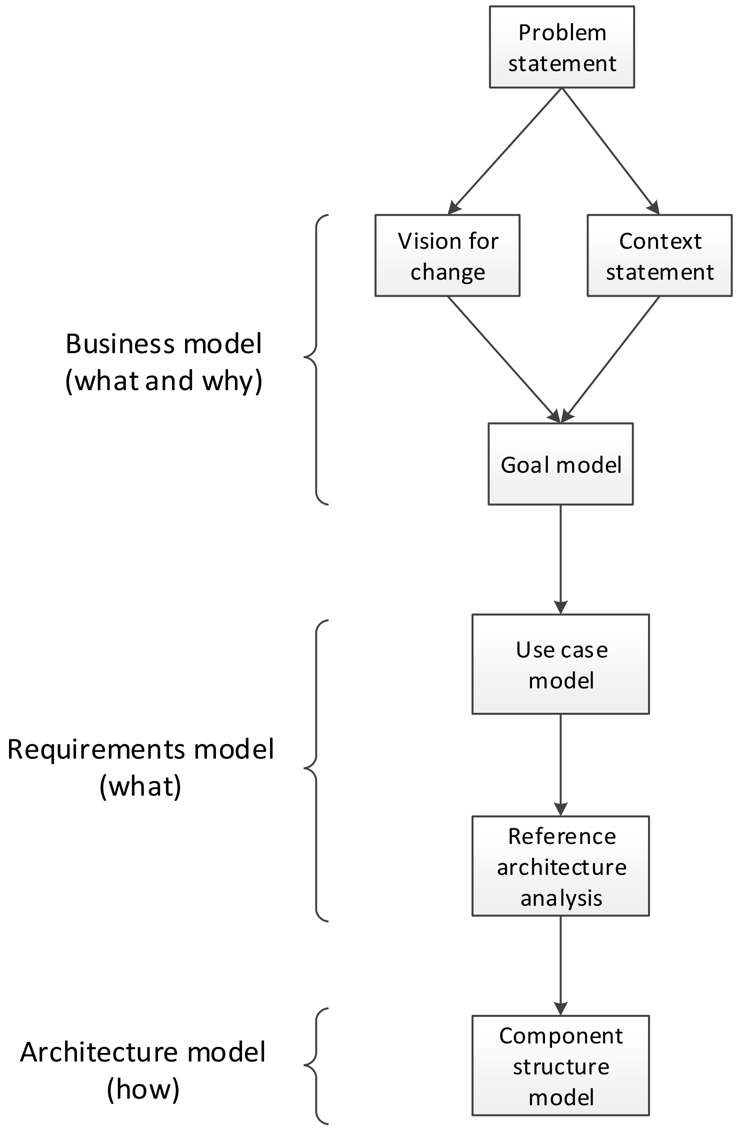

Figure 2: Relationships between COMET models

1.5 Report structure

In this section, we briefly describe the structure of the rest of this report.

Section 2 contains our literature review.

Section 3 contains the generic business model of the SBPM system.

Section 4 contains the generic requirements model of the SBPM system.

Section 5 contains the generic reference model of the SBPM system.

Section 6 contains an evaluation of the SBPM system through a scenario from the smart energy industry.

Section 7 contains an evaluation of the SBPM system through interviews with experts.

Section 8 contains our conclusions and ideas for future research.

2 Background

In this section, we briefly review literature related to smart objects, business processes, information flow processing, complex event processing, business activity monitoring and event-driven business process management. Our main sources of information are high-quality scientific journals, conference papers and books on the topics. In order to discover information for this study, we used search engines such as Web of Knowledge, Scopus, IEEE Xplore and Google Scholar. The main criteria for selecting an article or book were the year of publishing and the number of citations it has. For journal papers the criteria also included the impact factor of the journal and if the journal is refereed. In order to check information about journals, we used the list of journals provided by the library of the Eindhoven University of Technology.

2.1 Smart objects

2.1.1 Conceptual model and definition

Although smart objects are a hot topic, there is still a lot of controversy about what exactly a smart objects is. In literature, scholars even name the concept of smart object in different ways. For example, some alternative names that can be found are *digital artefact*, *smart artefact* and *smart item*. When it comes to the definition of the concept, most authors agree that a smart object is a physical object augmented with ICT technology in order to obtain some kind of "*smartness*" [5]–[7], [10], [15]. However, they view the meaning of smartness differently. The majority of authors define smartness by describing a set of abstract capabilities which smart objects can have. In Table 1, we provide a list of the most popular capabilities. In our opinion, most of these capabilities have a technical flavor and lead to technology-oriented definitions of smart object.

Table 1: Smart capabilities

Capability	Description
Storage	A smart object is able to store data related to its own identity or some other relevant data [6], [10].
Sensing	A smart object is able to use sensor technology in order to capture data about its own physical state and/or the state of its environment [6], [10].
Actuation	A smart object is able to use actuator technology to change its own physical state and/or the state of its environment [6], [10].
Computation	A smart object is able to use computational power in order to aggregation data, recognize events, making decisions or some other data processing activity [6], [10].
Communication	A smart object is able to perform wired or wireless communication of data. Communication can be initiated by the smart object or by an external entity [6], [10].

In Figure 3, we present a conceptual model of a smart object which is based on two recent studies conducted as a part of the Seventh Framework Programme (FP7) of the European Union [16], [17]. In the rest of this study, we will use this model when talking about smart objects.

Naturally, the most central concept in the model is the smart object. A smart object is the composition of a physical entity and its corresponding digital entity [17]. In some cases, it can also be an aggregation of several other smart objects.

A physical entity is defined as a "*discrete, identifiable part of the physical environment*" [17]. This could be any tangible object or environment. For example people, cars, washing machines, computers and public spaces are all physical entities. Note that one physical entity can be an aggregation of several other physical entities.

A digital entity is a representation of certain aspects of a physical entity in the digital world [17]. It can be any piece of software or digital information. Some examples of digital entities are 3D models, objects in object-oriented programming languages and social network accounts. There are two important requirements regarding digital entities. First, a digital entity always represents one and only one physical entity, and second, a digital entity should be synchronized with its physical entity. Moreover, note that one digital entity can be an aggregation of several other digital entities.

Physical and digital entities are synchronized with each other with the help of ICT devices [17]. ICT devices are usually embedded into, attached to or placed near the physical entity and they serve as an interface between the digital and physical environment. Devices can be divided into 3 main types: *sensors*, *tags* and *actuators*. Each of these types of devices are discussed in more detail in section 2.1.2. Sensors and tags enable the capturing of information about physical entities while actuators enable the modification of physical entities. Note that one device can be an aggregation of several different types of devices. For example, a smart meters is a device which contains both sensors and actuators.

Another important concept in the model are resources. They are software components which provide functionality for interacting with physical entities [17]. Two types of resource can be distinguished: on-device and network resources. Naturally, on-device resource are software components hosted on devices. They are usually related to collecting, storing and processing of sensor information or controlling actuators. One example of on-device resource would be a local, temporary storage in a smart meter which stores consumption information. Network resources are software components hosted somewhere on back-end systems. Unlike on-device resource, they usually run on general purpose hardware with more computational and storage capabilities. One example of a network resource would be a back-end database which stores consumption information about customers which has been collected by smart meters.

Since resources can be heterogeneous and their implementations can be highly dependent on the underlying hardware, they are made accessible via services [17]. A service provides a well-defined and standardized interface which can be used to access resources and, in this way, interact with physical entities. Moreover, a service can also be an aggregation of other services. This is useful in order to create services which provide higher-level functionality. For example, one service can invoke a number of other resource-related services. A digital entity can be related to services which enable it to interact with its corresponding physical entity.

Based on the conceptual model from Figure 3, a smart object can be defined as "*the extension of a physical entity with its associated digital entity*" [16], [17].

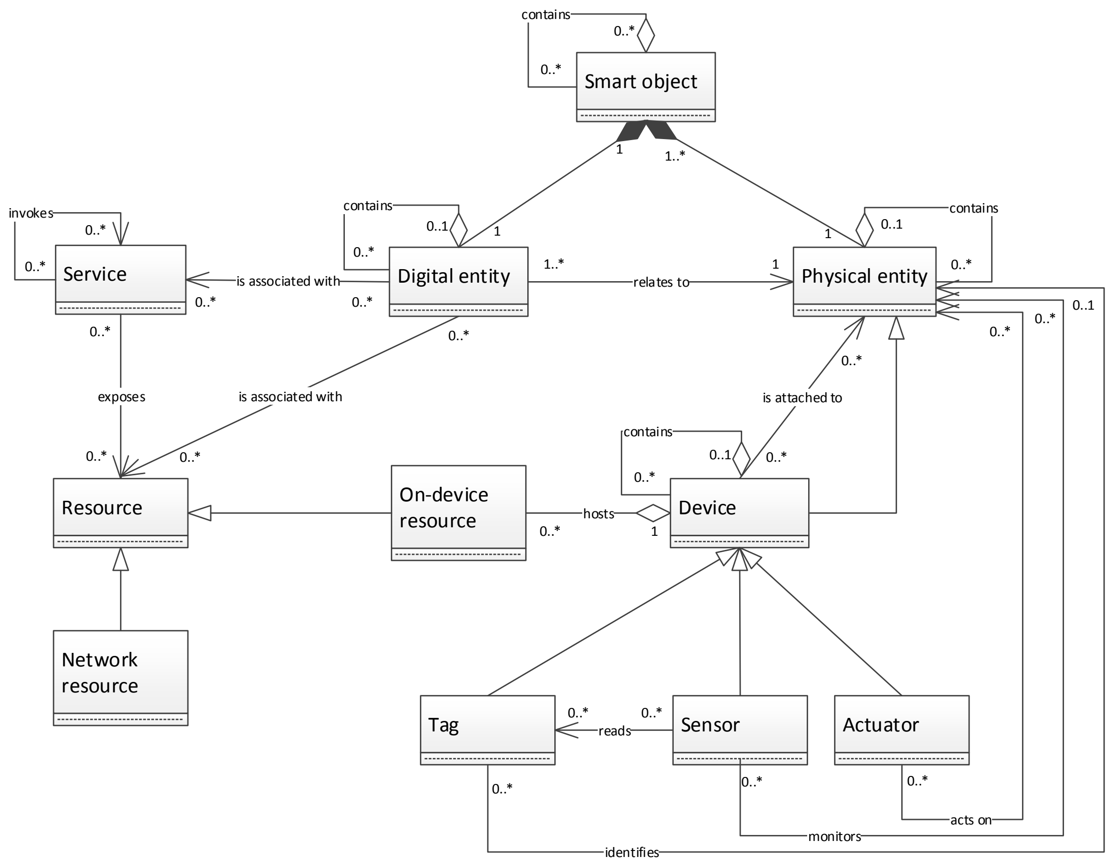

Figure 3: Conceptual model of a smart object [16], [17]

2.1.2 Types of devices

As it is described in the conceptual model illustrated in Figure 3, there are three main types of devices: sensors, tags and actuators. In this section, we briefly review each of them.

2.1.2.1 Sensors

A sensor (sometimes also called detector) is a device which is able to monitor its physical environment and capture some relevant information about it. Many different types of sensors exist and usually one sensor is focused on the monitoring of one particular aspect of the surrounding environment. Just a few trivial examples of such parameters are humidity, vibration, temperature and light [18]. The information captured by sensors can be used for various purposes such as the automatic identification, localization and status monitoring of physical entities (animate or inanimate). Sensors can be considered as “*interfaces*” through which information about the physical world is automatically flowing into the digital world [18].

2.1.2.2 Tags

Tags are machine-readable and sometimes machine-writable devices or labels which are attached to physical entities. They store relevant but usually very limited amount of information about these entities. The amount and purpose of the stored information can differ. It can be as simple as a string identifying an object or linking to more information stored elsewhere but it can also be a file containing detailed information about an object [19]. Tag and sensor technologies are closely related to each other. The reason for this is that the information stored on tags is usually read by special sensors called readers. In some cases, readers can also modify the information stored on tags. Nowadays, the most commonly used tags are based on optical (barcodes or QR-code [20]) and radio-frequency identification technologies (RFID tags [21]). We describe these two technologies in more detail in section 10.2.1.

2.1.2.3 Actuators

An actuator is a devices which can modify the state of the physical environment. The idea of actuators can be regarded, in some sense, as opposite to the idea of sensors. While sensors can be considered as “*interfaces*” through which information about the state of the physical environment is collected, actuators can be regarded as “*interfaces*” through which the state of the physical environment is modified. Just a few example of actuators are a robotic arm moving objects, electric motor with which cause a phone to vibrate and a laser used for cutting materials.

2.1.3 Related paradigms

There are a number of paradigms related to smart objects. Some of the most popular ones are *Ubiquitous Computing* [22], *Internet of Things (IoT)* [4], [7] and *Intranet of Things* [23].

Ubiquitous Computing is a computing paradigm that was first coined and described by Mark Weiser at Xerox Corporation [22]. The definition he provided is the following:

“Ubiquitous Computing has as its goal the nonintrusive availability of computers throughout the physical environment, virtually, if not effectively, invisible to the user.” [22]

The paradigm of *Internet of Things* was first coined by Kevin Ashton at Procter & Gamble in 1999. However, it really started to gain popularity after 2004 [7], [24]. Nowadays, it is one of the hottest topics and a buzzword, often used to describe one very promising future for the Internet and the world. As a result of the huge interest in the topic, a large number of definitions of IoT exist, coming from both academia and industry. Compared to Ubiquitous Computing, IoT is a bit more concrete and technically oriented. Basically, it describes a global network (Internet) to which not only computers but any objects (thing) will be able to connect. In the context of IoT, things are very similar to what we call smart objects.

The concept of Intranet of Things is similar to the concept of Internet of Things [23]. The main difference between the two is their scope. While the Internet of Things has a global scope and aims to connect "*things*" owned by various organizations around the world, the Intranet of Things has a local scope and aims to connect "*things*" within one or a few organizations. In Intranet of Things, information and services provided by smart objects stay closed and rarely get exposed so that other parties can also benefit from them. Nowadays most solutions involving smart objects fall into this paradigm. The main reason for this is the lack of standardization which will enable interoperability between the different solutions. Although Intranet of Things solutions usually fulfill well their requirements, the exponential number of possibilities and benefits which would exist if they were connected into one integrated Internet of things are never achieved. Nevertheless, with the help of a common vision and standardization, in the future these Intranets could start joining together to create a global Internet of Things.

2.2 Business processes

2.2.1 Definition

The first definition of a *business process* dates back to 18th century and the work of the father of modern economics, Adam Smith. Nowadays, some of the most widely accepted definitions agree that a business process is a set of logically related steps (called *tasks*, *procedures* or *activities*) which are performed in order to achieve a specific business outcome [25], [26]. Participants in business processes are, usually, a part of an organizational structure with defined roles and relationships [26].

2.2.2 Workflow management and business process management

Workflow Management (WFM) or just *Workflow* is a concept which is closely related to the concept of a business process [26]. It has a technical orientation and it can be defined as the "*automation of a business process*". A software system which is used to "*define, create and manage the execution of workflows*" is called a *Workflow Management System* (WFMS).

Business Process Management (BPM) is consider to be the "*next step*" after WFM [27]. It can be defined in the following way:

"*Business process management includes concepts, methods, and techniques to support the design, administration, configuration, enactment, and analysis of business processes.*" [3]

The main difference between WFM and BPM is that BPM is broader [27]. It is concerned not only with the automation of business processes but also with their continuous improvement and optimization over time. This difference between WFM and BPM is also visible in Figure 4 which illustrates the BPM lifecycle. As it can be seen, WFM focuses on the lower part of the lifecycle (design, configuration and enactment) while BPM is broader and supports all phases. A software system which supports BPM is called a *Business Process Management System* (BPMS).

2.2.3 Business process management lifecycle

The BPM lifecycle comprises 4 general phases: *process design and analysis*, *process configuration*, *process enactment* and *process evaluation* [3]. As it is depicted in Figure 4, these phases and their logical relationships form a cyclic structure. Although they are ordered logically in a cycle, this does not necessarily mean that they have the same temporal order. It is possible that the phases are executed in parallel or in any arbitrary order. Next, we discuss each of the 4 phases in more detail.

The design phase is where, usually, the BPM lifecycle starts from. People specialized in carrying out this phase are called process designers. The main responsibilities of process designers are related the identification, modeling, validation, simulation and verification of business processes. Once a business process has been designed, the configuration phase is where it has to be implemented. Usually, the business process model has to be extended with concrete technical and organizational information which will enable its enactment by a WFM/BPM system. Once a business process is implemented, its implementation should be tested and, at the end, deployed. During the enactment phase, instances of an implemented business process are actually being executed and monitored. People involved in this phase are process responsibles and participants. During the evaluation phase, the data gathered by a WFM/BPM system during the enactment phase is used to evaluate and eventually improve business process models and implementations. We describe these four BPM lifecycle phases in more detail in section 10.2.6.

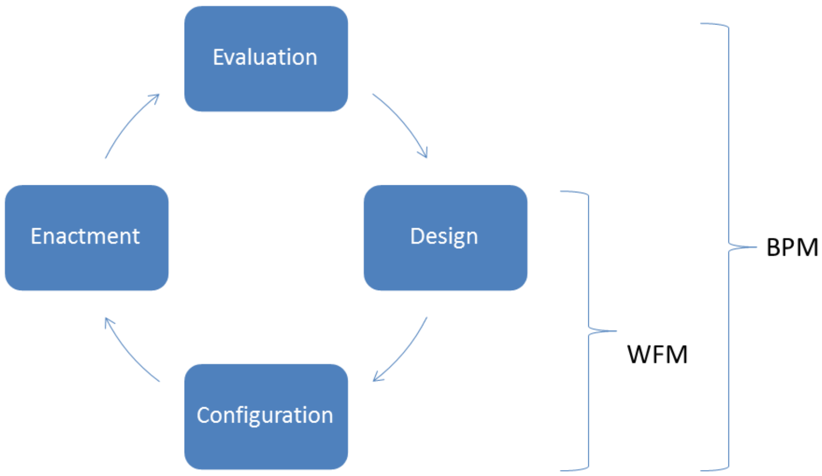

Figure 4: The BPM lifecycle [3], [27]

2.3 Information flow processing

The domain of *information flow processing* (IFP) deals with the collection of information from a large number of sources and its processing in a near real-time fashion in order to extract new information [28]. Examples of IFP systems are *complex event processing* (CEP), *active database* and *data stream management* systems. A high-level view of an IFP system is presented in Figure 5. The core of the IFP system is the IFP engine. This engine interfaces with three different types of external entities: *sources*, *rule managers* and *sinks*. Sources are entities which provide the IFP engine with flows of incoming information which it has to process. Rule managers are entities which define processing rules according to which the IFP engine has to process the flows of incoming information. Rules specify how the IFP engine should filter, combine and aggregate the incoming information flows in order to generate new outgoing flows. Sinks are entities which consume these outgoing flows of information. In our opinion, IFP systems can be very useful in the context of smart objects which can provide streams of real-world information.

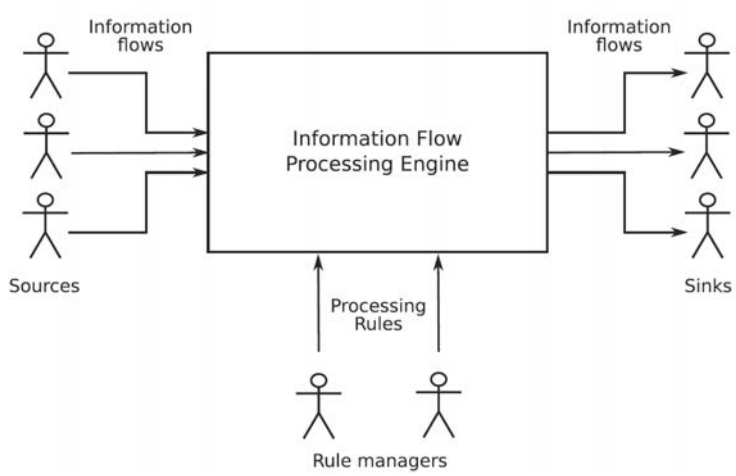

Figure 5: High-level view on an IFP system [28]

2.3.1 Complex event processing

Complex event processing (CEP) is a type of IFP. It deals with the real-time analysis of large numbers of events coming from various sources [28]. Its goal is to search for patterns in these events in order to discover higher-level events, called complex events. When searching for patterns, CEP analyses the casual, temporal and spatial correlations between events. The core of a CEP system is the *CEP engine*. *Event observers* (sources) provide the engine with a flow of notifications about events they have detected. In turn, the CEP engine uses *event patterns* (rules) in order to detect complex events based on these notifications. Once a complex event is detected, the engine sends a notification about that to interested *event consumers* (sinks).

2.4 Business activity monitoring

Business Activity Monitoring (BAM) is a concept introduced by Gartner [29]. It is related to the monitoring of business related parameters and events. The goal of BAM systems is to enable organizations to make informed decisions, quickly respond to problems and spot new opportunities. Typically, BAM systems use charts and notifications in order to communicate near real-time information to decision makers. Dashboards with comprehensive charts are used to present information about various business parameters (e.g. KPIs). Notifications are used to alert about the occurrence of important business events. In the context of an IFP system, a BAM system can play the role of a sink.

2.5 Event-driven BPM reference model

Event-Driven BPM (EDBPM) systems combine BPM, CEP and BAM technologies together [30]. In Figure 6, we outline the main components of an EDBPM system and their interconnections. Gray components are related to the BPM functionality of the system while blue components are related to the CEP functionality of the system.

Low-level event streams are illustrated at the bottom of the reference model. Events in these streams can come from various sources. For example, they can be generated by smart objects. A central place in the reference model is taken by the application server. It is a middleware platform which links the different components of the system. Besides that, it is responsible for aspects such as high availability, scalability, failover and transparency of heterogeneous infrastructures. The application server is connected to the streams of low-level events via adapters. The purpose of adapters is to convert heterogeneous events into a format with which the EDBPM system can work. Usually, there is one adapter for each type of events. The reference model illustrates two different types of specialists working with the system: workflow and event modelers. Workflow modelers are responsible for designing and partially for configuring business processes. During the configuration, they usually have to cooperate with IT specialists. They work with the workflow modeler component which corresponds to the process definition tools in the WFRM (see Figure 21). The result of their work are workflows (process definitions) which can be enacted by the workflow engine. This workflow engine corresponds to the workflow engine in the WFRM. Event modelers are responsible for identifying complex events of interest and modeling them. Event modelers work with the event modeler component. They model complex events in terms of an *Event Processing Language* (EPL). The result of their work are event scenarios which can be used by the rules engine in order to automatically detect complex events. Here, we have to note that the rule engine is a CEP engine. Another responsibility of event modelers is to determine how different complex events should be handled. They can define BAM views based on events as well as define responses to events.

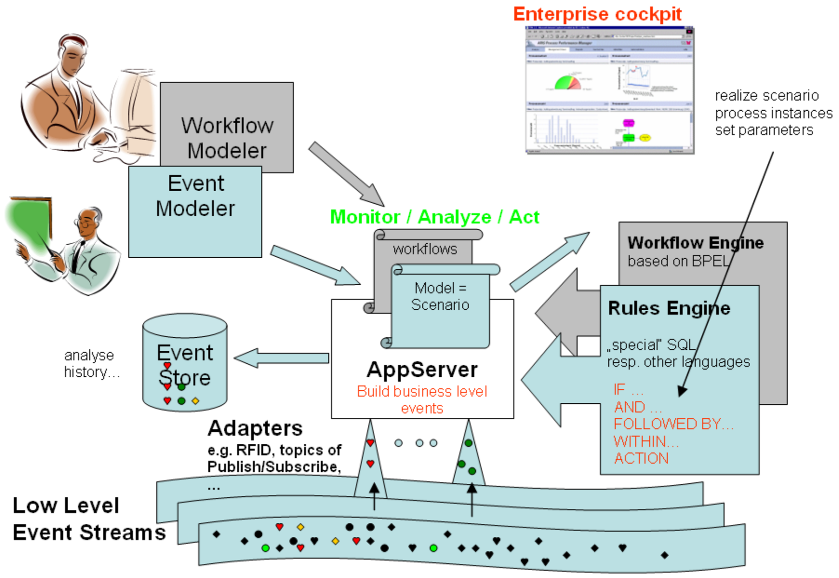

Figure 6: Event-driven BPM reference model [30]

2.6 Conclusions

As we already mentioned in section 2.1.1, smart objects can provide various types of functionality and they can expose it to the external world via services. In this study, we reviewed a number of technologies which can be used to incorporate these services in BPM systems in order to improve the real-world visibility, control and responsiveness of organizations. IFP technology can be used to process streams of information coming from a large number of smart object services. BAM technology can be used to present important information from smart object to decision makers. In the context of smart objects, both of these technologies can be used to improve the real-world visibility and responsiveness of organizations.

The most complete solution related to the integration of smart objects and BPM systems, we could identify in literature, are the EDBPM systems. In EDBPM systems, events generated by smart object services can be used to automatically trigger business processes (via CEP) as well as to provide decision makers with real-time insights about the physical environment (via BAM). Therefore, we conclude that an EDBPM system can utilize smart objects in order to improve the real-world visibility and responsiveness of organizations. However, EDBPM systems are limited to events generated by smart objects. They can neither utilize other types of information coming from smart objects nor use smart object services to control the state of the physical environment.

3 Business model

As we mentioned in the introduction, our goal is not to design a concrete BPM system for a concrete organizations but to design a new class of BPM systems. Therefore, in this section, we do not focus on a concrete business model but on a generic business model which describes the generic business environment and context in which the SBPM system will be used. The idea is that the concrete business models of various organizations from various industries should fit into this generic business model. We have to note that the business model we design in this section differs from the traditional definition of a business model. The business model we develop here is based on the definition of a business model presented in the COMET methodology [1].

3.1 Vision for change

In this section, we present our vision for the SBPM system. More precisely, we describe what the system should do and what benefits it will bring to organizations. This section contains both our own ideas and ideas inspired from literature. We provide argumentation and references wherever it is needed.

As we already mentioned in the introduction, one problem with traditional BPM systems is that they rely only on people in order to interact with their physical environment. Therefore, our goal is to design a SBPM system which can utilize smart objects in order to gather information about and modify the physical environment and, in this way, improve the real-world visibility, control and responsiveness of organizations.

Before we can specify how exactly the SBPM system should utilize smart objects, we first have to analyze what the similarities and differences between smart objects and people are. As we have shown in section 2.1, with the help of devices, smart objects are able to perform many tasks related to the physical environment which are usually assigned to people. Such tasks involve capturing, storing, processing and communication of information as well as actuation. So, we can say that there is some intersection between what people and smart objects can do. However, there is a big difference how well they can do certain tasks. Since smart objects are based on information technology, they inherit its strengths but also its limitations. On the one hand, compared to people, smart objects can work more accurately, quickly, cheaply and continuously. However, on the other hand, they are not as flexible as people. They cannot react and adapt easily to new, unseen situations. Having said that, we conclude that the SBPM system should rely on smart objects mainly when it comes to highly predictable, labor intensive tasks where speed and accuracy are important. This idea is also illustrated in Figure 7. Having in mind this general idea, next, we describe and motivate more precisely what the system should do.

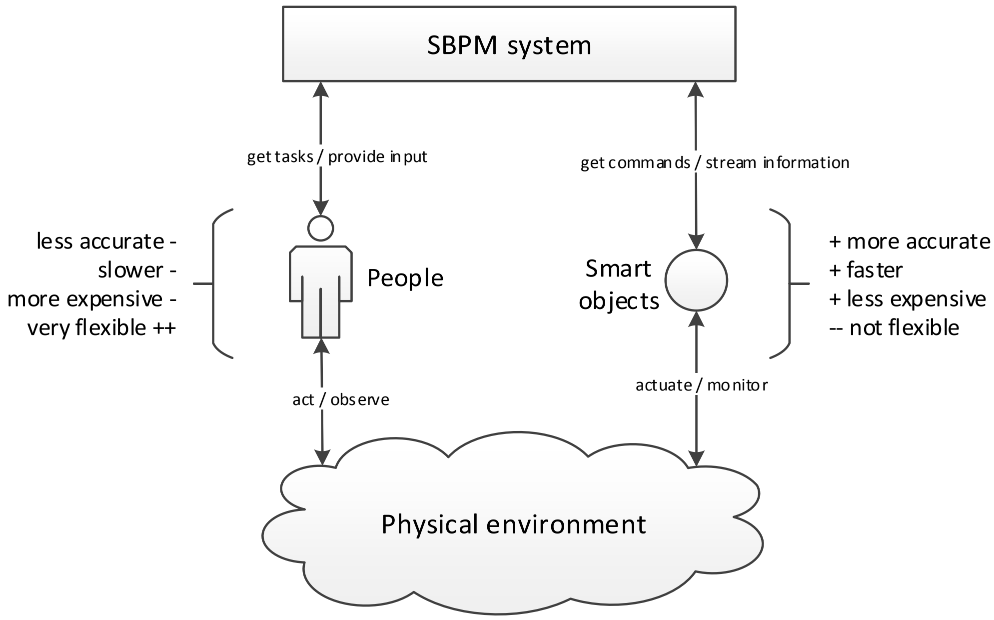

Figure 7: People and smart objects as mediators

3.1.1 Providing traditional BPM support

Since we defined the SBPM system as a type of BPM system, it has to support the management of business processes. More precisely, it should provide all functionalities that traditional BPM systems do. In this study, we assume that the WF reference model (described in section 10.2.5) represents a traditional BPM systems. The reason to make this assumption is the popularity of the model and the fact that most modern BPM systems today are based on it.

3.1.2 Involving smart objects in business processes

The SBPM system should support the involvement of smart objects in business processes. More precisely, it should support the automatic execution of business process activities using the services provided by smart objects. This idea is inspired by literature [9], [31], [32]. In order to achieve this, the system should support a number of things. First, various devices which can be used in smart objects should be able to connect to the system. Second, the system should support the definition of smart object services which are based on the services provided by the devices. In doing so, the system should take into account that services provided by devices are not always available because devices can often get disconnected [32]. The reason for this is the mobile nature of devices. In order to address this problem, we propose that smart object services are defined in a generic way so that they can be automatically implemented using only the currently available device services. In this way, the implementation of the smart object service will not depend on the availability of concrete device services. Third, the system should support the definition and enactment of business processes where smart object services can execute activities.

3.1.3 Utilizing information from smart objects

As we know from section 2.1.1, device services can provide streams of information related to the physical environment. For example, a temperature sensor can provide a stream of measurements. The SBPM system should be able to utilize this information in order to improve the organizational real-world visibility and responsiveness [9], [33]. Inspired from IFP technology described in section 2.3, we propose that the system should be able to process flows of information coming from various device services using rules. By doing so, it should be able to extract useful information related to smart objects. The question is how exactly this information can be used to improve the real-world visibility and responsiveness of organizations. Here, there are two main ideas. First, it should be possible to use rules to detect events in the physical environment and it should be possible to use these events as triggers in business processes. That means that the system should support the definition of business processes where events are assigned as triggers of activities. This will enable organizations to react automatically to business events from their physical environment and, therefore, it will improve their real-time responsiveness. This idea is inspired by CEP (described in section 2.3.1) and EDBPM (described in section 2.5) technology. Second, the useful information extracted with the help of rules should be presented to decision makers in a comprehensive way. This idea is inspired by BAM (described in section 2.4) and EDBPM technology.

3.2 Context statement

The goal of this section is to define the context the SBPM system in terms of stakeholders who are interested in the system and existing external systems which will interface with the system. First, in Table 2, we present a list of the system stakeholders. This list has been derived based on analysis of the WF and IFP reference models, literature.

Table 2: Stakeholders

Stakeholder	Description
Manager	Managers are concerned with the overall performance of the organization. They are typical for any organization.
Process designer	Process designers are responsible for the identification and definition of business processes [3]. Their work involves communication with process responsibles and process participants. Process designers has been identified as stakeholders from the WF reference model. There, they work with the process definition tools.
Process participant	Process participants are workers who actually perform the activities related to the enactment of a business process [3]. Usually, they are knowledgeable about the domain of the business process they participate in and the tasks they have to execute. Process participants are identified as stakeholders based on the WF reference model. There, they work with the workflow client applications.
Process responsible	Process responsibles are the people who are concerned with the correct and efficient enactment of process instances [3]. They administer and monitor the enactment of business processes instances. Process responsibles are identified as stakeholders based on the WF reference model. There, they work with the administration and monitoring tools.

Service designer	Service designers are concerned with the definition of smart object services based on the services provided by devices. They are stakeholders of the system because they need to define smart object services which will be used as actors in business processes.
Data analyst	Data analysts are concerned with the definition of information processing rules [28]. Their work involves communication with process responsibles and process participants. Data analysts are identified as stakeholders based on the IFP reference model. There, they are responsible for defining IFP rules.

In order to identify the external systems with which the SBPM system has to interface, we first have to understand how the SBPM system fits in the overall organizational IT landscape. In Figure 8, we show the different information technology layers which can generally be distinguished in organizations and position the SBPM system in their context. In Table 3, we describe briefly each layer, starting from the bottom one and moving up.

Table 3: Layers in the IT landscape

Layer	Description
Hardware	This layer consists of general purpose hardware (e.g. servers and PCs). It provides data storage, processing and communication resources to the operating system layer.
Operating system	This layer consists of system software which manages the resources provided by the hardware layer. It provides common services to the software layers above it.
Device	This layer contains heterogeneous devices which provide services to the infrastructure software layer. We do not distinguish between a device hardware layer and device software layer because device software is usually embedded into device hardware. In other words, it is developed specifically for the concrete hardware and cannot run on another hardware. Compared to general purpose hardware, device hardware provides more specialized but limited resources. For example, a sensor might be able to measure temperature but store only the last 100 measurements. Usually, services provided by devices are accessed by the infrastructure software via a communication network.
Infrastructure software	This layer contains general-purpose software systems which are not specific to any organizational business function. They provide common functionalities which can be (re)used by various applications. For example, a DBMS, BPMS and middleware software would often fit into this layer.
Application software	This layer contains application software systems (called applications) which are specific to organizational business functions. For example, a specialized financial software system (application) would fit into this layer.
User interface software	This layer provides interactive interfaces via which users can work with the software provided by the application and infrastructure software layers.

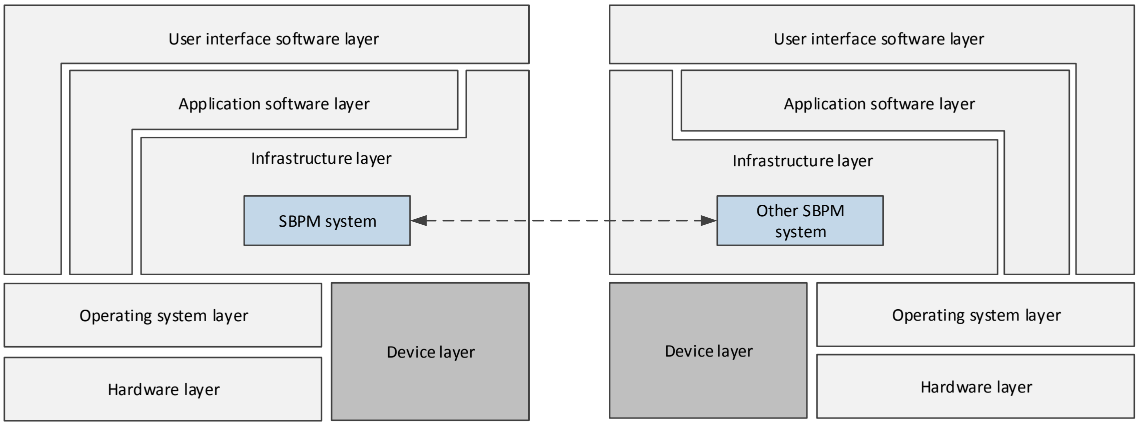

Figure 8: IT landscape of the SBPM system

As it can be seen in Figure 8, we position the SBPM in the infrastructure software layer. We do that, because it should be a software system which provides common functionality that can be (re)used by various software applications. More precisely, the SBPM system should be used to define and enact the flow of work between the individual tasks supported by applications. In this way, process logic is not hardcoded in applications but it is managed separately. This gives greater flexibility to organizations when they need to change their business processes. Furthermore, the SBPM system should also interface with the user interface, operating system and device layers. User interface modules should expose the functionality provided by the SBPM system to users (e.g. process designers and responsibles), while the SBPM system should utilize the services provided by operating systems and devices. Compared to the IT landscape in which a traditional BPM systems fit [34], the only new thing is the device layer. Finally, as illustrated, the SBPM should also interface with other SBPM systems. Of course, they should not interface directly but through their underlying layers (operating system and hardware layers). The goal of the last requirement is to ensure interoperability between different implementations of the SBPM system, potentially, hosted by different parties.

Finally, in Figure 9, we illustrate the SBPM system in the context of all stakeholders and external systems that we have identified in this section.

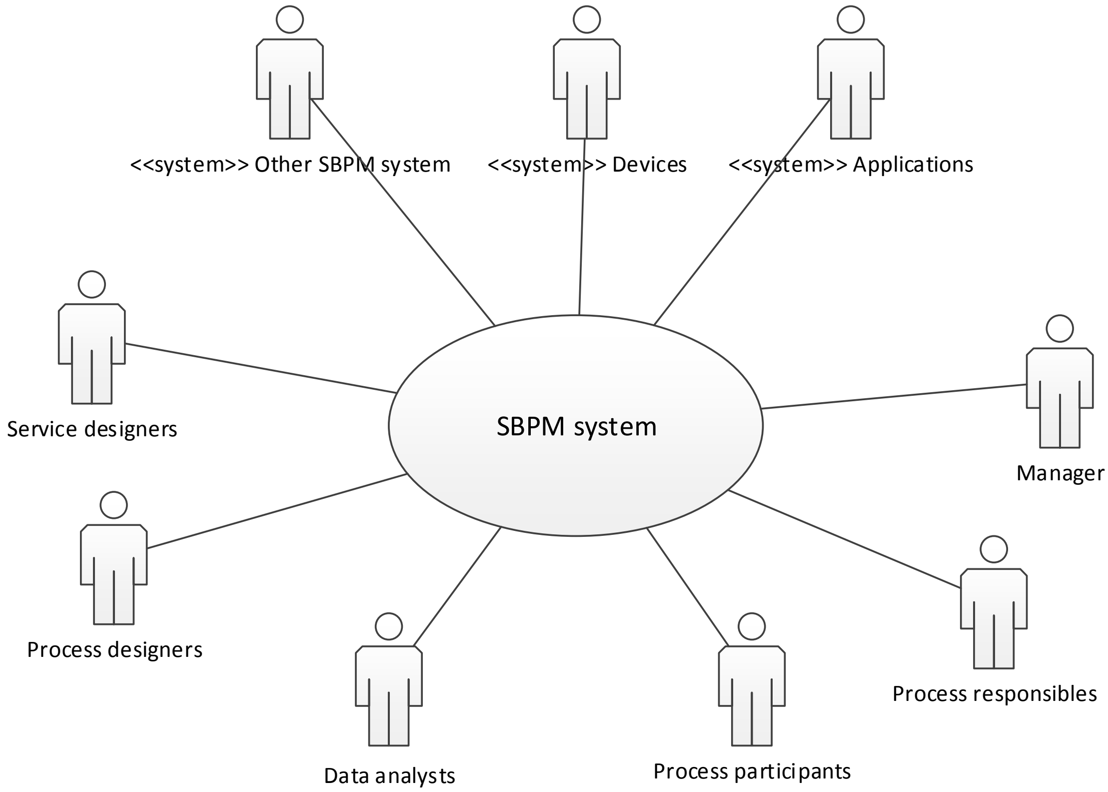

Figure 9: Context statement

3.3 Goal model

In this section, we describe the goals that the different stakeholders would like to achieve by using the SBPM system. We distinguish between two groups of stakeholders: stakeholders that govern the system and stakeholders that work directly with the system. Since the former group of stakeholders are dealing with governing the system, their viewpoint coincide with the viewpoint of the organization as a whole. Therefore, we call their goals – organizational goals. Since the latter group of stakeholders work directly with the system, we call their goals – user goals.

3.3.1 Organizational goals

In Table 4 we list and describe the organizational goals and in Figure 10 we show how they form a goal network. In order to identify these goals, we followed a bottom-up approach. Initially, we started with goals OG6, OG7, OG8, OG9, OG10 and OG11 (illustrated with darker color) which come from the initial definition of the SBPM system and the vision for change. Next, we analyzed how these goals related to each other. After that, iteratively, we identified higher-level goals which explain their existence. We did that by asking "Why?" questions. In Figure 10, we illustrate how higher-level goals relate to their sub-goals. The "+" sign on arrows indicates support, while the "-" sign indicates conflict. As it can be seen in Figure 10, the highest-level organizational goal is OG1 because all other goals support it directly or indirectly. In contrast, OG8 and OG11 are the lowest-level organizational goals because there are no other organizational goals which support them. OG10 is the only goal which is not supported by and does not support another goal. The reason for this is that this goal is not related to improving anything but it is there to express that since the SBPM system is a type of BPM system, organizations would expect it to provide traditional BPM functionality. Moreover, OG8, OG10 and OG11 might look more like solutions rather than goals. However, this is not the case because they are derived from the initial definition of SBPM system. This means that they are part of the problem, not part of the solution. Moreover, we have to note that there are relationships between goals OG2, OG3, OG4 and OG5 but they are not illustrated in Figure 10. The relationship between these four goals are described by the devil's quadrangle (see Figure 20).

Table 4: Organizational goals

Goal	Description
OG1: Improve customer satisfaction	By using the SBPM system, the organization wants to improve the satisfaction of its customers.
OG2: Improve process quality	By using the system, the organization wants to improve the performance of its operational business processes in terms of external and internal quality. Process quality performance is part of the devil's quadrangle [35] and it is further described in section 10.2.4. Improving external process quality may lead to improvement of customer satisfaction. For example, if an organization is providing its customers with better quality services, naturally, they will be more satisfied.
OG3: Improve process time	By using the system, the organization wants to improve the performance of its operational business processes in terms of time. Process time performance is part of the devil's quadrangle [35] and it is further described in section 10.2.4.
OG4: Improve process cost	By using the system, the organization wants to improve the performance of its operational business processes in terms of cost. Process cost performance is part of the devil's quadrangle [35] and it is further described in section 10.2.4.
OG5: Improve process flexibility	By using the system, the organization wants to improve the performance of its operational business processes in terms of flexibility. Process flexibility performance is part of the devil's quadrangle [35] and it is further described in section 10.2.4.
OG6: Real-world control	By using the system, the organization wants to improve its real-world control by improving its capability to modify the state of the physical environment in terms of accuracy, timeliness and completeness. This may lead to improvement in business process external quality and time [10].
OG7: Real-world visibility	By using the system, the organization wants to improve its real-world visibility by improving its capability to capture information about the state of the physical environment in terms of accuracy, timeliness and completeness. Such real-world visibility may enable management to make more optimal decision and, therefore, improve the business process performance in terms of external quality, time and cost [10].

OG8: Involve smart objects in processes	The organization wants to involve smart objects in business processes as proposed in the vision for change. This may lead to better real-world visibility and control because smart objects are able to collect information and actuate in more accurate, timely and complete way than people [10].
OG9: Organizational responsiveness	By using the SBPM system, the organization wants to improve the speed with which it detects and responds to the occurrence of important events in its physical environment. This goal is derived from the main goal of this study described in the introduction. Improving organizational responsiveness may lead to improvement in customer satisfaction. For example, if an electricity company is able to detect and resolve outages faster, customers will be more satisfied.
OG10: Traditional BPM support	By using the SBPM system, the organization wants to receive traditional BPM support as proposed in the vision for change. Such support may lead to improvement of business process performance in terms of quality, time [36] and cost but may lead to less process flexibility.
OG11: Utilize smart object information	By using the SBPM system, the organization wants to make use of the real-time information coming from smart objects as proposed in the vision for change. As we already mentioned, we have identified two ways to do that. The first one is to provide this information to decision makers in a comprehensive manner. Clearly, this may lead to improvement of real-world visibility. The second one is to automatically trigger business processes based on this information. Clearly, this may lead to improvement of real-time responsiveness.
OG12: Reduce costs	By using the SBPM system, the organization wants to reduce its costs. Here, we mean not only costs related to running business processes. The decision to add this goal has been taken based on interviews with experts (see section 7.2).

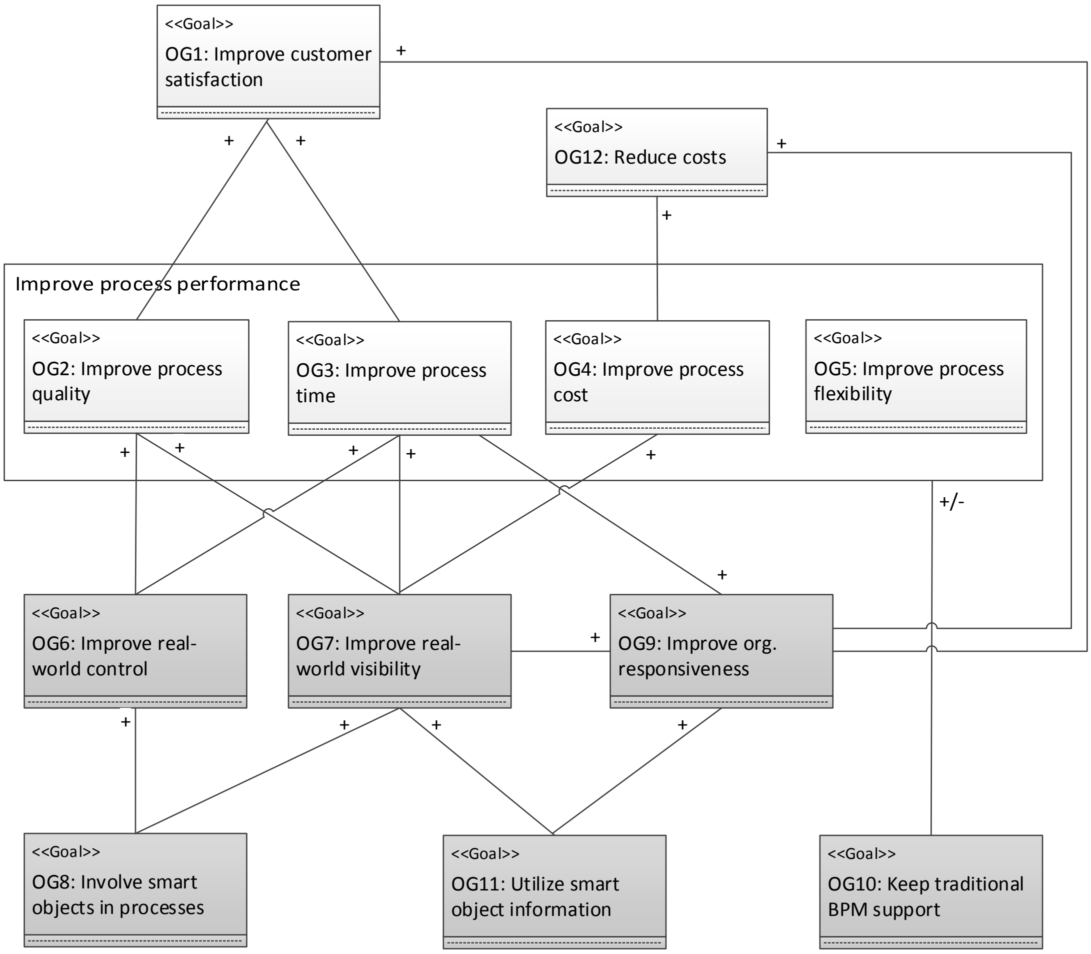

Figure 10: Organizational goal network

3.3.2 User and system goals

In Table 2, we present the goals which users want to achieve by using the system. In order to identify these goals we followed a top-down approach. Initially, we started with the lowest-level organizational goals OG8, OG11 and OG10. Then, by asking "How?" questions we derived the user goals which explain how the organizational goals are supported. So, we can say that the user goals logically follow from the organizational ones. Besides that, in order to identify the user goals, we have used literature and analyzed the functionality of WF, IFP, BAM and smart object technology. Note that, goals marked with "*" are intentionally described in a very generic way. In principle, they can be decomposed to more concrete goals, however, since they are related to traditional workflow support which is already well studied, we omit their detailed description.

Table 5: User goals

User	Goal	Description	Org. goal
Data analyst	UG1: Define information processing	The user wants to define how information coming from devices is going to be processed. This goal is inspired by IFP technology.	OG11
Data analyst	UG2: Define information presentation	The user wants to define how processed information can be presented to process responsibles in a comprehensive way. This goal is inspired by BAM technology.	OG11
Service designer	UG3: Define smart object services	The user wants to define smart objects services based on the services provided by devices. This goal is inspired by the conceptual model of a smart objects illustrated in Figure 3. More precisely, it is inspired by the possibility to have higher-level services which use resource-level services.	OG8
Process designer	UG4: Define business processes*	The user wants to define business processes as in a traditional BPM system. This goal is identified by analyzing the functionality of the process definition tools in the WF reference model.	OG10
Process designer	UG5: Involve smart objects services	The user wants to define business processes where certain activities can be executed by smart objects services. This goal is inspired by literature [37].	OG8
Process designer	UG6: Define automatic reactions	The user wants to define how business processes should react automatically to real-world information.	OG11
Process responsibles	UG7: Administer and monitor processes*	The user wants to administer and monitor business processes instances as in a traditional BPM system. This goal is identified by analyzing the functionality of the administration and monitoring tools in the WF reference model.	OG10
Process responsibles	UG8: Monitor physical environment	The user wants to monitor the physical environment. This goal is inspired by literature [9].	OG11
Process participant	UG9: Execute tasks*	The user wants to execute tasks related to business process instances as in a traditional BPM system. This goal is identified by analyzing the functionality provided by the client applications in the WF reference model.	OG10

4 Requirements model

4.1 Use case model

4.1.1 System boundary model

In this section, we capture the functional requirements for the SBPM system using use cases. At this stage, we look at the system as a black box. First, we provide a list of the use cases that the system should support. After that, we illustrate the use cases and their relationships in the system boundary model show in Figure 11.

The actors in the use cases are the users and existing systems which we presented in the context statement. We identify the use cases presented in this section, based on our vision for change as well as WF, IFP and BAM technology. More precisely, in order to define the content of Table 6, we decompose the high-level functionality proposed in the vision for change into a set of more concrete use cases. In the description of each use case, we provide more concrete motivation for its existence. We also relate each use case to a user goal from the business model. This additionally provides motivation for the existence of the identified use cases. In order to verify the completeness of the list of use cases we have identified here, we conduct a number of interviews with experts. A summary of the results from these interviews can be found in section 7.3.

Use cases marked with "*" are intentionally described in a very generic way. In principle, they can be decomposed to more concrete use cases, however, since they are derived from the analysis of the WF reference model, we omit their detailed description. The reader can refer to the WF reference model for more details.

Table 6: Use cases

Use case	Actors	Description	Goals
Define IFP rule	Data analyst	The actor uses the system in order to define a rule according to which information coming from devices will be processed. This use case includes the "Process information" use case. This use case is inspired by IFP technology. There, the actors in the use case are the rule managers. We describe this use case in more detail in Table 14.	UG1
Process information	Devices	The system uses the already defined information processing rules in order to automatically process the information flows coming from one or more devices. The result of the processing are new flows of information. This use case is inspired by IFP technology where it is supported by the IFP engine.	UG6
Define monitoring object	Data analyst	The actor uses the system in order to define a monitoring object (e.g. chart, notification) which can present information about smart objects (e.g. KPIs, events) to process responsibles. This use case is inspired by BAM technology [38].	UG1
Define service	Service designer	The actor uses the system in order to define a smart object service. Since devices are often mobile and have lower availability	UG3

		compared to information systems [32], we propose that service definitions should be independent from the concrete services provided by devices. This will enable the implementation of services based only on currently available device services.	
Define process*	Process designer	The actor uses the system in order to define a business process. This use case is inspired from the WF reference model where it is supported by the process definition tools.	UG4
Assign service	Process designer	This use case can extend the "Define process" use case. During the definition of a business process, the actor assigns the execution of an activity to one of the already defined smart object services.	UG5
Assign trigger	Process designer	This use case can extend the "Define process" use case. The actor uses the system in order to assign an already defined IFP rule as a trigger of a business process. This means that the business process will be automatically triggered once the rule is satisfied.	UG6
Monitor process*	Process responsible	The actor views information related to the status of running business process instances and their activities. This use case is inspired from the WF reference model where it is supported by the monitoring and administration tools.	UG7
Administer process*	Process responsible	The actor is managing the execution of business processes instances. He can manage users, roles and other resources related to them. Moreover, he can start, terminate and, in general, control the state of business process instances. This use case is inspired from the WF reference model where it is supported by the monitoring and administration tools.	UG7
View monitoring objects	Process responsible	The actor selects some of the already defined monitoring objects and view them. Selected monitoring objects can be presented in a dashboard [38]. This use case is inspired by BAM technology. We describe this use case in more detail in Table 15.	UG8
Start process	Process responsible	The actor selects one of the already defined business process definitions and start its execution.	UG7
Execute process	-	The system executes the business process instance by interpreting its corresponding business process definition. Based on this definition, it assigns tasks to process participants, invokes application software, calls smart object services, and control the flow of work. This use case includes the "Start process", "Invoke application", "Execute task" and "Compose service" use cases.	UG7, UG9
Invoke application*	Application	The system invokes an application in order to automatically execute a business process activity. This use case is inspired from the WF reference model where it is supported by workflow engines within the workflow enactment service.	UG7, UG9
Execute task*	Process participant	The actor executes a task related to a business process instance based on a task description. When the actor is ready, he notifies the system that the tasks is completed. This use case is inspired from the WF reference model where it is supported by the workflow client applications.	UG7, UG9
Compose service	Devices	The system composes and invokes a smart object service in order to automatically execute a business process activity. Based on the definition of the service, the system composes the currently	UG7

		available services provided by devices. The motivation for this use case is the same as for the "Define service" use case.	

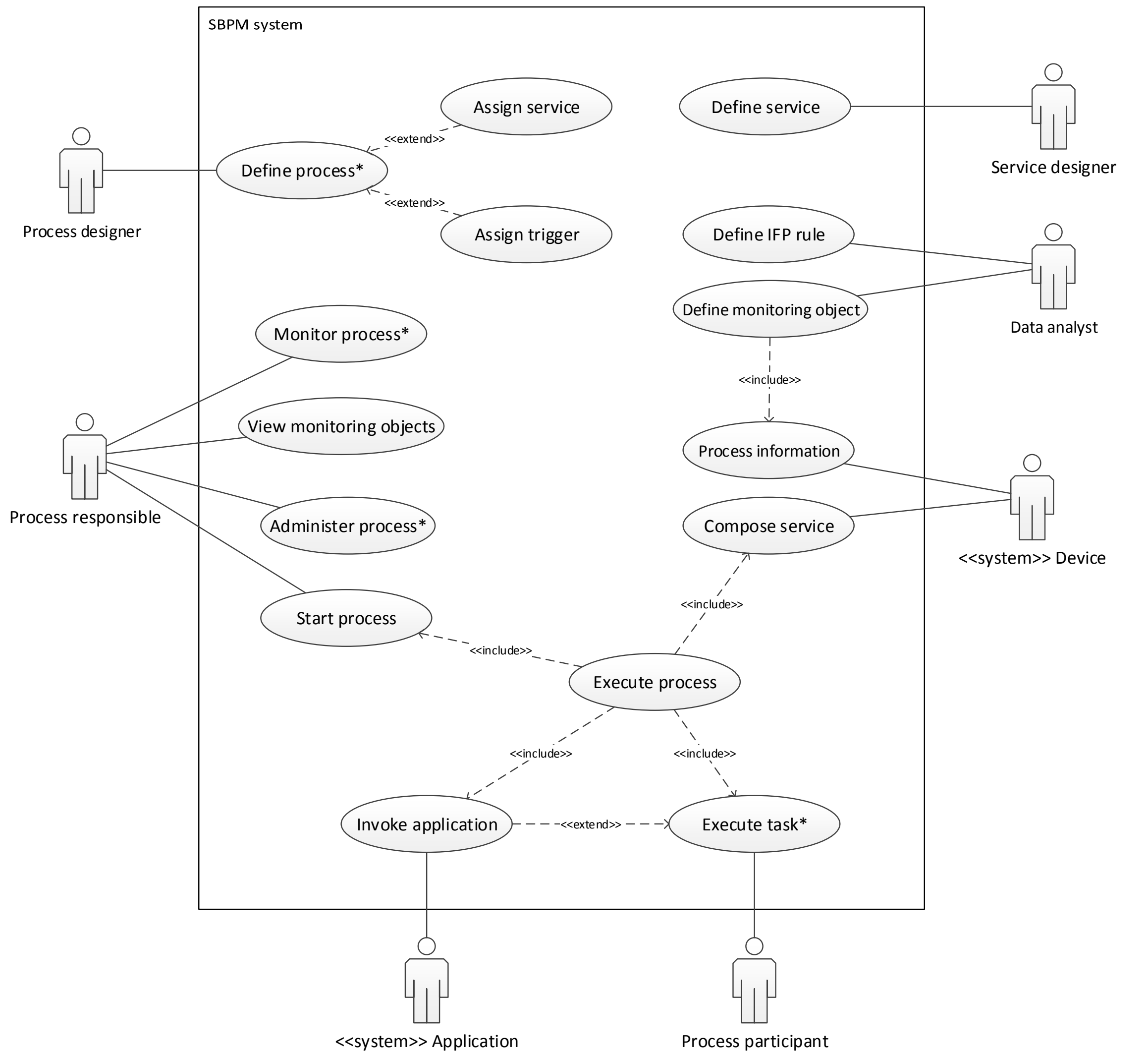

Figure 11: System boundary model

4.2 Architecture analysis

In this section, we analyze the use cases which the SBPM system should support in order to identify the modules of the system. This is an intermediary step which links the requirements model and the reference architecture model. In order to identify different modules, we group use cases together based on similar actors and similar functionality. For each group of use cases, we create one module which supports it. In order to simplify this grouping of use cases, in section 4.1.1, we intentionally separated certain use cases into different parts linked by a <<include>> dependency.

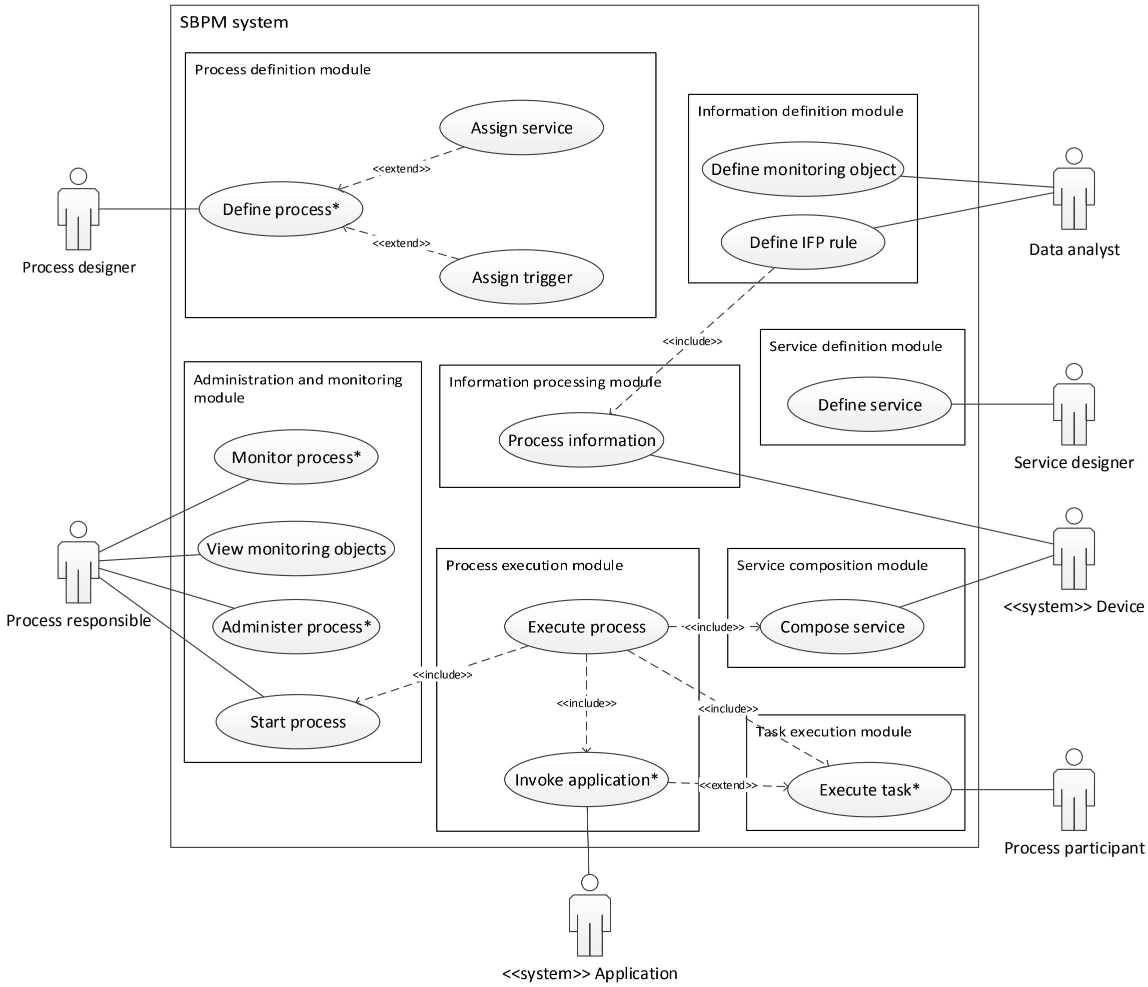

Figure 12: Grouping use cases into modules

In our opinion, the most interesting modules in Figure 12 are the information processing, process execution and service composition modules because they need to provide highly automated functionality. The information processing module supports the automatic processing of information coming from devices based on IFP rules. This functionality is in a separate module because we want to decouple the definition of IFP rules from the automated information processing based on these rules. The process execution module supports the automatic execution of business process definitions. This also involves the invocation of various external (to the system) applications. This functionality is in a separate module because we want to decouple the definition, monitoring and administration of business processes from their automatic execution. The service composition module supports the automatic composition and invocation of smart object services based on currently available device services. This functionality is quite specialized and, therefore, it is separated into a distinct module.

5 Reference model

Usually, an architecture is seen as a set of views, each of which addresses some stakeholder concerns [39], [40]. Rozanski and Woods proposed a framework distinguishing between 6 different views: functional, informational, concurrency, development, deployment and operational [39]. Alternatively, Kruchten proposed a framework describing 4 views: logical, process, physical and development [40].

In this section, our focus is on the functional view proposed by Rozanski and Woods which corresponds to the logical view proposed by Kruchten. This view is relevant to all stakeholders and its goal is to describe the functional elements of the system, their responsibilities, interfaces, and primary interactions [39]. The functional view has a central role in an architectural description because it greatly influences the shape of all other views. For that reason, it is often the first view to be designed in an architectural initiative.

In COMET, this view is captured by the component structure model. The goal of this model is to describe the high-level components of the system and their interdependencies. In this section, we propose a reference component structure model for the SBPM system. The goal of this model is to facilitate the design of the functional (logical) view of future, concrete SBPM systems. Most of the components in the component structure model are derived from the architectural analysis described in section 4.2. The rest of them are related to the integration patterns that we have chosen to use in order to interconnect the main components. Although COMET suggests the usage of UML to develop the model, in this study, we decided to use our own notation. The reason for this decision is that we want to use a less-restrictive notation when designing the reference model. Besides the component structure model, COMET also suggests the development of component interaction, component interface and component information models. However, in our study we do not develop these modules because they deal with concerns which are out of our scope. The actual reference model is illustrated in Figure 13.

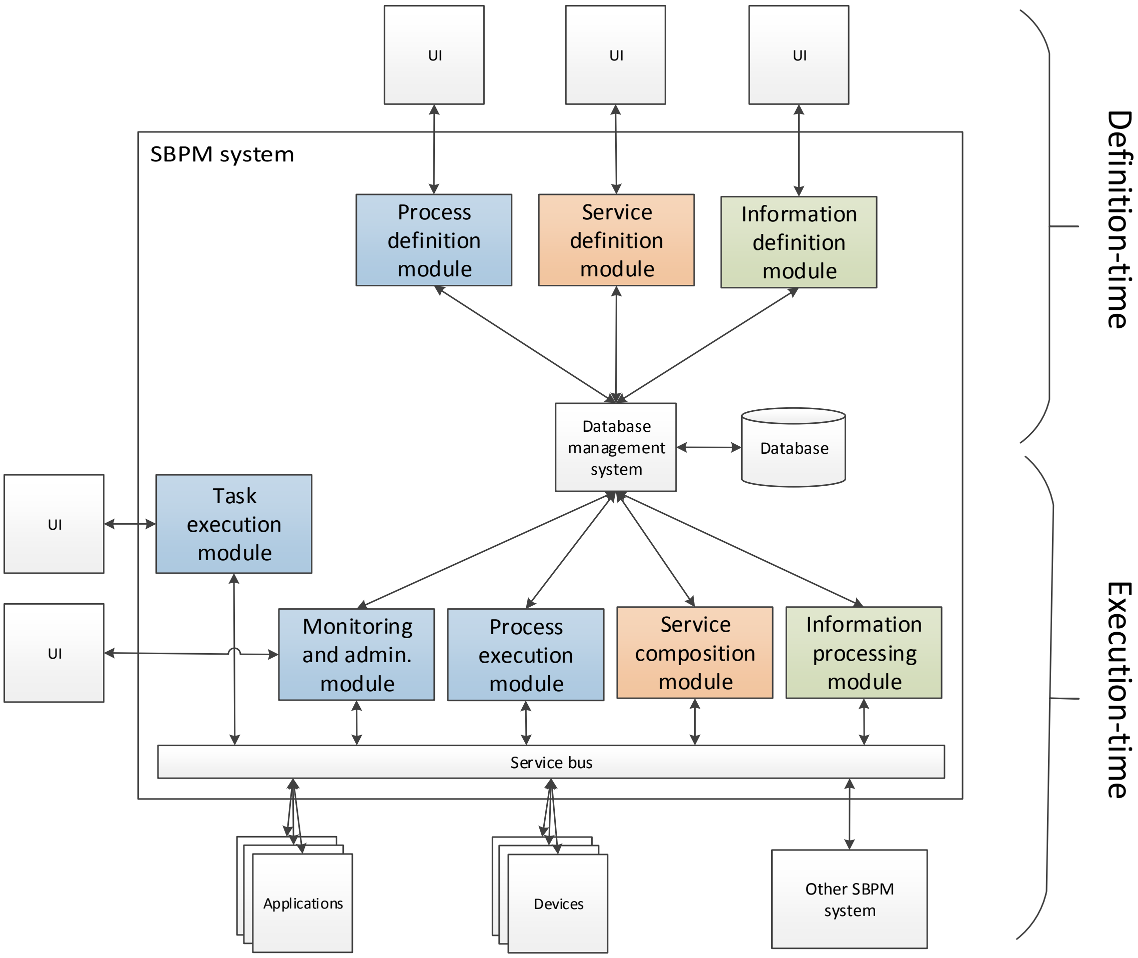

Figure 13: SBPM system reference model (functional view)

5.1 General structure of the model

In Figure 13, we show the component structure model of the SBPM system. This view illustrates the internal modules and external systems of the SBPM system as well as the interfaces between them. In order to clearly separate what is in and what is out of the system, the system boundary is shown. According to their type of functionality, the internal modules and external systems shown in the model are separated in two groups. The first group involves the modules and systems which support the definition of business processes, smart object services and information. We refer to this group as definition-time modules and systems. The second group involves the modules and systems which actually interpret the already created definitions. We refer to this group as execution-time modules and systems. We have to note that there is also similar division between build-time and run-time modules in the WF reference model [41].

We use two different interaction pattern in order to interconnect the internal modules of the SBPM system. The internal modules which belong to the definition-time group are connected with each other using a shared database pattern. We use this pattern in order to enable the permanent storage of the definitions produced by these modules. One the one hand, this enables definitions to be reused and changed rather than created from scratch every time. On the other hand, it enables flexible and asynchronous interaction between the three definition-time modules. For example, at any point in time, the process definition module can use the already created smart object service and processing rule definitions. Furthermore, the same shared database connects the definition-time modules with some of the execution-time modules. Again, the reason for this design decision is to enable flexible and asynchronous interaction between the modules which produce and the modules which consume the different definitions. Besides all of this, database technology is a good choice because it is one of the most widely used technologies in enterprises today. As far as the modules and systems from the execution-time group are concerned, we interconnect them using a shared service bus pattern. We use this pattern in order to enable synchronous interaction between the modules and systems. This communication pattern is the most logical one because execution-time modules and systems need to be active simultaneously and collaborate synchronously with each other in order to efficiently support the enactment of business processes and processing of real-world information.

Furthermore, we use coloring in order to group the internal modules of the system according to their functionality. Modules colored in blue are related to the definition and enactment of business processes. Modules colored in red are related to the definition and composition of smart object services. Modules colored in green are related to the definition and processing of information.

Compared to traditional BPM systems, there are a few new things to which we will pay extra attention. Internally, the modules related to the definition and composition of services as well as the modules related to definition and processing of rules are new.

5.2 Modules

In section 4.2, we already described the functionality which the modules of the SBPM system should support. In this section, our focus is on how they can be realized.

5.2.1 Database management system

As we already explained in section 5.1, the database management system plays a central role in our architecture since it provides a permanent data storage and facilitates the asynchronous communication between a number of different modules. As it is illustrated in Figure 13, the DBMS system is connected to a database. The database is a passive entity where data is actually stored. The database management system is an active module which manages the database. A more precise description of what data should be stored in the database of a traditional WFM systems is provided in the Mercurius reference architecture [34]. Typically, this includes organizational, product, workflow definition, process and application data. In the context of the SBPM system, the database should also store IFP rules, monitoring objects and smart object service definitions. Although the database management system might sound as a very technology-oriented module, in our opinion, this is not the case. It does not suggest any concrete technology but a wide class of technologies. Just a few examples are relational, object-oriented [42] and NoSQL [43] database management system.

5.2.2 Service bus

As we already explained in section 5.1, the service bus plays a central role in our architecture since it facilitates the synchronous communication between the execution-time modules and systems. In essence, it is a service-oriented middleware which focuses on interconnecting the services of these modules and system. It deals with routing of messages between services, data and protocol transformation, message queuing, quality of service monitoring, security monitoring, transaction management, etc. [44]. On the one hand, the service bus hides a lot of the complexity related to interconnecting a large number modules and systems together. On the other hand, it enables the interconnection of modules and systems implemented with heterogeneous technologies, running on heterogeneous platforms and using heterogeneous standards. Enterprise service bus technology is well known and widely used for the implementation of Service-Oriented Architecture (SOA) in large organizations today. The novel application of the service bus in our architecture is related to the integration of services hosted on devices. There are already a few studies related to this topic [4], [45], [46].

5.2.3 Service definition and composition modules

The smart object service definitions developed by data analysts using the service definition module are stored in the shared database. Once they are stored there, they can be used in the process definition module and involved in business process definitions. Before a smart object service is invoked by the process execution module, it has to be composed by the service composition module. The module composes the service based on its definition and the currently available on-device services.

In Figure 14, we illustrate a high-level model of an *Automated Service Composition* (ASC) system from literature [47]. In order to demonstrate the buildability of our architecture, in Table 7, we map the elements of the ASC system to the elements of the SBPM system.

Table 7: Mapping ASC and SBPM system elements

ASC system elements	Mapping to SBPM system elements
Service provider	In our architecture, service providers are devices.
Service repository	In our architecture, the service repository can be part of the service bus or the service composition module.
Execution engine, evaluator, translator and process generator	All of these modules deal with the actual composition of smart object services based on a service specification [47]. Therefore, in our architecture, they all are part of the service composition module.
Service requester	In our architecture, the service requester is the process execution module which actually initiates the invocation of smart object services.

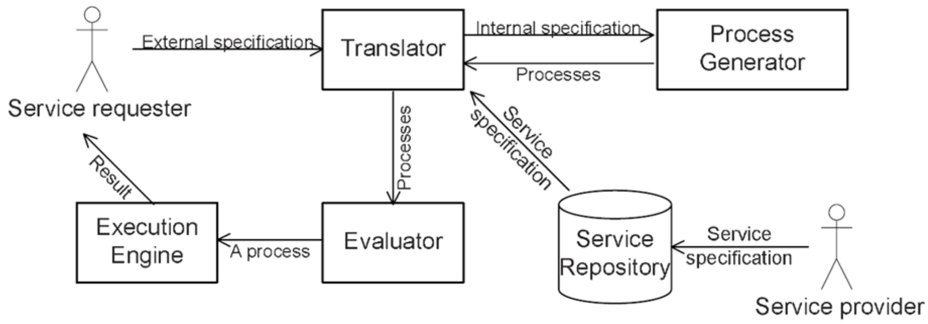

Figure 14: High-level model of an ASC system [47]

5.2.4 Information definition and processing modules

The information flow processing rules and monitoring objects developed by data analysts using the information definition module are stored in the shared database. Then, from there, they can be accessed and utilized by the information processing and administration and monitoring modules, respectively.

In Figure 15, we present a high-level model of an *Information Flow Processing* (IFP) system from literature [28]. In order to demonstrate the buildability of our architecture, in Table 8, we map the elements of the IFP system to the elements of the SBPM system.

Table 8: Mapping IFP and SBPM system elements

IFP system elements	Mapping to SBPM system elements
Rules	As we already mentioned, in our architecture, IFP rules are developed with the help of the information definition module and they are stored in the shared database.
Receiver, clock, decider, producer, forwarder and knowledge base	All of these elements deal with the processing of incoming flows of information based on the defined IFP rules [28]. Therefore, in our architecture, they all are part of the information processing module.
Sources of information flows	In our architecture, the sources of information flows are the services provided by devices which stream information to the SBPM system.
Sinks of information flows	In our architecture, the sinks of information flows are the process execution and administration and monitoring modules. These two modules directly consume the information flows produced by the information processing module.

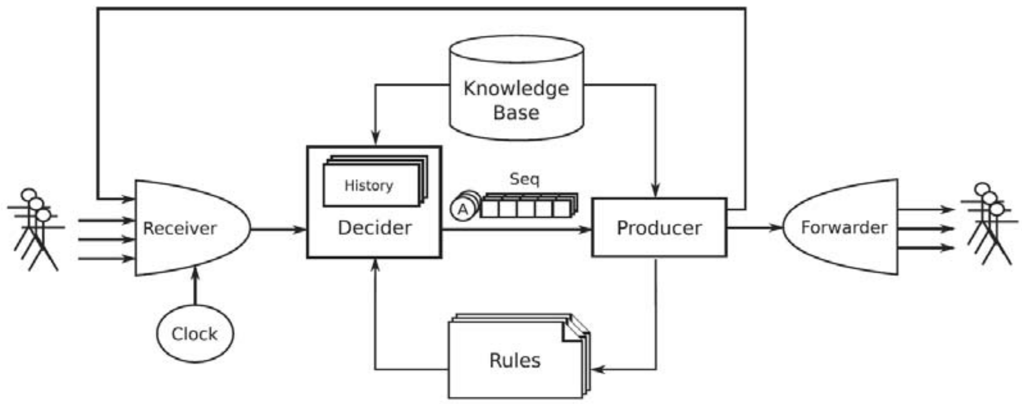

Figure 15: High-level model of an IFP system [28]

5.2.5 Process definition module

According to its functionality, this module corresponds to the process definition tools in the WF reference model. An architectural model of such module has already been described in the Mercurius reference architecture [34]. The novelty of the process definition module in our case is that it should support the involvement of smart object services in business processes and the definition of business process triggers related to IFP rules. This requires the module to support a new or extended modeling notation. Process definitions developed with the process definition module are stored by the database management system. The interface between the process definition module and the database management system corresponds to interface 1 in the WF reference model.

5.2.6 Task execution module

According to its functionality, the task execution module corresponds to the workflow client applications in the WF reference model. It is a component which manages the interaction between process participants and the process execution module. It provides process participants with information about tasks they can perform and inform the process execution module when they are completed. This module does not need to provide any specialized functionality for actually performing the tasks. The task execution module should communicate with the process execution module via the shared service bus. The interface between these two modules corresponds to interface 2 in the WF reference model. The design of this module can be based on the architectural model of the workflow client described in the Mercurius reference architecture [34].

5.2.7 Monitoring and administration module

According to its functionality, this module is similar to the administration and monitoring tools described in the WF reference model. On the one hand, the module should support functionality related to business process administration and monitoring. For that reason, it should communicate with the process execution module via the shared service bus. The interface between these two modules corresponds to interface 5 in the WF reference model. The novelty of the module is that it should support not only the monitoring of business processes but also the monitoring of the physical environment using information from devices. In order to do this, our monitoring and administration module should communicate with the information processing module via the service bus. With the help of monitoring objects, the module should be able to present information about the physical environment to the process responsibles. The design of the monitoring and administration module can be based on the design of the workflow client described in the Mercurius reference architecture [34]. As far as the monitoring part of the module is concerned, ideas from the architecture of BAM systems can be used [38].

5.2.8 Process execution module

According to its functionality, this module corresponds to the workflow enactment service in the WF reference model. It contains a number of process execution engines which correspond to the workflow engines in the WF reference model. The module uses these engines in order to execute business process definitions stored in the shared database. The novelty of the module is mainly related to the workflow engines. Besides the traditional functionality related to workflow engines such as interpreting process definitions and invoking applications, in our architecture they should also be able to invoke smart object services by interfacing with the service composition module. The design of the process execution module can be based on the architectural model of the workflow enactment server described in the Mercurius reference architecture [34].

6 Evaluation through a scenario

In this section, we use a descriptive approach to evaluate the SBPM system in the context of a more concrete scenario from the smart energy sector. In section 1.4, we already explain our motivation for choosing exactly the smart energy sector. More precisely, in this section, we design a business and requirements models for a SBPM system which uses smart meters in order to improve the visibility and responsiveness of electricity companies with regards to restoration of power outages. We have identified and described this concrete smart energy scenario mainly based on literature [48]–[50]. By designing a SBPM system for this concrete scenario, we want to show how the generic, industry-independent SBPM system that we have already designed in sections 0, 4 and 5 can be applied in a more concrete, industry specific scenario. Everywhere we show how the elements of the energy specific business and requirements models relate to the elements of the generic business and requirements models.

6.1 Business model

6.1.1 Problem statement

Electricity meters are devices used by electricity companies in order to measure the amount of electric energy consumed by their customers. Nowadays, internationally, the most widely used type of electricity meters for residential and commercial customers are the *electromechanical induction watt-hour meters* [51]. In the rest of the study, we will call them *conventional meters*.

Power outages are failures in the power grid which leave customers of electricity companies without power [50]. There are various events which can cause an outage. Just a few examples are a short circuit, network overload and damage to distribution lines. A major challenge for electricity companies which still rely on conventional meters is to detect outages and restore power quickly [48]–[50]. Such companies usually rely on their customers to call and complain when they do not have electricity [49]. There are at least a few problems with this approach. First, it can be slow because customers are not always immediately aware of the outage or they might intentionally wait for some time before calling. This can delay the detection of the outage with hours. Second, about 70 – 75% of calls are related to single customer outages and more than 30% of them are customer problems that can be solved without involving a technical crew. Third, information which can be collected from customer calls is not always accurate and complete due to the limited knowledge of customers. Finally, when companies dispatch technical crews to restore power, it takes additional time to confirm that the restoration indeed has been successful. Often such confirmation requires an additional phone conversation with the customer. The ability of electricity companies to cope with outages quickly is important because it can lead to penalties for them [50] and influence the level of satisfaction of their customers [49].

6.1.2 Vision for change

Smart meters are electronic devices which have sensing, computation, storage, communication and actuation capabilities. Compared to conventional meters, they have a number of advantages [50], [52]:

- Smart meters can be connected to the information systems used by companies and have a two-way communication with them. They can send data to information systems as well as receive commands from them.
- Besides consumption information, smart meters can also collect, store and communicate power quality information. For example, they can collect information related to electrical voltage, current and frequency.
- Smart meters can be used to control remotely the consumption of customers. More precisely, they can be used to limit, stop and restore their consumption.
- Smart meters usually have user interface through which they can provide customers with useful information related to consumption and tariffs.
- Smart meters are often equipped with batteries so that they can continue their work during a power outage.

Smart meters are devices which can be used to transform individual customers into smart objects. In this study, such smart objects will be called *smart customers*. Moreover, smart meters can be used to transform clusters of closely located customers into smart objects. In this study, such smart objects will be called *smart clusters*. In the context of the conceptual model of a smart object described in section 2.1.1, a smart cluster is the composition of a physical and digital entity. The closely located customers can be seen as a physical entity while the data about them stored in information systems can be seen as a digital entity. Smart meters can be seen as the devices which links the first two together. We have to note that each smart meter is a composite device which contain both sensor and actuator devices. Using the services provided by smart meters, electricity companies can monitor smart clusters.

In Figure 16, we provide a high-level view of a typical outage restoration business process from literature [50]. Everything starts with a number of trouble calls from customers who report their outages. Next, the data collected from the calls is analyzed in order to locate and diagnose the outage. After that, a technical crew is dispatched by an outage dispatcher in order to confirm and, eventually, restore the power. Once the technical crew arrives at the location of the outage, it confirms or rejects the occurrence of an outage. In the latter case, the customer is notified and the restoration process ends. Otherwise, the technical crew performs the required reparation and restores the power supply. At the end, the technical crew confirms that power is restored and the customer is notified (usually by phone).

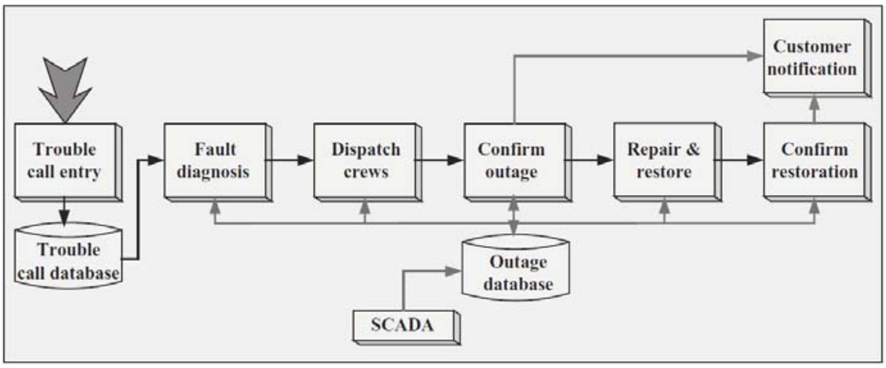

Figure 16: Outage restoration process [49], [50]

In order to improve the real-world visibility and responsiveness of electricity companies in the context of outage restoration, the SBPM system can utilize the services provided by smart clusters.

First, the system can improve the speed with which electricity companies detect outages if it utilizes the power quality information which smart meters can provide about smart clusters [49]. In order to do that, it should be able to gather and process information coming from a large number of smart meters. Inspired by CEP technology, we propose that the SBPM system should do this by supporting the definition and detection of complex events representing outages. In the context of the outage restoration process from Figure 16, this will lead to speed improvements in the first step where electricity companies await phone calls from customers. In the new situation, the SBPM system will automatically detect power outages based on data from smart meters. Second, the SBPM can improve the speed with which electricity companies react to outages. In order to do that, it should be able to inform the right people about outages and, if needed, automatically trigger business processes which will handle their restoration [49]. Finally, during the power restoration process, the system should be able to use the power quality information form smart clusters in order to confirm power outages and restorations [49]. In the context of the outage restoration process from Figure 16, this will lead to speed and cost improvements in the "*Confirm outage*" and "*Confirm restoration*" steps.

6.1.3 Context statement

In this section, we identify the stakeholders and external systems related to the SBPM system in the context of outage restoration and link them to the generic stakeholders and external systems.

Table 9: Stakeholders

Stakeholder	Description	Relates to
Manager	Managers are concerned with the overall performance of the organization regarding power outage restoration.	**Manager**
Outage dispatcher	Outage dispatchers are responsible for the quick detection and restoration of power outages [49], [50]. They use available information in order to detect outages, diagnose and locate	**Process responsible**

	outages, dispatch technical crews and monitor the whole process of outage restoration.	**and participant**
Restoration process designer	Restoration process designers are concerned with the definition of restoration business processes.	**Process designer**
Outage designer	Complex event designers are concerned with the definition of complex events patterns which can be used to automatically detect power outages.	**Data analyst**
Service designer	The service designer is concerned with designing smart cluster services which are based on the services provided by smart meters.	**Service designer**
Technician	Technician are concerned with the technical work related to an outage restoration [49], [50]. Technicians are usually grouped together into technical crews which are dispatched in order to restore a power outage.	**Process participant**
Customer	The customer is concerned with notifying the electricity company about outages by phone.	**Process participant**

Table 10: External systems

System	Description	Relates to
Smart meters	Smart meters are devices which provide data related to power consumption [50], [52].	**Devices**
Outage management system	An outage management system (OMS) is a software system used by electricity companies to support the process of power restoration [50]. What is important to note, however, is that these system are not process aware. In other words, processes are hardcoded in them.	**Applications**

6.1.4 Goals model

Similarly to the generic goal model, we distinguish between organizational and user goals. Moreover, we show the relation of the goals to the goals in the generic goal model.

Table 11: Organizational goals

Goal	Description	Relates to
SOG1: Improve customer satisfaction	By using the SBPM system, the organization wants to improve the satisfaction of its customers related to outage restoration.	**OG1**
SOG2: Improve restoration process quality	By using the system, the organization wants to improve the quality of the outage restoration business process.	**OG2**
SOG3: Improve restoration process time	By using the system, the organization wants to speed up the outage restoration business process. In turn, this can lead to improvement in customer satisfaction, since customers will have to stay less without electricity.	**OG3**

SOG4: Improve restoration process cost	By using the system, the organization wants to reduce loses and costs related to the outage restoration business process.	**OG4**
SOG5: Improve process flexibility	By using the system, the organization wants to improve the performance of its operational business processes in terms of flexibility.	**OG5**
SOG7: Improve power quality visibility	By using the system, the organization wants to have more accurate, timely and complete power quality information about its customers. Such improved power quality visibility may enable the organization to be more efficient in detecting and restoring power outages in terms of time and cost.	**OG7**
SOG8: Involve smart meters in processes	The organization wants to involve smart meters in the outage restoration business processes in order to confirm outages and outage restorations. This will lead to better power quality visibility because smart meters can provide more accurate, timely and complete information than people.	**OG8**
SOG9: Improve outage responsiveness	By using the SBPM system, the organization wants to improve the speed with which it detects and responds to the occurrence of outages. In turn, this can lead to improvement in customer satisfaction.	**OG9**
SOG10: Traditional BPM support	By using the SBPM system, the organization wants to receive traditional BPM support as proposed in the vision for change. Such support may lead to improvement of business process performance in terms of quality, time [36] and cost but may lead to less process flexibility.	**OG10**
SOG11: Utilize power quality information	By using the SBPM system, the organization wants to make use of the power quality information which can be collected by smart meters. On the one hand, this information should be presented to decision makers. On the other hand, it should be possible to automatically trigger an outage restoration business process based on the information. In turn, this may lead to improvement in power quality visibility and outage responsiveness.	**OG11**
SOG12: Reduce costs	By using the SBPM system, the organization wants to reduce costs related to outage restoration. For example, it might want to reduce costs related to penalties due to delayed power restoration [49].	**OG12**

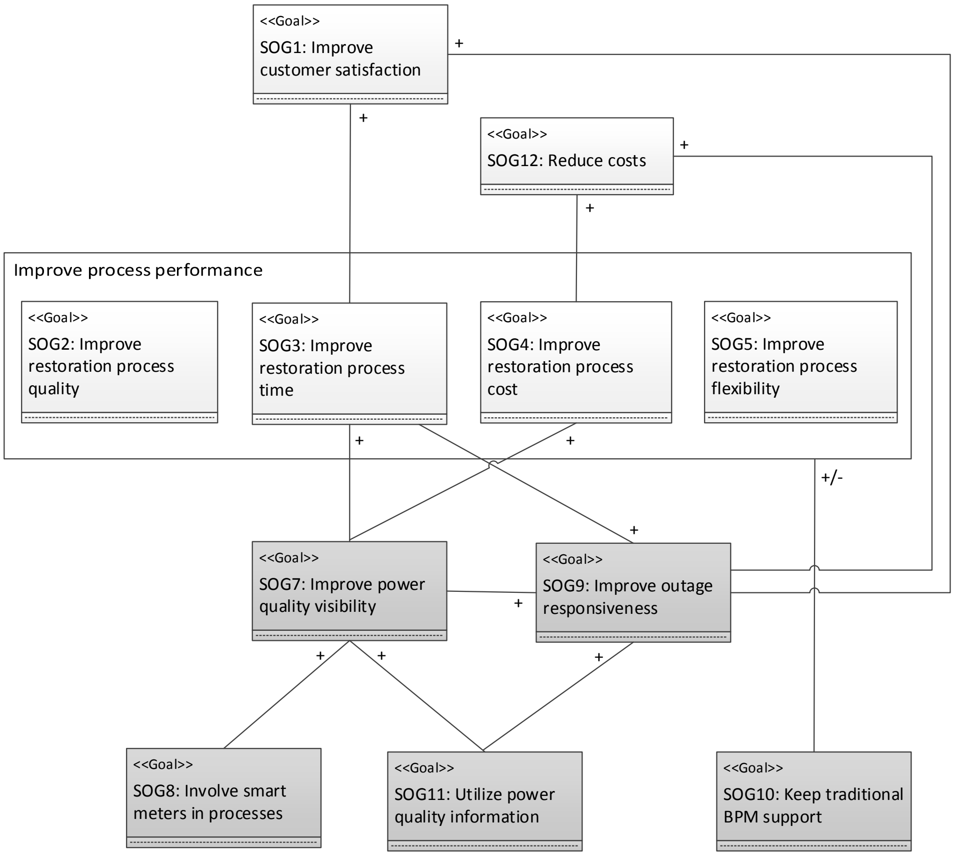

Figure 17: Goal network

Table 12: User goals

Stakeholder	Goal	Description	Org. goals	Relates to
Outage designer	SUG1: Define outage events	The user wants to define how outages corresponding to smart clusters should be detected automatically based on information from smart meters.	SOG11	**UG1**
Outage designer	SUG2: Define outage notifications	The user wants to define how outage dispatchers will be notified about smart cluster outages.	SOG11	**UG2**
Service designer	SUG3: Define smart cluster service	The user wants to define smart cluster services based on the services provided by smart meters.	SOG8	**UG3**
Restoration process designer	SUG4: Define outage restoration process*	The user wants to define outage restoration business process as in a traditional BPM system.	SOG10	**UG4**

Restoration process designer	SUG5: Involve smart cluster services	The user wants to define business processes where certain activities can be executed by smart cluster services.	SOg8	**UG5**
Restoration process designer	SUG6: Define outage reaction	The user wants to define how business processes should react automatically to outages.	SOG11	**UG6**
Outage dispatcher	UG7: Administer and monitor processes*	The user wants to administer and monitor outage restoration business processes instances as in a traditional BPM system.	SOG10	**UG7**
Outage dispatcher	UG8: View outage notifications	The user wants to view notifications related to smart cluster outages.	SOG11	**UG8**
Outage dispatcher	UG9: Execute tasks*	The user wants to execute tasks related to the outage restoration business process instances as in a traditional BPM system.	SOG10	**UG9**

6.2 Requirements model

In this section, we define the uses cases which have to be supported by the SBPM system and map them to the modules of the generic reference architecture.

Table 13: Use cases

Use case	Actors	Description	Goals	Relates to
Define outage	Outage designer	The actor uses the system in order to define an event which corresponds to a smart cluster outage. This event can be automatically detected by the system based on smart meter information. This use case includes the "Detect outage" use case. This use case is inspired by CEP technology.	SUG1	**Define IFP rule**
Detect outage	Smart meters	The system use the already defined events in order to automatically detect outages based on the information flows coming from smart meters. This use case is inspired by CEP technology.	SUG6	**Process information**
Define outage notification	Outage designer	The actor uses the system in order to define a notification which will be shown to outage dispatchers when an outage is detected. This use case is inspired by BAM technology.	SUG1	**Define monitoring object**
Define service	Service designer	The actor uses the system in order to define a smart cluster service based on the services provided by smart meters.	SUG3	**Define service**
Define restoration process*	Restoration process designer	The actor uses the system in order to define an outage restoration business process.	SUG4	**Define process***

Assign cluster service	Restoration process designer	This use case can extend the "Define restoration process" use case. During the definition of a restoration business process, the actor assigns the execution of an activity to one of the already defined smart cluster services.	SUG5	**Assign service**
Assign event trigger	Restoration process designer	This use case can extend the "Define restoration process" use case. The actor uses the system in order to assign an already defined outage event rule as a trigger of a business process. This means that the business process will be automatically triggered once the event is detected.	SUG6	**Assign trigger**
Monitor restoration process*	Outage dispatcher	The actor views information related to the status of running outage restoration business process instances and their activities.	SUG7	**Monitor process***
Administer restoration process*	Outage dispatcher	The actor is managing the execution of outage restoration business processes instances.	SUG7	**Administer process***
View outage notifications	Outage dispatcher	The actor views outage notifications and the related to them information. This use case is inspired by BAM technology.	SUG8	**View monitoring objects**
Start restoration process	Outage dispatcher	The actor starts the execution of the outage restoration business process.	SUG7	**Start process**
Execute restoration process	-	The system executes the outage restoration business process instance by interpreting its corresponding business process definition. This use case includes the "Start restoration process", "Invoke OM system", "Execute task" and "Compose service" use cases.	SUG7, SUG9	**Execute process**
Invoke OM system*	OM system	The system invokes a service of the outage management system in order to automatically execute a business process activity.	SUG7, SUG9	**Invoke application***
Execute task	Outage dispatcher	The actor executes a task related to an outage restoration business process instance. The execution of the tasks can be supported by the OM system which gets invoked by the system.	SUG7, SUG9	**Execute task**
Compose service	Smart meters	The system composes and invokes a smart cluster service in order to automatically execute an outage restoration business process activity. Based on the definition of the service, the system composes the currently available services provided by smart meters.	SUG7	**Compose service**

Based on the mapping between specific and generic use cases, we can also map the specific use cases to the modules in the reference model from section 5. The reason for this is that the modules in the reference model are identified based on grouping of generic use cases (see section 4.2). Due to the triviality of this mapping, we will not present it explicitly in this study.

6.3 Conclusions

In section 6, we showed how the generic business and requirements models related to the SBPM system can be specialized and applied in the context of outage restoration. On the one hand, this demonstrates the applicability of the generic SBPM system in a concrete scenario from industry. On the other hand, it gives evidence for the completeness of the generic business and requirements models, since all elements from the specific models can be mapped to elements from the generic ones. One limitation of the outage restoration scenario that we developed is that it does not cover functionality related to real-world control of the physical environment. Therefore, in section 6, we illustrated only the applicability of the generic SBPM system regarding the improvement of real-world visibility and responsiveness of organizations.

In order to provide additional confidence in the outage restoration scenario illustrated in section 6, we conducted an interview with a project manager from Essent. Essent is the biggest electricity company in Netherlands with more than 2 million electricity customers. The interview did not have the purpose to validate the details in every single model we have created but rather provide us with general feedback on our ideas. The interviewee confirmed the need for automation in the process of outage restoration and he was positive about the idea of using smart clusters as a way to gather real-world information related to outages. Moreover, the interviewee mentioned that SCADA [53] systems are currently widely used to monitor information coming from various devices in their infrastructure. However, they do not provide any process support. The interviewee pointed out that the SBPM system would be superior to SCADA systems and there is a need for it.

7 Evaluation through interviews

In order to further validate our work, we used interviews with experts from Capgemini Nederland. In total, we conducted 4 evaluation interviews where each of them was between 1 and 1.5 hours long. Due to the limited time that we had, the focus of the interviews was on evaluating the quality of only the most important generic models in our study. We have to note that our goal was not to perform a thorough quantitative evaluation but a concise qualitative one.

In order to prepare the experts for the interviews, we sent them a document to read in advance. The document was describing our research problem and the concept of a smart object. Our purpose was to provide the experts with context for the interviews and give them some time to think over the problem. As it turned out, this was a successful strategy. All of the interviewees had actually read the document before the interviews and they had already formulated some questions and opinions. This saved us a lot of precious time during the interviews and, in general, improved the overall quality of the interviews.

The focus of the interviews was to validate the smart object conceptual model as well as the SBPM organizational goal, requirements and reference models. Each interview consisted of 7 questions which the experts had to answer. In order to test the completeness of the models, we used open-ended questions which we asked before actually showing the models to the interviewees. We chose this strategy because we did not want to direct or influence the thinking process of the experts. In order to test the soundness of the models, we again used open-ended questions. However, this time we asked them after showing the models we have designed. The complete interviewing procedure and the results of the interviews are described in detail in section 10.1.

7.1 Smart object conceptual model

Although the smart object conceptual model, illustrated in Figure 3, is not part of our original research, we decided to validate it with experts because our study is heavily based on it. Based on the results from the interviews, we make a number of conclusions about the quality of the model.

First of all, we have to note that all interviewees had achieved a good understanding of the model before the interviews based on the document we had sent them. Moreover, two of them explicitly mentioned that the model is clear and understandable. All of this indicates that the model has a good level of understandability among experts.

In general, the interviewees had a very positive opinion about the model. Interviewee B explicitly mentioned that the separation of a smart object into a physical and digital entity is a good design decision. Moreover, according to all experts, the model is able to capture the variety of smart object they could think of. Interviewees A and D even gave examples of smart objects which fit into the model. All of this gives us evidence to conclude that the model is sound, complete and applicable to the industry.

Interviewee C noticed that the model does not show how smart objects can directly interact with each other. However, we do not consider this as a limitation of the model, because our study is focusing on designing a centralized system which orchestrate smart objects. In our context, we are not interested in the direct communication between smart objects. However, an answer to this question can be found in literature [17].

7.2 Organizational goal model

The organizational goal model illustrated in Figure 10 is important for our study because it provides motivation for the development of the SBPM system. For that reason, we decided to validate the quality of the model with experts.

According to the interviewees, the main drivers for organizations to improve their real-world visibility, control and responsiveness are related to improving business process performance in terms of quality, time and costs as well as improving customer experience. These results directly support the completeness of the upper part of our organizational goal model. However, we have to note that, when answering the question, interviewees were talking more about real-world visibility and responsiveness and less about real-world control. Based on this we can conclude that industry understands the need for real-world visibility and responsiveness, but still does not consider real-world control as an important capability.

When asked how the SBPM system can utilize smart object in order to improve the real-world visibility, control and responsiveness of organizations, interviewees mentioned our main ideas related to utilizing smart object data and involving smart object services in business processes. In addition, they also suggested some new ideas such as adapting business processes on the fly based on smart object data and applying data mining on historical data. However, we consider both of these ideas out of our scope. The first idea is not relevant to our study because it is not aligned with improving the real-world visibility, control or responsiveness of organizations. The second idea is irrelevant because our study does not deal with the evaluation step of the BPM lifecycle. In conclusion, we can say that the results also support the completeness of the lower part of the organizational goal model.

According to the experts, the organization goal model realistically captures the possible goals of organizations. Most of them emphasized on the importance of reducing costs, since this is one of the main goals of the clients that Capgemini serves. Therefore, we decided to include OG12 ("Reduce costs") in our generic goal model (see Figure 10). In conclusion, we can say that the organizational goal model is sound and it captures realistically the goals of organizations.

7.3 Requirements model

In order to validate the completeness of the requirements model, we asked the experts about their expectations related to the functionality of the SBPM system. Most of the answers that we received were related to big data on the fly, event stream processing, business process monitoring and involvement of smart object services in business processes. The only new idea that we received was related to supporting declarative processes modeling. However, since the goal of the SBPM system is to support highly predictable business processes, we find imperative process modeling to be more appropriate. More information about the different between imperative and declarative modelling can be found in literature [54]. Based on all said, we can conclude that the requirements model describes a complete set of use cases that the SBPM system should support.

7.4 External systems

According to interviewees, the external systems with which the SBPM system should interface include applications as well as Enterprise Resource Planning (ERP), Customer Relationship Management (CRM), geographical information, Business Intelligence (BI) and business monitoring systems. In our architecture, we consider the ERP and CRM systems as application systems because they relate to specific business functions. As far as the business monitoring is concerned, it is integrated as an internal module in our architecture. More precisely, it is part of the administration and monitoring module. We do not include a business intelligence system in our architecture because supporting the evaluation phase of the BPM lifecycle is out of our scope. However, if we had to include it, it would be as an internal module. Having said that, we consider the list of external systems we have identified to be complete.

7.5 Reference model

In order to validate the reference model of the SBPM system, we asked the experts about their general comments about it. Although, most of the comments we received were positive, there were a few remarks. One issue that interviewee B noticed is the lack of business rule support in the SBPM system. According to him, business rules are an important part of modern BPM system. The reason for this problem is that we based our architecture on the WF reference model which does not specify functionality related to business rules. Furthermore, according to interviewee C, the DBMS is a too concrete and technically-oriented component compared to the total level of abstraction of the reference model. However, we disagree with this remark because, in our opinion, a DBMS is a quite abstract module which does not suggest any particular technology. There are a number of different technologies which can be used to implement a DBMS (see section 5.2.1).

8 Conclusions, limitations and future work

In this section, we summarize our results, point out the limitations of our study and propose ideas for future research.

8.1 Conclusions

Generally speaking, in this study, we linked together two different fields. The first one is the field of BPM technology which already exists for a couple of decades and has already reached a relatively mature state. The second one is the field of smart objects which is developing rapidly in the last several years but is still in its infant state. The reason to link these two fields is that, as it turned out from our literature review, there is still no clear vision how BPM systems can utilize smart objects in order to benefit organizations.

The main goal of this study is to design an architectural blueprint of a Smart Business Process Management (SBPM) system which can utilize smart objects in order to improve the real-world visibility, control and responsiveness of organizations. On the one hand, the SBPM system we proposed improves the real-world visibility and control of organizations by enabling the involvement of smart objects as actors in business processes. For example, a step in a business process, which requires a measurement or action, can be performed automatically by a smart object service. This will reduce the cost and increase the accuracy and speed associated with the step. For that purpose, the system enables the definition and composition of smart object services based on the services provided by devices. On the other hand, the SBPM system we proposed improves the real-world visibility and responsiveness of organizations by enabling the processing and visualization of streaming information coming from devices. For that purpose, it enables the definition of Information Flow Processing (IFP) rules and monitoring objects. IFP rules can be used to detect events and aggregate information based on information flows while the monitoring objects can be used to present important information to decision makers.

In order to achieve our main research goal, we decomposed it into three sub-goals. The first one is to capture the organizational context of the SBPM system. We achieved that by developing a business model as proposed by the COMET methodology. The second one is to capture the functional requirements for the system. We achieved that by developing a requirements model as proposed by the COMET methodology. The third one is to design a reference model which will facilitate the development of future, concrete SBPM systems. We achieved this by developing a component structure model as proposed by COMET, but using our own notation.

In order to validate the quality of the models which we have developed in this study, we used two different approaches. The first one involved the description of the SBPM system in the context of a concrete industry scenario related to outage restoration. On the one hand, this evaluation gave us evidence for the completeness of our generic business and requirements models. On the other hand, it illustrated how the SBPM system can be applied successfully for the improvement of real-world visibility and responsiveness of organizations in a concrete scenario from the energy sector. As far as the second approach is concerned, it involved 4 interviews with experts from Capgemini Nederland. The focus of the interviews was on evaluating the quality of only the most important generic models in our study. Based on the positive results from the interviews, we concluded that, in general, the models we have developed have a good level of understandability, completeness and soundness.

By designing an architectural blueprint of the SBPM system, we make the first steps towards the design of concrete SBPM systems. The architectural blueprint consists of a business, requirements and reference model, each of which can help organizations in a different way. First, the generic business model can help organizations decide if they actually need a SBPM system. The focus of the model is on organizational context and the business motivation for developing a SBPM system. As it turned out, the two most important reasons for using a SBPM system are improving customer satisfaction and reducing costs. Second, the generic requirements model can help organizations decide what functionality they need for a concrete SBPM system. As it turned out, a SBPM system combines functionality from BPM, IFP and BAM systems. Finally, the goal of the reference model is to facilitate the design of future concrete SBPM systems. Therefore, in the reference model, our focus is on the functional view of the system which has a central role in any architectural initiative. It is relevant to all stakeholders and it greatly influences the shape of all other views. Nevertheless, we have to note that due to the high abstraction level of the architectural blueprint (and reference model in particular) as well as the limited evaluation we have performed, we cannot provide a complete guarantee that this blueprint can actually be used successfully by organizations to design concrete SBPM systems.

When it comes to the paradigms related to smart objects (see section 2.1.3), the SBPM system fits best into the idea of Intranet of Things. The reason for this is that it focuses on bringing benefits to individual organizations with the help of smart objects (things). However, the capability of the SBPM system to interface with other SBPM system might be used to create networks of SBPM system which enact business processes in tandem. In the future, this might also contribute to the development of the Internet of Things.

8.2 Limitations and future work

As we already described in section 1.4, in order to limit our scope, we omitted a number of COMET models while designing the SBPM system. A logical future research would be to fill these gaps and build a more complete architectural description of the SBPM system. In the context of the business model, the risk analysis, community model and work analysis refinement model have to be created. In the context of the requirements model, the non-functional requirements for the system should be identified. Just a few examples of non-functional requirements relevant to the SBPM system are security, fault tolerance and information reliability. As far as the reference model is concerned, it should not be limited only to the functional view but it should also cover other relevant views. More precisely, the component interaction, interface and information models proposed by COMET have to be developed. For more information about the COMET models we mentioned above, the reader can refer to the COMET methodology [1]. Moreover, besides designing new COMET models, a direction for future research is the refinements and further detailing of the models.

As a future research, the architectural blueprint of the SBPM system can be extended to support the evaluation phase from the BPM cycle. This will enable the utilization of historic data from smart objects in order to improve business process definitions [55].

Another important topic which is not discussed in detail in this study are the data interchange formats used by the different elements of the system. In our opinion, especially interesting are the formats used for communication between the process execution, service composition and information processing modules as well as the devices. Moreover, it should be investigated what standards currently exist and how the service bus can convert between different interchange formats in order to enable interoperability between modules.

9 References

[1] A. Berre, B. Elvesæter, J. Ø. Aagedal, and J. Oldevik, "Component and model-based development methodology 2.6," 2006.

[2] W. M. P. van der Aalst, "Business Process Management: A Comprehensive Survey," *ISRN Softw. Eng.*, vol. 2013, pp. 1–37, 2013.

[3] M. Weske, *Business process management: concepts, languages, architectures*. Springer Berlin Heidelberg, 2007.

[4] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[5] G. Kortuem, F. Kawsar, D. Fitton, and V. Sundramoorthy, "Smart objects as building blocks for the Internet of things," *IEEE Internet Comput.*, vol. 14, no. 1, pp. 44–51, Jan. 2010.

[6] T. S. López, D. C. Ranasinghe, B. Patkai, and D. McFarlane, "Taxonomy, technology and applications of smart objects," *Inf. Syst. Front.*, vol. 13, no. 2, pp. 281–300, Aug. 2009.

[7] A. Serbanati, C. M. Medaglia, and U. B. Ceipidor, "Building blocks of the internet of things: State of the art and beyond," in *Deploying RFID-Challenges, Solutions, and Open Issues*, 2011, pp. 351–366.

[8] "August Smart Lock." [Online]. Available: http://www.august.com/. [Accessed: 29-Oct-2013].

[9] S. Haller, S. Karnouskos, and C. Schroth, "The internet of things in an enterprise context," in *Future Internet–FIS 2008*, 2009, pp. 1–15.

[10] N. Oertel, J. Dibbern, and Z. Nochta, "Assessing the potential of ubiquitous computing for improving business process performance," *Inf. Syst. E-bus. Manag.*, vol. 8, no. 4, pp. 415–438, Nov. 2009.

[11] E. Fleisch, "Business perspectives on ubiquitous computing," *M-Lab Work. Pap.*, pp. 1–22, 2001.

[12] A. Hevner, S. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Q.*, vol. 28, no. 1, pp. 75–105, 2004.

[13] M. Iacob, H. Jonkers, and M. Lankhorst, "ArchiMate 2.0 Specification," 2012.

[14] S. Deliverable, S. Renner, and C. Heinemann, "European Smart Metering Landscape Report," Vienna, 2011.

[15] M. Beigl, H.-W. Gellersen, and A. Schmidt, "Mediacups: experience with design and use of computer-augmented everyday artefacts," *Comput. Networks*, vol. 35, no. 4, pp. 401–409, Mar. 2001.

[16] G. Kortuem, F. Kawsar, D. Fitton, and V. Sundramoorthy, "Smart objects as building blocks for the Internet of things," *IEEE Internet Comput.*, vol. 14, no. 1, pp. 44–51, Jan. 2010.

[17] N. Bui and J. De Loof, "Internet of Things – Architecture Converged architectural reference model for the IoT Overview of the IoT-A project partners," no. 257521, 2013.

[18] J. Wilson, *Sensor technology handbook*. Newnes, 2004.

[19] A. Furness, "Machine-readable data carriers–a brief introduction to automatic identification and data capture," *Assem. Autom.*, vol. 20, no. 1, 2000.

[20] H. Kato and K. T. Tan, "Pervasive 2D Barcodes for Camera Phone Applications," *IEEE Pervasive Comput.*, vol. 6, no. 4, pp. 76–85, Oct. 2007.

[21] C. M. Roberts, "Radio frequency identification (RFID)," *Comput. Secur.*, vol. 25, no. 1, pp. 18–26, Feb. 2006.

[22] M. Weiser, "Hot topics-ubiquitous computing," *Computer (Long. Beach. Calif).*, vol. 26, no. 10, pp. 71–72, 1993.

[23] M. Zorzi and A. Gluhak, "From today's intranet of things to a future internet of things: a wireless-and mobility-related view," *Wirel. Commun. IEEE*, vol. 17, no. 6, pp. 44–51, 2010.

[24] H. Sundmaeker, P. Guillemin, P. Friess, and S. Woelfflé, *Vision and challenges for realising the Internet of Things*. European Commision, 2010.

[25] T. Davenport and J. Short, "The New Industrial Engineering: Information Technology and Business Process Redesign," *Sloan Manage. Rev.*, vol. 31, no. 4, pp. 11–27, Oct. 1990.

[26] *Workflow Management Coalition Terminology & Glossary*. Workflow Management Coalition, 1999, pp. 1–65.

[27] W. Van der Aalst, A. ter Hofstede, and M. Weske, "Business process management: A survey," in in *Business Process Management*, Springer Berlin Heidelberg, 2003, pp. 1–12.

[28] G. Cugola and A. Margara, "Processing flows of information: From data stream to complex event processing," *ACM Comput. Surv.*, vol. 44, no. 3, pp. 1–62, Jun. 2012.

[29] D. W. McCoy, "Business Activity Monitoring : Calm Before the Storm." Gartner, pp. 1–3, 2002.

[30] R. Von Ammon, "Event-Driven Business Process Management and its Practical Application Taking the Example of DHL," in *Future Internet Symposium*, 2008.

[31] K. Sperner, S. Meyer, and C. Magerkurth, "Introducing Entity-Based Concepts to Business Process Modeling," pp. 166–171, 2011.

[32] S. Meyer, K. Sperner, J. Pasquier, and C. Magerkurth, "Towards modeling real-world aware business processes," in *Proceedings of the Second International Workshop on Web of Things - WoT '11*, 2011, no. June, pp. 1 – 6.

[33] M. Strassner and T. Schoch, "Today's impact of ubiquitous computing on business processes," *First Int. Conf. Pervasive Comput.*, 2002.

[34] P. Grefen and R. R. de Vries, "A reference architecture for workflow management systems," *Data Knowl. Eng.*, vol. 27, no. 1, pp. 31–57, Aug. 1998.

[35] M. H. Jansen-Vullers, M. W. N. C. Loosschilder, P. A. M. Kleingeld, and H. A. Reijers, "Performance Measures to evaluate the impact of Best Practices," *Proc. Work. Dr. Consort. 19th Int. Conf. Adv. Inf. Syst. Eng. (BPMDS Work.*, vol. 1, pp. 359–368, 2007.

[36] H. a. Reijers and W. M. P. van der Aalst, "The effectiveness of workflow management systems: Predictions and lessons learned," *Int. J. Inf. Manage.*, vol. 25, no. 5, pp. 458–472, Oct. 2005.

[37] S. Haller and C. Magerkurth, "The Real-time Enterprise : IoT-enabled Business Processes," in *IAB and IETF Interconnecting Smart Objects with the Internet Workshop 2011*, 2011, no. March, pp. 1–3.

[38] J. G. Kang and K. H. Han, "A Business Activity Monitoring System Supporting Real-Time Business Performance Management," *2008 Third Int. Conf. Converg. Hybrid Inf. Technol.*, pp. 473–478, Nov. 2008.

[39] N. Rozanski and E. Woods, *Software Systems Architecture*. Addison-Wesley, 2005.

[40] P. B. Kruchten, "The 4+1 View Model of architecture," *IEEE Softw.*, vol. 12, no. 6, pp. 42–50, 1995.

[41] D. Hollingsworth, "Workflow management coalition the workflow reference model," *Workflow Management Coalition*. Workflow Management Coalition, pp. 1–55, 1995.

[42] E. Bertino and L. Martino, "Object-oriented database management systems: concepts and issues," *Computer (Long. Beach. Calif).*, vol. 24, no. 4, pp. 33–47, Apr. 1991.

[43] J. Han, E. Haihong, G. Le, and J. Du, "Survey on NoSQL database," *Pervasive Comput. Appl.*, pp. 363–366, 2011.

[44] B. Hutchison, P. Lambros, and R. Phippen, "The Enterprise Service Bus : Making service-oriented architecture real," vol. 44, no. 4, pp. 781–797, 2005.

[45] S. Bandyopadhyay, M. Sengupta, S. Maiti, and S. Dutta, "A Survey of Middleware for Internet of Things," pp. 288–296, 2011.

[46] P. Spiess, S. Karnouskos, D. Guinard, D. Savio, O. Baecker, L. M. S. De Souza, and V. Trifa, "SOA-Based Integration of the Internet of Things in Enterprise Services," *2009 IEEE Int. Conf. Web Serv.*, pp. 968–975, Jul. 2009.

[47] J. Rao and X. Su, "A Survey of Automated Web Service," *Lect. Notes Comput. Sci.*, vol. 3387, pp. 43–54, 2005.

[48] S. S. S. R. Depuru, L. Wang, and V. Devabhaktuni, "Smart meters for power grid: Challenges, issues, advantages and status," *Renew. Sustain. Energy Rev.*, vol. 15, no. 6, pp. 2736–2742, Aug. 2011.

[49] H. Tram and S. Member, "Technical and Operation Considerations in Using Smart Metering for Outage Management," pp. 1–3, 2008.

[50] J. Ekanayake, N. Jenkins, and K. Liyanage, *Smart grid: technology and applications*. Wiley, 2012.

[51] A. B. Haney, T. Jamasb, and G. Pollitt, "Technology , Economics and International Experience Smart Metering and Electricity Demand : Technology , Economics and International," no. February, 2009.

[52] "Dutch Smart Meter Requirements 2.2." KEMA Consulting, pp. 1–137, 2008.

[53] A. Daneels and W. Salter, "What is SCADA," *Int. Conf. Accel. Large Exp. Phys. Control Syst.*, pp. 339–343, 1999.

[54] P. Pichler, B. Weber, S. Zugal, J. Pinggera, and H. A. Reijers, "Imperative versus Declarative Process Modeling Languages : An Empirical Investigation."

[55] R. K. L. Ko, S. S. G. Lee, and E. W. Lee, "Business process management (BPM) standards: a survey," *Bus. Process Manag. J.*, vol. 15, no. 5, pp. 744–791, 2009.

[56] R. Want, "An Introduction to RFID Technology," *IEEE Pervasive Comput.*, vol. 5, no. 1, pp. 25–33, Jan. 2006.

[57] W. Van der Aalst, M. Weske, and D. Grünbauer, "Case handling: a new paradigm for business process support," *Data Knowl. Eng.*, vol. 53, no. 2, pp. 129–162, 2005.

[58] "Business Process Model and Notation (BPMN)." Object Management Group, 2011.

[59] J. Rumbaugh, I. Jacobson, and G. Booch, *The unified modeling language reference manual*. Addison-Wesley, 1999.

[60] W. M. P. van der Aalst and A. H. M. ter Hofstede, "YAWL: yet another workflow language," *Inf. Syst.*, vol. 30, no. 4, pp. 245–275, Jun. 2005.

[61] I. Sommerville, *Software Engineering*, 9th ed. Addison-Wesley, 2010.

[62] P. Kruchten, H. Obbink, and J. Stafford, "The past, present, and future for software architecture," *Software, IEEE*, vol. 23, no. 2, pp. 22–30, 2006.

[63] J. N.Buxton and B. Randell, "Software engineering techniques," Rome, 1969.

[64] M. Shaw and D. Garlan, *Software architecture: perspectives on an emerging discipline*. Prentice Hall, 1996, p. 242.

[65] S. Alter, "Defining information systems as work systems: implications for the IS field," *Eur. J. Inf. Syst.*, vol. 17, no. 5, pp. 448–469, Oct. 2008.

[66] P. Grefen, *Business Information System Architecture - 1BM41 Lecture Reader*. Eindhoven: Eindhoven University of Technology, 2012.

[67] S. Alter, *Information systems: A management perspective*. Addison-Wesley, 2000.

[68] M. Lankhorst, *Enterprise architecture at work: Modelling, communication and analysis*. Springer-Verlag Berlin Heidelberg, 2013.

[69] S. Angelov, J. Trienekens, and P. Grefen, "Towards a method for the evaluation of reference architectures: Experiences from a case," in in *Software Architecture*, Springer Berlin Heidelberg, 2008, pp. 225–240.

[70] S. Angelov and P. Grefen, "An e-contracting reference architecture," *J. Syst. Softw.*, vol. 81, no. 11, pp. 1816–1844, Nov. 2008.

[71] O. Etzion and P. Niblett, *Event processing in action*. Manning Publications Co., 2010.

[72] M. Goetz, "Integration of Business Process Management and Complex Event Processing." Nuremberg, pp. 1–14, 2011.

10 Appendices

10.1 Interviews

10.1.1 Interviewing procedure

In this section, we describe step by step the procedure which we have followed during the interviews with experts from Capgemini Nederland B.V.

Step 1. One week before the interview, we send a description of the smart object conceptual model to the interviewee. The goal is to let the interviewee familiarize himself with the model beforehand, since, otherwise, it would take a lot of time to do this during the interview.

Step 2. In the beginning of the interview, we present the main goal of our study to the interviewee. The goal is to provide the interview with some basic context for the interview.

Step 3. We ask the interviewee: *Do you think that the smart object conceptual model is able to capture the variety of smart objects?* The goal of the questions is to verify the quality of the conceptual model.

Step 4. We ask the interviewee: “*What are the main business drivers for organizations to improve their real-world visibility, control and responsiveness?*” The goal of this question is to check the completeness of the organizational goal model.

Step 5. We ask the interviewee: *How can smart objects be utilized by BPM systems in order to improve the real-world visibility, control and responsiveness of organizations?* Again, the goal of the question is to check the completeness of the organizational goal model.

Step 6. We introduce our own organizational goal model to the interviewee.

Step 7. We ask the interviewee: *Do you think that our organizational goal model realistically captures the possible goals of organizations?*

Step 8. We ask the interviewee: *What is the most important functionality that you expect from the SBPM system?* The goal of the question is to verify the completeness of the requirements model.

Step 9. We ask the interviewee: *With what external system do you expect the SBPM system to interface?*

Step 10. We introduce the reference model to the interviewee.

Step 11. We ask the interviewee: *What are your general comments on the reference model?*

10.1.2 Interview A

The interview was conducted with a system analyst from Capgemini Nederland B.V. on 28.10.2013.

- Do you think that the smart object conceptual model is able to capture the variety of smart objects?
  - Yes
  - One example of smart object which fits well in this model is a GPS device.
- What are the main business drivers for organizations to improve their real-world visibility, control and responsiveness?
  - Organizations need responsiveness in order to avoid losing business value. For example, if an organization do not respond in time, it might miss a business opportunity
  - Visibility is useful in improving total quality management and optimizing business processes
- How can smart objects be utilized by BPM systems in order to improve the real-world visibility, control and responsiveness of organizations?
  - Data from smart objects can be analyzed on the fly. For example, in order to improve traffic control, sensors which detect cars can be integrated in roads.
  - Data from smart objects can provide BPM systems with a feedback loop which can be used to optimize the execution of business processes on the fly.
  - Data from smart objects collected for large periods of time can be analyzed using data mining
  - One example is from the field of sport. Sensors can be located at certain points in a marathon track in order to monitor the flow of marathon runners. Then, data from this sensors can be used to detect delays which might signal for a problems on the track. This is more cost-efficient than using people for this purpose.
- Do you think that our organizational goal model realistically captures the possible goals of organizations?
  - Not able to say
- What is the most important functionality that you expect from the SBPM system?
  - Support the modeling and execution of declarative processes
  - Event stream processing
  - Big data on the fly
- With what external system do you expect the SBPM system to interface?
  - Applications
- What are your general comments on the reference model?
  - The notion of rules can be confused with business rules
  - Seems that all types of external systems are covered
  - The service composition and rule processing modules are the most interesting

10.1.3 Interview B

The interview was conducted with an enterprise architect from Capgemini Nederland B.V. on 06.11.2013.

- Do you think that the smart object conceptual model is able to capture the variety of smart objects?
  - Yes
  - The model is understandable
  - The idea to separate digital and physical part of a smart object is good
  - The purpose of the model is not self-evident (it is not clear if the model just describe a smart object or how a smart object interacts with the rest of the world)
- What are the main business drivers for organizations to improve their real-world visibility, control and responsiveness?
  - Improve operational business processes
  - Improve resource utilization by having more information and control
  - Bring more value to customers
  - Reduce costs
  - Improve profit
- How can smart objects be utilized by BPM systems in order to improve the real-world visibility, control and responsiveness of organizations?
  - Define and monitor KPIs
  - Adapt processes based on data from smart objects
  - Optimize processes
  - Changing the way of work can be faster with smart objects rather than with people
- Do you think that our organizational goal model realistically captures the possible goals of organizations?
  - Yes
  - Look at the benefit analysis methodology of Capgemini
  - Lowering costs and increasing value leads to more profit
- What is the most important functionality that you expect from the SBPM system?
  - Provide accurate insight in real world behavior of business processes
  - Provide means to reach to events in the real-world
- With what external system do you expect the SBPM system to interface?
  - Applications
  - ERP systems
- What are your general comments on the reference model?
  - Business rules are missing
  - It should be possible to evaluate business rules based on data from smart objects
  - Standardization is important in order to integrate various devices

10.1.4 Interview C

The interview was conducted with an enterprise architect from Capgemini Nederland B.V. on 06.11.2013.

- Do you think that the smart object conceptual model is able to capture the variety of smart objects?
    - The idea of the model is clear
    - It is questionable if a tag is device. In case of an RFID tag, it is. In case of a barcode, it is not very clear.
    - Most devices fit in the model since it is quite generic
    - The model does not show how smart objects can interact with each other
- What are the main business drivers for organizations to improve their real-world visibility, control and responsiveness?
    - Provide personalized customer experience/ personalized interaction with the customer
    - Cost reduction
    - Decouple the physical presence of organization from the physical customer presence in terms of time and space
- How can smart objects be utilized by BPM systems in order to improve the real-world visibility, control and responsiveness of organizations?
    - Use smart objects as input gatherers
    - Use smart objects as event sources
- Do you think that our organizational goal model realistically captures the possible goals of organizations?
    - Improve profit can be added
    - Providing individual user experience is missing
- What is the most important functionality that you expect from the SBPM system?
    - It should be possible to model and execute actuation
    - It should be possible to model and detect events
    - It should be possible to create a feedback loop. In other words, changes made in real-world should be detected by the system.
- With what external system do you expect the SBPM system to interface?
    - Business intelligence systems
    - Business monitoring systems
    - Existing information systems
- What are your general comments on the reference model?
    - DBMS is too technically oriented compared to other modules
    - The model makes sense

  - Traditional BPM systems are still lacking standardization for integrating applications. This is inherited by the SBPM system.

10.1.5 Interview D

The interview was conducted with a data architect from Capgemini Nederland B.V. on 14.11.2013.

- Do you think that the smart object conceptual model is able to capture the variety of smart objects?
  - The model is quite generic and therefore it can probably capture the variety of smart objects
  - The interviewee gave some example of smart objects which fit into the model
- What are the main business drivers for organizations to improve their real-world visibility, control and responsiveness?
  - Improve efficiency in terms of costs
  - Reduce manual work
  - Real-world visibility, control and responsiveness are interdependent
  - Reduce efforts related to maintenance of machines
  - Responsiveness can help to reduce loses
- How can smart objects be utilized by BPM systems in order to improve the real-world visibility, control and responsiveness of organizations?
  - Real-time data from smart objects can be used in BPM systems
  - Business processes can be started as a reaction to data
  - Use data from smart object to monitor real-world
- Do you think that our organizational goal model realistically captures the possible goals of organizations?
  - The model seems ok
  - Costs are really important because clients always look for cost reduction
- What is the most important functionality that you expect from the SBPM system?
  - Interaction with smart objects through services
  - Use services provided by smart objects to automate business processes
  - Filter huge amount of information coming from smart objects
  - Define how data from smart objects is used
  - For example, there are printers which automatically notify responsible people when they are out of ink
- With what external system do you expect the SBPM system to interface?
  - ERP systems
  - CRM systems
  - Geographical Information Systems (GIS), for example, in the context of smart energy
  - Many other types of systems
- What are your general comments on the reference model?

- The models is very understandable because it is not too complex
- The model seem to cover all aspects
- Interfacing with other SBPM systems is very important because it enables cooperation between organizations on business process level

10.2 Extended literature review

10.2.1 Types of tagging technology

10.2.1.1 Optical technology

In the case of optical technology, usually the tags are labels such as barcodes or QR-code and readers are some kind of image capturing and processing devices (e.g. a smart phone [20]). One limitation of this type of technology is that readers require a direct line of sight in order to read the labels. Usually this requires the label to cover part of the surface of the object it is describing. Another limitation is that once they are printed, the data on the labels cannot be modified. Nevertheless, their simplicity and low cost compensates for their drawbacks and they are widely used in many industries such as retail, logistics, manufacturing and advertising.

10.2.1.2 Radio frequency identification technology

Radio-frequency identification (RFID) technology exists already for more than 60 years. Its core idea is the usage of radio-frequency electromagnetic fields for transferring of data. There are two main classes of RFID tags: *passive* and *active* [21], [56]. Active tags need a source of energy such as a battery or connection to the power grid in order to work. Their advantages are that they have more memory and it can be accessed from a larger distance, compared to passive ones. Their drawbacks are that they are larger in size, more expensive and their lifetime depends on the battery [21]. Unlike active RFID tags, passive tags are usually read-only and they do not require to be connected to a power source. They are able to capture energy from the transmissions of a reader and use it for sending back a response. The absence of battery gives them longer lifetime and reduces their need for maintenance. Compared to optical tags, RFID tags are more expensive but they have a number of advantages. The biggest one is that they do not require a direct line of sight in order to be read.

10.2.2 Paradigms related to smart objects

There are a number of paradigms related to smart objects. Some of the most popular ones are Ubiquitous Computing, Internet of Things, Intranet of Things and Industrial Internet.

10.2.2.1 Ubiquitous computing

Ubiquitous computing is a computing paradigm that was first coined and described by Mark Weiser at Xerox Corporation [22]. The definition he provided is the following:

"Ubiquitous computing has as its goal the nonintrusive availability of computers throughout the physical environment, virtually, if not effectively, invisible to the user." [22]

This definition focuses on a few main things. Phrases such as "*nonintrusive*" and "*invisible to the user*" clearly suggest that in the ubiquitous computing paradigm, users should not be aware that they are actually using computer technology. Second, the word "*availability*" suggests that the computer technology has to be anywhere and anytime accessible and usable by users. Finally, "*throughout the physical environment*" emphasizes that computers have to be a part of the physical world. They have to be seamlessly integrated into it. Clearly, smart objects will play a central role in achieving this vision.

10.2.2.2 Internet-of-things

The concept of *Internet-of-Things* (IoT) was first coined by Kevin Ashton in a presentation at Procter & Gamble in 1999. Back then, it was used to link the RFID technology applied in supply chains to the popular by that time topic of the Internet. However, IoT really started to gain popularity after 2004 [7], [24]. Nowadays, it is one of the hottest topics and a buzzword, often used to describe one very promising future for the Internet and the world. As a result of the huge interest in the topic, a large number of definitions of IoT can be found in the literature. They are coming both from academia and industry. Some of them are more generic and some of them are more technically oriented.

The phrase "*Internet-of-things*" is a composition of two terms. Each term gives a perspective from which the meaning of the whole phrase can be viewed. On the one hand, the term "*internet*" gives to some degree a technical nuance of the concept. It suggests that the IoT is related with the development of a new global communication infrastructure or an extension of the current Internet. On the other hand, the term "*things*" suggests a generalization of the participants in this Internet of the future. Not only ICT devices but also any physical entity (including people) will be connected and represented in this new global infrastructure [4], [7]. There will be no limit who and what can participate in the IoT and everything will be all called a "*thing*". Note that the concept of thing resembles the concept of a smart object, defined in 2.1. Furthermore, an additional semantic perspective can be added to the picture. Since the amount of information generated by "*things*" in IoT will be tremendous, one of the challenges is how to represent, store, interconnect, search, and organize it. A key role in meeting these challenges could be played by semantic technologies [4]. As a paradigm, Internet-of-things is very similar to ubiquitous computing. It also has the goal to integrate the digital and the physical worlds. The resulting world will be invisibly augmented with the capabilities provided by computer technology.

In our opinion, one important aspect of IoT which has to be analyzed is the hype around it. According to Gartner's hype cycle for emerging technologies for 2012 (see Figure 18), IoT is just entering the phase called "*Peak of Inflated Expectations*". This means that currently it experiences high media attention and publicity. Unfortunately, often this is also accompanied by unrealistic expectations for its future applications and impact. Moreover, in their analysis, Gartner predicts that as a technology, IoT needs more than 10 years to reach maturity and wide adoption.

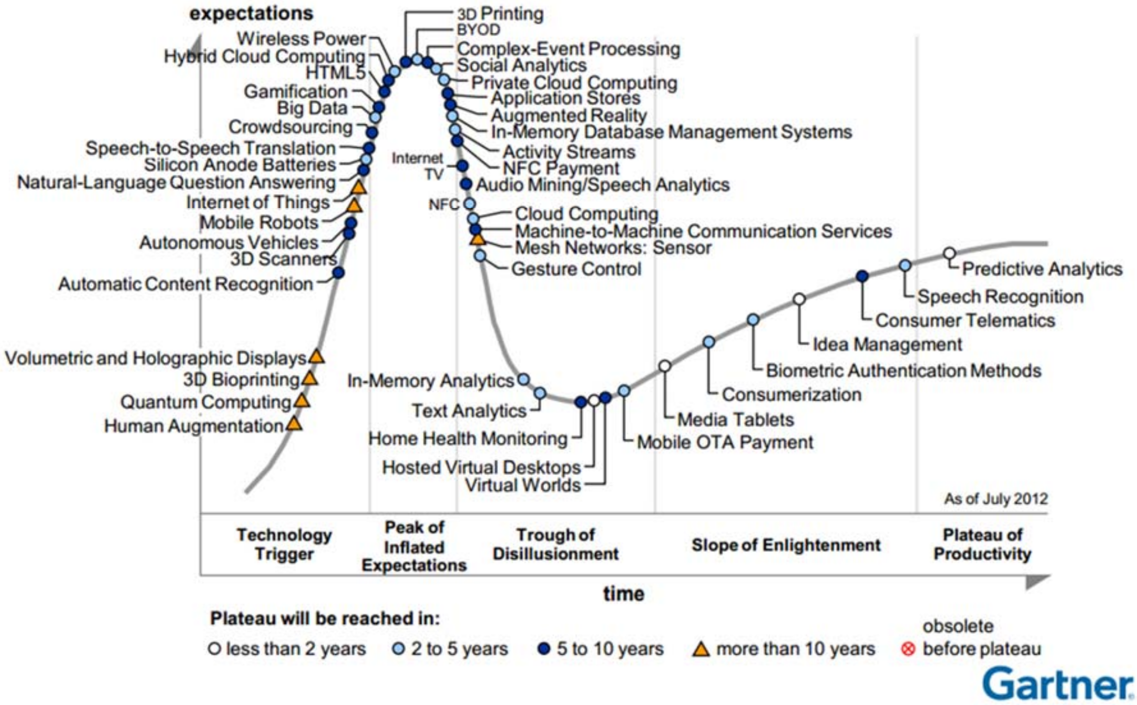

Figure 18: Gartner's 2012 Hype Cycle for Emerging Technologies

10.2.2.3 Intranet-of-things

The concept of Intranet-of-things is similar to the concept of Internet-of-things [23]. The main difference between the two is their scope. While the Internet-of-things has a global scope and aims to connect "*things*" owned by many different parties, the Intranet-of-things has a local scope and aims to connect "*things*" within one or a few organization. Unlike the Internet-of-things, the Intranet-of-things focuses on the current situation. Nowadays, systems interconnecting things are usually implemented for a specific application, in a specific industry, working with a limited number of technologies and proprietary standards. Although they can be highly optimized in achieving their specific goals, they do not have a global scope and remain closed for the outside world. The information they store and the services they provided usually never get exposed so that other parties can also benefit from them. This leads to the creation of vertical "*silos*" solutions which are called Intranets-of-things. Despite the fact that Intranets-of-things might fulfill well their specific tasks, the exponential number of possibilities and benefits which they could bring if they were connected into one integrated Internet-of-things are never achieved. Nevertheless, it is possible that the current disconnected Intranets-of-things can be the foundation for the future Internet-of-things. With the help of a common vision and standardization, in the future, these intranets could start joining together and create a global Internet-of-things.

10.2.3 Classification of business processes

10.2.3.1 Level of abstraction

Business processes differ in their level of abstraction. Generally, we can talk about tree main types of business processes (see Figure 19): *organizational*, *operational* and *implemented* [3]. Organizational business processes capture high-level business functionality and, respectively, they have the highest level of abstraction. Usually, they are specified in an informal (e.g. textual) or semi-formal (e.g. diagrammatic) way. For instance, an organizational process can describe the general way in which a project is managed. Typically, one organizational business process is realized by a number of operational ones. Operational processes have lower level of abstraction. Usually, they are specified by business process models which show concrete activities and relationships between them. For example, an operational business process can describe the concrete steps to initiate a project. Operational business processes can be implemented so that they can be enacted by a WFM/BPM system. This is done in the configuration phase of the BPM lifecycle as described in 10.2.10.2. Implemented business processes are the least abstract of the tree types. Their models are extended with concrete technical and organizational information which enables their enactment by a WFM/BPM system.

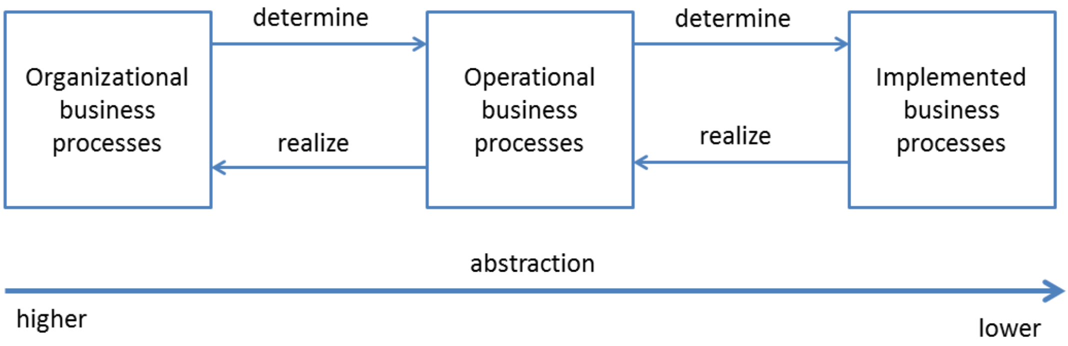

Figure 19: Types of business processes [3]

10.2.3.2 Level of structure

Business processes differ in their level of structure. Generally, we can classify them in two main categories: *structured* and *unstructured* [3]. Structured are business processes which can be specified completely during their design. This means that their activities and flow control have to be known beforehand. Structured processes are predictable and this makes them easier to enact with WFM/BPM systems. They are typically related to more trivial types of work. Unstructured are business processes which cannot be specified completely during their design. Usually, such processes are knowledge intensive and their enactment depends to a large degree on the knowledge of the process participants. Since unstructured business processes have to deal with uncertainty, their enactment needs a lot of flexibility. This makes them difficult to support with regular WFM/BPM systems. A widely used approach for supporting unstructured business processes is *Case Handling* [57].

10.2.3.3 Level of automation

Business processes differ in their level of automation [3]. Not only their enactment can be automated using a WFM/BPM system, but also their individual steps can be automated using technology. The main purpose of automation is to cut costs and flow time of business processes by reducing the involvement of human resources. In the extremes, business processes can be completely automated or completely manual. However, in most cases, they are somewhere in the between. The trend related to complete process automation is called *Straight Through Processing* (STP) [27]. STP is usually possible for highly structured processes where each activity can be automated. The gains from STP are the biggest when the process is highly repeatable.

10.2.4 Business process performance measurement

The British physicist and engineer, Load Kelvin, has said: "*If you can't measure it, you can't improve it.*" This statement holds true also when it comes to business processes. Once a business process is designed or redesigned, it is important that its performance can be measured. Otherwise, it is difficult to compare objectively between alternative business process designs and, therefore, it is difficult to improve them in a systematic way. A framework for measuring business process performance, with respect to time, cost, quality and flexibility, has been proposed [35]. The core of the framework is captured by the so-called *devil's quadrangle* illustrated on Figure 20. The devil's quadrangle suggests that there are always trade-offs between the different dimensions of the framework. For example, improving one of the dimensions would probably worsen some of the others.

Time dimension. There are two main performance measures related to the time dimension: *lead* and *throughput* time [35]. Lead time is the time needed to enact a complete business processes instance. Throughput time refers to the time needed to complete a step in the processes. Throughput time can be further decomposed into service, queue, wait, move, and setup times.

Quality dimension. We can distinguish between two types of quality: *external* and *internal* [35]. External is the quality perceived from the viewpoint of the customer. The customer is an individual or organization that will benefit from the output of the business processes. The external quality can be measured as the satisfaction of the customer with the process and its output. The satisfaction of the customer with the process depends on the match between his expectations and the actual enactment of the process. The satisfaction of the customer with the output of the process depends on the degree to which he believes it corresponds to specifications. Internal is the quality perceived from the viewpoint of the business process participants (employees). When this quality is high, it results in high motivation, high job satisfaction and low absenteeism of employees. In turn, this increases the external quality of the business process. Internal quality can be measured in terms of work skill variety, task identity, task significance, autonomy, feedback, co-worker relations, etc.

Cost dimension. There are at least four performance measures related to the cost dimension: *running*, *inventory*, *transport* and *administrative* costs [35]. Running costs are associated with the enactment of the business process. They can be further decomposed into labor, machinery and training costs. Inventory costs are related to storing products and data. Transport costs are related to moving products and disseminating data. Administrative cost are related to all activities needed to keep the business processes in intact. Furthermore, the cost dimension has relations with the other dimensions. For instance, a delay in process lead time will result in additional costs, poor quality of process output can result in expensive rework and low process flexibility might result in cost-inefficient enactment.

Flexibility dimension. Flexibility is "*the ability to react to changes*". It can be measured with respect to individual resources, individual tasks or the business process as a whole. There are five main performance measures related to flexibility: *mix*, *labor*, *routing*, *volume* and *process modification* flexibility [35]. Mix flexibility is the ability to handle different types of cases (per resource, task or process). Labor flexibility is the ability to perform different types of tasks (per resource or process). Routing flexibility is the ability to handle a case using different routes within a business process. Volume flexibility is the ability to handle varying volumes of input. Process modification flexibility is the ability to change the structure of a business process.

The performance of a business process is usually evaluated during the design and evaluation phases of the BPM lifecycle. At the design phase, an estimation of the process performance can be made using simulation techniques. At the evaluation phase, the data gathered during the enactment of the process can be used to estimate its performance.

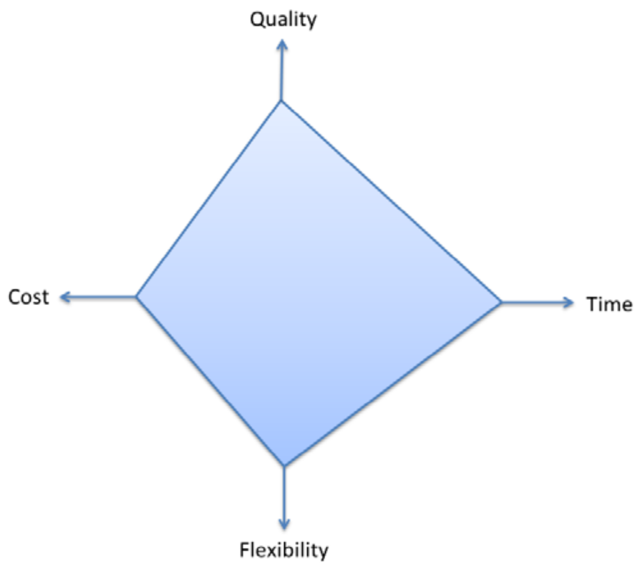

Figure 20: The devil's quadrangle [35]

10.2.5 Workflow reference model

The *Workflow Reference Model* (WFRM) was developed by the Workflow Management Coalition and it was first published in 1995 [41]. It was created in response to the lack of standardization in the design of WFM systems at that time. The WFRM identifies common terminology, characteristics and components of WFM systems developed by different vendors and serves as a template for the design of new ones. Since it is based on accumulated experience in the domain, we can classify it as a practice-driven reference model. Although the WFRM was developed a long time ago and it focuses on WFM systems, it is still widely used today. The reason for this is that modern BPM systems extend the functionality of WFM systems, and therefore their design is still, to a large degree, based on the WFRM. The model is shown in Figure 21. It illustrates the major components and interfaces which are typical for any WFM system. Next, we briefly explain the idea of each component and how it connects to the others.

A **workflow engine** is a software service which manages the enactment of business processes. In other words, it manages the execution of business process instances in accordance with their business process definitions. This includes tasks such as navigation between process activities, assigning of tasks to participants, invoking external applications and many others.

The **workflow enactment service** is a software service which consists of one or more workflow engines. It supports the enactment phase of the BPM lifecycle. More specifically, it provides a run-time environment in which business process instances can be managed. The workflow enactment service interacts with external components via the workflow application programming interface (WAPI). The WAPI is a common core of API calls and interchange formats which is extended by the five interfaces. Different workflow enactment service can connect, communicate and work together via Interface 4. For example, a workflow enactment service can invoke an activity or sub-process in another one.

The **process definition tools** support the design and configuration phases of the BPM lifecycle. Their output is a process definition which can be interpreted by a workflow engine. Process definitions are exchanged between the process definition tools and the workflow enactment service via Interface 1.

The **workflow client applications** are related to the enactment phase of the BPM lifecycle. They enable workflow engines and human resources to interact with each other. More precisely, with the help of workflow client applications, workflow engines can assign tasks to people and then track their execution progress. The exchange of data between the workflow enactment service and workflow client applications is done via Interface 2.

The **invoked applications** are related to the enactment phase of the BPM lifecycle. They are software applications which people need to use in order to complete specific business process activities. Workflow engines can directly invoke and communicate with them via Interface 3.

The **administration and monitoring** tools are related to the enactment phase of the BPM lifecycle. Typical administration tasks supported by them are related to the management of business process instances, users, roles and resources. Typical monitoring tasks are related to tracking the status and progress of process instances. The administration and monitoring tools exchange data with the workflow enactment service via Interface 5.

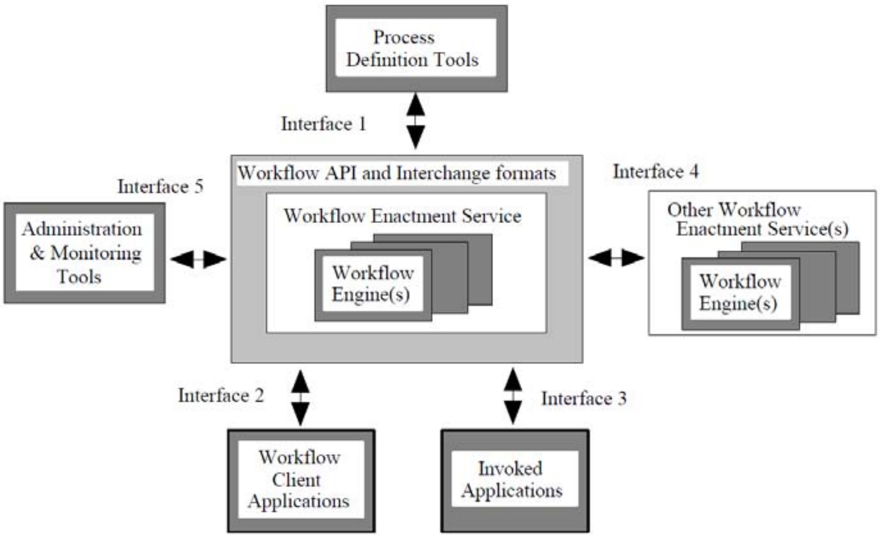

Figure 21: Workflow reference model [41]

10.2.6 BPM lifecycle

10.2.6.1 Design phase

The design phase is where, usually, the BPM lifecycle starts from [3]. People specialized in carrying out this phase are called process designers. In order to design business processes, process designers have to examine existing business processes as well as their organizational and technical environment. This includes their identification and modeling. When business processes are modeled, they are expressed in a graphical notation such as BPMN [58], UML [59] or YAWL [60]. In this way they are made explicit which facilitates their future improvement. Once the initial design of a business process is ready, it has to be analyzed and, based on the results, eventually, improved. Usually, the analysis includes validation and verification of the process model. The idea of the validation is to assure that the business process model is actually reflecting the real-world business process as closely as possible. Collaboration with the actual business process participants and simulation techniques are often used in validation. The idea of verification is to assure that the process model does not contain any undesired properties (e.g. deadlocks). Again, simulations are commonly used to do this.

10.2.6.2 Configuration phase

Once a business process has been designed, it has to be implemented [3]. Usually, this is done with the help of a WFM or BPM system. The business process model has to be extended with concrete technical and organizational information which will enable its enactment by a WFM/BPM system. The BPM/WFM system must also be configured to realize the business processes. This configured should take into account the organizational and technical environment of the business processes. At a minimum, it should specify how business process participants interact with the system and how it integrates with other existing software systems. In some cases, it can also include development work related to the integration of legacy systems as well as to the implementation of new ones. Once a business process is implemented, its implementation should be tested. It is important to perform tests on activity and process level in order to assure the proper functioning of the process. At the end, the process is deployed in its target environment.

10.2.6.3 Enactment phase

During the enactment phase, instances of an implemented business process are actually being executed. A business process instance is a "representation of a single enactment of a business process" [26]. It is "created, managed and terminated" by the WFM/BPM system, in accordance with its process definition [26]. Each instance works with separate data and is managed independently. Usually, the start of a process instance is triggered by an event from the external environment (e.g. receiving an insurance claim) [3]. An important part of the enactment phase is monitoring. It is a mechanism for providing information on the status of different process instances. This information is especially valuable for process participants and responsibles because it helps them to track the progress of process instances. Furthermore, it is common that BPM systems gather and store detailed data about the enactment of business processes. This data is used in the evaluation phase for business processes improvement.

10.2.6.4 Evaluation phase

In literature, the evaluation phase is, sometimes, also called diagnosis phase [27]. In this phase, the data gathered by a WFM/BPM system during the enactment phase is used to evaluate and eventually improve business process models and implementations. Two main techniques are used to do that: business activity monitoring (BAM) and process mining. BAM refers to the aggregation, analysis, and presentation of near real-time information related to business process instances. Often, this information is presented visually, for instance, as a set of charts on a dashboard, each corresponding to a key performance indicator. With the help of BAM, processes responsibles can detect problems (e.g. delays and bottlenecks) in near real-time fashion during the business processes enactment. The idea of process mining is to extract knowledge from the data gathered during processes enactment. One popular application is to check the level of conformity between business process models and their actual execution in reality. If the gap between the two is too large, this is a problem which might require the redesign of the process.

10.2.7 Types of architecture

Architecture, as an art and science for describe buildings and other physical structures, exists already for many centuries. However, in the field of information technology, the concept of architecture was introduced quite recently. In this section, we will first introduce some of the main types of IT architectures which exists today. Our purpose is to clarify their meanings and show how they differ from each other. After that, we discuss a famous architectural model in the field of BPM which we consider fundamental to our study.

10.2.7.1 Software architecture

Before defining a software architecture, first we have to clarify the meaning of a software system (or just software). According to a well-known book in the field of software engineering, software systems are "*abstract and intangible*" [61]. That means that many physical barriers which exist for tangible systems, do not apply to software systems. For example, once it is developed, it can be easily duplicated many times without significant additional costs. The same book defines software system as a system consisting of a number of related computer programs, configuration files and documentation.

The term "*software architecture*" was coined for the first time in 1969 at a NATO conference on software engineering [62]. As it becomes clear from the report from the conference [63], the need for architecture came from the increasing complexity of software by that time. However, it took software architecture more than 10 additional years before it starts to emerge as a distinct discipline. Nowadays, its importance in designing complex software systems is well recognized by large companies and academia. A large number of definitions of software architecture can be found in literature. Nevertheless, they all agree to a large degree on its meaning. Therefore, in this study, we present only one of the most popular definitions:

"The architecture of a software system defines that system in terms of computational components and interactions between those components." [64]

10.2.7.2 Information systems architecture

We start with defining what an information system (IS) is and how it is different from a software system. Many definitions of information system can be found in literature but most of them do not cover all possible types of such systems [65]. One of the most complete definitions is:

"An information system is a particular type of work system that uses information technology to capture, transmit, store, retrieve, manipulate, or display information, thereby supporting one or more other work systems." [65]

Here, we have to note that the definition of information system is based on the concept of work system which is defined in the following way:

"A work system is a system in which human participants and/or machines perform work (processes and activities) using information, technology, and other resources to produce specific products and/or services for specific internal or external customers." [65]

According to the definition presented above, the concept of information system is broader than the concept of software system. As depicted in Figure 22, it involves participants, information, technology, business processes, products/services and customers. In fact, an information system can include one or more software systems (e.g. as components). Clearly, the scope and complexity of information systems can be large and their architecture has to take this into account. IS architecture can be defined in the following way:

"The architecture of an information system defines that system in terms of functional components and interactions between those components, from the viewpoint of specific aspects of that system, possibly organized into multiple levels, and based on specific structuring principles." [66]

This definitions proposes three main ways for coping with complexity: describing the information system from different viewpoints, organizing the descriptions in multiple levels and making use of structuring principles (e.g. architectural styles and patterns) [66].

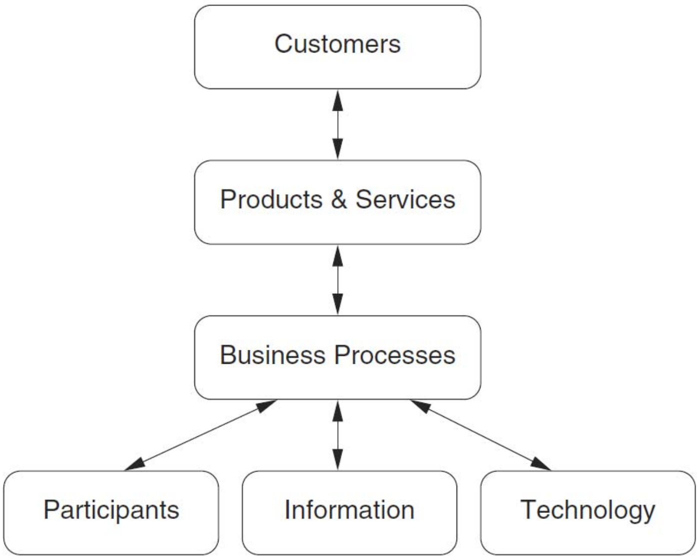

Figure 22: An integrated view of an information system [67]

10.2.7.3 Enterprise architecture

In a recent but already widely accepted book on the topic of enterprise architectures, the concept of enterprise is defined as *"any collection of organizations that has a common set of goals and/or a single bottom line"* [68]. Note that this concept has much broader nature than the concept of information system. One enterprise can have multiple supporting information systems. The same book defines enterprise architecture in the following way:

"Enterprise architecture is a coherent whole of principles, methods, and models that are used in the design and realization of an enterprise's organizational structure, business processes, information systems, and infrastructure." [68]

10.2.7.4 Reference and concrete architecture

Generally speaking, the purpose of a reference architecture is to provide a high-level architectural template in a particular domain which can be elaborated to obtain concrete architectures for specific situations. This template can be defined for a software, information system or enterprise architecture. In this study, we focus only on information system reference architectures and we leave the other two types out since we consider them as irrelevant to our research. In the case of information systems, reference architecture can be defined as "*a general design (abstract blueprint) of a structure for a specific class of information systems*" [66]. Figure 23 shows the high level process of designing a reference architecture and then obtaining a concrete architecture from it. A reference architecture is created by applying domain knowledge, architectural styles and architectural patterns. Next, the reference architecture can be elaborated into a concrete architecture using context details and again architectural styles and patterns.

Two main types of IS reference architectures can be identified: *practice-driven* and *futuristic* reference architectures [69]. Practice-driven reference architectures (PRAs) are usually based on accumulated domain experience. For example, sources of such experience could be existing concrete architectures. Due to their practical nature, usually, the design of such reference architectures follow a bottom-up approach. Examples of PRAs in the workflow management domain are the architectures proposed by Grefen and Vries [34] as well as Hollingsworth and Hampshire [41]. Futuristic reference architectures (FRAs) are usually inspired by research and they are defined before the existence of practical experiences from concrete architectures. Their goal is, based on current research, to provide a vision for the future design of concrete architecture in a specific domain. Due to their research-driven nature, usually they are designed in a top-down fashion. An example of a FRA is the architecture proposed by Angelov and Grefen [70].

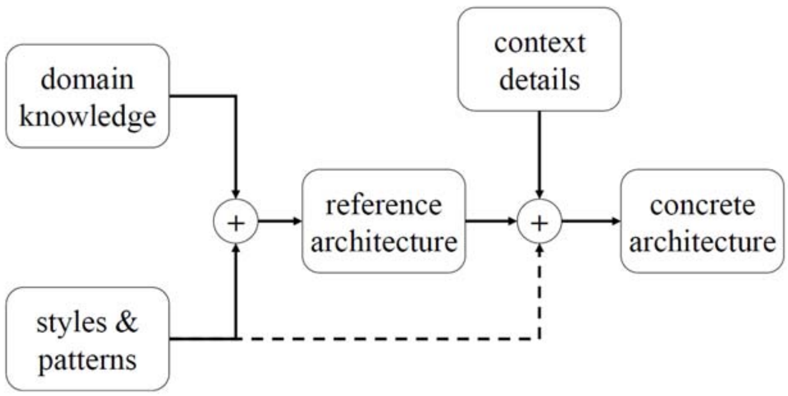

Figure 23: Reference architecture process [66]

10.2.8 Events

An event can be defined as a notable thing that happens in the physical or digital environment of an organization [71]. It can signify a change, problem or opportunity. Moreover, events can have different levels of abstraction. For example, in the context of an AMI (see 10.3.2), the power outage of a house can be considered a low-level event. A smart meter can easily detect it and notify the electricity company. However, the failure of a power line is a more abstract event and therefore more difficult to recognize. It cannot be detected based on a single check/measurement of a sensor. It can only be detected as a result of a number of lower-level events. For example, a number of power outages which happened at the same time, in a certain area.

One problem with smart objects is that they generate a lot of real-time events and data which can lead to information overload. Therefore, organizations need an automated way to filter important from unimportant events and data. CEP and BAM technologies can be a solution to this problem. CEP technologies can detect complex business events based on low-level events generated by smart objects. BAM technologies can enable organizations to react to these events in time by notifying the right people about them. Moreover, BAM technologies can improve decision making by providing people with comprehensive views over the data generated by smart objects.

10.2.9 Integrating CEP, BAM and BPM

Although CEP, BAM and BPM have very different goals, they can complement each other very well if integrated together [30], [72]. The result of their integration is often called *Event-Driven BPM* (EDBPM). EDBPM systems are BPM systems which utilize the potential of CEP and BAM technologies. They are relevant to our study because of their ability to cope with the information overload created by smart objects. In this section, first, we review two different approaches for integrating CEP and BPM systems. After that, we review a reference model describing the architecture of EDBPM systems.

10.2.9.1 Using CEP in BPM

A CEP system can be used to analyze events coming from a BPM system in real-time [72]. Usually, these are events coming from running business processes instances. In this way, complex event patterns related to business process execution are found during the enactment phase rather than during the evaluation phase. These patterns can be a useful insight when monitoring business process instances. Note that in this case, the BPM system is an event observer.

10.2.9.2 Using BPM to define CEP reactions

A BPM system can be used to support reactions to events detected by a CEP system [72]. In other words, when the CEP system detects a certain situation, this can lead to the triggering of a business process which handles it. This is especially useful when the handling of situations is not trivial and it requires the execution of some complex logic which might also involve people. Moreover, defining reactions as business processes also gives a lot of flexibility because business processes can be easily modified when needed. Note that in this case, the BPM system is an event consumer.

10.2.10 Business process management lifecycle

The BPM lifecycle comprises 4 general phases: *process design and analysis*, *process configuration*, *process enactment* and *process evaluation* [3]. As it is depicted in Figure 4, these phases and their logical relationships form a cyclic structure. Although they are ordered logically in a cycle, this does not necessarily mean that they have the same temporal order. It is possible that the phases are executed in parallel or in any arbitrary order. Next, we discuss each of the 4 phases in more detail.

10.2.10.1 Design phase

The design phase is where, usually, the BPM lifecycle starts from [3]. People specialized in carrying out this phase are called process designers. In order to design business processes, process designers have to examine existing business processes as well as their organizational and technical environment. This includes their identification and modeling. When business processes are modeled, they are expressed in a graphical notation such as BPMN [58], UML [59] or YAWL [60]. In this way they are made explicit which facilitates their future improvement. Once the initial design of a business process is ready, it has to be analyzed and, based on the results, eventually, improved. Usually, the analysis includes validation and verification of the process model. The idea of the validation is to assure that the business process model is actually reflecting the real-world business process as closely as possible. Collaboration with the actual business process participants and simulation techniques are often used in validation. The idea of verification is to assure that the process model does not contain any undesired properties (e.g. deadlocks). Again, simulations are commonly used to do this.

10.2.10.2 Configuration phase

Once a business process has been designed, it has to be implemented [3]. Usually, this is done with the help of a WFM or BPM system. The business process model has to be extended with concrete technical and organizational information which will enable its enactment by a WFM/BPM system. The BPM/WFM system must also be configured to realize the business processes. This configured should take into account the organizational and technical environment of the business processes. At a minimum, it should specify how business process participants interact with the system and how it integrates with other existing software systems. In some cases, it can also include development work related to the integration of legacy systems as well as to the implementation of new ones. Once a business process is implemented, its implementation should be tested. It is important to perform tests on activity and process level in order to assure the proper functioning of the process. At the end, the process is deployed in its target environment.

10.2.10.3 Enactment phase

During the enactment phase, instances of an implemented business process are actually being executed. A business process instance is a "representation of a single enactment of a business process" [26]. It is "created, managed and terminated" by the WFM/BPM system, in accordance with its process definition [26]. Each instance works with separate data and is managed independently. Usually, the start of a process instance is triggered by an event from the external environment (e.g. receiving an insurance claim) [3]. An important part of the enactment phase is monitoring. It is a mechanism for providing information on the status of different process instances. This information is especially valuable for process participants and responsibles because it helps them to track the progress of process instances. Furthermore, it is common that BPM systems gather and store detailed data about the enactment of business processes. This data is used in the evaluation phase for business processes improvement.

10.2.10.4 Evaluation phase

In literature, the evaluation phase is, sometimes, also called diagnosis phase [27]. In this phase, the data gathered by a WFM/BPM system during the enactment phase is used to evaluate and eventually improve business process models and implementations. Two main techniques are used to do that: business activity monitoring (BAM) and process mining. BAM refers to the aggregation, analysis, and presentation of near real-time information related to business process instances. Often, this information is presented visually, for instance, as a set of charts on a dashboard, each corresponding to a key performance indicator. With the help of BAM, processes responsibles can detect problems (e.g. delays and bottlenecks) in near real-time fashion during the business processes enactment. The idea of process mining is to extract knowledge from the data gathered during processes enactment. One popular application is to check the level of conformity between business process models and their actual execution in reality. If the gap between the two is too large, this is a problem which might require the redesign of the process.

10.3 Smart energy

In this section, our goal is to briefly introduce the field of smart energy. We start by defining the concept of smart grid and explaining its main characteristics. After that, we discuss the idea of advanced metering infrastructure since it is the first major milestone towards the development of the smart grid.

10.3.1 Smart grid

Before defining the concept of smart grid, we first discuss the concept of power grid. Note that in literature, this concept can be found with different names such as electrical grid, electrical system and electricity network. A power grid is an interconnected network for delivering electricity. Generally speaking, it consists of five main components: producers which generate power, substations where power voltage is stepped up or down, transmission lines responsible for moving electricity over long distances, distribution lines responsible for delivering electricity to customers and consumers who actually use the power. Power is delivered from producers to consumers.

Smart grid is a relatively new concept in the energy field. It is defined by United States Department of Energy in the following way:

"A smart grid uses digital technology to improve reliability, security, and efficiency (both economic and energy) of the electric system from large generation, through the delivery systems to electricity consumers and a growing number of distributed-generation and storage resources." [50]

A few important things become apparent from this definition. First, the smart grid is an extension of the current electrical system (power grid). Second, its main goal is to improve the reliability, security and efficiency of the electrical system using digital technology. Third, it has to cope with the challenges brought by the "*growing number of distributed-generation and storage resources".* Next, we list 6 main characteristics of the smart grid which can be identified from literature [50].

Enables informed participation by customers. Consumers become active participants in the smart grid. They can help to balance the supply and demand of power in the smart grid by modifying the own way of using power. New technologies such as smart meters and appliances help them to understand and control their own consumption.

Accommodates all generation and storage options. The smart grid accommodates not only large centralized power generation but also other smaller and distributed energy resources such as renewables, micro-generators and power storages. In this way, some consumers become also producers. This distribution of power generation improves the robustness of the smart grid.

Enables new products, services and market. On the one hand, the large number and variety of producers will increase competition. On the other hand, the informed participation of customers will enable them to choose product and services which meet best their needs. These two factors will lead to the development of new products, services and markets.

Provides the power quality for the range of needs. Not all consumers need the same quality of power. The smart grid is able to supply power with different quality and price to meet the needs of different consumers. It relies on predictive maintenance and self-healing in order to avoid and recover from events which impact the power quality.

Optimizes asset utilization and operating efficiency. A smart grid applies the latest technologies to optimize the use of its assets and to improve its total efficiency.

Provides resiliency to disturbances, attacks and natural disasters. The smart grid is able to predict and self-heal from unexpected harmful events. Self-healing consists of detecting, analyzing, responding and restoring from such events. Usually, the strategy to deal with failures is to isolate the problematic part of the grid while the rest of it is restored to normal operation.

10.3.2 Advanced metering infrastructure

The development of *Advanced Metering Infrastructure* (AMI) is considered to be the first major milestone towards the development of a smart grid. Its main goal is to provide an automated, intelligent and two-way connection between consumers and power companies. It can be considered as an extension of the current *Advanced Meter Reading* (AMR) which enables automatic communication in only one direction (from consumers to electricity companies). On the one hand, AMI enables electricity companies to remotely collect information about consumers. They can collect information related to consumption and power quality (e.g. voltage, current, frequency) as well as receive notifications about important events (e.g. a power outage). On the other hand, AMI enables electricity companies to communicate information to consumers and control their consumption. By doing the former, they can provide consumers with pricing and billing information. By doing the latter, companies can limit the consumption of certain consumers or even completely stop their power supply.

AMI is not a single technology but rather a combination of many technologies. This includes embedded, communication, data management and many other technologies. However, in its core are special electronic devices called *smart meters*. They are uniquely identifiable, communication enabled devices with sensing and actuation capabilities. A consumer having a smart meter can be considered a smart object according to the definition presented in section 2.1.1. Moreover, it can be classified as a ISAN-smart object according to the model presented in the same section. We are going to call such consumers: *smart consumers*. Smart meters collect consumption and power quality data and periodically send it to electricity companies. They also notify electricity companies about important events such as power outages. Note that smart meters usually have batteries so that they can continue their work after a power failure. Electricity companies can control smart meters remotely by sending commands to them. For instance, they can remotely use the smart meter to stop the power supply of a smart consumer. Many smart meters possess also user interface through which they provide consumers with useful information related to tariffs, power usage and power quality.

Currently, many countries within the EU as well as outside of it are involved in large projects aiming to replace conventional meters with smart ones and, respectively, develop an AMI. Different countries have different visions how and when this modernization has to be done. So far, the leading European countries in smart meter deployment are Sweden and Italy [14]. More than 90% of the electricity consumers in these two countries have already installed smart meters. In the Netherlands, so far, a large-scale deployment of smart meters has not started yet [14]. In contrast to many other countries, in 2010 the Dutch government has decided that smart meter installation will be voluntary. Moreover, if a customer decides to install a smart meter, he can choose if and how often the data it collects will be transferred to the utility companies. In the Netherlands, smart meters will measure not only electricity but also gas.

10.3.3 Scenarios

10.3.3.1 Smart metering scenario

Here we present a realistic use case from the field of electricity smart metering related to the billing of customers. We have identified this use case with the help of Capgemini Netherland. Let XYZ be a fictitious electricity company which operates an AMI. All of its customers have smart meters and the company collects automatically information about their consumption on a daily basis. XYZ bills its customers every month based on the collected data. Unfortunately, some customers fail to pay their bills on time. In such cases, the company notifies each customer by mail that he has to pay within one week. If he fails to do so, the company uses his smart meter to limit his power consumption to several hours per day. If the customer has not paid his bills within two more weeks, the company uses his smart meter to disconnect him completely from the power grid. Of course, if the customer pays, his normal power supply is restored.

10.3.3.2 Self-healing smart grid scenario

Here we present a use case related to outage management in a self-healing smart grids. We have identified this use case with the help of Capgemini Netherland. Let XYZ be a fictitious electricity company which operates a smart grid. In order to detect outages in its section of the smart grid, the company relies on the smart meters of its customers. Once a consumer is left without power, his smart meter automatically sends an outage notification to the electricity company. Compared to waiting for the customers to call and complain about an outage, this approach enables much faster reaction to problems. Based on the provided outage data, the company estimates the place of the failure and tries to isolate the affected sector of the grid in order to prevent potential spread of the disruption. The problem is isolated with the help of smart switches (circuit breakers) which are closed or opened remotely in order to take out of service the damaged part of the grid and keep the normal operation in the rest of it. Once this is done, the electricity company sends a crew which goes on the spot and repairs the problem. When the crew is ready, the isolated section is brought back online with the help of smart switches. When the power supply of individual consumers is restored, their smart meters send notifications to the electricity company. In this way the company gets feedback if the problem was resolved successfully.

10.4 Use cases

In this section, we describe two of the generic use cases in more detail. Still, our focus is on the main scenarios and we do not describe the possible alternative scenarios.

Table 14: Use case - Define IFP rule

Name	Define IFP rule
Goal	UG1
Actors	Data analyst
Preconditions	It is clear what type of information will be provided by devices
Postconditions	The IFP rule is stored permanently in the system and the system is utilizing the rule to process information from devices
Trigger	The actor decides to define a new IFP rule
Main scenario	1. The actor selects a name for the new IFP rule 2. The actor uses a special notation/language to express the condition and action parts of the rule [28] 3. The actor tests if the rule is functioning as expected 4. The actor chooses to save the rule 5. The system stores the rule and starts using it

Table 15: Use case - View monitoring objects

Name	View monitoring objects
Goal	UG8

Actors	Process responsible
Preconditions	• There are some monitoring objects already defined in the system • There are devices connected to the system • There are running business process instances
Postconditions	The system stores permanently the dashboard arrangements of the actor
Trigger	The actor decides to monitor the physical environment
Main scenario	1. The actor selects a business process instance 2. The actor selects a number of monitoring objects related to this instance 3. The actor arranges them into a dashboard 4. The system populates the monitoring objects with information 5. The actor saves the dashboard